UNIVERSITY OF THE BASQUE COUNTRY
UPV/EHU

DOCTORAL THESIS

On the Design, Implementation and Application of Novel Multi-disciplinary Techniques for explaining Artificial Intelligence Models

Author:
Alejandro Barredo Arrieta

Supervisors:
Prof. Dr. Javier Del Ser
Dr. Sergio Gil

A Thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy in the

Department of Communications Engineering

December 10, 2021

"Success is not final, failure is not fatal, it is the courage to continue that counts."

Winston Churchill

ABSTRACT

Artificial Intelligence is a non-stopping field of research that has experienced an incredible growth last decades. Some of the reasons for this apparently exponential growth are the improvements in computational power, sensing capabilities and data storage which results in a huge increment on data availability. However, this growth has been mostly led by a performance based mindset that has pushed models towards a *black-box* nature. The performance prowess of these methods along with the rising demand for their implementation has triggered the birth of a new research field. Explainable Artificial Intelligence. As any new field, XAI falls short in cohesiveness. Added the consequences of dealing with concepts that are not from natural sciences (*explanations*) the tumultuous scene is palpable. This thesis contributes to the field from two different perspectives. A theoretical one and a practical one. The former is based on a profound literature review that resulted in two main contributions: 1) the proposition of a new definition for Explainable Artificial Intelligence and 2) the creation of a new taxonomy for the field. The latter is composed of two XAI frameworks that accommodate in some of the raging gaps found field, namely: 1) XAI framework for Echo State Networks and 2) XAI framework for the generation of counterfactuals. The first accounts for the gap concerning Randomized neural networks since they have never been considered within the field of XAI, although some of the main concerns for not pursuing their application is related to the mistrust generated by their *black-box* nature. The second presents a new paradigm to treat counterfactual generation. The search for counterfactuals is governed by three different objectives as opposed to the classical approach in which counterfactuals are just generated following a minimum distance approach of some type. This framework allows for an in depth analysis of a target model by means of counterfactuals responding to: *Adversarial Power*, *Plausibility* and *Change Intensity*. All in all, the achievements this Thesis reports contribute to the general knowledge on model explainability in a momentum when Artificial Intelligence model must proliferate and become trustworthy in almost all disciplines and fields.

My dear, here we must run as fast as we can, just to stay in place.
And if you wish to go anywhere you must run twice as fast as that.

— Lewis Carroll - Alice's Adventures in Wonderland

ACKNOWLEDGEMENTS

To *Natalia Díaz-Rodriguez*, *Javier Del Ser*, *Adrien Bennetot*, *Siham Tabik*, *Alberto Barbado*, *Salvador Garcia*, *Sergio Gil-Lopez*, *Daniel Molina*, *Richard Benjamins*, *Raja Chatila*, *Ibai Laña*, *Miren Nekane Bilbao* and *Francisco Herrera* for their contributions to this thesis.

This journey started three years ago when I entered Tecnalia Research and Innovation. After a tough wait to get the papers right and during a cold winter I finally met most of the people to whom I must thank.
This Thesis would have never been possible without the guidance, help and indisputable contribution of *Javi* to me and the quality of this Thesis. It has been a pleasure to witness the level to which this professional is able to perform and pull the rest around him. I also have to recognize the work of my other supervisor, *Sergio*, who has been the source of great inspiration and some of the most interesting talks I have ever had.
I could not forget *Esther* for being such a hard working colleague that has never turn a cold shoulder. From great discussions during lunch hours to unforgettable lessons whose teacher she has been. She has become a beacon to what I want for myself. I have to also mention *Iratxe* since this has been a shared path among us and I will never forget the countless arguments she lost over the years (just kidding). To the rest of my OPTIMA colleagues I owe the great environment that I have been lucky to feel part of during these years. To *Isidoro*, *Elena*, *Joseba* and *Iñigo* for giving me this opportunity within ICT.
To my friends from Palado I owe many life lessons we have learned together for they have been probably the people with whom I navigated some of the most crazy adventures in my life. To my brothers, *Xabier* and *Iñigo*, for not letting the bar fall, keeping me on my toes and being an inextinguishable source of inspiration and pride.
Finally, to my parents. For being the reason of all my successes. For being a beacon of support and motivation even in times I did not believe in myself. I could never thank them enough for all the sacrifices done for me and the relentless hand I felt supporting me through all my life.

CONTENTS

1 INTRODUCTION 1
- 1.1 Context . . . 1
- 1.2 Motivation and Objectives . . . 3
- 1.3 Outline and Contributions . . . 4
  - 1.3.1 Chapter 2: Background . . . 4
  - 1.3.2 Chapter 3: On the Post-hoc Explainability of Deep Echo State Networks . . . 4
  - 1.3.3 Chapter 4: Exploring the Trade-off in Counterfactual Explanations 5
  - 1.3.4 Chapter 5: Concluding Remarks . . . 5
- 1.4 Reading this Thesis . . . 5

2 BACKGROUND 7
- 2.1 Explainability: What, Why, What For and How? . . . 8
  - 2.1.1 Terminology Clarification . . . 8
  - 2.1.2 What? . . . 9
  - 2.1.3 Why? . . . 11
  - 2.1.4 What For? . . . 12
  - 2.1.5 How? . . . 15
- 2.2 Transparent Machine Learning Models . . . 19
  - 2.2.1 Logistic/Linear Regression . . . 20
  - 2.2.2 Decision Trees . . . 21
  - 2.2.3 K Nearest Neighbors . . . 22
  - 2.2.4 Rule Based Learning . . . 23
  - 2.2.5 General Additive Models . . . 23
  - 2.2.6 Bayesian Models . . . 24
- 2.3 Post-hoc Explainability Techniques for Machine Learning Models . . . 24
  - 2.3.1 Model-agnostic Techniques for Post-hoc Explainability . . . 27
  - 2.3.2 Post-hoc Explainability in Shallow ML Models . . . 29
  - 2.3.3 Tree Ensembles, Random Forests and Multiple Classifier Systems 29
  - 2.3.4 Support Vector Machines . . . 30
  - 2.3.5 Explainability in Deep Learning . . . 32
  - 2.3.6 Hybrid Transparent and *black-box* Methods . . . 37
  - 2.3.7 Alternative Taxonomy of Post-hoc Explainability Techniques for Deep Learning . . . 39
- 2.4 Open Issues in the Field of XAI . . . 40
  - 2.4.1 On the Tradeoff between Interpretability and Performance . . . 41
  - 2.4.2 On the Concept and Metrics . . . 42
  - 2.4.3 Challenges to Achieve Explainable Deep Learning . . . 43
  - 2.4.4 Explanations for AI Security: XAI and Adversarial Machine Learning . . . 44
  - 2.4.5 XAI and Output Confidence . . . 46
  - 2.4.6 XAI in Randomized Neural Networks . . . 46
  - 2.4.7 XAI, Rationale Explanation and Critical Data Studies . . . 47

2.4.8 XAI and Theory-guided Data Science . . . 47
2.4.9 Guidelines for ensuring Interpretable AI Models . . . 48
2.5 Summary . . . 49
3 ON THE POST-HOC EXPLAINABILITY OF DEEP ECHO STATE NETWORKS 51
3.1 Related Work . . . 53
3.1.1 Echo State Network: Fundamentals . . . 53
3.1.2 Explainability of Recurrent Neural Networks . . . 56
3.1.3 Importance Attribution Methods . . . 57
3.2 Proposed Framework . . . 57
3.2.1 Potential Memory . . . 58
3.2.2 Temporal Patterns . . . 60
3.2.3 Pixel Absence Effect . . . 64
3.2.4 Benefits of the framework for Different Audiences . . . 66
3.3 Experimental Setup . . . 67
3.3.1 Time Series Forecasting . . . 67
3.3.2 Image Classification . . . 68
3.3.3 Video Classification . . . 69
3.4 Results and Discussion . . . 71
3.4.1 Time Series Analysis . . . 71
3.4.2 Image Classification . . . 75
3.4.3 Video Classification . . . 77
3.5 Summary . . . 79
4 EXPLORING THE TRADE-OFF IN COUNTERFACTUAL EXPLANATIONS 81
4.1 Related Work . . . 83
4.1.1 Deep Learning for Image Classification and Generative Modeling 83
4.1.2 Explainable Artificial Intelligence (XAI) and Counterfactual Explanations . . . 84
4.1.3 Multi-objective Optimization . . . 86
4.2 Proposed Framework . . . 87
4.2.1 Design Rationale . . . 87
4.2.2 Structure and Modules . . . 88
4.2.3 Target Audiences and Examples of Use Cases . . . 92
4.3 Experimental Setup . . . 92
4.3.1 Considered GAN Architectures . . . 93
4.3.2 Audited Classification Models . . . 96
4.3.3 Multi-objective Optimization Algorithm . . . 96
4.4 Results and Discussion . . . 98
4.4.1 Experiment #1: BicycleGAN-based Counterfactual Generation for Auditing a `Shoe` Versus `NoShoe` footwear classifier . . . 99
4.4.2 Experiment #2: AttGAN-based Counterfactual Generation for Auditing a `Man` Versus `Woman` gender classifier . . . 100
4.4.3 Experiment #3: ShapeHDGAN-based Counterfactual Generation for Auditing a `Chair` Versus `Xbox` voxel classifier . . . 101
4.4.4 Experiment #4: StyleGAN2-based Counterfactual Generation for Auditing a classifier of `Cathedral` Versus `Office` classifier . . . 103

4.4.5 Experiment #5: CGAN-based Counterfactual Generation for Auditing a `MNIST` Classifier . . . 104
4.5 Summary . . . 105
5 CONCLUDING REMARKS 109
5.1 List of Publications . . . 111
5.2 Future Research Lines . . . 111
BIBLIOGRAPHY 113

LIST OF FIGURES

Figure 2.1 Diagram showing the different purposes of explainability in ML models sought by different audience profiles 11
Figure 2.2 Conceptual diagram exemplifying the different levels of transparency characterizing a ML model 16
Figure 2.3 Conceptual diagram showing the different post-hoc explainability approaches available for a ML model M_φ. 18
Figure 2.4 Graphical illustration of the levels of transparency of different ML models considered in this overview 20
Figure 2.5 Taxonomy of the reviewed literature and trends identified for explainability techniques related to different ML models 26
Figure 2.6 Examples of rendering for different XAI visualization techniques on images. 35
Figure 2.7 Feature visualization at different levels of a certain network [298]. 36
Figure 2.8 Three different examples of explanation when using LIME on images [74]. 36
Figure 2.9 Pictorial representation of a hybrid model combining different paradigms 38
Figure 2.10 Alternative Deep Learning specific taxonomy extended from [16] and its connection to previous taxonomy Figure 2.5 40
Figure 2.11 Trade-off between model interpretability and performance, and a representation of the potential area of improvement 42
Figure 3.1 Schematic diagram showing a canonical ESN, and the multi-layered stacking architecture of a Deep ESN model 54
Figure 3.2 Input and target sinusoids used for training an ESN model with $\alpha = 0.5$ and a single reservoir with $N \in \{5, 50, 500\}$ neurons 60
Figure 3.3 Representation of an ESN model by means of recurrence plots with three different layer configurations 62
Figure 3.4 Mean absolute error as a function of the number of reservoirs and average difference evolution between layer configurations . 63
Figure 3.5 Schematic diagram showing the process of converting an RGB image to a $K = 3$ time series $\mathbf{u}(t)$. 64
Figure 3.6 Segments of the different time series datasets used in the experiments 65
Figure 3.7 Schematic diagram showing the process of converting a video to time series 70
Figure 3.8 Potential memory analysis of the ESN model trained over the different datasets 72
Figure 3.9 Temporal patterns analysis over the NARMA, battery and traffic datasets 74

Figure 3.10 Pixel absence effect over a Deep ESN model trained to classify MNIST digits for single pixels. 76
Figure 3.11 Pixel absence analysis of a video frame that gives importance to sections of the image that apparently should not be important 78
Figure 3.12 Pixel absence analysis of videos presenting spatially disjoint and overlapping classes 79
Figure 4.1 Conceptual representation of the rationale behind the core of the proposed framework 86
Figure 4.2 Block diagram of the proposed framework based on three criteria: plausibility, adversarial power and change intensity. . . . 89
Figure 4.3 Block diagram of the proposed systems: BicycleGAN, AttGAN, ShapeHDGAN, StyleGAN2; and Conditional GAN (CGAN). . . 93
Figure 4.4 Pareto front of the counterfactual examples generated for a `Shoe` example and the BicycleGAN model 99
Figure 4.5 Analysis of the average RGB luminance of the `Shoe` vs `NoShoe` dataset for the BicycleGAN experiment 100
Figure 4.6 Pareto front of the counterfactual examples generated for a `male` example and the AttGAN model 101
Figure 4.7 Diagram showing the occurrence of different feature combinations, split between `male` and `female` examples 102
Figure 4.8 Pareto front of the counterfactual examples generated for a `chair` example and the Shape3DGAN model 102
Figure 4.9 Local explanations corresponding to the anchor voxel and two of the counterfactuals depicted in Figure 4.8. 103
Figure 4.10 Pareto front of the counterfactual examples generated for a `Cathedral` example and the StyleGAN2 model 104
Figure 4.11 Comparison between the original counterfactuals and the anchor image following the colored markers in Figure 4.6 105
Figure 4.12 Pareto front of the counterfactual examples generated for an MNIST digit classification model and the CGAN model 106
Figure 4.13 Sample of the unfinished digits generated for supplementing the MNIST dataset and the output generated when trained with these samples 106

LIST OF TABLES

Table 2.1 Goals pursued in the reviewed literature toward reaching explainability, and their main target audience. 12
Table 2.2 Overall picture of the classification of ML models attending to their level of explainability. 19
Table 3.1 List of action recognition video datasets considered in the experiments, along with their characteristics 69
Table 3.2 Table presenting the parameters of the considered ESN and Conv2DLSTM models. 71
Table 3.3 Accuracy and number of trainable parameters of models tested over each video classification dataset. 76
Table 4.1 Structure and training parameters of the models audited by the proposed framework. 97
Table 4.2 Dataset and accuracy of the different classifiers put to the test . 97

ACRONYMS

xAI	Explainable Artificial Intelligence
LR	Logistic Regression
DT	Decision Tree
GAM	Generative Additive Models
RL	Rule List
KNN	K-Nearest Neighbors
RF	Random Forest
SVM	Support Vector Machine
MLP	Multi-Layer Perceptron
CNN	Convolutional Neural Network
RNN	Recurrent Neural Network
ESN	Echo State Network
DESN	Deep Echo State Network
NN	Neural Network

DNN Deep Neural Network

LRP Layer-wise Relevance Propagation

Grad-CAM Gradient-weighted Class Activation Mapping

BP Back Propagation

DGN Deep Generator Network

LIME

HMM Hidden Markov Models

KB Knowledge Base

KF Kalman Filter

DKF Deep Kalman Filter

DVBF Deep Variational Bayes Filters

VAE Variational Auto-Encoder

SVAE Structural Variational Auto-Encoder

DKNN Deep K-Nearest Neighbor

CBR Case Based Reasoning

GAN Generative Adversarial Network

LSTM Long Short-Term Memory

NARMA Non-linear Auto-Regressive Moving Average

RGB Red Green Blue

Conv Convolution

cVAE-GAN Conditional Variational Auto-Encoder - Generative Adversarial Network

LISP List-Processing

1 INTRODUCTION

1.1 CONTEXT

Society has been witness of the different rises and falls of Artificial Intelligence (AI) in history [2]. From the promises and abandonment of early machine translation and connectionism 1950-1970. Followed by the conquest and collapse of the reign of LISP (list-processing) 1980-1990 to the current heights with DL (Deep Learning) 1997-2021. Currently, AI accounts for 3.8% of the whole peer-review research corpus in 2020 from a 1.3% in 2011 [3]. To match this wonderful trend, the amount of journal publications grew by 34.5% from 2019 to 2020. Now it constitutes a body of research of over eighty thousand articles [3]. AI community has the obligation to strengthen its vulnerabilities so that another winter does not happen.

In the beginning, AI systems were tractable and easy to understand, however, this trend has long been broken. The upsurge of Deep Learning (DL) comprises a restless search for empirical prowess that obscures the internals of the models. The vastness of the parametric space of such systems make them less intuitive and opaque. Hence the coining of the term *black-box* [4] which successfully transmits the overall feeling of mistrust. The opposition to this frenzy of performance is governed by the search for *transparency* which invokes a full understanding of the internals of a given model [5].

The continuous increment in AI's performance has been pushing the implementation of algorithms in increasingly more fields. The interaction of this fact with the growing nature of opaqueness has started to grow concerns among users [6]. Concerns that revolve around the usage of models that do not allow for detailed explanations about their behaviour [7]. Most fields of application would benefit of, if not demand, explanations that support a model's output. Something that eases the user's task of welcoming this new tool to their toolbox. Furthermore, AI has always been closely linked to ethics due to its connection to human life. Nowadays, aspects of this sort are starting to arise in national and international regulations which enforces the need for Explainable Artificial Intelligence.

Interpretability, tractability and trustworthiness are characteristics that humans usually seek prior to depositing their trust in something. These characteristics could be derived from general ethics or Machine Learning ethics when specifically directed towards the usage of ML [8–10]. AI, as other fields, cannot base its advancement in performance solely. A single-objective driven research leaves other interesting properties aside and starts to drift from its primary intent. Performance driven research is one of the main reasons for the existence of a tradeoff between model interpretability and performance. The allusion to this tradeoff is played frequently to advocate for Neural Networks. However, the existence of such a tradeoff is not as straightforward as it appears. Creating an example where a interpretable model (e.g. linear regression) fails to learn the pattern in a given dataset, while a *black-box* (e.g. neural

network) does learn it accurately is quite simple. However, the opposite is also true, it is not hard to devise an example where the opposite happens. A neural network overfits a dataset and is unable to predict correctly while the interpretable model does. These two cases would present opposite examples of the tradeoff theory. Being there a tradeoff between performance and interpretability does not mean that a more complex model always results in more performance and less interpretability. It means that, given a setting where highly complex relations are to be learned, the selection among a more complex model (better performance) and a less complex one (worse performance) will rise the dilemma of the tradeoff. Consequently, the tradeoff is not solely linked to the concept of performance. It is largely linked to the context of the model itself. Problems where features are intractable, non-relatable and highly diverse require complex models to learn the patterns contained within. In such problems is where this tradeoff appears, a simpler, interpretable model is not able to capture the whole of it and fails to perform adequately. A more complex *black-box* model is able to capture the extent of the problem but is non interpretable.

One way of easing this problem is to lay a bit of attention at interpretability. Machine Learning (ML) models could greatly benefit from it. By providing subsystems or techniques that improve the understanding of a model's predictions, visualize its learned rules or inform about the vulnerabilities that might divert its prediction, ML models could improve their interpretation by three main points:

- Bias could be minimized when a better understanding of the workings is present.
- Robustness could be improved by means of a deeper knowledge about the involved parameters and their repercussion
- Guaranties about the real reasons driving the output of the model could be verified.

These three outcomes are closely related to AI ethics, namely: *Equity, Reproducibility and Responsibility.*

The field of *eXplainable AI* (XAI) [7] is in charge of giving birth to the set of techniques that will allow to 1) create highly performing models that are explainable and 2) effectively aid humans to trust, understand and manage the upcoming generation of artificial intelligent partners. However, this is still a toddler of a field that has many discrepancies that need to be sorted out. The literature about XAI, although, growing significantly, has not still agreed upon the common grounds in the context of Explainable Artificial Intelligence. Due to its intersection with the nature of explanations its definitions are still a bit vague and need to be addressed. Connected to this gap in the agreement of the common definitions, a taxonomy should be built. One that accounts for all the work conducted in the field that would serve as a departing point to any new-comer interested in the field. Product also of these differences in terms and concepts, a space that needs filling is devised. A space to accommodate for the metrics needed to evaluate the advances of the field. Finally, coupled with all this a plethora of methods and frameworks are needed to fulfill the needs of researchers and users alike

1.2 MOTIVATION AND OBJECTIVES

As stated before, AI is a growing field that that has seen its troops more than doubled in the last years. This growth has come hand in hand with the breaking of almost all the barriers found in the field until the date. However, the paths needed to be taken in order to achieve these marvels have also brought opaqueness in their functioning, followed by mistrust from users. Deep Learning is one by one conquering all the performance milestones in the field, from surpassing humans ability in object detection thanks to ImageNet [11] to the conquest of Go, what was deemed the hardest game in the world [12], to mastering human machine translation [13] and surpassing the ability to generate human photo realistic images [14].

All these successful milestones had one thing in common. They are all part of a context of unbelievable complexity, data diversity and dimension, hence the trade-off mentioned earlier (interpretability vs accuracy) comes into play again. There is no doubt that these achievements are a product of the development in computation power and Deep Learning. Which means, they would not have been possible without *"black box"* models. However, such models always bring up concerns of *"what if?"*. What if a model such as this was deployed in a sensible environment. An environment in which these model's decisions would impact human lives. There are two possible routes to take from this dilemma if we sort out the ones in which the models are discarded. On the one hand, the problem could be solved by bypassing the model with a human observer that will always take the final decision. Although the solution might be effective, it would not be efficient since many of the strengths of the model would be limited (speed and autonomy). On the other hand, given the right tools, the internals of a model should be understood. Its boundaries and deficiencies minimized and its working made compliant with regulations. Given that this situation is getting more and more common each day, XAI's field is getting strength and it is the cornerstone of AI implementation in the real world.

Being the development of AI as it has been these last years, most world governments and agencies have put plans in place to address the gap found in the explainability of the models. Many works [15–21] have summarized the development of the field of XAI, however, they have not focused in the search for a unified framework of concepts and metrics that will empower scrutiny and analysis over the field and its proposed methods. This thesis focuses in two main things. First, the creation of a framework that will serve its purpose as a guide. A guide meant to recollect and categorize all the work that has been done during the years under a common set of definitions and concepts that will ease the future development of the field. Second, the addressing of the challenges devised in the field by means of the creation of new techniques and frameworks. Hence, the focus of this thesis can be divided in four different objectives:

- **Surveying the field of XAI**: A systematic review of the state of the art that goes past the simple recollection of the advancements of the field. Attempting to understand the different reasons involved in the search for XAI. *"What is it?", "Why is it done?", "What for is it done?" and "How is it done?"*

- **Proposition of a new set of concept definitions**: A set of concepts that summarize the diverse references found in the literature. There are many concepts and reasons related to the field of XAI that have been mentioned but not summarized nor agreed upon. Concepts of: *understandability, comprehesibility, interpretabiliy, explainability* and *transparency*. And reasons covering: *trustworthiness, causality, transferability, informativeness, confidence, fairness, accessibility, interactivity* and *privacy awareness*.
- **Creation of a new taxonomy for the field based on these definitions**: A structure that, based on set of agreed upon terms, is able to account for all the works conducted in the field.
- **Creation of a framework that improves a vulnerability of the field**: Given the analysis obtained from the previous three objectives. This thesis contributes to filling some technical gaps with the creation of two XAI frameworks. This frameworks are centered around the idea of including the audience in the loop of XAI generation. One will attempt at focusing out of the mainstream arena while the other proposes a new approach that strengthens one of the core methodologies of XAI by paying a closer attention to humans.

1.3 OUTLINE AND CONTRIBUTIONS

This thesis is outlined by following the previously mentioned overall structure. One chapter accounts for the first three, more theoretical, objectives previously mentioned. Followed by two other chapters concerning the technical (application) contributions of the thesis. Finally a last chapter containing the final remarks and future lines of work. A brief summary of each chapter is described below.

1.3.1 *Chapter 2: Background*

This contains the theoretical study and contribution of the thesis. It firsts introduces an analysis of the main concepts of the search for XAI, namely: *"What"*, *"Why"*, *"What for"* and *"How"*. This is where a proposition for the definition of Explainable Artificial Intelligence is risen. Followed by the proposition of a new taxonomy that categorizes every technique generated during the years. Then the most important works concerning each of the categories of the taxonomy are described.

1.3.2 *Chapter 3: On the Post-hoc Explainability of Deep Echo State Networks*

This chapter presents the first technical contribution concerning the field of randomized neural networks, which have no methods nor frameworks at disposal for seeking explainability. It briefly presets the underlying theoretical concepts needed for understanding randomized neural networks, specifically Echo State Networks (ESN). Then presents a novel framework to address explainability in the context of ESN and Deep ESNs with its corresponding experiments proving its validity for different audiences.

1.3.3 *Chapter 4: Exploring the Trade-off in Counterfactual Explanations*

The forth chapter introduces the second technical contribution of this thesis. It proposes the paradigm of treating counterfactual generation as a multi-objective problem by means of generative adversarial networks. This contribution presents a framework in which counterfactual generation is utilized to audit a target model. Aside from what it has been proposed previously, this framework considers three main objectives for this search, namely: *Adversarial power*, *Plausibility of the solution* and *Change intensity*.

1.3.4 *Chapter 5: Concluding Remarks*

This last chapter summarizes the concluding remarks product of the study carried throughout this thesis. This chapter also presents the outcomes of this thesis in a quantifiable manner accounted for in a set of research contributions submitted to specialized journals and conferences. Finally, it describes the future lines of research that could be of great importance for the field.

1.4 READING THIS THESIS

As mentioned before, this thesis contemplates two different branches of research. The first one focuses on a theoretical analysis of the field. The second gathers the technical contributions presented. Due to the different sub-fields these contributions relate to, a reader may want to jump around the content of this thesis as it fits their needs. Chapter 2 contains the theoretical part of the thesis and serves as a backbone to the motivations for the subsequent two chapters. Its reading could be beneficial to the overall understanding. However, this is not necessary, specially for the audience with a certain expertise in the field of Explainable Artificial Intelligence. Then, Chapter 3 and 4 cover the specific analysis carried out upon the fields of Echo State Networks (former) and counterfactual explanations (latter). Finally, Chapter 5 summarizes the resulting conclusions of all the work so alternating each chapter with the last one can be a reasonable way of approaching the reading.

2

BACKGROUND

This first chapter covers the theoretical contribution of this thesis. As introduced before, the rising trend of contributions and concerns around XAI and related concepts motivates an in depth analysis of the field. This literature outbreak shares its rationale with the research agendas of national governments and agencies. Although some recent surveys [15–21] summarize the upsurge of activity in XAI across sectors and disciplines, this first chapter aims to cover the creation of a complete unified framework of categories and concepts that allow for scrutiny and understanding of the field of XAI methods. As we will later show in detail, model explainability is among the most crucial aspects to be ensured for the fruitful implementation of AI in the real world. All in all, the novel contributions of this chapter can be summarized as follows:

1. Grounded on a first elaboration of concepts and terms used in XAI-related research, we propose a novel definition of explainability that places *audience* (Figure 2.1) as a key aspect to be considered when explaining a ML model. We also elaborate on the diverse purposes sought when using XAI techniques, from trustworthiness to privacy awareness, which round up the claimed importance of purpose and targeted audience in model explainability.

2. We define and examine the different levels of transparency that a ML model can feature by itself, as well as the diverse approaches to post-hoc explainability, namely, the explanation of ML models that are not transparent by design.

3. We thoroughly analyze the literature on XAI and related concepts published to date, covering approximately 400 contributions arranged into two different taxonomies. The first taxonomy addresses the explainability of ML models using the previously made distinction between transparency and post-hoc explainability, including models that are transparent by themselves, Deep and non-Deep (i.e., *shallow*) learning models. The second taxonomy deals with XAI methods suited for the explanation of Deep Learning models, using classification criteria closely linked to this family of ML methods (e.g. layerwise explanations, representation vectors, attention).

4. We enumerate a series of challenges of XAI that still remain insufficiently addressed to date. Specifically, we identify research needs around the concepts and metrics to evaluate the explainability of ML models, and outline research directions toward making Deep Learning models more understandable. We further augment the scope of our prospects toward the implications of XAI techniques in regards to confidentiality, robustness in adversarial settings, data diversity, and other areas intersecting with explainability.

The remainder of this chapter is structured as follows: first, Section 2.1 and subsections therein open a discussion on the terminology and concepts revolving around

explainability and interpretability in AI, ending up with the aforementioned novel definition of interpretability (Subsections 2.1.1 and 2.1.2), and a general criterion to categorize and analyze ML models from the XAI perspective. Sections 2.1.5.1 and 2.1.5.2 proceed by reviewing recent findings on XAI for ML models (on transparent models and post-hoc techniques respectively) that comprise the main division in the aforementioned taxonomy. We also include a review on hybrid approaches among the two, to attain XAI. Benefits and caveats of the synergies among the families of methods are discussed in Section 2.4.3, where we present a prospect of general challenges and some consequences to be cautious about. Finally, Section 2.5 concludes the chapter with an outlook aimed at engaging the community around this vibrant research area, which has the potential to impact society, in particular those sectors that have progressively embraced ML as a core technology of their activity.

2.1 EXPLAINABILITY: WHAT, WHY, WHAT FOR AND HOW?

Prior to any further analysis of the literature. It is of paramount importance to firstly set the basis for what *explainability* stands for in the field of AI and ML more precisely. This section revisits the many definitions (what?), reasons (why?, what for?) and the paths to achieve it (how?) and builds the grounds for the rest of this thesis.

2.1.1 *Terminology Clarification*

The first issue that impedes the establishment of a common ground in the field of XAI is the interchangeable use of the terms *interpretability* and *explainability*. These two concepts are distinct and therefore they should be differentiated. On the one hand, *interpretability* acts as a passive property of a given model. This property, also called transparency some times, refers to the level of sense a given model makes for a human observer. On the other hand, *explainability* represents an active property. *Explainability* describes the actions taken by a given model with the intent of making its functions clearer or easier to understand for a human observer. Among the most commonly used nomenclature, five terms are distinguishable, namely: *Understandability*, *Comprehensibility*, *Interpretability*, *Explainability* and *Transparency*.

- *Understandability* refers to the ability a given model has to ease a human's understanding of its workings. Also called *intelligibility*, does not need to explain its internal functioning nor the manner in which the data is processed internally [22].

- *Comprehensibility* stands for the ability of a given model to present its internal knowledge in an understandable way for a human observer [23–25]. This brief notion can be extracted from the work of Michalski [26]. From the author itself, "*the results of computer induction should be symbolic descriptions of given entities, semantically and structurally similar to those a human expert might produce observing the same entities. Components of these descriptions should be comprehensible as single 'chunks' of information, directly interpretable in natural language, and should relate quantitative and qualitative concepts in an integrated fashion*". Given the difficulty of

its quantification, complexity evaluation is usually closely placed to a model's *comprehensibility* [21].

- *Interpretability* acts as the inherent ability of a model to present its meaning in an understandable manner for humans.
- *Explainability* refers to the interface between a model and a human. The explanation works as an accurate proxy of the model that is also comprehensible to humans [21].
- *Transparency* responds to the property of a model that is understandable. Given that understandability can be measured in different levels, transparent models are divided in three different categories: simulatable models, decomposable models and algorithmically transparent models [5].

From the definitions presented above, *understandability* places itself as the most relevant one for XAI. All the definitions are somehow tied to this notion. *Transparency* and *interpretability* refer to the property of a model of being inherently understandable. *Comprehensibility* moves the scope towards the audience and comprehends their capability to understand a model. Understandability links these concepts in that it measures the degree to which a human can understand a decision made by a model (*Transparency* and *interpretability*) or the knowledge contained within the model (*comprehensibility*). This differentiation brings up the importance of introducing the audience in the definition of XAI. The audience bridges understandability by its two fronts and makes it the backbone of XAI.

2.1.2 *What?*

The field of philosophy is still discussing the existence for a unified theory of explanation. A theory able to form an approximation of a common structure and intent around an explanation [27]. However, such an achievement is still uncompleted. The best of efforts have brought together different approaches extracted from different knowledge disciplines. The same has happened when addressing interpretability in AI. There is not a consensus of what these terms (interpretability and explainability) really mean yet. However, this fact has not stalled the claims of achievements referring to interpretable models and techniques that brandish explainability. As a starting point to from which to build upon it might be helpful to use the definition of Explainable Artificial Intelligence (XAI) coined by D: Gunning [7].

> *"XAI will create a suite of machine learning techniques that enables human users to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners"*

This definition is composed by two main elements, namely, understanding and trust. These two concepts must be addressed beforehand, however it faults at considering other concerns that revolve around this issue. Causality, transferability, informativeness, fairness and confidence [5, 28–30]. These concepts will be covered thoroughly later, although it seemed important to mention them in here to support the claim of incompleteness of the prior definition of XAI.

As lightly introduced in the paragraph before, a clear and complete definition of Explainable Artificial Intelligence is still out of reach. To work towards the completeness of this definition, first a clear denotation of explanation is required, since a broader reformulation of the definition would fall short of compelling the extent of it.

An explanation can be defined as *"the details and reasons that someone gives to make something clear or easy to understand"* (definition by the Cambridge Dictionary of English [31]). By translating this into the context of Machine Learning: *"the details or reason a model gives to make its functioning clear or easy to understand"*, two main stress point can be devised. First, details or reasons ought to be linked to an audience in order to confer any meaning. Second, the degree to which these details or reasons have completely resolved the doubts from the issue being explained is also completely dependent on the audience to which they have been presented. It follows logically that in order to solve these discrepancies, a reference to the audience must be included within the definition of explanations in the context of machine learning models. This could read as:

> *Given a certain audience, explainability refers to the details and reasons a model gives to make its functioning clear or easy to understand.*

Explanations are somehow necessarily linked to the concepts of weighting, comparing or convincing an audience by means of logical formalizations or counter arguments [32], hence, its relation to the field of cognitive science and the *psychology of explanations* [7] is clear. The validity of whether something has been understood or not is hard to asses objectively. However, there are accounts in which this gauging is done easier. The reduction of complexity of a given model can be seen as explanations about the internals of such a model. This reduction should be considered as an XAI approach, measured by the amount of complexity that has been reduced from the model as a proxy for how much explanation has been done. Contrarily, the amount of interpretability that has been gained for these types of approaches is difficult to assess. For example: a model simplification process can be measured by counting the amount of structural elements that have been removed from the model (usually done in DNNs). However, the amount of interpretability gained by means of visualization techniques when explaining a model is hard to assess. This brings up the open challenge of conforming general metrics to evaluate these issues. This challenge is addressed further in Section 2.4.2.

Explainability, as a difference with interpretability, is closely linked to post-hoc technique since it covers the techniques to transform a non-interpretable model into an explainable one. The rest of the work will focus on explainability as the main objective, given the broader scope of the concept. The interpretability of a model comes from the design of the model itself, while in any case, a model can be the target to be explained. Trying to convey all the aforementioned, explainable AI can be defined as:

> *Given an audience, an **explainable** Artificial Intelligence is one that produces details or reasons to make its functioning clear or easy to understand.*

This definition presents the first contribution of this thesis and implicitly assumes that the ease of understanding and clarity targeted by XAI techniques for the model

at hand reverts on different application purposes, such as a better trustworthiness of the model's output by the audience.

2.1.3 *Why?*

The introduction has presented explainability as one of the main barriers AI is facing regarding its practical implementation nowadays. The inability to back state-of-the-art machine learning algorithms with explanations that help understand why they do as well as they do, is a problem that hinders in two main causes illustrated in Figure 2.1

The gap between the research community and business sectors conforms the first cause impeding the full penetration of the newest ML models in sectors that traditionally have lagged behind the digital transformation of their processes. Banking, finances, security and health are some of the fields that encounter this problem hand in hand with their strict regulations and the fear of implementing techniques that may put at risk their assets.

The second cause revolves around knowledge. AI has empowered human cognitive abilities by helping infer relations otherwise impossible. Every field dealing with huge amounts of reliable data has largely benefited from the adoption of AI and ML techniques. However, performance and results are starting to be considered as the only interests when staring to latest research studies. There are cases in which this might be fair, although it is far from the real interests for science and society. The search for understanding is what opens the gate to further model improvement and its practical utility.

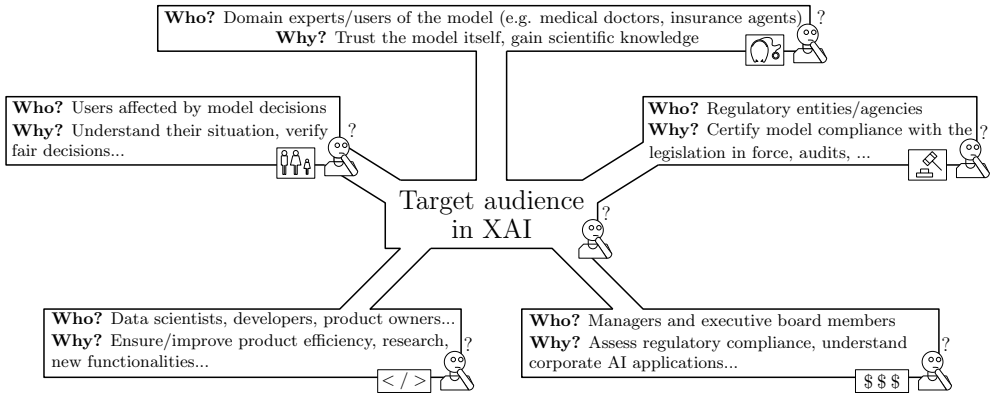


Figure 2.1: Diagram showing the different purposes of explainability in ML models sought by different audience profiles. Two goals occur to prevail across them: need for model understanding, and regulatory compliance. Image partly inspired by the one presented in [33], used with permission from IBM.

The following section analyses the motivation and goals behind the search for explainable AI models.

Table 2.1: Goals pursued in the reviewed literature toward reaching explainability, and their main target audience.

XAI Goal	**Main target audience** (Fig. 2)	**References**
Trustworthiness	Domain experts, users of the model affected by decisions	[5, 17, 28, 34–39]
Causality	Domain experts, managers and executive board members, regulatory entities/agencies	[37, 40–45]
Transferability	Domain experts, data scientists	[5, 25, 30, 34, 39–41, 46–88]
Informativeness	All	[5, 25, 29, 30, 34, 36, 37, 39, 40, 43, 46–49, 52–63, 66–69, 71–82, 89–157]
Confidence	Domain experts, developers, managers, regulatory entities/agencies	[5, 37, 47, 49, 51, 57, 64, 75, 91, 92, 99, 111, 120, 122, 158]
Fairness	Users affected by model decisions, regulatory entities/agencies	[5, 28, 37, 47, 50, 102–104, 123, 124, 131, 159–161]
Accessibility	Product owners, managers, users affected by model decisions	[25, 30, 34, 39, 48, 53, 56, 58, 65, 70–74, 77–79, 89, 96, 97, 106, 108, 110, 111, 114–118, 127, 132]
Interactivity	Domain experts, users affected by model decisions	[39, 53, 62, 68, 70, 77, 89, 127]
Privacy awareness	Users affected by model decisions, regulatory entities/agencies	[92]

2.1.4 *What For?*

Many different goals have been described when motivating the search for an explainable model. Most papers disagree in these goals, and which of them should an explainable model compel. However, all these different goals might help discriminate the purpose for which a given exercise of Machine Learning explainability is performed. Unfortunately, scarce contributions have attempted to define such foals from a conceptual perspective [5, 16, 28, 48]. To settle a first classification criteria for the full suit of papers covered in this background, the encountered definitions are hereby synthesized an enumerated:

- *Trustworthiness:* Trustworthiness is found by many as a principal aim when imagining an explainable Artificial Intelligence model [34, 162]. However, declaring a model explainable as per its capabilities of inducing trust might not be fully compliant with the requirement of model explainability. Trustworthiness might be considered as the confidence of whether a model will act as intended when facing a given problem. Although it should most certainly be a property of any explainable model, it does not imply that every trustworthy model can be considered explainable on its own, nor is trustworthiness a property easy

to quantify. Trust might be far from being the only purpose of an explainable model since the relation among the two, if agreed upon, is not reciprocal. Part of the reviewed papers mention the concept of trust when stating their purpose for achieving explainability. However, as seen in Table 2.1, they do not amount to a large share of the recent contributions related to XAI.

- *Causality:* another common goal for explainability is that of finding causality among data variables. Several authors argue that explainable models might ease the task of finding relationships that, should they occur, could be tested further for a stronger causal link between the involved variables [163, 164]. The inference of causal relationships from observational data is a field that has been broadly studied over time [165]. As widely acknowledged by the community working on this topic, causality requires a wide frame of prior knowledge to prove that observed effects are causal. A ML model only discovers correlations among the data it learns from, and therefore might not suffice for unveiling a cause-effect relationship. However, causation involves correlation, so an explainable ML model could validate the results provided by causality inference techniques, or provide a first intuition of possible causal relationships within the available data. Again, Table 2.1 reveals that causality is not among the most important goals if we attend to the amount of papers that state it explicitly as their goal.

- *Transferability:* models are always bounded by constraints that should allow for their seamless transferability. This is the main reason why a training-testing approach is used when dealing with ML problems [166, 167]. Explainability is also an advocate for transferability, since it may ease the task of elucidating the boundaries that might affect a model, allowing for a better understanding and implementation. Similarly, the mere understanding of the inner relations taking place within a model facilitates the ability of a user to reuse this knowledge in another problem. There are cases in which the lack of a proper understanding of the model might drive the user toward incorrect assumptions and fatal consequences [46, 168]. Transferability should also fall between the resulting properties of an explainable model, but again, not every transferable model should be considered as explainable. As observed in Table 2.1, the amount of papers stating that the ability of rendering a model explainable is to better understand the concepts needed to reuse it or to improve its performance is the second most used reason for pursuing model explainability.

- *Informativeness:* ML models are used with the ultimate intention of supporting decision making [95]. However, it should not be forgotten that the problem being solved by the model is not equal to that being faced by its human counterpart. Hence, a great deal of information is needed in order to be able to relate the user's decision to the solution given by the model, and to avoid falling in misconception pitfalls. For this purpose, explainable ML models should give information about the problem being tackled. Most of the reasons found among the papers reviewed is that of extracting information about the inner relations of a model. Almost all rule extraction techniques substantiate their approach on

the search for a simpler understanding of what the model internally does, stating that the knowledge (information) can be expressed in these simpler proxies that they consider explaining the antecedent. This is the most used argument found among the reviewed papers to back up what they expect from reaching explainable models.

- *Confidence:* as a generalization of robustness and stability, confidence should always be assessed on a model in which reliability is expected. The methods to maintain confidence under control are different depending on the model. As stated in [169–171], stability is a must-have when drawing interpretations from a certain model. Trustworthy interpretations should not be produced by models that are not stable. Hence, an explainable model should contain information about the confidence of its working regime.

- *Fairness:* from a social standpoint, explainability can be considered as the capacity to reach and guarantee fairness in ML models. In a certain literature strand, an explainable ML model suggests a clear visualization of the relations affecting a result, allowing for a fairness or ethical analysis of the model at hand [103, 172]. Likewise, a related objective of XAI is highlighting bias in the data a model was exposed to [173, 174]. The support of algorithms and models is growing fast in fields that involve human lives, hence explainability should be considered as a bridge to avoid the unfair or unethical use of algorithm's outputs.

- *Accessibility:* a minor subset of the reviewed contributions argues for explainability as the property that allows end users to get more involved in the process of improving and developing a certain ML model [39, 89] . It seems clear that explainable models will ease the burden felt by non-technical or non-expert users when having to deal with algorithms that seem incomprehensible at first sight. This concept is expressed as the third most considered goal among the surveyed literature.

- *Interactivity:* some contributions [53, 62] include the ability of a model to be interactive with the user as one of the goals targeted by an explainable ML model. Once again, this goal is related to fields in which the end users are of great importance, and their ability to tweak and interact with the models is what ensures success.

- *Privacy awareness:* almost forgotten in the reviewed literature, one of the byproducts enabled by explainability in ML models is its ability to assess privacy. ML models may have complex representations of their learned patterns. Not being able to understand what has been captured by the model [4] and stored in its internal representation may entail a privacy breach. Contrarily, the ability to explain the inner relations of a trained model by non-authorized third parties may also compromise the differential privacy of the data origin. Due to its criticality in sectors where XAI is foreseen to play a crucial role, confidentiality and privacy issues will be covered further in Subsection 2.4.4.

2.1.5 *How?*

The literature makes a clear distinction among models that are interpretable by design, and those that can be explained by means of external XAI techniques. This duality could also be regarded as the difference between interpretable models and model interpretability techniques; a more widely accepted classification is that of *transparent* models and post-hoc explainability. This same duality also appears in the paper presented in [21]. The distinction its authors make refers to the methods to solve the transparent box design problem against the problem of explaining the *black-box* problem. This work, further extends the distinction made among transparent models including the different levels of transparency considered.

Within transparency, three levels are contemplated: algorithmic transparency, decomposability and simulatability[1]. Among post-hoc techniques we may distinguish: *text explanations*, *visualizations*, *local explanations*, *explanations by example*, *explanations by simplification* and *feature relevance*. In this context, there is a broader distinction proposed by [28] discerning between 1) opaque systems, where the mappings from input to output are invisible to the user; 2) interpretable systems, in which users can mathematically analyze the mappings; and 3) comprehensible systems, in which the models should output symbols or rules along with their specific output to aid in the understanding process of the rationale behind the mappings being made. This last classification criterion could be considered included within the one proposed earlier, hence this paper will attempt at following the more specific one.

2.1.5.1 *Levels of Transparency in Machine Learning Models*

Transparent models convey some degree of interpretability by themselves. Models belonging to this category can be also approached in terms of the domain in which they are interpretable, namely, algorithmic transparency, decomposability and simulatability. As we elaborate next in connection to Figure 2.2, each of these classes contains its predecessors, e.g. a *simulatable* model is at the same time a model that is decomposable and algorithmically transparent:

- *Simulatability* denotes the ability of a model of being simulated or thought about strictly by a human, hence complexity takes a dominant place in this class. This being said, simple but extensive (i.e., with *too large* amount of rules) rule based systems fall out of this characteristic, whereas a single perceptron neural network falls within. This aspect aligns with the claim that sparse linear models are more interpretable than dense ones [175], and that an interpretable model is one that can be easily presented to a human by means of text and *visualizations* [34]. Again, endowing a decomposable model with simulatability requires that the model has to be self-contained enough for a human to think and reason about it as a whole.

- *Decomposability* stands for the ability to explain each of the parts of a model (input, parameter and calculation). It can be considered as intelligibility as stated

1 The alternative term *simulability* is also used in the literature to refer to the capacity of a system or process to be simulated. However, we note that this term does not appear in current English dictionaries.

in [176]. This characteristic might empower the ability to understand, interpret or explain the behavior of a model. However, as occurs with algorithmic transparency, not every model can fulfill this property. Decomposability requires every input to be readily interpretable (e.g. cumbersome features will not fit the premise). The added constraint for an algorithmically transparent model to become decomposable is that every part of the model must be understandable by a human without the need for additional tools.

- *Algorithmic Transparency* can be seen in different ways. It deals with the ability of the user to understand the process followed by the model to produce any given output from its input data. Put it differently, a linear model is deemed transparent because its error surface can be understood and reasoned about, allowing the user to understand how the model will act in every situation it may face [167]. Contrarily, it is not possible to understand it in deep architectures as the loss landscape might be opaque. [177, 178] Hence, given it cannot be fully observed, the solution has to be approximated through heuristic optimization (e.g. through stochastic gradient descent). The main constraint for algorithmically transparent models is that the model has to be fully explorable by means of mathematical analysis and methods.

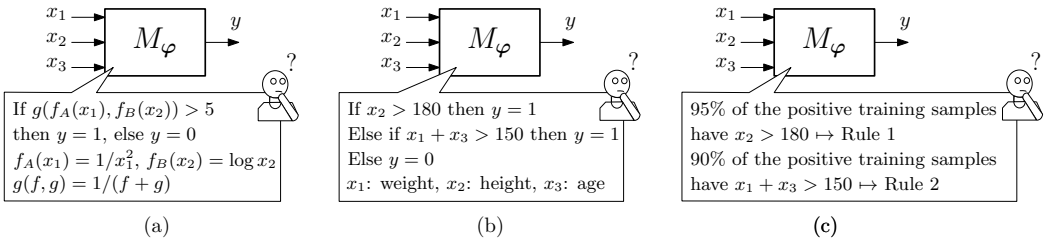


Figure 2.2: Conceptual diagram exemplifying the different levels of transparency characterizing a ML model M_φ, with φ denoting the parameter set of the model at hand: (a) simulatability; (b) decomposability; (c) algorithmic transparency. Without loss of generality, the example focuses on the ML model as the explanation target. However, other targets for explainability may include a given example, the output classes or the dataset itself.

2.1.5.2 *Post-hoc Explainability Techniques for Machine Learning Models*

Post-hoc explainability targets models that are not readily interpretable by design by resorting to diverse means to enhance their interpretability, such as *text explanations*, *visual explanations*, *local explanations*, *explanations by example*, *explanations by simplification* and *feature relevance explanations*. Each of these techniques covers one of the most common ways humans explain systems and processes by themselves.

Further along this river, actual techniques, or better put, actual group of techniques are specified to ease the future work of any researcher that intends to look up for an specific technique that suits its knowledge. Not ending there, the classification also includes the type of data in which the techniques has been applied. Note that many techniques might be suitable for many different types of data, although the categorization only considers the type used by the authors that proposed such techniques.

Overall, post-hoc explainability techniques are divided first by the intention of the author (explanation technique e.g. Explanation by simplification), then, by the method utilized (actual technique e.g. sensitivity analysis) and finally by the type of data in which it was applied (e.g. images).

- *Text explanations* deal with the problem of bringing explainability for a model by means of learning to generate *text explanations* that help explaining the results from the model [174]. *Text explanations* also include every model that generate symbols as proxies for the model's functioning. These symbols may portrait the rationale of the algorithm by means of a semantic mapping from model to symbols.

- *Visual explanation* techniques for post-hoc explainability aim at visualizing the model's behavior. Many of the visualization methods existing in the literature come along with dimensionality reduction techniques that allow for human interpretable visualizations. Visualizations are considered as the most fruitful approaches to introduce complex interactions within the variables involved in the model to users not acquainted to ML modeling. *Visual explanations* are many times found coupled with other techniques to improve their understanding.

- *Local explanations* tackle explainability by segmenting the solution space and giving explanations to less complex solution subspaces that are relevant for the whole model. These explanations can be formed by means of techniques with the differentiating property that these only explain part of the whole system's functioning.

- *Explanations by example* consider the extraction of data examples that relate to the result generated by a certain model, enabling to get a better understanding of the model itself. Similarly to how humans behave when attempting to explain a given process, *explanations by example* are mainly centered in extracting representative examples that grasp the inner relationships and correlations found by the model being analyzed.

- *Explanations by simplification* collectively denote those techniques in which a whole new system is rebuilt based on the trained model to be explained. This new, simplified model usually attempts at optimizing its resemblance to its antecedent functioning, while reducing its complexity, and keeping a similar performance score. An interesting byproduct of this family of post-hoc techniques is that the simplified model is, in general, easier to be implemented due to its reduced complexity with respect to the model it represents.

- Finally, *feature relevance explanation* methods for post-hoc explainability clarify the inner functioning of a model by computing a relevance score for its managed variables. These scores quantify the affection (sensitivity) a feature has upon the output of the model. A comparison of the scores among different variables unveils the importance granted by the model to each of such variables when producing its output. *Feature relevance* methods can be thought to be an indirect method to explain a model.

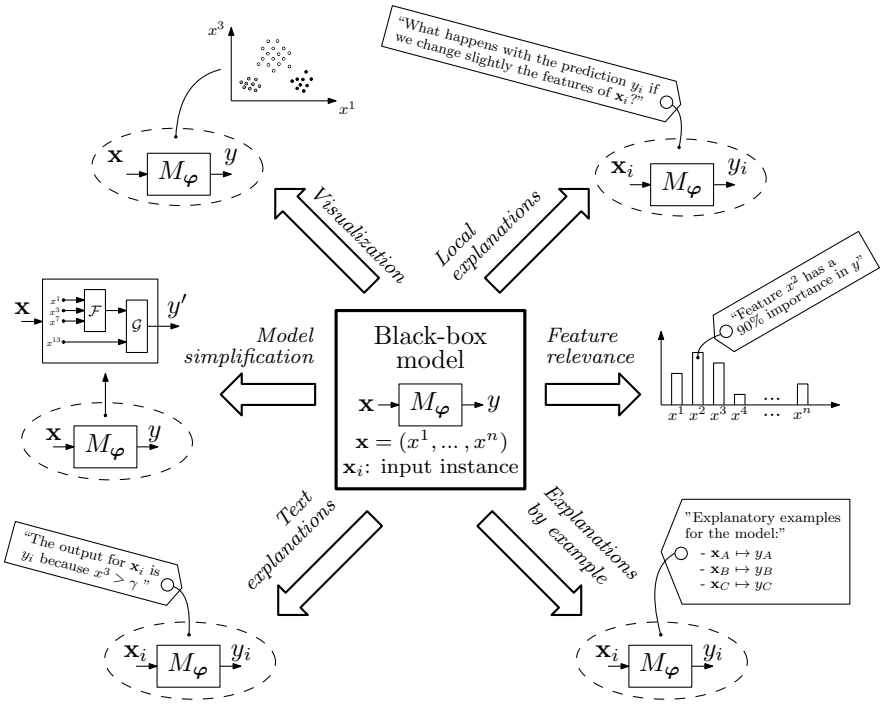


Figure 2.3: Conceptual diagram showing the different post-hoc explainability approaches available for a ML model M_φ.

The above classification (portrayed graphically in Figure 2.3) will be used when reviewing specific/agnostic XAI techniques for ML models in the following sections (Table 2.2). For each ML model, a distinction of the propositions to each of these categories is presented in order to pose an overall image of the field's trends.

Table 2.2: Overall picture of the classification of ML models attending to their level of explainability.

Model	Transparent ML Models			Post-hoc analysis
	Simulatability	**Decomposability**	**Algorithmic Transparency**	
Linear/Logistic Regression	Predictors are human readable and interactions among them are kept to a minimum	Variables are still readable, but the number of interactions and predictors involved in them have grown to force decomposition	Variables and interactions are too complex to be analyzed without mathematical tools	Not needed
Decision Trees	A human can simulate and obtain the prediction of a decision tree on his/her own, without requiring any mathematical background	The model comprises rules that do not alter data whatsoever, and preserves their readability	Human-readable rules that explain the knowledge learned from data and allows for a direct understanding of the prediction process	Not needed
K-Nearest Neighbors	The complexity of the model (number of variables, their understandability and the similarity measure under use) matches human naive capabilities for simulation	The amount of variables is too high and/or the similarity measure is too complex to be able to simulate the model completely, but the similarity measure and the set of variables can be decomposed and analyzed separately	The similarity measure cannot be decomposed and/or the number of variables is so high that the user has to rely on mathematical and statistical tools to analyze the model	Not needed
Rule Based Learners	Variables included in rules are readable, and the size of the rule set is manageable by a human user without external help	The size of the rule set becomes too large to be analyzed without decomposing it into small rule chunks	Rules have become so complicated (and the rule set size has grown so much) that mathematical tools are needed for inspecting the model behaviour	Not needed
General Additive Models	Variables and the interaction among them as per the smooth functions involved in the model must be constrained within human capabilities for understanding	Interactions become too complex to be simulated, so decomposition techniques are required for analyzing the model	Due to their complexity, variables and interactions cannot be analyzed without the application of mathematical and statistical tools	Not needed
Bayesian Models	Statistical relationships modeled among variables and the variables themselves should be directly understandable by the target audience	Statistical relationships involve so many variables that they must be decomposed in marginals so as to ease their analysis	Statistical relationships cannot be interpreted even if already decomposed, and predictors are so complex that model can be only analyzed with mathematical tools	Not needed
Tree Ensembles				Needed: Usually *Model simplification* or *Feature relevance* techniques
Support Vector Machines				Needed: Usually *Model simplification* or *Local explanations* techniques
Multi–layer Neural Network				Needed: Usually *Model simplification*, *Feature relevance* or *Visualization* techniques
Convolutional Neural Network				Needed: Usually *Feature relevance* or *Visualization* techniques
Recurrent Neural Network				Needed: Usually *Feature relevance* techniques

2.2 TRANSPARENT MACHINE LEARNING MODELS

The previous section introduced the concept of *transparent* models. A model is considered to be transparent if by itself it is understandable. The models surveyed in this section are a suit of transparent models that can fall in one or all of the levels of model transparency described previously (namely, simulatability, decomposability and algorithmic transparency). In what follows we provide reasons for this statement, with graphical support given in Figure 2.4.

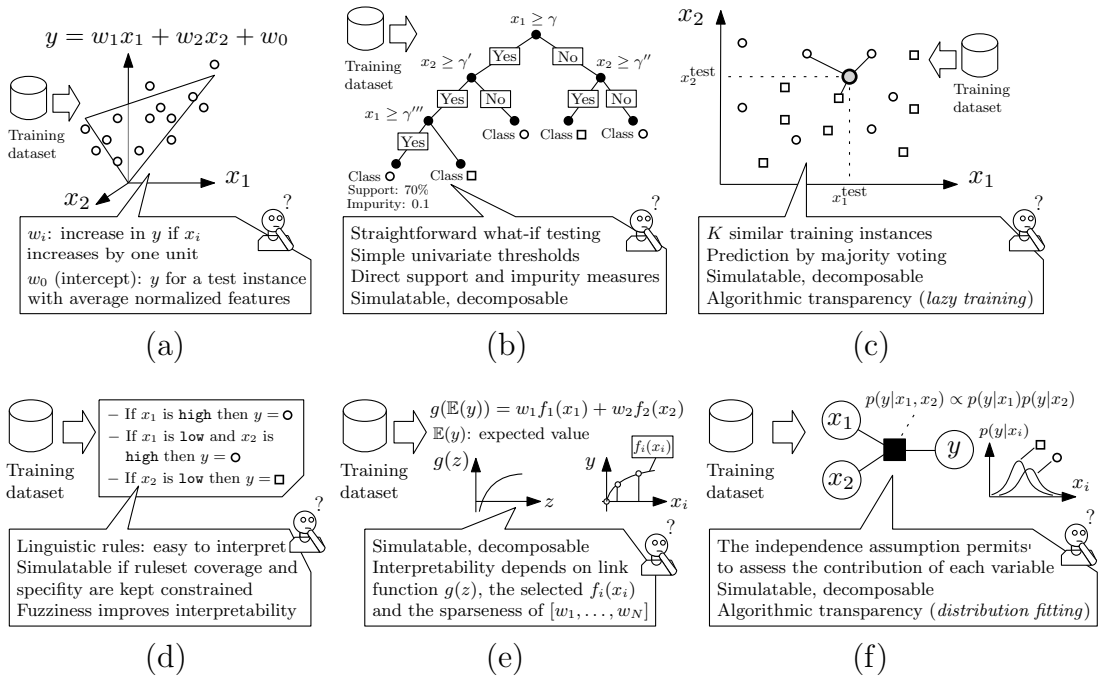


Figure 2.4: Graphical illustration of the levels of transparency of different ML models considered in this overview: (a) Linear regression; (b) Decision trees; (c) K-Nearest Neighbors; (d) Rule-based Learners; (e) Generalized Additive Models; (f) Bayesian Models.

2.2.1 *Logistic/Linear Regression*

Logistic Regression (LR) is a classification model trained to predict a dependent variable (category) that is dichotomous (binary). However, when the dependent variable is continuous, linear regression would be its homonym. This model takes the assumption of linear dependence between the predictors and the predicted variables. This specific reason (stiffness of the model) is the one that maintains the model under the umbrella of transparent methods. However, as stated in Section 2, explainability is linked to a certain audience, which makes a model fall under both categories depending who is to interpret it. This way, logistic and linear regression, although clearly meeting the characteristics of transparent models (algorithmic transparency, decomposability and simulatability), may also demand post-hoc explainability techniques (mainly, visualization), particularly when the model is to be explained to non-expert audiences.

The usage of this model has been largely applied within Social Sciences for quite a long time, which has pushed researchers to create ways of explaining the results of the models to non-expert users. Most authors agree on the different techniques used to analyze and express the soundness of LR [179–182], including the overall model evaluation, statistical tests of individual predictors, goodness-of-fit statistics and validation of the predicted probabilities. The overall model evaluation shows the improvement of the applied model over a baseline, showing if it is in fact improving the model without predictions. The statistical significance of single predictors is shown by calculating the Wald chi-square statistic. The goodness-of-fit statistics show

the quality of fitness of the model to the data and how significant this is. This can be achieved by resorting to different techniques e.g. the so-called Hosmer-Lemeshow (H-L) statistic. The validation of predicted probabilities involves testing whether the output of the model corresponds to what is shown by the data. These techniques show mathematical ways of representing the fitness of the model and its behavior.

Other techniques from other disciplines besides Statistics can be adopted for explaining these regression models. Visualization techniques are very powerful when presenting statistical conclusions to users not well-versed in statistics. For instance, the work in [183] shows that the usage of probabilities to communicate the results, implied that the users where able to estimate the outcomes correctly in 10% of the cases, as opposed to 46% of the cases when using natural frequencies. Although logistic regression is among the simplest classification models in supervised learning, there are concepts that must be taken care of.

In this line of reasoning, the authors of [184] unveil some concerns with the interpretations derived from LR. They first mention how dangerous it might be to interpret log odds ratios and odd ratios as substantive effects, since they also represent unobserved heterogeneity. Linked to this first concern, [184] also states that a comparison between these ratios across models with different variables might be problematic, since the unobserved heterogeneity is likely to vary, thereby invalidating the comparison. Finally they also mention that the comparison of these odds across different samples, groups and time is also risky, since the variation of the heterogeneity is not known across samples, groups and time points. This last paper serves the purpose of visualizing the problems a model's interpretation might entail, even when its construction is as simple as that of LR.

Also interesting is to note that, for a model such as logistic or linear regression to maintain decomposability and simulatability, its size must be limited, and the variables used must be understandable by their users. As stated in Section 2.1.5.1, if inputs to the model are highly engineered features that are complex or difficult to understand, the model at hand will be far from being *decomposable*. Similarly, if the model is so large that a human cannot think of the model as a whole, its simulatability will be put to question.

2.2.2 *Decision Trees*

Decision trees are another example of a model that can easily fulfill every constraint for transparency. Decision trees are hierarchical structures for decision making used to support regression and classification problems [135, 185]. In the simplest of their flavors, decision trees are *simulatable* models. However, their properties can render them *decomposable* or *algorithmically transparent*.

Decision trees have always lingered in between the different categories of transparent models. Their utilization has been closely linked to decision making contexts, being the reason why their complexity and understandability have always been considered a paramount matter. A proof of this relevance can be found in the upsurge of contributions to the literature dealing with decision tree simplification and generation [135, 185–187]. As noted above, although being capable of fitting every category within transparent models, the individual characteristics of decision trees can push

them toward the category of algorithmically transparent models. A *simulatable* decision tree is one that is manageable by a human user. This means its size is somewhat small and the amount of features and their meaning are easily understandable. An increment in size transforms the model into a *decomposable* one since its size impedes its full evaluation (simulation) by a human. Finally, further increasing its size and using complex feature relations will make the model *algorithmically transparent* loosing the previous characteristics.

Decision trees have long been used in decision support contexts due to their off-the-shelf transparency. Many applications of these models fall out of the fields of computation and AI (even information technologies), meaning that experts from other fields usually feel comfortable interpreting the outputs of these models [188–190]. However, their poor generalization properties in comparison with other models make this model family less interesting for their application to scenarios where a balance between predictive performance is a design driver of utmost importance. Tree ensembles aim at overcoming such a poor performance by aggregating the predictions performed by trees learned on different subsets of training data. Unfortunately, the combination of decision trees looses every transparent property, calling for the adoption of post-hoc explainability techniques as the ones reviewed later in the chapter.

2.2.3 *K Nearest Neighbors*

Another method that falls within transparent models is that of K-Nearest Neighbors (KNN), which deals with classification problems in a methodologically simple way: it predicts the class of a test sample by voting the classes of its K nearest neighbors (where the neighborhood relation is induced by a measure of distance between samples). When used in the context of regression problems, the voting is replaced by an aggregation (e.g. average) of the target values associated with the nearest neighbors.

In terms of model explainability, it is important to observe that predictions generated by KNN models rely on the notion of distance and similarity between examples, which can be tailored depending on the specific problem being tackled. Interestingly, this prediction approach resembles that of experience-based human decision making, which decides upon the result of past similar cases. There lies the rationale of why KNN has also been adopted widely in contexts in which model interpretability is a requirement [191–194]. Furthermore, aside from being simple to explain, the ability to inspect the reasons by which a new sample has been classified inside a group and to examine how these predictions evolve when the number of neighbors K is increased or decreased empowers the interaction between the users and the model.

One must keep in mind that as mentioned before, KNN's class of transparency depends on the features, the number of neighbors and the distance function used to measure the similarity between data instances. A very high K impedes a full simulation of the model performance by a human user. Similarly, the usage of complex features and/or distance functions would hinder the decomposability of the model, restricting its interpretability solely to the transparency of its algorithmic operations.

2.2.4 *Rule Based Learning*

Rule-based learning refers to every model that generates rules to characterize the data it is intended to learn from. Rules can take the form of simple conditional *if-then* rules or more complex combinations of simple rules to form their knowledge. Also connected to this general family of models, fuzzy rule based systems are designed for a broader scope of action, allowing for the definition of verbally formulated rules over imprecise domains. Fuzzy systems improve two main axes relevant for this thesis. First, they empower more understandable models since they operate in linguistic terms. Second, they perform better than classic rule systems in contexts with certain degrees of uncertainty. Rule based learners are clearly transparent models that have been often used to explain complex models by generating rules that explain their predictions [129, 130, 195, 196].

Rule learning approaches have been extensively used for knowledge representation in expert systems [197]. However, a central problem with rule generation approaches is the coverage (amount) and the specificity (length) of the rules generated. This problem relates directly to the intention for their use in the first place. When building a rule database, a typical design goal sought by the user is to be able to analyze and understand the model. The amount of rules in a model will clearly improve the performance of the model at the stake of compromising its intepretability. Similarly, the specificity of the rules plays also against interpretability, since a rule with a high number of antecedents an/or consequences might become difficult to interpret. In this same line of reasoning, these two features of a rule based learner play along with the classes of transparent models presented in Section 2. The greater the coverage or the specificity is, the closer the model will be to being just *algorithmically transparent*. Sometimes, the reason to transition from classical rules to fuzzy rules is to relax the constraints of rule sizes, since a greater range can be covered with less stress on interpretability.

Rule based learners are great models in terms of interpretability across fields. Their natural and seamless relation to human behaviour makes them very suitable to understand and explain other models. If a certain threshold of coverage is acquired, a rule wrapper can be thought to contain enough information about a model to explain its behavior to a non-expert user, without forfeiting the possibility of using the generated rules as an standalone prediction model.

2.2.5 *General Additive Models*

In statistics, a Generalized Additive Model (GAM) is a linear model in which the value of the variable to be predicted is given by the aggregation of a number of unknown smooth functions defined for the predictor variables. The purpose of such model is to infer the smooth functions whose aggregate composition approximates the predicted variable. This structure is easily interpretable, since it allows the user to verify the importance of each variable, namely, how it affects (through its corresponding function) the predicted output.

Similarly to every other transparent model, the literature is replete with case studies where GAMs are in use, specially in fields related to risk assessment. When com-

pared to other models, these are understandable enough to make users feel confident on using them for practical applications in finance [198–200], environmental studies [201], geology [202], healthcare [46], biology [203, 204] and energy [205]. Most of these contributions use visualization methods to further ease the interpretation of the model. GAMs might be also considered as *simulatable* and *decomposable* models if the properties mentioned in its definitions are fulfilled, but to an extent that depends roughly on eventual modifications to the baseline GAM model, such as the introduction of link functions to relate the aggregation with the predicted output, or the consideration of interactions between predictors.

All in all, applications of GAMs like the ones exemplified above share one common factor: understandability. The main driver for conducting these studies with GAMs is to understand the underlying relationships that build up the cases for scrutiny. In those cases the research goal is not accuracy for its own sake, but rather the need for understanding the problem behind and the relationship underneath the variables involved in data. This is why GAMs have been accepted in certain communities as their *de facto* modeling choice, despite their acknowledged misperforming behavior when compared to more complex counterparts.

2.2.6 *Bayesian Models*

A Bayesian model usually takes the form of a probabilistic directed acyclic graphical model whose links represent the conditional dependencies between a set of variables. For example, a Bayesian network could represent the probabilistic relationships between diseases and symptoms. Given symptoms, the network can be used to compute the probabilities of the presence of various diseases. Similar to GAMs, these models also convey a clear representation of the relationships between features and the target, which in this case are given explicitly by the connections linking variables to each other.

Once again, Bayesian models fall below the ceiling of Transparent models. Its categorization leaves it under *simulatable*, *decomposable* and *algorithmically transparent*. However, it is worth noting that under certain circumstances (overly complex or cumbersome variables), a model may loose these first two properties. Bayesian models have been shown to lead to great insights in assorted applications such as cognitive modeling [206, 207], fishery [201, 208], gaming [209], climate [210], econometrics [211] and robotics [212]. Furthermore, they have also been utilized to explain other models, such as averaging tree ensembles [213].

2.3 POST-HOC EXPLAINABILITY TECHNIQUES FOR MACHINE LEARNING MODELS: TAXONOMY, SHALLOW MODELS AND DEEP LEARNING

When ML models do not meet any of the criteria imposed to declare them transparent, a separate method must be devised and applied to the model to explain its decisions. This is the purpose of post-hoc explainability techniques (also referred to as post-modeling explainability), which aim at communicating understandable information about how an already developed model produces its predictions for any given input. In this section we categorize and review different algorithmic approaches for

post-hoc explainability, discriminating among: 1) those that are designed for their application to ML models of any kind; and 2) those that are designed for a specific ML model and thus, can not be directly extrapolated to any other learner. We now elaborate on the trends identified around post-hoc explainability for different ML models, which are illustrated in Figure 2.5 in the form of hierarchical bibliographic categories and summarized next:

- Model-agnostic techniques for post-hoc explainability (Subsection 2.3.1), which can be applied seamlessly to any ML model disregarding its inner processing or internal representations.
- Post-hoc explainability that are tailored or specifically designed to explain certain ML models. We divide our literature analysis into two main branches: contributions dealing with post-hoc explainability of *shallow* ML models, which collectively refers to all ML models that do not hinge on layered structures of neural processing units (Subsection 2.3.2); and techniques devised for *deep* learning models, which correspondingly denote the family of neural networks and related variants, such as convolutional neural networks, recurrent neural networks (Subsection 2.3.5) and hybrid schemes encompassing deep neural networks and transparent models. For each model we perform a thorough review of the latest post-hoc methods proposed by the research community, along with an identification of trends followed by such contributions.
- We end our literature analysis with Subsection 2.3.7, where we present a second taxonomy that complements the more general one in Figure 2.5 by classifying contributions dealing with the post-hoc explanation of Deep Learning models. To this end we focus on particular aspects related to this family of *black-box* ML methods, and expose how they link to the classification criteria used in the first taxonomy.

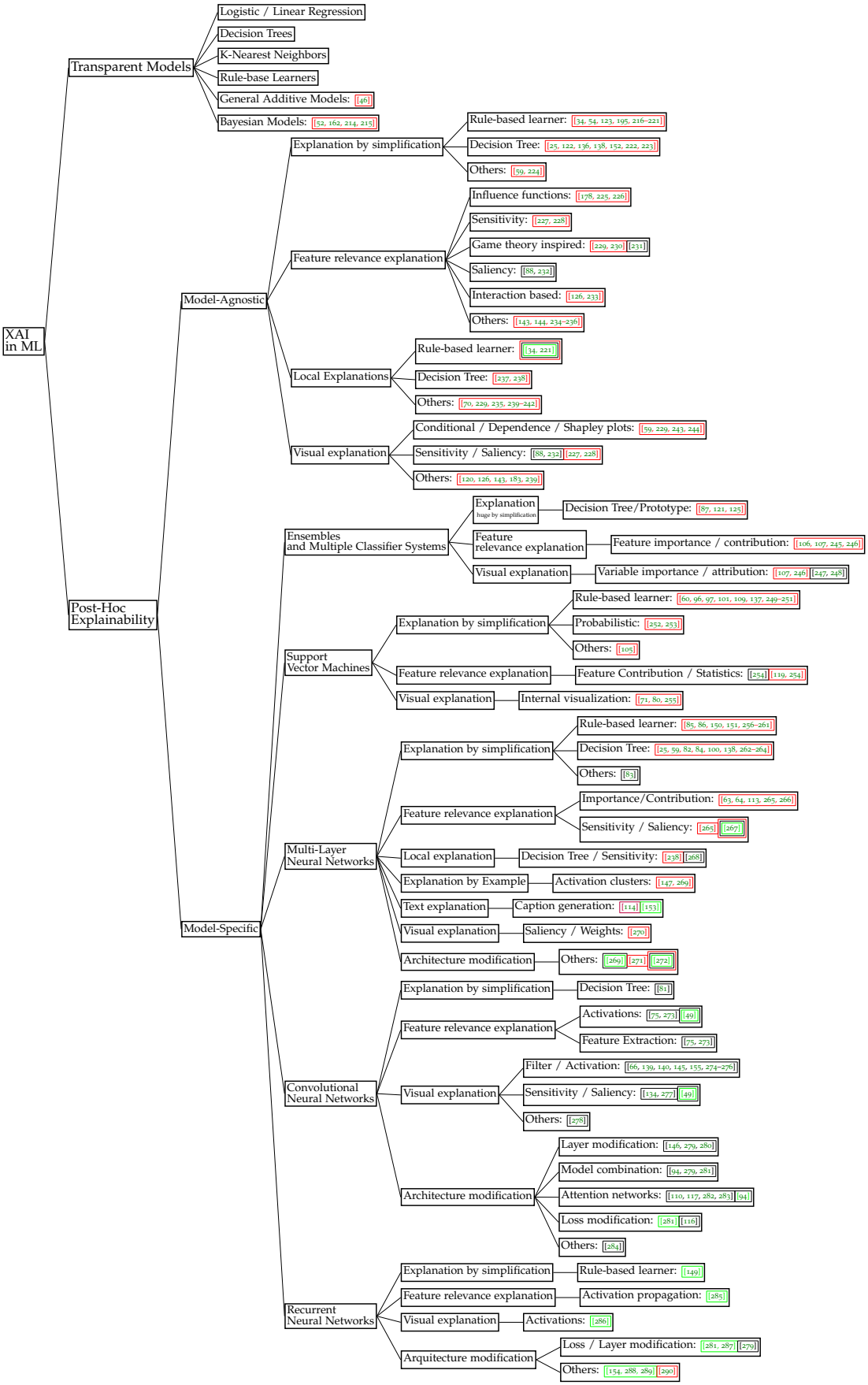


Figure 2.5: Taxonomy of the reviewed literature and trends identified for explainability techniques related to different ML models. References boxed in black, green and red correspond to XAI techniques using image, text or tabular data, respectively. In order to build this taxonomy, the literature has been analyzed in depth to discriminate whether a post-hoc technique can be seamlessly applied to any ML model, even if, e.g., explicitly mentions *Deep Learning* in its title and/or abstract.

2.3.1 *Model-agnostic Techniques for Post-hoc Explainability*

Model-agnostic techniques for post-hoc explainability are designed to be plugged to any model with the intent of extracting some information from its prediction procedure. Sometimes, simplification techniques are used to generate proxies that mimic their antecedents with the purpose of having something tractable and of reduced complexity. Other times, the intent focuses on extracting knowledge directly from the models or simply visualizing them to ease the interpretation of their behavior. Following the taxonomy introduced in Section 2.1, model-agnostic techniques may rely on *model simplification*, *feature relevance* estimation and *visualization* techniques:

- *Explanation by simplification*. They are arguably the broadest technique under the category of model agnostic post-hoc methods. *Local explanations* are also present within this category, since sometimes, simplified models are only representative of certain sections of a solution space. Almost all techniques taking this path for *model simplification* are based on rule extraction techniques. Among the most known contributions to this approach we encounter the technique of Local Interpretable Model-Agnostic Explanations (LIME) [34] and all its variations [219, 221]. LIME builds locally linear models around the predictions of an opaque model to explain it. These contributions fall under explanations by simplification as well as under *local explanations*. Besides LIME and related flavors, another approach to rule extraction is G-REX [217]. Although it was not originally intended for extracting rules from opaque models, the generic proposition of G-REX has been extended to also account for model explainability purposes [195, 216]. In line with rule extraction methods, the work in [220] presents a novel approach to learn rules in CNF (Conjunctive Normal Form) or DNF (Disjunctive Normal Form) to bridge from a complex model to a human-interpretable model. Another contribution that falls off the same branch is that in [223], where the authors formulate *model simplification* as a model extraction process by approximating a transparent model to the complex one. Simplification is approached from a different perspective in [123], where an approach to distill and audit black box models is presented. In it, two main ideas are exposed: a method for model distillation and comparison to audit *black-box* risk scoring models; and an statistical test to check if the auditing data is missing key features it was trained with. The popularity of *model simplification* is evident, given it temporally coincides with the most recent literature on XAI, including techniques such as LIME or G-REX. This symptomatically reveals that this post-hoc explainability approach is envisaged to continue playing a central role on XAI.

- *Feature relevance explanation* techniques aim to describe the functioning of an opaque model by ranking or measuring the influence, relevance or importance each feature has in the prediction output by the model to be explained. An amalgam of propositions are found within this category, each resorting to different algorithmic approaches with the same targeted goal. One fruitful contribution to this path is that of [229] called SHAP (SHapley Additive exPlanations). Its authors presented a method to calculate an additive feature importance score for

each particular prediction with a set of desirable properties (local accuracy, *missingness* and consistency) that its antecedents lacked. Another approach to tackle the contribution of each feature to predictions has been coalitional Game Theory [230] and local gradients [239]. Similarly, by means of local gradients [235] test the changes needed in each feature to produce a change in the output of the model. In [233] the authors analyze the relations and dependencies found in the model by grouping features, that combined, bring insights about the data. The work in [178] presents a broad variety of measures to tackle the quantification of the degree of influence of inputs on outputs of systems. Their QII (Quantitative Input Influence) measures account for correlated inputs while measuring influence. In contrast, in [227] the authors build upon the existing SA (Sensitivity Analysis) to construct a Global SA which extends the applicability of the existing methods. In [232] a real-time image saliency method is proposed, which is applicable to differentiable image classifiers. The study in [126] presents the so-called Automatic STRucture IDentification method (ASTRID) to inspect which attributes are exploited by a classifier to generate a prediction. This method finds the largest subset of features such that the accuracy of a classifier trained with this subset of features cannot be distinguished in terms of accuracy from a classifier built on the original feature set. In [226] the authors use influence functions to trace a model's prediction back to the training data, by only requiring an oracle version of the model with access to gradients and Hessian-vector products. Heuristics for creating counterfactual examples by modifying the input of the model have been also found to contribute to its explainability [241, 242]. Compared to those attempting explanations by simplification, a similar amount of publications were found tackling explainability by means of *feature relevance* techniques. Many of the contributions date from 2017 and some from 2018, implying that as with *model simplification* techniques, *feature relevance* has also become a vibrant subject study in the current XAI landscape.

- *Visual explanation* techniques are a vehicle to achieve model-agnostic explanations. Representative works in this area can be found in [227], which present a portfolio of visualization techniques to help in the explanation of a *black-box* ML model built upon the set of extended techniques mentioned earlier (Global Sensitivity Analysis). Another set of visualization techniques is presented in [228]. The authors present three novel SA methods (data based SA, Monte-Carlo SA, cluster-based SA) and one novel input importance measure (Average Absolute Deviation). Finally, [243] presents ICE (Individual Conditional Expectation) plots as a tool for visualizing the model estimated by any supervised learning algorithm. Visual explanations are less common in the field of model-agnostic techniques for post-hoc explainability. Since the design of these methods must ensure that they can be seamlessly applied to any ML model disregarding its inner structure, creating *visualizations* from just inputs and outputs from an opaque model is a complex task. This is why almost all visualization methods falling in this category work along with *feature relevance* techniques, which provide the information that is eventually displayed to the end user.

Several trends emerge from our literature analysis. To begin with, rule extraction techniques prevail in model-agnostic contributions under the umbrella of post-hoc explainability. This could have been intuitively expected if we bear in mind the wide use of rule based learning as explainability wrappers anticipated in Section 2.2.4, and the complexity imposed by not being able to *get into* the model itself. Similarly, another large group of contributions deals with *feature relevance*. Lately these techniques are gathering much attention by the community when dealing with DL models, with hybrid approaches that utilize particular aspects of this class of models and therefore, compromise the independence of the *feature relevance* method on the model being explained. Finally, visualization techniques propose interesting ways for visualizing the output of *feature relevance* techniques to ease the task of model's interpretation. By contrast, visualization techniques for other aspects of the trained model (e.g. its structure, operations, etc) are tightly linked to the specific model to be explained.

2.3.2 *Post-hoc Explainability in Shallow ML Models*

Shallow ML covers a diversity of supervised learning models. Within these models, there are strictly interpretable (transparent) approaches (e.g. KNN and Decision Trees, already discussed in Section 2.2). However, other shallow ML models rely on more sophisticated learning algorithms that require additional layers of explanation. Given their prominence and notable performance in predictive tasks, this section concentrates on two popular shallow ML models (tree ensembles and Support Vector Machines, SVMs) that require the adoption of post-hoc explainability techniques for explaining their decisions.

2.3.3 *Tree Ensembles, Random Forests and Multiple Classifier Systems*

Tree ensembles are arguably among the most accurate ML models in use nowadays. Their advent came as an efficient means to improve the generalization capability of single decision trees, which are usually prone to overfitting. To circumvent this issue, tree ensembles combine different trees to obtain an aggregated prediction/regression. While it results to be effective against overfitting, the combination of models makes the interpretation of the overall ensemble more complex than each of its compounding tree learners, forcing the user to draw from post-hoc explainability techniques. For tree ensembles, techniques found in the literature are explanation by simplification and *feature relevance* techniques; we next examine recent advances in these techniques.

To begin with, many contributions have been presented to simplify tree ensembles while maintaining part of the accuracy accounted for the added complexity. The author from [122] poses the idea of training a single albeit less complex model from a set of random samples from the data (ideally following the real data distribution) labeled by the ensemble model. Another approach for simplification is that in [121], in which authors create a Simplified Tree Ensemble Learner (STEL). Likewise, [125] presents the usage of two models (simple and complex) being the former the one in charge of interpretation and the latter of prediction by means of Expectation-Maximization and Kullback-Leibler divergence. As opposed to what was seen in model-agnostic tech-

niques, not that many techniques to board explainability in tree ensembles come by means of *model simplification*. It derives from this that either the proposed techniques are good enough, or model-agnostic techniques do cover the scope of simplification already.

Following simplification procedures, *feature relevance* techniques are also used in the field of tree ensembles. Breiman [291] was the first to analyze the variable importance within Random Forests. His method is based on measuring MDA (Mean Decrease Accuracy) or MIE (Mean Increase Error) of the forest when a certain variable is randomly permuted in the out-of-bag samples. Following this contribution [246] shows, in an real setting, how the usage of variable importance reflects the underlying relationships of a complex system modeled by a Random Forest. Finally, a crosswise technique among post-hoc explainability, [245] proposes a framework that poses recommendations that, if taken, would convert an example from one class to another. This idea attempts to disentangle the variables importance in a way that is further descriptive. In the article, the authors show how these methods can be used to elevate recommendations to improve malicious online ads to make them rank higher in paying rates.

Similar to the trend shown in model-agnostic techniques, for tree ensembles again, *simplification* and *feature relevance* techniques seem to be the most used schemes. However, contrarily to what was observed before, most papers date back from 2017 and place their focus mostly on bagging ensembles. When shifting the focus towards other ensemble strategies, scarce activity has been recently noted around the explainability of boosting and stacking classifiers. Among the latter, it is worth highlighting the connection between the reason why a compounding learner of the ensemble produces an specific prediction on a given data, and its contribution to the output of the ensemble. The so-called Stacking With Auxiliary Features (SWAF) approach proposed in [247] points in this direction by harnessing and integrating explanations in stacking ensembles to improve their generalization. This strategy allows not only relying on the output of the compounding learners, but also on the origin of that output and its consensus across the entire ensemble. Other interesting studies on the explainability of ensemble techniques include model-agnostic schemes such as DeepSHAP [231], put into practice with stacking ensembles and multiple classifier systems in addition to Deep Learning models; the combination of explanation maps of multiple classifiers to produce improved explanations of the ensemble to which they belong [248]; and recent insights dealing with traditional and gradient boosting ensembles [292, 293].

2.3.4 *Support Vector Machines*

Another shallow ML model with historical presence in the literature is the Support Vector Machine. SVM models are more complex than tree ensembles, with a much opaquer structure. Many implementations of post-hoc explainability techniques have been proposed to relate what is mathematically described internally in these models, to what different authors considered explanations about the problem at hand. Technically, an SVM constructs a hyper-plane or set of hyper-planes in a high or infinite-dimensional space, which can be used for classification, regression, or other tasks such as outlier detection. Intuitively, a good separation is achieved by the hy-

perplane that has the largest distance (so-called functional margin) to the nearest training-data point of any class, since in general, the larger the margin, the lower the generalization error of the classifier. SVMs are among the most used ML models due to their excellent prediction and generalization capabilities. From the techniques stated in Section 2, post-hoc explainability applied to SVMs covers explanation by *simplification*, *local explanations*, *visualizations* and *explanations by example*.

Among explanation by simplification, four classes of simplifications are made. Each of them differentiates from the other by how deep they go into the algorithm inner structure. First, some authors propose techniques to build rule based models only from the support vectors of a trained model. This is the approach of [96], which proposes a method that extracts rules directly from the support vectors of a trained SVM using a modified sequential covering algorithm. In [60] the same authors propose eclectic rule extraction, still considering only the support vectors of a trained model. The work in [97] generates fuzzy rules instead of classical propositional rules. Here, the authors argue that long antecedents reduce comprehensibility, hence, a fuzzy approach allows for a more linguistically understandable result. The second class of simplifications can be exemplified by [101], which proposed the addition of the SVM's hyperplane, along with the support vectors, to the components in charge of creating the rules. His method relies on the creation of hyper-rectangles from the intersections between the support vectors and the hyper-plane. In a third approach to *model simplification*, another group of authors considered adding the actual training data as a component for building the rules. In [129, 249, 251] the authors proposed a clustering method to group prototype vectors for each class. By combining them with the support vectors, it allowed defining ellipsoids and hyper-rectangles in the input space. Similarly in [109], the authors proposed the so-called Hyper-rectangle Rule Extraction, an algorithm based on SVC (Support Vector Clustering) to find prototype vectors for each class and then define small hyper-rectangles around. In [108], the authors formulate the rule extraction problem as a multi-constrained optimization to create a set of non-overlapping rules. Each rule conveys a non-empty hyper-cube with a shared edge with the hyper-plane. In a similar study conducted in [250], extracting rules for gene expression data, the authors presented a novel technique as a component of a multi-kernel SVM. This multi-kernel method consists of feature selection, prediction modeling and rule extraction. Finally, the study in [137] makes use of a growing SVC to give an interpretation to SVM decisions in terms of linear rules that define the space in Voronoi sections from the extracted prototypes.

Leaving aside rule extraction, the literature has also contemplated some other techniques to contribute to the interpretation of SVMs. Three of them (visualization techniques) are clearly used toward explaining SVM models when used for concrete applications. For instance, [80] presents an innovative approach to visualize trained SVM to extract the information content from the kernel matrix. They center the study on Support Vector Regression models. They show the ability of the algorithm to visualize which of the input variables are actually related with the associated output data. In [71] a visual way combines the output of the SVM with heatmaps to guide the modification of compounds in late stages of drug discovery. They assign colors to atoms based on the weights of a trained linear SVM that allows for a much more comprehensive way of debugging the process. In [119] the authors argue that many

of the presented studies for interpreting SVMs only account for the weight vectors, leaving the margin aside. In their study they show how this margin is important, and they create an statistic that explicitly accounts for the SVM margin. The authors show how this statistic is specific enough to explain the multivariate patterns shown in neuroimaging.

Noteworthy is also the intersection between SVMs and Bayesian systems, the latter being adopted as a post-hoc technique to explain decisions made by the SVM model. This is the case of [253] and [252], which are studies where SVMs are interpreted as MAP (Maximum A Posteriori) solutions to inference problems with Gaussian Process Priors. This framework makes tuning the hyper-parameters comprehensible and gives the capability of predicting class probabilities instead of the classical binary classification of SVMs. Interpretability of SVM models becomes even more involved when dealing with non-CPD (Conditional Positive Definite) kernels that are usually harder to interpret due to missing geometrical and theoretical understanding. The work in [105] revolves around this issue with a geometrical interpretation of indefinite kernel SVMs, showing that these do not classify by hyper-plane margin optimization. Instead, they minimize the distance between convex hulls in pseudo-Euclidean spaces.

A difference might be appreciated between the post-hoc techniques applied to other models and those noted for SVMs. In previous models, *model simplification* in a broad sense was the prominent method for post-hoc explainability. In SVMs, *local explanations* have started to take some weight among the propositions. However, simplification based methods are, on average, much older than local explanations.

As a final remark, none of the reviewed methods treating SVM explainability are dated beyond 2017, which might be due to the progressive proliferation of DL models in almost all disciplines. Another plausible reason is that these models are already understood, so it is hard to improve upon what has already been done.

2.3.5 *Explainability in Deep Learning*

Post-hoc *local explanations* and *feature relevance* techniques are increasingly the most adopted methods for explaining DNNs. This section reviews explainability studies proposed for the most used DL models, namely multi-layer neural networks, Convolutional Neural Networks (CNN) and Recurrent Neural Networks (RNN).

2.3.5.1 *Multi Layer Neural Networks*

From their inception, multi-layer neural networks (also known as multi-layer perceptrons) have been warmly welcomed by the academic community due to their huge ability to infer complex relations among variables. However, as stated in the introduction, developers and engineers in charge of deploying these models in real-life production find in their questionable explainability a common reason for reluctance. That is why neural networks have been always considered as *black-box* models. The fact that explainability is often a must for the model to be of practical value, forced the community to generate multiple explainability techniques for multi-layer neural

networks, including *model simplification* approaches, *feature relevance* estimators, *text explanations*, *local explanations* and model *visualizations*.

Several *model simplification* techniques have been proposed for neural networks with one single hidden layer, however very few works have been presented for neural networks with multiple hidden layers. One of these few works is DeepRED algorithm [262], which extends the decompositional approach to rule extraction (splitting at neuron level) presented in [264] for multi-layer neural network by adding more decision trees and rules.

Some other works use *model simplification* as a post-hoc explainability approach. For instance, [59] presents a simple distillation method called *Interpretable Mimic Learning* to extract an interpretable model by means of gradient boosting trees. In the same direction, the authors in [138] propose a hierarchical partitioning of the feature space that reveals the iterative rejection of unlikely class labels, until association is predicted. In addition, several works addressed the distillation of knowledge from an ensemble of models into a single model [83, 294, 295] .

Given the fact that the simplification of multi-layer neural networks is more complex as the number of layers increases, explaining these models by *feature relevance* methods has become progressively more popular. One of the representative works in this area is [63], which presents a method to decompose the network classification decision into contributions of its input elements. They consider each neuron as an object that can be decomposed and expanded then aggregate and back-propagate these decompositions through the network, resulting in a *deep Taylor decomposition*. In the same direction, the authors in [113] proposed DeepLIFT, an approach for computing importance scores in a multi-layer neural network. Their method compares the activation of a neuron to the reference activation and assigns the score according to the difference.

On the other hand, some works try to verify the theoretical soundness of current explainability methods. For example, the authors in [267], bring up a fundamental problem of most *feature relevance* techniques, designed for multi-layer networks. They showed that two axioms that such techniques ought to fulfill namely, *sensitivity* and *implementation invariance*, are violated in practice by most approaches. Following these axioms, the authors of [267] created *integrated gradients*, a new *feature relevance* method proven to meet the aforementioned axioms. Similarly, the authors in [64] analyzed the correctness of current *feature relevance* explanation approaches designed for Deep Neural Networks (e.g.: DeConvNet, Guided BackProp and LRP) on simple linear neural networks. Their analysis showed that these methods do not produce the theoretically correct explanations and presented two new explanation methods *PatternNet* and *PatternAttribution* that are more theoretically sound for both, simple and deep neural networks.

2.3.5.2 *Convolutional Neural Networks*

Currently, CNNs constitute the state-of-art models in all fundamental computer vision tasks, from image classification and object detection to instance segmentation. Typically, these models are built as a sequence of convolutional layers and pooling layers to automatically learn increasingly higher level features. At the end of the sequence, one or multiple fully connected layers are used to map the output features

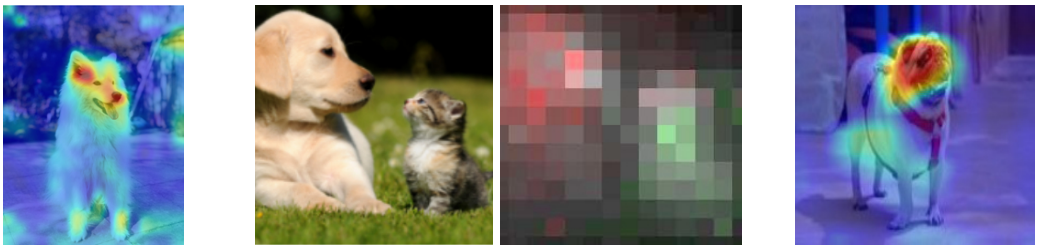
map into scores. This structure entails extremely complex internal relations that are very difficult to explain. Fortunately, the road to explainability for CNNs is easier than for other types of models, as the human cognitive skills favors the understanding of visual data.

Existing works that aim at understanding what CNNs learn can be divided into two broad categories: 1) those that try to understand the decision process by mapping back the output in the input space to see which parts of the input were discriminative for the output; and 2) those that try to delve inside the network and interpret how the intermediate layers see the external world, not necessarily related to any specific input, but in general.

One of the seminal works in the first category was [296]. When an input image runs feed-forward through a CNN, each layer outputs a number of feature maps with strong and soft activations. The authors in [296] used Deconvnet, a network designed previously by the same authors [145] that, when fed with a feature map from a selected layer, reconstructs the maximum activations. These reconstructions can give an idea about the parts of the image that produced that effect. To visualize these strongest activations in the input image, the same authors used the occlusion sensitivity method to generate a saliency map [139], which consists of iteratively forwarding the same image through the network occluding a different region at a time.

To improve the quality of the mapping on the input space, several subsequent papers proposed simplifying both the CNN architecture and the visualization method. In particular, [99] included a global average pooling layer between the last convolutional layer of the CNN and the fully-connected layer that predicts the object class. With this simple architectural modification of the CNN, the authors built a class activation map that helps identify the image regions that were particularly important for a specific object class by projecting back the weights of the output layer on the convolutional feature maps. Later, in [146], the authors showed that max-pooling layers can be used to replace convolutional layers with a large stride without loss in accuracy on several image recognition benchmarks. They obtained a cleaner visualization than Deconvnet by using a guided backpropagation method.

To increase the interpretability of classical CNNs, the authors in [116] used a loss for each filter in high level convolutional layers to force each filter to learn very specific object components. The obtained activation patterns are much more interpretable for their exclusiveness with respect to the different labels to be predicted. The authors in [75] proposed visualizing the contribution to the prediction of each single pixel of the input image in the form of a heatmap. They used a Layer-wise Relevance Propagation (LRP) technique, which relies on a Taylor series close to the prediction point rather than partial derivatives at the prediction point itself. To further improve the quality of the visualization, attribution methods such as heatmaps, saliency maps or class activation methods (*GradCAM* [297]) are used (see Figure 2.6). In particular, the authors in [297] proposed a Gradient-weighted Class Activation Mapping (Grad-CAM), which uses the gradients of any target concept, flowing into the final convolutional layer to produce a coarse localization map, highlighting the important regions in the image for predicting the concept.



(a) Heatmap [173] (b) Attribution [298] (c) Grad-CAM [297]

Figure 2.6: Examples of rendering for different XAI visualization techniques on images.

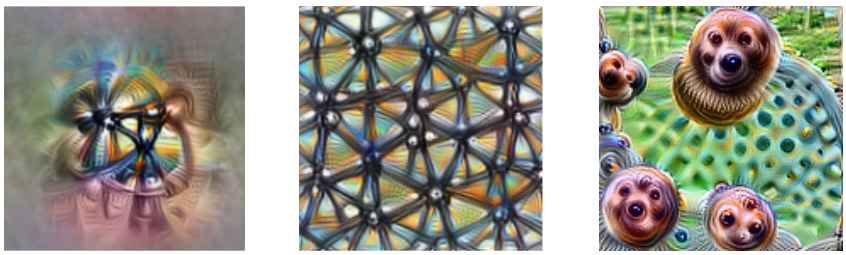
In addition to the aforementioned *feature relevance* and *visual* explanation methods, some works proposed generating *text explanations* of the visual content of the image. For example, the authors in [94] combined a CNN feature extractor with an RNN attention model to automatically learn to describe the content of images. In the same line, [283] presented a three-level attention model to perform a fine-grained classification task. The general model is a pipeline that integrates three types of attention: the object level attention model proposes candidate image regions or patches from the input image, the part-level attention model filters out non-relevant patches to a certain object, and the last attention model localizes discriminative patches. In the task of video captioning, the authors in [114] use a CNN model combined with a bi-directional LSTM model as encoder to extract video features and then feed these features to an LSTM decoder to generate textual descriptions.

One of the seminal works in the second category is [140]. In order to analyse the visual information contained inside the CNN, the authors proposed a general framework that reconstruct an image from the CNN internal representations and showed that several layers retain photographically accurate information about the image, with different degrees of geometric and photometric invariance. To visualize the notion of a class captured by a CNN, the same authors created an image that maximizes the class score based on computing the gradient of the class score with respect to the input image [277]. In the same direction, the authors in [273] introduced a Deep Generator Network (DGN) that generates the most representative image for a given output neuron in a CNN.

For quantifying the interpretability of the latent representations of CNNs, the authors in [128] used a different approach called network dissection. They run a large number of images through a CNN and then analyze the top activated images by considering each unit as a concept detector to further evaluate each unit for semantic segmentation. This paper also examines the effects of classical training techniques on the interpretability of the learned model.

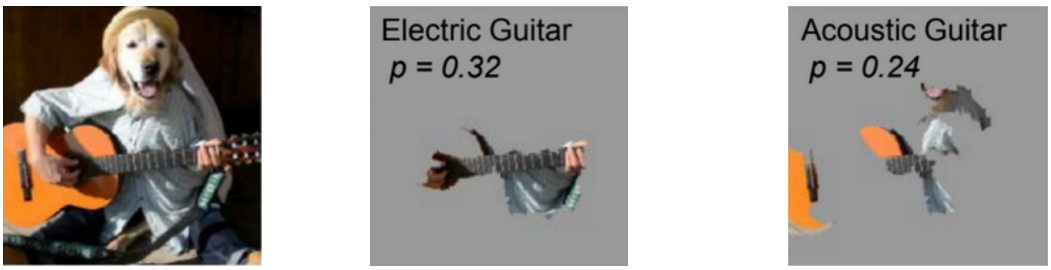
Although many of the techniques examined above utilize *local explanations* to achieve an overall explanation of a CNN model, others explicitly focus on building global explanations based on locally found prototypes. In [268, 299], the authors empirically showed how *local explanations* in deep networks are strongly dominated by their lower level features. They demonstrated that deep architectures provide strong priors that prevent the altering of how these low-level representations are captured. All in all, *visualization* mixed with *feature relevance* methods are arguably the most adopted approach to explainability in CNNs.

Instead of using one single interpretability technique, the framework proposed in [300] combines several methods to provide much more information about the network. For example, combining feature visualization (*what is a neuron looking for?*) with attribution (*how does it affect the output?*) allows exploring how the network decides between labels. This visual interpretability interface displays different blocks such as feature visualization and attribution depending on the visualization goal. This interface can be thought of as a union of individual elements that belong to layers (input, hidden, output), atoms (a neuron, channel, spatial or neuron group), content (activations – the amount a neuron fires, attribution – which classes a spatial position most contributes to, which tends to be more meaningful in later layers), and presentation (information visualization, feature visualization). Figure 2.7 shows some examples. Attribution methods normally rely on pixel association, displaying what part of an input example is responsible for the network activating in a particular way [298].



(a) Neuron (b) Channel (c) Layer

Figure 2.7: Feature visualization at different levels of a certain network [298].



(a) Original image (b) Explaining *electric guitar* (c) Explaining *acoustic guitar*

Figure 2.8: Three different examples of explanation when using LIME on images [74].

A much simpler approach to all the previously cited methods was proposed in LIME framework [74], as was described in Subsection 2.3.1 LIME perturbs the input and sees how the predictions change. In image classification, LIME creates a set of perturbed instances by dividing the input image into interpretable components (contiguous *superpixels*), and runs each perturbed instance through the model to get a probability. A simple linear model learns on this data set, which is locally weighted. At the end of the process, LIME presents the superpixels with highest positive weights as an explanation (see Figure 2.8).

A completely different explainability approach is proposed in adversarial detection. To understand model failures in detecting adversarial examples, the authors in [269] apply the k-nearest neighbors algorithm on the representations of the data learned

by each layer of the CNN. A test input image is considered as adversarial if its representations are far from the representations of the training images.

2.3.5.3 *Recurrent Neural Networks*

As occurs with CNNs in the visual domain, RNNs have lately been used extensively for predictive problems defined over inherently sequential data, with a notable presence in natural language processing and time series analysis. These types of data exhibit long-term dependencies that are complex to be captured by a ML model. RNNs are able to retrieve such time-dependent relationships by formulating the retention of knowledge in the neuron as another parametric characteristic that can be learned from data.

Few contributions have been made for explaining RNN models. These studies can be divided into two groups: 1) explainability by understanding what a RNN model has learned (mainly via *feature relevance* methods); and 2) explainability by modifying RNN architectures to provide insights about the decisions they make (*local explanations*).

In the first group, the authors in [285] extend the usage of LRP to RNNs. They propose a specific propagation rule that works with multiplicative connections as those in LSTMs (Long Short Term Memory) units and GRUs (Gated Recurrent Units). The authors in [286] propose a visualization technique based on finite horizon n-grams that discriminates interpretable cells within LSTM and GRU networks. Following the premise of not altering the architecture, [301] extends the interpretable mimic learning distillation method used for CNN models to LSTM networks, so that interpretable features are learned by fitting Gradient Boosting Trees to the trained LSTM network under focus.

Aside from the approaches that do not change the inner workings of the RNNs, [290] presents RETAIN (REverse Time AttentIoN) model, which detects influential past patterns by means of a two-level neural attention model. To create an interpretable RNN, the authors in [288] propose an RNN based on SISTA (Sequential Iterative Soft-Thresholding Algorithm) that models a sequence of correlated observations with a sequence of sparse latent vectors, making its weights interpretable as the parameters of a principled statistical model. Finally, in [289], its authors construct a combination of an HMM (Hidden Markov Model) and an RNN, so that the overall model approach harnesses the interpretability of the HMM and the accuracy of the RNN model.

2.3.6 *Hybrid Transparent and black-box Methods*

The use of background knowledge in the form of logical statements or constraints in Knowledge Bases (KBs) has shown to not only improve explainability but also performance with respect to purely data-driven approaches [302–304]. A positive side effect shown is that this hybrid approach provides robustness to the learning system when errors are present in the training data labels. Other approaches have shown to be able to jointly learn and reason with both symbolic and sub-symbolic representations and inference. The interesting aspect is that this blend allows for

expressive probabilistic-logical reasoning in an end-to-end fashion [305]. A successful use case is on dietary recommendations, where explanations are extracted from the reasoning behind (non-deep but KB-based) models [306].

Future data fusion approaches may thus consider endowing DL models with explainability by externalizing other domain information sources. Deep formulation of classical ML models has been done, e.g. in Deep Kalman filters (DKFs) [307], Deep Variational Bayes Filters (DVBFs) [308], Structural Variational Autoencoders (SVAE) [309], or conditional random fields as RNNs [310]. These approaches provide deep models with the interpretability inherent to probabilistic graphical models. For instance, SVAE combines probabilistic graphical models in the embedding space with neural networks to enhance the interpretability of DKFs. A particular example of classical ML model enhanced with its DL counterpart is Deep K-Nearest Neighbors DkNN [269], where the neighbors constitute human-interpretable explanations of predictions. The intuition is based on the rationalization of a DNN prediction based on evidence. This evidence consists of a characterization of confidence termed *credibility* that spans the hierarchy of representations within a DNN, that must be supported by the training data [269].

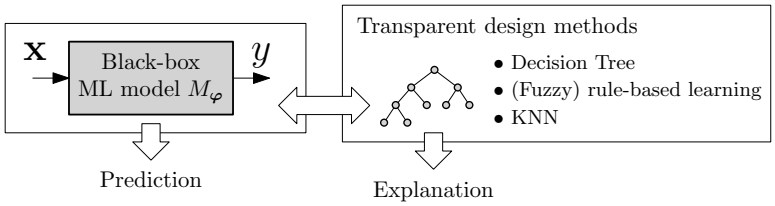


Figure 2.9: Pictorial representation of a hybrid model. A neural network considered as a *black-box* can be explained by associating it to a more interpretable model such as a Decision Tree [311], a (fuzzy) rule-based system [23] or KNN [269].

A different perspective on hybrid XAI models consists of enriching *black-box* models knowledge with that one of transparent ones, as proposed in [28] and further refined in [174] and [312]. In particular, this can be done by constraining the neural network thanks to a semantic KB and bias-prone concepts [174], or by stacking ensembles jointly encompassing white- and *black-box* models [312].

Other examples of hybrid symbolic and sub-symbolic methods where a knowledge-base tool or graph-perspective enhances the neural (e.g., language [313]) model are in [314, 315]. In reinforcement learning, very few examples of symbolic (graphical [316] or relational [78, 317]) hybrid models exist, while in recommendation systems, for instance, explainable autoencoders are proposed [318]. A specific transformer architecture symbolic visualization method (applied to music) pictorially shows how soft-max attention works [319]. By visualizing self-reference, i.e., the last layer of attention weights, arcs show which notes in the past are informing the future and how attention is skip over less relevant sections. Transformers can also help explain image captions visually [320].

Another hybrid approach consists of mapping an uninterpretable *black-box* system to a white-box *twin* that is more interpretable. For example, an opaque neural network can be combined with a transparent Case Based Reasoning (CBR) system [321, 322]. In [323], the DNN and the CBR (in this case a kNN) are paired in order to

improve interpretability while keeping the same accuracy. The *explanation by example* consists of analyzing the feature weights of the DNN which are then used in the CBR, in order to retrieve nearest-neighbor cases to explain the DNN's prediction.

2.3.7 *Alternative Taxonomy of Post-hoc Explainability Techniques for Deep Learning*

DL is the model family where most research has been concentrated in recent times and they have become central for most of the recent literature on XAI. While the division between model-agnostic and model-specific is the most common distinction made, the community has not only relied on this criteria to classify XAI methods. For instance, some model-agnostic methods such as *SHAP* [229] are widely used to explain DL models. That is why several XAI methods can be easily categorized in different taxonomy branches depending on the angle the method is looked at. An example is LIME which can also be used over CNNs, despite not being exclusive to deal with images. Searching within the alternative DL taxonomy shows us that LIME can explicitly be used for *Explaining a Deep Network Processing*, as a kind of *Linear Proxy Model*. Another type of classification is indeed proposed in [16] with a segmentation based on 3 categories. The first category groups methods explaining the processing of data by the network, thus answering to the question *"why does this particular input lead to this particular output?"*. The second one concerns methods explaining the representation of data inside the network, i.e., answering to the question *"what information does the network contain?"*. The third approach concerns models specifically designed to simplify the interpretation of their own behavior. Such a multiplicity of classification possibilities leads to different ways of constructing XAI taxonomies.

Figure 2.10 shows the alternative Deep Learning taxonomy inferred from [16]. From the latter, it can be deduced the complementarity and overlapping of this taxonomy to Figure 2.5 as:

- Some methods [277, 285] classified in distinct categories (namely *feature relevance for CNN* and *feature relevance for RNN*) in Figure 2.5 are included in a single category (*Explanation of Deep Network Processing with Salience Mapping*) when considering the classification from [16].
- Some methods [85, 147] are classified on a single category (*Explanation by simplification for Multi-Layer Neural Network*) in Figure 2.5 while being in 2 different categories (namely, *Explanation of Deep Network Processing with Decision Trees* and *Explanation of Deep Network Representation with the Role of Representation Vectors*) in [16], as shown in Figure 2.10.

A classification based on explanations of model processing and explanations of model representation is relevant, as it leads to a differentiation between the execution trace of the model and its internal data structure. This means that depending of the failure reasons of a complex model, it would be possible to pick-up the right XAI method according to the information needed: the execution trace or the data structure. This idea is analogous to testing and debugging methods used in regular programming paradigms [351].

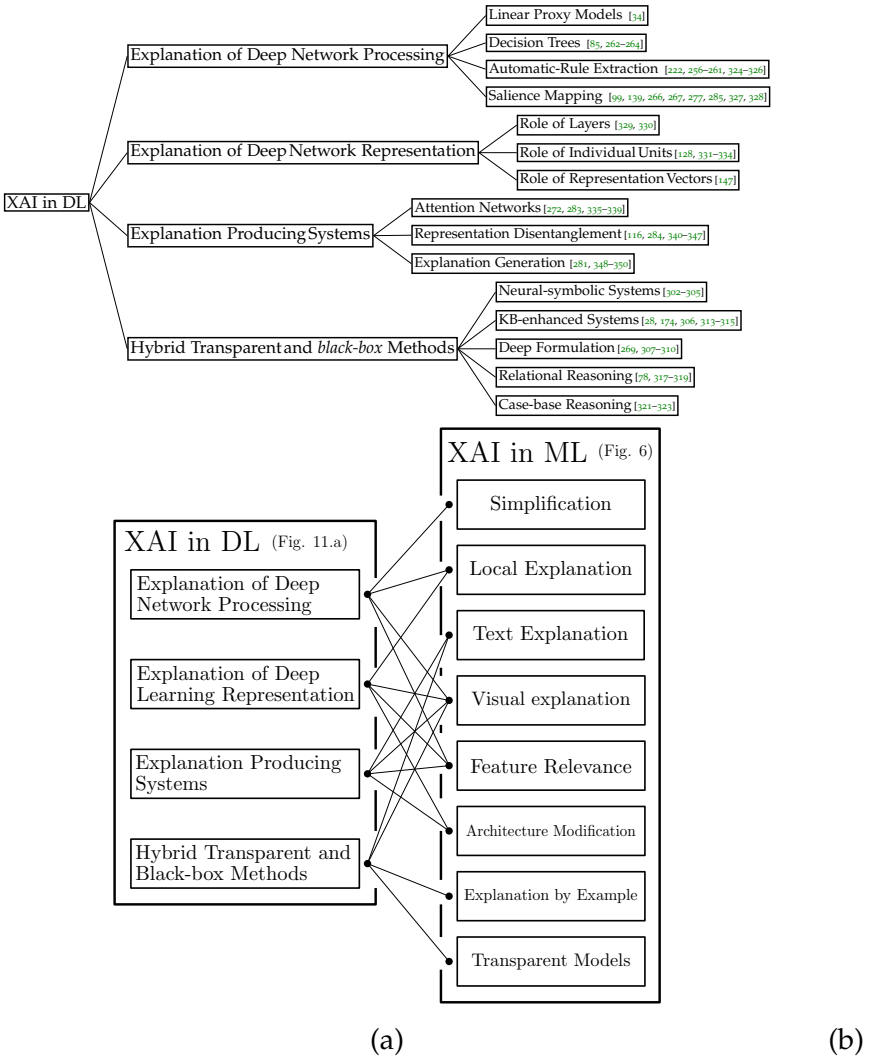


Figure 2.10: (a) Alternative Deep Learning specific taxonomy extended from the categorization from [16]; and (b) its connection to the taxonomy in Figure 2.5.

2.4 OPEN ISSUES IN THE FIELD OF XAI

We now capitalize on the performed literature review to put forward a critique of the achievements, trends and challenges that are still to be addressed in the field of explainability of ML and data fusion models. Actually our discussion on the advances taken so far in this field has already anticipated some of these challenges. In this section we revisit them and explore new research opportunities for XAI, identifying possible research paths that can be followed to address them effectively in years to come:

- When introducing this thesis in Section 1.1 we already mentioned the existence of a tradeoff between model interpretability and performance, in the sense that making a ML model more understandable could eventually degrade the quality of its produced decisions. In Subsection 2.4.1 we will stress on the potential

of XAI developments to effectively achieve an optimal balance between the interpretability and performance of ML models.

- In Subsection 2.1.2 we stressed on the imperative need for reaching a consensus on *what* explainability entails within the AI realm. Reasons for pursuing explainability are also assorted and, under our own assessment of the literature so far, not unambiguously mentioned throughout related works. In Subsection 2.4.2 we will further delve into this important issue.

- Given its notable prevalence in the XAI literature, Subsections 2.3.5 and 2.3.7 revolved on the explainability of Deep Learning models, examining advances reported so far around a specific bibliographic taxonomy. We go in this same direction with Subsection 2.4.3, which exposes several challenges that hold in regards to the explainability of this family of models.

- Finally, we close up this prospective discussion with Subsections 2.4.4 to 2.4.9, which place on the table several research niches that despite its connection to model explainability, remain insufficiently studied by the community.

2.4.1 *On the Tradeoff between Interpretability and Performance*

The matter of interpretability versus performance is one that repeats itself through time, but as any other big statement, has its surroundings filled with myths and misconceptions.

As perfectly stated in [352], it is not necessarily true that models that are more complex are inherently more accurate. This statement is false in cases in which the data is well structured and features at our disposal are of great quality and value. This case is somewhat common in some industry environments, since features being analyzed are constrained within very controlled physical problems, in which all of the features are highly correlated, and not much of the possible landscape of values can be explored in the data [353]. What can be hold as true, is that more complex models enjoy much more flexibility than their simpler counterparts, allowing for more complex functions to be approximated. Now, returning to the statement *"models that are more complex are more accurate"*, given the premise that the function to be approximated entails certain complexity, that the data available for study is greatly widespread among the world of suitable values for each variable and that there is enough data to harness a complex model, the statement presents itself as a true statement. It is in this situation that the trade-off between performance and interpretability can be observed. It should be noted that the attempt at solving problems that do not respect the aforementioned premises will fall on the trap of attempting to solve a problem that does not provide enough data diversity (variance). Hence, the added complexity of the model will only fight against the task of accurately solving the problem.

In this path toward performance, when the performance comes hand in hand with complexity, interpretability encounters itself on a downward slope that until now appeared unavoidable. However, the apparition of more sophisticated methods for explainability could invert or at least cancel that slope. Figure 2.11 shows a tentative representation inspired by previous works [7], in which XAI shows its power

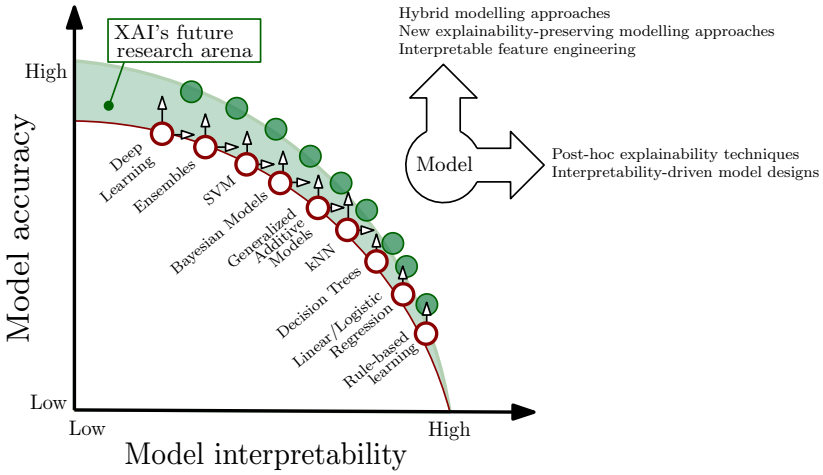


Figure 2.11: Trade-off between model interpretability and performance, and a representation of the area of improvement where the potential of XAI techniques and tools resides.

to improve the common trade-off between model interpretability and performance. Another aspect worth mentioning at this point due to its close link to model interpretability and performance is the *approximation dilemma*: explanations made for a ML model must be made drastic and approximate enough to match the requirements of the audience for which they are sought, ensuring that explanations are representative of the studied model and do not oversimplify its essential features.

2.4.2 *On the Concept and Metrics*

The literature clearly asks for an unified concept of explainability. In order for the field to thrive, it is imperative to place a common ground upon which the community is enabled to contribute new techniques and methods. A common concept must convey the needs expressed in the field. It should propose a common structure for every XAI system. This paper attempted a new proposition of a concept of explainability that is built upon that from Gunning [7]. In that proposition and the following strokes to complete it (Subsection 2.1.2), explainability is defined as the ability a model has to make its functioning clearer to an audience. To address it, post-hoc type methods exist. The concept portrayed in this chapter might not be complete but as it stands, allows for a first common ground and reference point to sustain a profitable discussion in this matter. It is paramount that the field of XAI reaches an agreement in this respect combining the shattered efforts of a widespread field behind the same banner.

Another key feature needed to relate a certain model to this concrete concept is the existence of a metric. A metric, or group of them should allow for a meaningful comparison of how well a model fits the definition of explainable. Without such tool, any claim in this respect dilutes among the literature, not providing a solid ground on which to stand. These metrics, as the classic ones (accuracy, F1, sensitivity...), should express how well the model performs in a certain aspect of explainability. Some attempts have been done recently around the measurement of XAI, as reviewed thor-

oughly in [354, 355]. In general, XAI measurements should evaluate the goodness, usefulness and satisfaction of explanations, the improvement of the mental model of the audience induced by model explanations, and the impact of explanations on the performance of the model and on the trust and reliance of the audience. Measurement techniques surveyed in [354] and [355] (e.g., goodness checklist, explanation satisfaction scale, elicitation methods for mental models, computational measures for explainer fidelity, explanation trustworthiness and model reliability) seem to be a good push in the direction of evaluating XAI techniques. Unfortunately, conclusions drawn from these overviews are aligned with our prospects on the field: more quantifiable, general XAI metrics are really needed to support the existing measurement procedures and tools proposed by the community.

This chapter does not tackle the problem of designing such a suite of metrics, since such a task should be approached by the community as a whole, prior acceptance of the broader concept of explainability, which on the other hand, is one of the aims of the current Thesis. Nevertheless, we advocate for further efforts towards new proposals to evaluate the performance of XAI techniques, as well as comparison methodologies among XAI approaches that allow contrasting them quantitatively under different application context, models and purposes.

2.4.3 *Challenges to Achieve Explainable Deep Learning*

While many efforts are currently being made in the area of XAI, there are still many challenges to be faced before being able to obtain explainability in DL models. First, as explained in Subsection 2.1.2, there is a lack of agreement on the vocabulary and the different definitions surrounding XAI. As an example, we often see the terms *feature importance* and *feature relevance* referring to the same concept. This is even more obvious for visualization methods, where there is absolutely no consistency behind what is known as saliency maps, salient masks, heatmaps, neuron activations, attribution, and other approaches alike. As XAI is a relatively young field, the community does not have a standardized terminology yet.

As it has been commented in Subsection 2.4.1, there is a trade-off between interpretability and accuracy [16], i.e., between the simplicity of the information given by the system on its internal functioning, and the exhaustiveness of this description. Whether the observer is an expert in the field, a policy-maker or a user without machine learning knowledge, intelligibility does not have to be at the same level in order to provide the *audience* an understanding [356]. This is one of the reasons why, as mentioned above, a challenge in XAI is establishing objective metrics on what constitutes a good explanation. A possibility to reduce this subjectivity is taking inspiration from experiments on human psychology, sociology or cognitive sciences to create objectively convincing explanations. Relevant findings to be considered when creating an explainable AI model are highlighted in [357]: First, explanations are better when *constrictive*, meaning that a prerequisite for a good explanation is that it does not only indicate why the model made a decision X, but also why it made decision X rather than decision Y. It is also explained that probabilities are not as important as causal links in order to provide a satisfying explanation. Considering that black box models tend to process data in a quantitative manner, it would be necessary to

translate the probabilistic results into qualitative notions containing causal links. In addition, they state that explanations are *selective*, meaning that focusing solely on the main causes of a decision-making process is sufficient. It was also shown that the use of counterfactual explanations can help the user to understand the decision of a model [42, 44, 358].

Combining connectionist and symbolic paradigms seems a favourable way to address this challenge [174, 304, 317, 359, 360]. On one hand, connectionist methods are more precise but opaque. On the other hand, symbolic methods are popularly considered less efficient, while they offer a greater explainability thus respecting the conditions mentioned above:

- The ability to refer to established reasoning rules allows symbolic methods to be constrictive.
- The use of a KB formalized e.g. by an ontology can allow data to be processed directly in a qualitative way.
- Being selective is less straightforward for connectionist models than for symbolic ones.

Recalling that a good explanation needs to influence the mental model of the user, i.e. the representation of the external reality using, among other things, symbols, it seems obvious that the use of the symbolic learning paradigm is appropriate to produce an explanation. Therefore, neural-symbolic interpretability could provide convincing explanations while keeping or improving generic performance [302].

As stated in [28], a truly explainable model should not leave explanation generation to the users as different explanations may be deduced depending on their background knowledge. Having a semantic representation of the knowledge can help a model to have the ability to produce explanations (e.g., in natural language [174]) combining common sense reasoning and human-understandable features.

Furthermore, until an objective metric has been adopted, it appears necessary to make an effort to rigorously formalize evaluation methods. One way may be drawing inspiration from the social sciences, e.g., by being consistent when choosing the evaluation questions and the population sample used [361].

A final challenge XAI methods for DL need to address is providing explanations that are accessible for society, policy makers and the law as a whole. In particular, conveying explanations that require non-technical expertise will be paramount to both handle ambiguities, and to develop the social right to the (not-yet available) right for explanation in the EU General Data Protection Regulation (GDPR) [362].

2.4.4 *Explanations for AI Security: XAI and Adversarial Machine Learning*

Nothing has been said about confidentiality concerns linked to XAI. One of the last surveys very briefly introduced the idea of algorithm property and trade secrets [18]. However, not much attention has been payed to these concepts. If *confidential* is the property that makes something *secret*, in the AI context many aspects involved in a model may hold this property. For example, imagine a model that some company has developed through many years of research in a specific field. The knowledge

synthesized in the model built might be considered to be confidential, and it may be compromised even by providing only input and output access [363]. The latter shows that, under minimal assumptions, *data model functionality stealing* is possible. An approach that has served to make DL models more robust against intellectual property exposure based on a sequence of non accessible queries is in [364]. This recent work exposes the need for further research toward the development of XAI tools capable of explaining ML models while keeping the model's confidentiality in mind.

Ideally, XAI should be able to explain the knowledge within an AI model and it should be able to reason about what the model acts upon. However, the information revealed by XAI techniques can be used both to generate more effective attacks in adversarial contexts aimed at confusing the model, at the same time as to develop techniques to better protect against private content exposure by using such information. Adversarial attacks [365] try to manipulate a ML algorithm after learning what is the specific information that should be fed to the system so as to lead it to a specific output. For instance, regarding a supervised ML classification model, adversarial attacks try to discover the minimum changes that should be applied to the input data in order to cause a different classification. This has happened regarding computer vision systems of autonomous vehicles; a minimal change in a stop signal, imperceptible to the human eye, led vehicles to detect it as a 45 mph signal [366]. For the particular case of DL models, available solutions such as Cleverhans [367] seek to detect adversarial vulnerabilities, and provide different approaches to harden the model against them. Other examples include AlfaSVMLib [368] for SVM models, and AdversarialLib [369] for evasion attacks. There are even available solutions for unsupervised ML, like clustering algorithms [370].

While XAI techniques can be used to furnish more effective adversarial attacks or to reveal confidential aspects of the model itself, some recent contributions have capitalized on the possibilities of Generative Adversarial Networks (GANs [371]), Variational Autoencoders [372] and other generative models towards explaining data-based decisions. Once trained, generative models can generate instances of what they have learned based on a noise input vector that can be interpreted as a latent representation of the data at hand. By manipulating this latent representation and examining its impact on the output of the generative model, it is possible to draw insights and discover specific patterns related to the class to be predicted. This generative framework has been adopted by several recent studies [373, 374] mainly as an attribution method to relate a particular output of a Deep Learning model to their input variables. Another interesting research direction is the use of generative models for the creation of counterfactuals, i.e., modifications to the input data that could eventually alter the original prediction of the model [375]. Counterfactual prototypes help the user understand the performance boundaries of the model under consideration for his/her improved trust and informed criticism. In light of this recent trend, we definitely believe that there is road ahead for generative ML models to take their part in scenarios demanding understandable machine decisions. However, there is still a gap in the conception of these adversarial generators.

Most of the models to date are focused on exploiting a given singular feature within the model. Lets say, the minimal change needed to turn a model's prediction.

Such techniques only treat adversarial analysis as a single objective optimization process. However, adversarial analysis could be presented as a multi-objective optimization process. This new paradigm would allow for a much broader analysis of the model at hand, enforcing the conditions that are thought to be possible in its application. This proposition is still unexploited in the literature and serves as one the motivating concepts for the rest of this thesis. The Chapter 4 covers a further analysis on the matter and presents a technical contribution to prove its relevance.

2.4.5 *XAI and Output Confidence*

Safety issues have also been studied in regards to processes that depend on the output of AI models, such as vehicular perception and self-driving in autonomous vehicles, automated surgery, data-based support for medical diagnosis, insurance risk assessment and cyber-physical systems in manufacturing, among others [376]. In all these scenarios erroneous model outputs can lead to harmful consequences, which has yielded comprehensive regulatory efforts aimed at ensuring that no decision is made solely on the basis of data processing [172].

In parallel, research has been conducted towards minimizing both risk and uncertainty of harms derived from decisions made on the output of a ML model. As a result, many techniques have been reported to reduce such a risk, among which we pause at the evaluation of the model's output confidence to decide upon. In this case, the inspection of the share of epistemic uncertainty (namely, the uncertainty due to lack of knowledge) of the input data and its correspondence with the model's output confidence can inform the user and eventually trigger his/her rejection of the model's output [377, 378]. To this end, explaining via XAI techniques which region of the input data the model is focused on when producing a given output can discriminate possible sources of epistemic uncertainty within the input domain.

2.4.6 *XAI in Randomized Neural Networks*

Within the field of NN, Randomized Neural Networks have seem some success from the performance perspective [379–385]. However, their xAI stance is non-existing. These models present an interesting tradeoff against classical Neural Networks. Randomized Neural Networks substitute iterative training for randomization. Such a change supposes a great advantage in training time which if backed up by great performance, should leave open a new path to high dimensional model training apart of classical back-propagation neural networks.

Among Randomized Neural Networks, three main families can be found [385], namely: Random Weight-Feed Forward Network (RW-FFN), Random Features for Kernel Methods (RF-KM) and Reservoir Computing (RC). Paying attention to the body of the taxonomy in Figure 2.5, the gap of research missing for this area can be found. These models have been always surrounded by a subtle mist of miss-trust due to their randomized essence. Understanding the reasons of why interchanging long iterative training with randomized initializations is even possible is paramount for these families of methods to thrive.

This gap in the literature ought to be filled. Randomized NNs are starting to grow in their application, although, they are still in need of supporting xAI methods and frameworks that help them match the necessities of their fields of application. This aspect represents the motivation for the next chapter of this thesis. The Chapter 3 of the present thesis will delve into this matter by proposing a new framework that stands as the first XAI contribution to the field of Randomized Neural Networks.

2.4.7 *XAI, Rationale Explanation and Critical Data Studies*

When shifting the focus to the research practices seen in Data Science, it has been noted that reproducibility is stringently subject not only to the mere sharing of data, models and results to the community, but also to the availability of information about the full discourse around data collection, understanding, assumptions held and insights drawn from model construction and results' analyses [386]. In other words, in order to transform data into a valuable actionable asset, individuals must engage in collaborative sense-making by sharing the context producing their findings, wherein context refers to sets of narrative stories around how data were processed, cleaned, modeled and analyzed. In this discourse we find also an interesting space for the adoption of XAI techniques due to their powerful ability to describe *black-box* models in an understandable, hence conveyable fashion towards colleagues from Social Science, Politics, Humanities and Legal fields.

XAI can effectively ease the process of explaining the reasons why a model reached a decision in an accessible way to non-expert users, i.e. the *rationale explanation*. This confluence of multi-disciplinary teams in projects related to Data Science and the search for methodologies to make them appraise the ethical implications of their data-based choices has been lately coined as Critical Data studies [387]. It is in this field where XAI can significantly boost the exchange of information among heterogeneous audiences about the knowledge learned by models.

2.4.8 *XAI and Theory-guided Data Science*

We envision an exciting synergy between the XAI realm and *Theory-guided Data Science*, a paradigm exposed in [388] that merges both Data Science and the classic theoretical principles underlying the application/context where data are produced. The rationale behind this rising paradigm is the need for data-based models to generate knowledge that is the prior knowledge brought by the field in which it operates. This means that the model type should be chosen according to the type of relations we intend to encounter. The structure should also follow what is previously known. Similarly, the training approach should not allow for the optimization process to enter regions that are not plausible. Accordingly, regularization terms should stand the prior premises of the field, avoiding the elimination of badly represented true relations for spurious and deceptive false relations. Finally, the output of the model should inform about everything the model has come to learn, allowing to reason and merge the new knowledge with what was already known in the field.

Many examples of the implementation of this approach are currently available with promising results. The studies in [389]-[396] were carried out in diverse fields, show-

casing the potential of this new paradigm for data science. Above all, it is relevant to notice the resemblance that all concepts and requirements of Theory-guided Data Science share with XAI. All the additions presented in [388] push toward techniques that would eventually render a model explainable, and furthermore, knowledge consistent. The concept of *knowledge from the beginning*, central to Theory-guided Data Science, must also consider how the knowledge captured by a model should be explained for assessing its compliance with theoretical principles known beforehand. This, again, opens a magnificent window of opportunity for XAI.

2.4.9 *Guidelines for ensuring Interpretable AI Models*

Recent surveys have emphasized on the multidisciplinary, inclusive nature of the process of making an AI-based model interpretable. Along this process, it is of utmost importance to scrutinize and take into proper account the interests, demands and requirements of all stakeholders interacting with the system to be explained. From the designers of the system to the decision makers consuming its produced outputs and users undergoing the consequences of decisions made therefrom.

Given the confluence of multiple criteria and the need for having the human in the loop, some attempts at establishing the procedural guidelines to implement and explain AI systems have been recently contributed. Among them, we pause at the thorough study in [397], which suggests that the incorporation and consideration of explainability in practical AI design and deployment workflows should comprise four major methodological steps:

1. Contextual factors, potential impacts and domain-specific needs must be taken into account when devising an approach to interpretability: These include a thorough understanding of the purpose for which the AI model is built, the complexity of explanations that are required by the audience, and the performance and interpretability levels of existing technology, models and methods. The latter pose a reference point for the AI system to be deployed in lieu thereof.

2. Interpretable techniques should be preferred when possible: when considering explainability in the development of an AI system, the decision of which XAI approach should be chosen should gauge domain-specific risks and needs, the available data resources and existing domain knowledge, and the suitability of the ML model to meet the requirements of the computational task to be addressed. It is in the confluence of these three design drivers where the guidelines postulated in [397] (and other studies in this same line of thinking [6]) recommend first the consideration of standard interpretable models rather than sophisticated yet opaque modeling methods. In practice, the aforementioned aspects (contextual factors, impacts and domain-specific needs) can make transparent models preferable over complex modeling alternatives whose interpretability require the application of post-hoc XAI techniques. By contrast, *black-box* models such as those reviewed in this chapter (namely, support vector machines, ensemble methods and neural networks) should be selected only when their superior modeling capabilities fit best the characteristics of the problem at hand.

3. If a *black-box* model has been chosen, the third guideline establishes that ethics-, fairness- and safety-related impacts should be weighed. Specifically, responsibility in the design and implementation of the AI system should be ensured by checking whether such identified impacts can be mitigated and counteracted by supplementing the system with XAI tools that provide the level of explainability required by the domain in which it is deployed. To this end, the third guideline suggests 1) a detailed articulation, examination and evaluation of the applicable explanatory strategies, 2) the analysis of whether the coverage and scope of the available explanatory approaches match the requirements of the domain and application context where the model is to be deployed; and 3) the formulation of an interpretability action plan that sets forth the explanation delivery strategy, including a detailed time frame for the execution of the plan, and a clearance of the roles and responsibilities of the team involved in the workflow.

4. Finally, the fourth guideline encourages to rethink interpretability in terms of the cognitive skills, capacities and limitations of the individual human. This is an important question on which studies on measures of explainability are intensively revolving by considering human mental models, the accessibility of the audience to vocabularies of explanatory outcomes, and other means to involve the expertise of the audience into the decision of what explanations should provide.

We foresee that the set of guidelines proposed in [397] and summarized above will be complemented and enriched further by future methodological studies, ultimately heading to a more *responsible* use of AI. Methodological principles ensure that the purpose for which explainability is pursued is met by bringing the manifold of requirements of all participants into the process, along with other universal aspects of equal relevance such as no discrimination, sustainability, privacy or accountability. A challenge remains in harnessing the potential of XAI to realize a *Responsible AI*, as we discuss in the next section.

2.5 SUMMARY

This chapter has revolved around eXplainable Artificial Intelligence (XAI), which has been identified in recent times as an utmost need for the adoption of ML methods in real-life applications. This first chapter has elaborated on this topic by first clarifying different concepts underlying model explainability, as well as by showing the diverse purposes that motivate the search for more interpretable ML methods. These conceptual remarks have served as a solid baseline for a systematic review of recent literature dealing with explainability, which has been approached from two different perspectives: 1) ML models that feature some degree of transparency, thereby interpretable to an extent by themselves; and 2) post-hoc XAI techniques devised to make ML models more interpretable. This literature analysis has yielded a global taxonomy of different proposals reported by the community, classifying them under uniform criteria. Given the prevalence of contributions dealing with the explainability of Deep Learning models, we have inspected in depth the literature dealing with this

family of models, giving rise to an alternative taxonomy that connects more closely with the specific domains in which explainability can be realized for Deep Learning models. Finally, this chapter has covered the most recognizable challenges still to be tackled. The following chapter will attempt to work towards the fulfillment of one of the gaps underlined in this chapter, namely, explainability for randomized neural networks. Specifically for Echo State Networks.

3

ON THE POST-HOC EXPLAINABILITY OF DEEP ECHO STATE NETWORKS FOR TIME SERIES FORECASTING AND IMAGE AND VIDEO CLASSIFICATION

Since their inception [398, 399], Echo State Networks (ESNs) have been frequently proposed as an efficient replacement for traditional Recurrent Neural Networks (RNNs). As opposed to conventional gradient-based RNN training, the recurrent part (reservoir) of ESNs is not updated via gradient backpropagation, it is simply initialized at random, given that certain mathematical properties are met. This randomized nature renders ESNs within the realm of randomized neural networks, along with random vector functional links, stochastic configuration networks and random features for kernel approximation [379, 380]. As a result, their training overhead for real life applications becomes much less computationally demanding than that of RNNs. Over the well-known Mackey-Glass chaotic time series prediction benchmark, ESN has been shown to improve the accuracy scores achieved by multi-layer perceptrons (MLPs), support vector machines (SVMs), backpropagation-based RNNs and other learning approaches by a factor of over 2000 [400]. These proven benefits have appointed ESN as a top contending dynamical model for performance and computational efficiency reasons when compared to other modeling counterparts.

Unfortunately, choosing the right parameters to initialize these reservoirs falls a bit on the side of luck and past experience of the scientist [401], and less on that of sound reasoning. As stated in [402] and often referred thereafter, the current approach for assessing whether a reservoir is suited for a particular task is to observe if it yields accurate results, either by handcrafting the values of the reservoir parameters or by automating their configuration via an external optimizer. All in all, this poses tough questions to address when developing an ESN for a certain application, since knowing whether the created structure is optimal for the problem at hand is not possible without actually training it. Furthermore, despite recent attempts made in this direction [403, 404], there is no clear consensus on how to guide the search for good reservoir based models.

Concerns in this matter go a step beyond the ones exposed above about the configuration of these models. Model design and development should orbit around a deep understanding of the multiple factors hidden below the surface of the model. For this purpose, a manifold of techniques have been proposed under the Explainable Artificial Intelligence (xAI) paradigm for easing the understanding of decisions issued by existing AI-based models. The information delivered by xAI techniques allow improving the design/configuration of AI models, extracting augmented knowledge about their outputs, accelerating debugging processes, or achieving a better outreach and adoption of this technology by non-specialized audience [405]. Although the activity in this research area has been vibrant for explaining many *black-box* machine learning models, there is no prior work on the development of techniques of this sort for dynamical approaches. The need for providing explanatory information about the

knowledge learned by ESNs remains unaddressed, even though recent advances on the construction of multi-layered reservoirs (Deep ESN [383]) that make these models more opaque than their single-layered counterparts have been proposed.

Given the above context and the prior information presented at Chapter 2, this part of the thesis takes a step ahead by presenting a novel suite of xAI techniques crafted to issue explanations of already trained Deep ESN models. Although these techniques are specifically designed for ESN explanation, they hardly fall under the paradigm of model specific techniques. However, they could be considered under model-family specific concerning dynamic or recurrent models. Within such a category, the methods proposed belong to Post-hoc Explainability. The proposed techniques elicit visual information that permit to assess the memory properties of these models, visualize their detected patterns over time, and analyze the importance of an individual input in the model's output. Mathematical definitions of the xAI factors involved in the tools of the proposed framework are given, complemented by examples that help illustrate their applicability and comprehensibility to the general audience. This introduction of the overall framework is complemented by several experiments showcasing the use of our xAI framework to three different real applications: 1) battery cell consumption prediction, 2) road traffic flow forecasting, 3) image classification and 4) video classification. The results are conclusive: the outputs of the proposed xAI techniques confirm, in a human-readable fashion, that the Deep ESN models capture temporal dependencies existing in data that could be expected due to prior knowledge, and that this captured information can be summarized to deepen the understanding of a general practitioner/user consuming the model's output. The novel ingredients of this chapter can be summarized as follows:

1. Three xAI methods for ESNs:
    - Potential memory, which permits to quantitatively assess the estimated memory retained in a trained ESN, and that can be extrapolated to any recursive model. This technique falls within *Visual Explanations* from Section 2.3.1.
    - Temporal patterns, which examines the correlative patterns captured by the ESN, and visualizes them to aid the training and explanation process. This tool allows checking whether the model captures from data what the practitioner could expect as per his/her expert knowledge or prior intuition told by the application at hand. This technique also falls within *Visual Explanations* from Section 2.3.1.
    - Pixel absence effect, which evaluates the effect of each dimension of the input instance on the output of the model. This improves the granularity of the analysis, uncovering which parts of the input are most influential for the prediction. This technique falls within amid *Visual Explanations* and *Feature Relevance* from Section 2.3.1.
2. The aforementioned tools are evaluated over data of distinct nature: time series, image and video data. The latter one (video) imply transforming images that flow over time into a multidimensional time series.

3. Explanations issued by the proposed techniques are proven not only to inform further future studies dealing with these recurrent randomized neural networks, but to also unveil important modeling issues (e.g. the presence of bias) that are often not easy to detect from sequential data.

The rest of this chapter is organized as follows: Section 3.1 provides the reader with the required background on ESN literature, and sets common mathematical grounds for the rest of the work. Section 3.2 introduces the framework and analyses each of the techniques proposed in its internal subsections. Section 3.3 presents the experiments designed to ensure the viability of this study with real data. Section 3.4 analyzes and discusses the obtained results. Finally Section 3.5 puts an end to the chapter by summarizing the contains of the chapter.

3.1 RELATED WORK

Before proceeding with the description of the proposed suite of xAI techniques, this section briefly revisits the fundamentals of ESN and Deep ESN models (Subsection 3.1.1), notable advances in the explainability of recurrent neural networks and models for time series (Subsection 3.1.2), and techniques used for quantifying the importance of features (Subsection 3.1.3).

3.1.1 *Echo State Network: Fundamentals*

In 2001, Wolfgang Maass and Herbert Jaeger independently introduced Liquid State Machines [406] and ESNs [407], respectively. The combination of these studies with research on computational neuroscience and machine learning [408, 409] brought up the field of Reservoir Computing. Methods belonging to this field consist of a set of sparsely connected, recurrent neurons capable of mapping high-dimensional sequential data to a low-dimensional space, over which a learning model can be trained to capture patterns that relate this low-dimensional space to a target output. This simple yet effective modeling strategy has been harnessed for regression and classification tasks in a diversity of applications, such as road traffic forecasting [410], human recognition [411] or smart grids [412], among others [383].

Besides their competitive modeling performance in terms of accuracy/error, Reservoir Computing models are characterized by a less computationally demanding training process than other recursive models: in these systems, only the learner mapping the output of the reservoir to the target variable of interest needs to be trained. Neurons composing the reservoir are initialized at random under some stability constraints. This alternative not only alleviates the computational complexity of recurrent neural networks, but also circumvents one of the downsides of gradient backpropagation, namely, exploding and vanishing gradients.

To support the subsequent explanation of the proposed xAI techniques, we now define mathematically the internals and procedure followed by ESNs to learn a mapping from a K-dimensional input $\mathbf{u}(t)$ (with t denoting index within the sequence) to

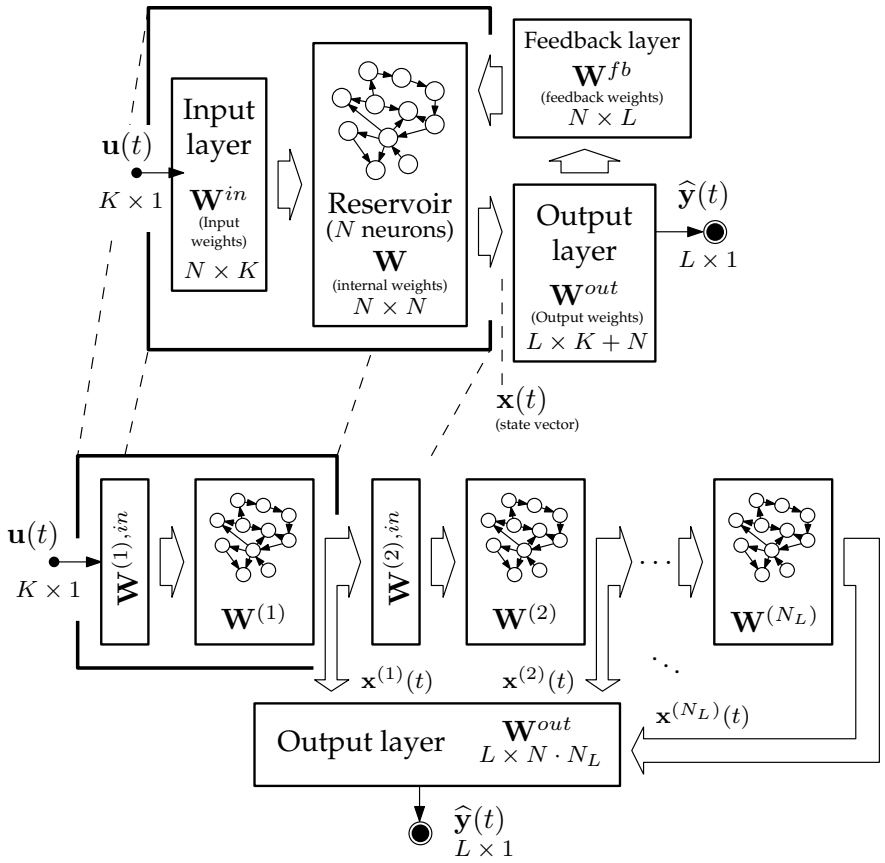


Figure 3.1: Schematic diagram showing a canonical ESN (upper plot), and the multi-layered stacking architecture of a Deep ESN model (below).

a L-dimensional output $\mathbf{y}(t)$ by a reservoir of N neurons. Following Figure 3.1, the reservoir state is updated as:

$$\mathbf{x}(t+1) = \\ \alpha f(\mathbf{W}^{in}\mathbf{u}(t+1) + \mathbf{W}\mathbf{x}(t) + \mathbf{W}^{fb}\mathbf{y}(t)) + (1-\alpha)\mathbf{x}(t), \tag{1}$$

where $\mathbf{x}(t)$ denotes the N-sized state vector of the reservoir, $f(\cdot)$ denotes an activation function, $\mathbf{W}_{N\times N}$ is the matrix of internal weights, $\mathbf{W}^{in}_{N\times K}$ are the input connection weights, and $\mathbf{W}^{fb}_{N\times L}$ is a feedback connection matrix. Parameter $\alpha \in \mathbb{R}(0,1]$ denotes the *leaking rate*, which allows to set different learning dynamics in the above recurrence [413]. The output of the ESN at index t can be computed once the state of the reservoir has been updated, yielding:

$$\widehat{\mathbf{y}}(t) = g\left(\mathbf{W}^{out}[\mathbf{x}(t);\mathbf{u}(t)]\right) = g(\mathbf{W}^{out}\mathbf{z}(t)), \tag{2}$$

where $[\mathbf{a};\mathbf{b}]$ is a concatenation operator between vectors $\mathbf{a}$ and $\mathbf{b}$, $\mathbf{W}^{out}_{L\times(K+N)}$ is a matrix containing the output weights, and $g(\cdot)$ denotes an activation function. Weights belonging to the aforementioned matrices can be adjusted as per a training data-set with examples $\{(\mathbf{u}(t),\mathbf{y}(t))\}$. However, as opposed to other recurrent neural networks, not all weight inside the matrices $\mathbf{W}^{in}$, $\mathbf{W}$, $\mathbf{W}^{fb}$ and $\mathbf{W}^{out}$ are adjusted. Instead, the weight values of input, hidden state, and feedback matrices are drawn initially at random, whereas those of the output matrix $\mathbf{W}^{out}$ are the only ones tuned during

the training phase, using Moore-Penrose pseudo-inversion (for classification) or regularized least squares (regression). In this latter case:

$$\min_{\mathbf{w}_l^{out}} \sum_{t=1}^{T} \left(\sum_{j=1}^{K+N} w_{l,j}^{out} \cdot z_j(t) - y_l(t) \right)^2 + \lambda \left\| \mathbf{w}_l^{out} \right\|_2^2, \tag{3}$$

where $l \in \{1, \ldots, L\}$; $\mathbf{w}_l^{out} \in \mathbb{R}^{K+N}$; $\|\cdot\|_2$ denotes L_2 norm; $z_j(t)$ is the j-th entry of $\mathbf{z}(t)$; and $\mathbf{w}_l^{out} = [w_{l,1}^{out}, ..., w_{l,K+N}^{out}]^T$ denotes the l-th row of $\mathbf{W}^{out}$; and $\lambda \in \mathbb{R}[0, \infty)$ permits to adjust the importance of the L_2 regularization term in the minimization.

More recently, a multi-layered architecture based on the leaky ESN model described above was introduced in [383]. In essence, the Deep ESN model embodies a stacking ensemble of N_L reservoirs, set one after the other, forming a hierarchically layered series of reservoirs. As a result of this concatenation, the state vector of the global Deep ESN model is given by $[\mathbf{x}^{(1)}(t); ...; \mathbf{x}^{(N_L)}(t)] \doteq [\mathbf{x}^{(l)}(t)]_{l=1}^{N_L}$, which can be conceived as a multi-scale representation of the input $\mathbf{u}(t)$. It is this property, together with other advantages such as a lower computational burden of their training algorithm and the enriched reservoir dynamics of stacked reservoirs [414], what lends Deep ESNs a competitive modeling performance when compared to other RNNs.

As shown in Figure 3.1, in what follows (l) indicates that the parameter featuring this index belongs to the l-th layer of the Deep ESN. The first stacked ESN is hence updated as:

$$\begin{aligned} \mathbf{x}^{(1)}(t+1) &= \left(1 - \alpha^{(1)}\right) \mathbf{x}^{(1)}(t) + \\ &\alpha^{(1)} f\left(\mathbf{W}^{(1)} \mathbf{x}^{(1)}(t) + \mathbf{W}^{(1),in} \mathbf{u}^{(1)}(t+1)\right), \end{aligned} \tag{4}$$

whereas the recurrence in layers $l \in \{2, \ldots, N_L\}$ is given by:

$$\begin{aligned} \mathbf{x}^{(l)}(t+1) &= \left(1 - \alpha^{(l)}\right) \mathbf{x}^{(l)}(t) + \\ &\alpha^{(l)} f\left(\mathbf{W}^{(l)} \mathbf{x}^{(l)}(t) + \mathbf{W}^{(l),in} \mathbf{x}^{(l-1)}(t+1)\right), \end{aligned} \tag{5}$$

and the Deep ESN output yields as:

$$\widehat{\mathbf{y}}(t) = g\left(\mathbf{W}^{out}[\mathbf{x}^{(1)}(t); \ldots; \mathbf{x}^{(N_L)}(t)]\right), \tag{6}$$

where $\mathbf{W}^{out}_{L \times (N_L \cdot N)}$ is the weight matrix that maps the concatenation of state vectors of the stacked reservoirs to the target output. Analogously to canonical single-layer ESN, weights in $\mathbf{W}^{(l)}$ and $\mathbf{W}^{(l),in}$ for each layer $l = 1, \ldots, N_L$ are initialized at random and re-scaled to fulfill the so-called Echo State Property [415], i.e.:

$$\max_{l \in \{1, \ldots, N_L\}} \rho\left((1 - \alpha^{(l)})\mathbf{I} + \alpha^{(l)} \mathbf{W}^{(l)}\right) < \rho_{max}, \tag{7}$$

with $\rho(\cdot)$ denoting the largest absolute eigenvalue of the matrix set at its argument, and $\rho_{max} < 1$ the so-called *spectral radius* of the model. Once $\mathbf{W}^{(l)}$ and $\mathbf{W}^{(l),in}$ have been set at random fulfilling the above property, they are kept fixed for the rest of the training process, whereas weights in $\mathbf{W}^{out}$ are tuned over the training data by

means of regularized least-squares regression as per Expression (3) (or any other low-complexity learning model alike). Although it has not been covered in the literature, the readout layer can be designed, selected or tailored as per the needs of the problem or task at hand. In this work, multiple readout layers are considered to tackle multi-class (random forests) and regression problems (linear regression).

Despite the simplicity and computational efficiency of the ESN and Deep ESN training process, the composition of the network itself – namely, the selection of the number of layers and the value of hyper-parameters such as $\alpha^{(l)}$ is a matter of study that remain so far without a clear answer [402, 407, 416]. Some automated approaches have been lately proposed, either relying on the study of the frequency spectrum of concatenated reservoirs [417] or by means of heuristic wrappers [418]. However, no previous work can be found on explainability measures that can be drawn from an already trained Deep ESN to elucidate diverse properties of its captured knowledge. When the audience for which such measures are produced is embodied by machine learning experts, the suite of xAI techniques can help them discern what they observe at its input, quantify relevant features (e.g. its memory depth) and thereby, ease the process of configuring them properly for the task at hand.

3.1.2 *Explainability of Recurrent Neural Networks*

This subsection dives deeper than Chapter 2 into the literature of explainability of RNN. Paying a closer attention to the most relevant concepts needed for this specific chapter. Several application domains have traditionally shown a harsh reluctance against embracing the latest advances in machine learning due to the opaque, *black-box* nature of models emerging over the years. This growing concern with the need for explaining *black-box* models has been mainly showcased in Deep Learning models for image classification, wherein explanations produced by xAI techniques can be inspected easily. However, the explainability of models developed for sequence data (e.g. time series) has also been studied at a significantly smaller corpus of literature. A notable milestone is the work in [419], where a post-hoc xAI technique originally developed for image classification was adopted for LSTM architectures, thereby allowing for the generation of local explanations in sequential data modeling and regression. Stimulated in part by this work, several contributions have hitherto proposed different approaches for the explainability of recurrent neural networks, including gradient-based approaches [420, 421], ablation-based estimations of the relevance of particular sequence variables for the predicted output [422, 423], or the partial linearization of the activations through the network, permitting to isolate the contribution at different time scales to the output [424].

In addition to standard Deep Learning approaches resorting to gradient backpropagation [425], time series data are also processed by means of other modeling approaches, mostly traditional data mining methods for time series analysis. By moving away from Deep Neural Networks, further possibilities emerge for achieving interpretable methods by design, which allow for a better understandability of its inner working by the audience without requiring external tools for the purpose [426]. Among them, Symbolic Aggregate Approximation (SAX) [427, 428] and Fuzzy Logic appear to be the best contenders. SAX works by first transforming the time series

into its piece-wise aggregate approximation [429]. Then the resulting transformation is converted into a string of symbols assigned equiprobably among the regions of the discretization. Finally the obtained representation permits a clear visualization of the recurrent patterns hand in hand with the stream-like implementation with minimal overheads. Fuzzy logic, fuzzy sets and computing with words [430–433] are methods that improve the interpretability of otherwise hard-to-understand systems, by bringing them closer to how human reasoning operate when inspecting multiple dimensions over time. By transforming continuous values into meaningful levels, fuzzy computing is able to tackle many types of problems without hindering the interpretability of data and operations held over them through the model of choice.

Despite the numerous attempts made recently at explaining dynamic systems, most of them are fabricated in a ad-hoc manner, thus leaving aside other flavors of time-series and recurrent neural computation whose intrinsically abstract nature also calls for studies on their explainability. Indeed, this noted lack of proposals to elicit explanations for alternative models is in close agreement with the conclusions and prospects made in comprehensive surveys on xAI [405, 434]. This is the purpose of the set of techniques presented in the following section, to provide different interpretable indicators of the properties of a trained Deep ESN model, as well as information that permit to visualize and summarize the knowledge captured by its reservoirs.

3.1.3 *Importance Attribution Methods*

Due to one of the techniques in the proposed framework (Pixel absence effect), it is compulsory that importance attribution methods are discussed in this background. Importance attribution methods were introduced in Chapter 2 under the umbrella of feature relevance explanation methods. Importance attribution methods are not exclusive to neural networks or randomized neural networks for that matter. There has been extended use of this concept for Tree Ensembles and Support Vector Machines [435, 436] constructing a well developed body of research around these techniques. The assessment of the validity of a machine learning model's output is paramount for a well structured system validation/improvement workflow. Feature attribution methods intend on giving interpretability cues by measuring the importance each feature has in the final prediction outcome. Many techniques have been proposed in the last few years to gauge the importance of features in different learning tasks [146, 297, 299, 437–443], yet none of them has been utilized with the recurrent randomized neural networks that are at the core of this study.

3.2 PROPOSED FRAMEWORK

The framework proposed in this work is composed by three xAI techniques, each tackling some of the most common aspects that arise when training an ESN model or understanding their captured knowledge. These techniques cover three main characteristics that help understand strengths and weaknesses of these models:

1. *Potential Memory*, which is a simple test that closely relates to the amount of sequential memory retained within the network. The potential memory can be quantified by addressing how the model behaves when its input fades abruptly. The intuition behind this concept recalls to the notion of *stored information*, i.e., the time needed by the system to return to its resting state.

2. *Temporal Patterns*, which target the visualization of patterns occurring inside the system by means of *recurrence plots* [444, 445]. Temporal patterns permit to examine the multiple scales at which patterns are captured by the Deep ESN model through their layers, easing the inspection of the knowledge within the data that is retained by the neuron reservoir. This technique also helps determine when the addition of a layer does not contribute to the predictive performance of the overall model.

3. *Pixel Absence Effect*, which leverages the concept of *local explanations* [405] and extends it to ESN and Deep ESN models. This technique computes the prediction difference resulting from the suppression of one of the inputs of the model. The generated matrix of deviations will be contained within the dimensions of the input signal, allowing for the evaluation of the importance of the absence of a single component over the input signal.

The following subsections provide rationale for the above set of xAI techniques, their potential uses, and examples with synthetic data previewing the explanatory information they produce.

3.2.1 *Potential Memory*

One of the main concerns when training an ESN model is to evaluate the potential memory the network retains within their reservoir(s). Quantifying this information becomes essential to ensure that the model possesses enough modeling complexity to capture the patterns that best correlate with the target to be predicted.

Intuitively, a reservoir is able to hold information to interact with the newest inputs being fed to the network in order to generate the current output. It follows that, upon the removal of these new inputs, the network should keep outputting the information retained in the reservoir, until it is flushed out. The multi-layered structure of the network does not allow relating the flow through the Deep ESN model directly to its memory, since the state in which this memory is conserved would most probably reside in an unknown abstract space. This is the purpose of the proposed potential memory indicator: to elicit a hint of the memory contained in the stacked reservoirs. When informed to the audience, the potential memory can improve the confidence of the user with respect to the application at hand, especially in those cases where a hypothesis on the memory that the model should possess can be drawn from intuition. For instance, in an order-ten nonlinear auto-regressive-moving average (NARMA) system, the model should be able to store at least ten past steps to produce a forecast for the next time step.

Before proceeding further, it is important to note that a high potential memory is not sufficient for the model to produce accurate predictions for the task under

consideration. However, it is a necessary condition to produce estimations based on inputs further away than the number of sequence steps upper bounded by this indicator.

For the sake of simplicity, let us assume a single-layer ESN model without leaking rate ($\alpha = 1$) nor feedback connection (i.e. $\mathbf{W}^{fb} = \mathbf{o}_{N\times L}$). These assumptions simplify the recurrence in Expression (1) to:

$$\mathbf{x}(t+1) = f(\mathbf{W}^{in}\mathbf{u}(t+1) + \mathbf{W}\mathbf{x}(t)), \tag{8}$$

which can be seen as an expansion of the current input $\mathbf{u}(t+1)$ and its history represented by $\mathbf{x}(t)$. If we further assume that the network is initially[1] set to $\widehat{\mathbf{y}}(0) = \mathbf{o}$ when $\mathbf{u}(0) = \mathbf{o}$, the system should evolve gradually to the same state when the input signal $\mathbf{u}(t)$ is set to $\mathbf{o}$ at time $t = T_0$. The time taken for the model to return to its initial state is the potential memory of the network, representing the information of the history kept by the model after training. Mathematically: [Potential memory] Given an already trained ESN model with parameters

$$\{\alpha^{(l)}, \mathbf{W}^{(l),in}, \mathbf{W}^{(l)}, \mathbf{W}^{fb}, \mathbf{W}^{out}\},$$

and initial state set to $\widehat{\mathbf{y}}(0) = \mathbf{y}_0 \in \mathbb{R}^L$ when $\mathbf{u}(0) = \mathbf{o}$, the potential memory (PM) of the model at evaluation time T_0 is given by:

$$PM(T_0) = T_0 - _{t>T_0} \|\widehat{\mathbf{y}}(t) - \mathbf{y}_0\|_2 < \epsilon, \tag{9}$$

where ϵ is the tolerance below which convergence of the measure is declared.

In order to illustrate the output of this first technique, we consider a single ESN model with varying number of neurons in its reservoir to learn the phase difference of 50 samples between an input and an output sinusoid. Plots nested in Figure 3.2 depict the fade dynamics of different ESN models trained with $N = 5$ (plots 3.2.a and 3.2.b), $N = 50$ (plots 3.2.c and 3.2.d) and $N = 500$ neurons (plots 3.2.e and 3.2.f) in their reservoirs. Specifically, the graph on the top represents the actual signal to be predicted, while the bottom graphs display the behavior of the output of each ESN model when the input signal is zeroed at time $T_0 = 10100$, along with the empirical distribution of the potential memory computed for $T_0 \in (10000, 10100]$.

When the reservoir is composed by just $N = 5$ neurons, the potential memory of the network is low given the quick transition of the output signal to its initial state after the input is set to $\mathbf{o}$. When increasing the size of the reservoir to 50 neurons, a rather different behavior is noted in the output of the ESN model when flushing its contained knowledge, not reaching its steady state until 20 time steps for the example depicted in Figure 3.2.c. Finally, an ESN comprising 500 neural units in its reservoir does not come with a significant increase of its potential memory, however, the model is able to compute the target output in a more precise fashion with respect to the previous case. This simple experiment shows that the potential memory is a useful metric to evaluate the past range of the sequence modeled and retained at its input. As later exposed in Section 3.4, this technique is of great help when fitting a model, since only when the potential memory is large enough, does the model succeed on the testing.

1 For clarity any consideration to the bias term is avoided in this statement.

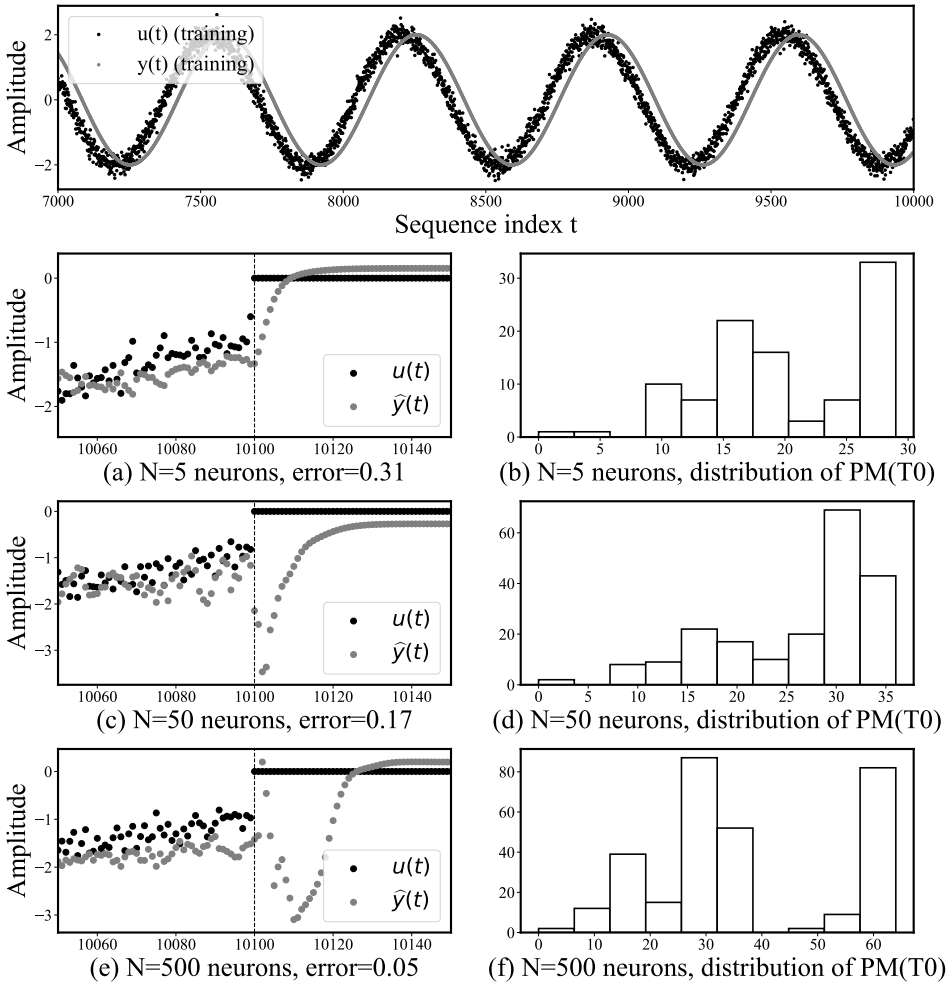


Figure 3.2: (Top) Input and target sinusoids used for training an ESN model with $\alpha = 0.5$ and a single reservoir with $N \in \{5, 50, 500\}$ neurons. In the rest of plots, reservoir memory dynamics when the input signal is set to $\mathbf{0}$ at $t = 10100$, along with the histogram of $PM(T_0)$ values obtained for $T_0 \in (10000, 10100]$ and $\epsilon = 0.05$.

3.2.2 *Temporal Patterns*

One important aspect when designing and building a model is to ascertain whether it has been able to capture the temporal patterns observed in the dynamics of the data being modeled. To shed light on this matter, the devised *temporal patterns* technique resorts to a well-established tool for the analysis of dynamical systems: Recurrence Plots [444, 445]. This component of the suite aims at tackling two main problems: 1) to determine whether the model has captured any temporal patterns; and 2) if the depth of the model (as per its number of stacked layers) is adding new knowledge that contributes to the predictive performance of the model.

Before proceeding further with the mathematical basis of this technique, we pause at some further motivating intuition behind temporal patterns. Given the *black-box* nature of neural networks, their training often brings up the question whether the model is capturing the patterns expected to faithfully model the provided data. From logic it can be deduced that, the deeper the network is, the more detail it will be able

to hold on to. However, this does not always result in a better predictive model. Since any given layer feeds on the previous layer's high dimension representation (with the exception of the first layer), the patterns found in these latter representations should potentially be more intricate, since their input is already more detailed.

This is best understood from a simile with the process of examining a footprint. The human cortex has much definition to extract information from the visualized footprint. However, when staring at an already enlarged image, by the effect of a magnifying glass, the human visual cortex can pick on features they could not detect before, hence obtaining a similar effect to that of layering multiple reservoirs. The depth of the network, as well as the magnification optics, have to be appropriately sized for the task: there is no point in searching for a car using a microscope, as there is no point on using a deep structure to model a simple phenomenon.

Following the findings from [383, 446] in which their authors already found how Deep ESN architectures developed multiple time-scale representations that grew with layering, it seems of utmost importance from an explainability perspective to dispose of a tool that will help us inspect this property in already trained models. For this purpose, recurrence plots are used to analyze the temporal patterns found within the different layers of the network. When the layers capture valuable information, their recurrence plots show more focused patterns than their previous layers. This accounts for the fact that each layer is able to better focus on patterns, leaving aside noise present in data. When this growing detail in the recurrence plots disappears at a point of the hierarchy of reservoirs, adding more layers seem counterproductive and does not potentially yield any modeling gain whatsoever.

Mathematically we embrace the definition of recurrence plots in [444]. Specifically, the recurrence plot $\mathbf{R}_{t,t'}^{(l,l')}$ between input $\mathbf{x}^{(l)}(t)$ and $\mathbf{x}^{(l')}(t')$ is given by:

$$\mathbf{R}_{t,t'}^{(l,l')} = \begin{cases} 1 & \text{if } \left\| \mathbf{x}^{(l)}(t) - \mathbf{x}^{(l')}(t') \right\|_2 < \psi, \\ 0 & \text{otherwise,} \end{cases} \tag{10}$$

i.e. as a $N \times N$ binary matrix in which value 1 is set for those indices corresponding to time steps (t, t') where the hidden state vectors of the reservoirs l and l' are equal up to an error ψ. Obviously, the case when $l = 1$ (i.e. the first signal is the input data $\mathbf{u}(t)$), dimensions of the recurrence plot can be enlarged to cover all its length. Likewise, the case with heterogeneously sized reservoirs also fits in the above definition by simply adjusting accordingly the size of the matrix. For our particular case, recurrence plots run for each of the states of a certain layer pair l, l' throughout the test history portrait the repetitions in phase space that happened throughout a given time window.

Figure 3.3 exemplifies the output of the recurrence plots when applied to the output layers of an ESN trained to map a noisy sinusoidal signal with an increasing additive trend to its noiseless version. The first thing that can be extracted from the figure is that the system being modeled is non-stationary (for the inspected scope). As shown in the signal and the recurrent plots, both present an upward slope. The second property to notice is the periodicity of the plots. By observing the processed and predicted signal (processed-gray, predicted-red), one may note that the four oscillations are captured by the model, and that they are mostly symmetric. A last feature

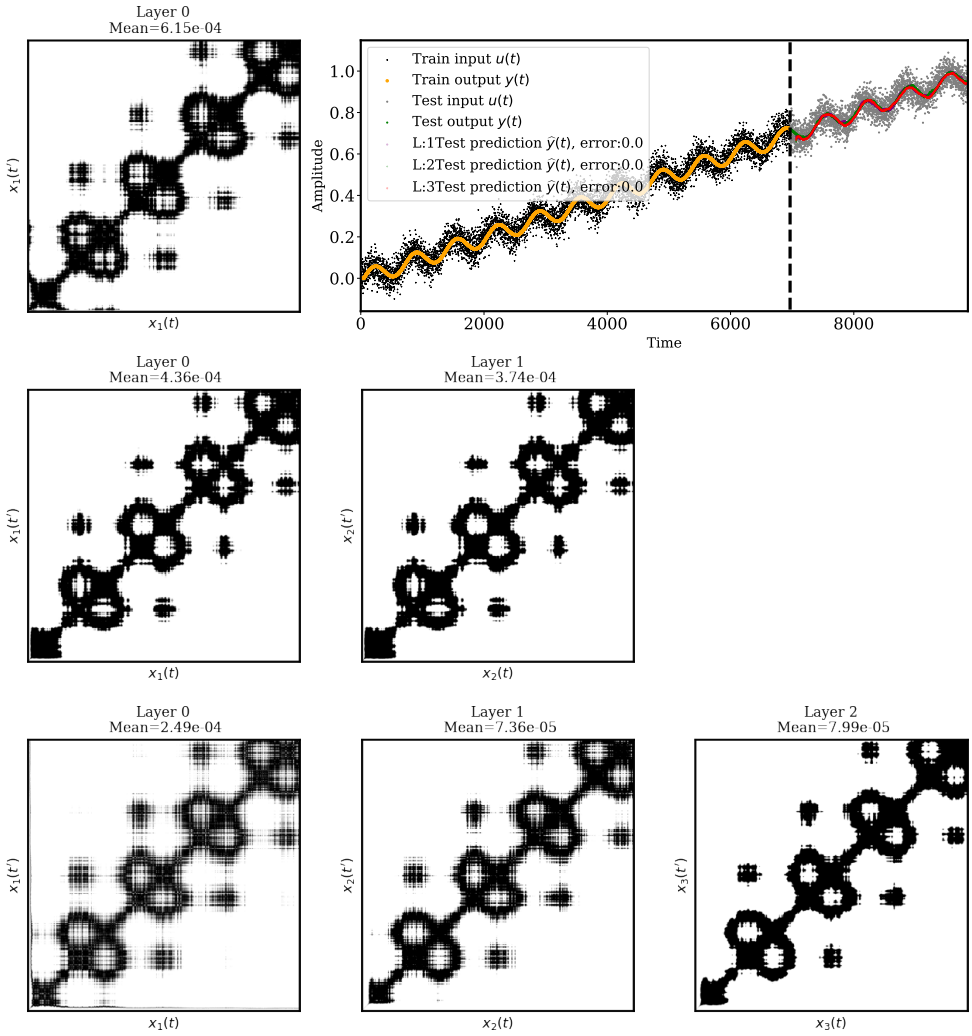


Figure 3.3: Representation of an ESN model trained to model a trended sinusoid by means of recurrence plots with three different layer configurations (1 layer: top, 2 layers: middle, 3 layers: bottom), displaying the evolution of the patterns seen on the recurrence plots along with an indication of the mean of the recurrence plots.

shown by the model is that the signal is noisy, hence those dots close to the valleys, meaning that there are agglomerations of readings in those spots that a clean signal would not have. The figure clearly shows that the recurrent behavior of the signal can be inspected on the recurrence plots of its hidden states. This type of analysis helps detect the properties of the phenomena producing the data under analysis: stationary or non-stationary nature, periodicity, trend, strong fluctuations, similar/inverse evolution through time, static states and similarity between states but at different rates. The extraction of such features from the recurrence plot perfectly couples with the notion of deepening our understanding of these models giving a further insight on the temporal patterns happening within the data.

This tool serves a two-way purpose that will be further discussed in the experiment section. First, it allows inspecting whether the model is actually capturing the features that could be expected from the system; secondly, once these features are captured

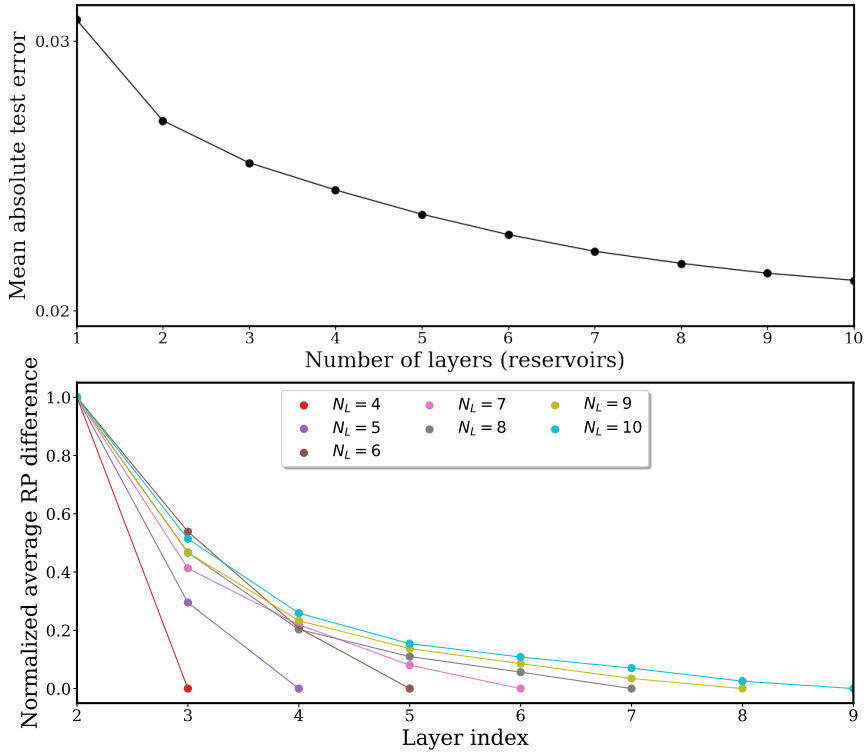


Figure 3.4: Mean absolute error as a function of the number of reservoirs N_L (top), and evolution of the average difference between recurrence plots corresponding to layers $l-1$ and $l+1$ for $N_L \in \{4,\ldots,10\}$ (bottom). There is a clear positive correlation between the test error and the RP difference, suggesting that performance improvements occur whenever subsequent reservoirs yield a more detailed recurrence plot.

correctly, it will allow for a better understanding of the system's dynamics through time. For example, the above image 3.3 clearly shows in each best configuration (two layers) that the model is clearly capturing the system non-stationary condition along with its low amplitude fluctuations that present a recurrent behavior through time. A further analysis of this tool when applied to real data will be covered in Section 3.4.

A byproduct derived from this tool is a numerical score that determines whether an added layer is contributing to the modeling process. A preliminary experiment is devised to support the claim that the *sharpness* of a certain layer's recurrent plot adds to the validity of such layer's addition to the predictive power of the model. Given input and output signals, an ESN of equal characteristics is run for a number of times and each number of layers. For each layer's recurrent plot $R^{l,l'}_{t,t'}$, its average value is computed, which relates to how sharp the recurrence plot is. Results in Figure 3.4 run for the example in Figure 3.3 reveals that the average test error computed over 50 trials correlates with the decrease of the average difference between recurrence plots, which validates the intuition about the relative usefulness of every layer. In summary, a clear positive correlation exists between the decrease in recurrent pattern's average and prediction error rate, which means that testing whether the average of the subsequent recurrent plots is decreasing could be of use when determining the contribution of adding the layer to the ESN.

3.2.3 *Pixel Absence Effect*

Finally, we delve into the third technique that comprises the proposed xAI suite for ESN-based models. The explanation departs from the thought that the analysis of a predictive model always brings up the question of whether an input is causing any effect on the model's prediction. Naturally, the goal is to find out how the inputs deemed to be the most important ones for the task at hand are actually the ones on which the model focuses most when eliciting its predictions. However, discovering whether this is actually true in the trained model is not an easy task.

Intuitively, if a step back is taken from the architecture to simply stare at the difference caused by the cancellation of a certain input, it should be possible to observe the individual influence of such input for the issued prediction. Similarly, inputs could be canceled in groups to evaluate the importance of each neighboring region of the input space. Given that the result of this technique can be laid directly over the input of the model, the results can be readily read and understood by any type of user, disregarding his/her technical background. This capability makes this xAI approach promising from an explainability standpoint. Indeed, the term *pixel* in the name of the technique comes purposely from the noted suitability of the technique to be *depicted* along with the input data, applying naturally when dealing with image or video data. However, this importance attribution technique can be applied to other input data flavors (e.g. time series) without requiring any modification of its steps.

Before delving further into this technique, it is of paramount importance to understand that the transformation of an image into a time series is needed to harness the modeling potential of ESN models. For this, the transformation presented by [447] is adopted: in accordance with Figure 3.5, an gray-scale (RGB) image with dimensions $W \times H$ (pixels) can be represented as a $W \cdot H$-length time series $u(t)$ (corr. $\mathbf{u}(t)$) by taking its rows/columns and concatenating them serially. For this work, images were transformed in a row basis, which means that for W rows, each of the pixels in the given row are left as series of pixels, stitching the last pixel of the given row with the first pixel of the next, until every row has been concatenated. Implicitly, this transformation is not unique to image data, since what is being processed is actually a time series. However, this technique is adopted over image data since it allows reverting all the computed importance levels back to the image without having to quantify the relevance of this attribution over time.

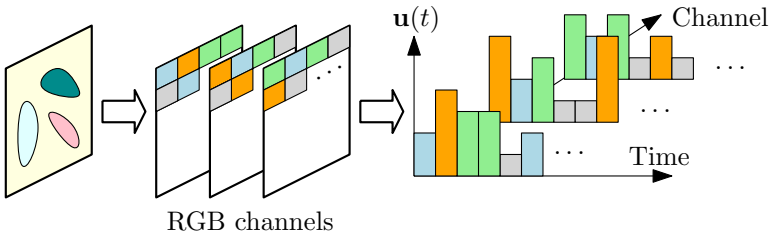


Figure 3.5: Schematic diagram showing the process of converting an RGB image to a $K = 3$ time series $\mathbf{u}(t)$.

In accordance with the previous notation, this technique is now mathematically described starting from the input-output a single-layered ESN model in Expression (2), which is included below for convenience:

$$\widehat{\mathbf{y}}(t) = g\left(\mathbf{W}^{out}[\mathbf{x}(t); \mathbf{u}(t)]\right) = g(\mathbf{W}^{out}\mathbf{z}(t)). \tag{11}$$

The cancellation of a certain point $T_{\boxtimes}$ in the input sequence $\mathbf{u}(t)$ would affect not only in the sequence itself, but also in future predictions $\widehat{\mathbf{y}}(t)$ $\forall t > T_{\boxtimes}$ due to the propagation of the altered value throughout the reservoir. Namely:

$$\widehat{\mathbf{y}}^{\boxtimes}(t) = g\left(\mathbf{W}^{out}[\mathbf{x}^{\boxtimes}(t); \mathbf{u}^{\boxtimes}(t)]\right), \tag{12}$$

where $\mathbf{u}^{\boxtimes}(t)$ denotes the input signal with $\mathbf{u}(T_{\boxtimes}) = \mathbf{0}_{K\times 1}$, and $\widehat{\mathbf{y}}^{\boxtimes}(t)$ results from applying recurrence (1) to this altered signal. The intensity of the effect of the suppression on the output signal at time $T_{\boxtimes}$ can be quantified as:

$$\mathbf{e}(t; T_{\boxtimes}) = [e_l(t; T_{\boxtimes})]_{l=1}^{L} = \widehat{\mathbf{y}}^{\boxtimes}(t) - \widehat{\mathbf{y}}(t), \ \forall t \geqslant T_{\boxtimes}. \tag{13}$$

where $\mathbf{e}(t; T_{\boxtimes}) \in \mathbb{R}^{L\times 1}$, and its sign indicates the direction in which the output is pushed as per the modification of the input. Clearly, when dealing with classification tasks (i.e. $\mathbf{y}(t) \in \mathbb{N}^{L\times 1}$), a similar rationale can be followed by conceiving the output of the ESN-based model as the class probabilities elicited for the input sequence. Therefore, the intensity computed as per (13) denotes whether the modification of the input sequence *increases* ($e_l(t; T_{\boxtimes}) > 0$) or *decreases* ($e_l(t; T_{\boxtimes}) < 0$) the probability associated to class $l \in \{1, \ldots, L\}$.

At this point it is important to remark that, given that this third xAI tool derives from the perturbation of individual pixels, possible correlations between pixels over the image are neglected, thereby quantifying importances that do not consider the plausibility of such perturbations. To solve this matter, the pixel absence analysis can be extended by boostrapping: pixels to be altered are sampled by boostrapping (in a sample size N_p established beforehand, e.g. ten at a time), for a total of N_b boostraps. The resulting probability changes are accumulated for each given pixel, giving rise to an enhanced quantification of the overall importance of every pixel, considering the effect of pixel-to-pixel interactions on the output of the ESN model. The description of this technique is summarized in the Pseudocode 1.

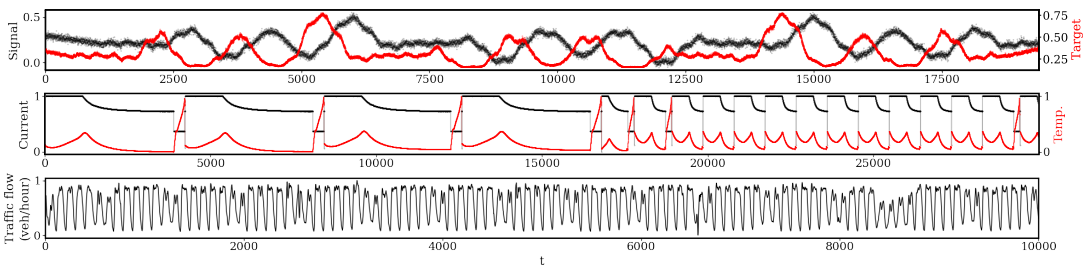


Figure 3.6: Segments of the different time series datasets used in the experiments: (top) input $u(t)$ (black) and output $y(t)$ (red) of a NARMA model of order 10; (middle) electrical current (black) and temperature (red) from a battery; and (bottom) road traffic flow (vehicles per hour) collected over time.

Algorithmus 1 : Bootstrap pixel absence effect

Data : Input series $\mathbf{u}(t)$ (e.g. coming from an image-to-time-series transformation), output $\hat{\mathbf{y}}(t)$ of the model at time t, time $T_{\boxtimes}$ at which the xAI tool is queried.

Result : Vector $\mathbf{e}(t; T_{\boxtimes})$ of accumulated relative output probability differences

1 **for** $n_b = 1, \ldots, N_b$ *(number of bootstraps)* **do**
2 Sample uniformly at random N_p positions from $\mathbf{u}(t)$ (sampled positions)
3 Cancel sampled positions in $\mathbf{u}(t)$ by interpolating their values as per their closest neighboring points in the time series
4 Compute ESN output $\hat{y}^{\boxtimes}(t)$ for perturbed input
5 Measure difference between old and new (perturbed) output as per Expression (13)
6 Update vector $\mathbf{e}(t; T_{\boxtimes})$ by accumulating differences at the sampled positions

3.2.4 *Benefits of the framework for Different Audiences*

All the xAI tools composing the proposed framework yield different insights and uses depending on the audience to which the issued explanations are presented. This section elaborates further on the benefits that such explanations can entail to different user profiles.

To begin with, the output of the *potential memory* tool can be thought to be a quantitative measure of the amount of information retained in the ESN reservoirs through time. For a data scientist, the size of the ESN's reservoir(s) is arguably the first parameter to be tuned when designing ESN architectures. This being said, this parameter has to be solved before focusing on other parameters. By being informed about the potential memory of the model, a data scientist can ascertain whether the reservoir is large enough for the application at hand whenever intuition and prior expert knowledge can give a hint of the minimum memory that an ESN should possess. A similar reasoning applies to the case of a general user without a background on machine learning: prior intuition and knowledge can tell whether an ESN can realistically model a pattern over time, increasing the confidence of the user on the model's outcomes.

In what refers to *recurrent patterns*, a data scientist can harness its generated recurrence plot to further validate the model's efficacy by matching the features found in the raw data with the patterns encountered in its recurrent patterns, including stationarity, recurrence, and other time-dependent properties alike. Likewise, a general user can discover information about the behavior of the system being modeled, since recurrent patterns showcase certain properties that, once identified from the information displayed in the recurrent plots, are easy to detect from data.

Finally, *pixel absence* gives further insights about the model's behavior, and allows a data scientist to tune the parameters of the model in a more effective manner. Most interestingly, the output of this tool helps discover biases hidden in data and propagated to the model's output through the architecture. When the computed feature importances are shown to a general user, they support the fast interpretation of the

most relevant parts of the input sequences with respect to the prediction, and the verification that such influential parts conform to prior knowledge of the user.

3.3 EXPERIMENTAL SETUP

We assess the performance of the set of proposed techniques described previously by means of several computer experiments where different ESN-based models are used to model datasets of diverse nature. All experiments have been run several times before showing the results, verifying that explanations issued by the developed xAI techniques are stable and do not vary under different randomized values of the reservoirs' parameters. As such, experiments are divided in three use cases:

- To begin with, experiments deal with the most explored nature of data in ESN modeling: time series. When dealing with tasks formulated over time series data (particularly forecasting), it is of utmost importance to understand the behavior of the ESN architecture. To this end, the output of potential memory and temporal patterns is exemplified when used to explain an ESN-based model for three different regression tasks: one comprising a variable-amplitude sinusoid, and two real-world datasets for battery consumption and traffic flow forecasting.
- Second, experiments focus on the explanation of ESN models when used for image classification. This application is not new [448–450], but considering this task as a benchmark for the developed xAI suite permits to show their utility to unveil strengths and weaknesses of these architecture in such a setup.
- Finally, video classification is approached with ESN-based models. Differently than the other two tasks under consideration, we will not only analyze the knowledge captured by the ESN-based classifier, but also compare it to a baseline Conv2DLSTM model [451] used for video classification.

The following subsections describe the setup and datasets used for the tasks described above.

3.3.1 *Time Series Forecasting*

As mentioned before, time series analysis stands at the core of the ESN architecture's main objective: modeling sequential data. Therefore, three experiments with different time series data sources are devised to gain understanding upon being modeled by means of ESNs. Specifically, such experiments consider 1) a synthetic NARMA system of order 10 fed with a variable noisy sinusoidal signal; 2) a dataset composed by real traffic flow data collected by different sensors deployed over the city of Madrid, Spain [452], and 3) a dataset of lithium-ion battery sensor readings. A regression task can be formulated over the above datasets: to predict the next value of the time series data, given the history of past values.

To begin with, we need a controllable dynamical system producing data to be modeled via a reservoir-based approach. Opting for this initial case ease the process of verifying whether the knowledge of the trained reservoir visualized with the xAI

tools conforms to what could be expected from the dynamical properties underneath. For this purpose, a first system is chosen featuring a recurrent behavior to see its implications on the potential memory and temporal patterns. A NARMA system governed by the following expression meets this recurrent nature sought for the time series:

$$y(t+1) = \tanh(0.3 \cdot y(t) + 0.05 \cdot y(t) \cdot \sum_{i=0}^{9} y(t-i) + 1.5 \cdot u(t-9) \cdot u(t) + 0.1), \tag{14}$$

i.e., a one-dimensional time series $u(t)$ given by:

$$u(t) = \sin(2\pi F t) + 0.02 \cdot n(t), \tag{15}$$

with $F = 4$ Hz, $n(t) \sim \mathcal{N}(0, 1)$, i.e. a Gaussian distribution with mean 0 and unit variance. The recurrence as per Expression (14) imposes that any model used for modeling the relationship between $u(t)$ and $y(t+1)$ should focus on at least ten prior time steps when producing its outcome, and should reflect the behavior of the signal amplitude changes over time. A segment of the time series resulting from the NARMA system is shown in Figure 3.6 (top plot).

The second and third datasets, however, deal with the task of forecasting the next value of time series data generated by a real-life recurrent phenomenon. On one hand, a battery dataset is built with recorded electric current and temperature measurements of a lithium-ion battery cycled to depletion following different discharging profiles. This type of data allows for an inspection of a multivariate ESN that presents a very stable working regime, featuring very abrupt changes (as the middle plot in Figure 3.6 clearly shows). On the other hand, traffic flow data collected by a road sensor in Madrid is used. This data source is interesting since it contains long-term recurrence dynamics (bottom plot in Figure 3.6), suited to be modeled via ESN-based approaches. These datasets have been made available in a public GitHub repository, along with all scripts needed for replicating the experiments (`https://github.com/alejandrobarredo/xAI4ESN`).

Methodologically the experiments with the time series datasets discussed in Subsection 3.3.1 are structured in the following way. First, a *potential memory* analysis is carried out for the three datasets, elaborating on how the results relate to the characteristics of the data being modeled. Then, a follow-up analysis is performed around the *temporal patterns* emerging from the reservoir dynamics, also crosschecking whether such patterns conform to intuition. All figures summarizing the outputs of these xAI techniques are depicted together in the same page for a better readability of the results and an easier verification of the claims made thereon.

3.3.2 *Image Classification*

As mentioned before, ESN models have been explored for image classification in the past, attaining competitive accuracy levels with a significantly reduced training complexity [448–450]. To validate whether an ESN-based model is capturing features from its input image that intuitively relate to the target class to be predicted, the second experiment focuses on the application of the developed *pixel absence* technique

Table 3.1: List of action recognition video datasets considered in the experiments, along with their characteristics

Dataset	# classes	# videos	Frame size	Description
KTH [453]	6	600	160×120	Videos capturing 6 human actions (walking, jogging, running, boxing, hand waving and hand clapping) performed several times by 25 subjects in 4 different scenarios
WEIZMANN [454]	10	90	180×144	Videos of 9 different people, each performing 10 natural actions (e.g. run, walk, skip, jumping-jack) divided in examples of 8 frames
IXMAS [455]	13	1650	320×240	Lab-generated multi-orientation videos of 5 calibrated and synchronized cameras recording common human actions (e.g. checking-watch, crossing-arms, scratching-head)
UCF11 [456]	11	3040	320×240	Action recognition dataset of realistic action videos, collected from YouTube, having a variable number of action categories
UCF50 [457]	50	6676		
UCF101 [458]	101	13320		
UCFSPORTS [459]	16	800	320×240	Sport videos, collected from YouTube, having 16 sport categories
HMDB51 [460]	51	6766	320×240	Videos collected from different sources (mainly YouTube and movies) divided into 51 actions that can be grouped in five types: general facial actions, facial actions with object, general body movement, body with object, human body interactions
SquaresVsCrosses (*new in this work*)	2	2000	28×28	Synthetic video dataset containing two classes (*squares* and *crosses*) with clear spatial separability. The *squares* class contains videos of a 2×2 pixel blob moving randomly in a square trajectory of equal horizontal and vertical length, and radius centered in the middle of the frame. The *crosses* class contains an equally sized 2×2 blob moving randomly in a cross motion through the center of the frame.

to an ESN image classifier trained to solve the well-known MNIST digits classification task [461]. This dataset comprises 60000 train images and 10000 test images of 28×28 pixels, each belonging to one among 10 different categories.

Since these recurrent models are rather used for modeling sequential data, there is a prior need for encoding image data to sequences, so that the resulting sequence preserves most of the spatial correlation that could be exploited for classification. This being said, a column encoding strategy is selected, so that the input sequence is built by the column-wise concatenation of the pixels of every image. This process yields, for every image, a one-dimensional sequence of $28 \times 28 = 784$ points, which is input to a Deep ESN model with $N_L = 4$ reservoirs of $N = 100$ neurons each. Once trained, the Deep ESN model achieves a test accuracy of 86%, which is certainly lower than CNN models used for the same task, but comes along with a significantly decrease in training complexity. Training and testing over this dataset was performed in less than 5 seconds using a parallelyzed Python implementation of the Deep ESN model and an i7 2.8 GHz processor with 16GB RAM.

In this case results and the discussion held on them focus on the *pixel absence* effect computed not only for single pixels (namely, points of the converted input sequence), but also to 2×2, 4×4 and 8×8 pixel blobs showing the importance the model granted to regions of the input image with increasing granularity. As elaborated in Section 3.3.2, the absence effects noted in this dataset exemplify further uses beyond explainability, connecting with the robustness of the model against adversarial attacks.

3.3.3 *Video Classification*

With this third kind of data a further step is taken beyond the state of the art, assessing the performance of ESN-based models for video classification. The transfor-

mation of an image to sequences paves the way towards exploring means to follow similar encoding strategies for the classification of stacked images, which essentially gives rise to the structure of a video.

In its seminal approach, the video classification task comprises not only predicting the class associated to an image (frame), but also aggregating the predictions issued for every frame over the length of every video. These two steps can be performed sequentially or, instead, jointly within the model, using for this purpose assorted algorithmic strategies such as joint learnable frame classification and aggregation models. In this case, similarly to image classification, each video (i.e. a series of frames) is encoded to a multidimensional sequence data that embeds the evolution of each pixel of the frames over the length of the video. Specifically, each component $u_k(t)$ of $\mathbf{u}(t)$ is a sequence denoting the value of the k-th pixel as a function of the frame index t. Therefore, the dimension K of the input $\mathbf{u}(t)$ to the ESN model is equal to the number of pixels of the frames composing the video under analysis. Figure 3.7 depicts a diagram of this video encoding process.

In order to assess its predictive accuracy, an ESN-based model with $N_L = 4$ reservoirs and $N = 500$ neurons each is run over several known video classification datasets and a synthetic dataset built in the attempt of showcasing, Pixel Absence's ability to discover biases. These datasets are thoroughly described in Table 3.1. However, achieving a fair comparison between other video classification models reported in the literature and the ESN-based model is not straightforward. Most existing counterparts are deep neural networks that exploit assorted video preprocessing strategies (e.g. local space-time features [453] or the so-called universal background model [462]) and/or massively pretrained weights [463], which permit to finely tune the network for the task at hand. Including these ingredients in the benchmark would not allow for a fair attribution of the noted performance gaps to the modeling power of one or another classification model.

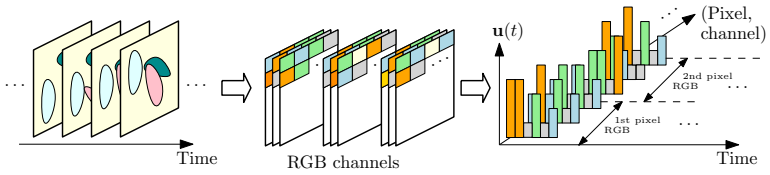


Figure 3.7: Schematic diagram showing the process of converting a sequence of images (video) to a multi-dimensional time series $\mathbf{u}(t)$.

To avoid this problem and focus the scope of the discussion, the ESN-based architecture is compared to a Conv2DLSTM deep neural network [451] having at its input the original, unprocessed sequence of video frames. The CNN part learns visual features from each of the frames, whereas the LSTM part learns to correlate the learned visual features over time with the target class. Models are trained with just the information available in each of the datasets, without data augmentation nor pretraining. This comparison allows comparing the strengths of these raw modeling architectures without any impact of other processing elements along the video classification pipeline. Table 3.2 summarizes the characteristics of these models under comparison, as well as the parameters of the ESN-based approaches used in the preceding tasks.

In regards to explainability, the developed pixel-absence technique is applied to the ESN model trained for video classification to extract insights about the model's learning abilities in this setup. The discussion about the results of the benchmark is presented in Subsection 3.3.3.

Table 3.2: Table presenting the parameters of the considered ESN and Conv2DLSTM models.

Experiment	N_L	N	α	ρ_{max}
NARMA	1	100	0.99	0.95
Battery	5	200	0.90	0.9
Traffic	2	200	0.99	0.9
MNIST	4	100	0.95	0.9
Video	4	500	0.99	0.9
Conv2DLSTM	64 3×3 kernels + LSTM, tanh activation + Dropout (0.2) + Dense (256 neurons), ReLu activation + Dropout (0.3) + Dense(# classes), softmax output Optimizer: SGD, $lr = 0.001$ Categorical cross-entropy, 20 epochs			

3.4 RESULTS AND DISCUSSION

This section goes through and discusses on the results obtained for the experiments introduced previously. The section is structured in three main parts: time series forecasting (Subsection 3.4.1), image classification (Subsection 3.4.2), and video classification (Subsection 3.4.3).

3.4.1 *Time Series Analysis*

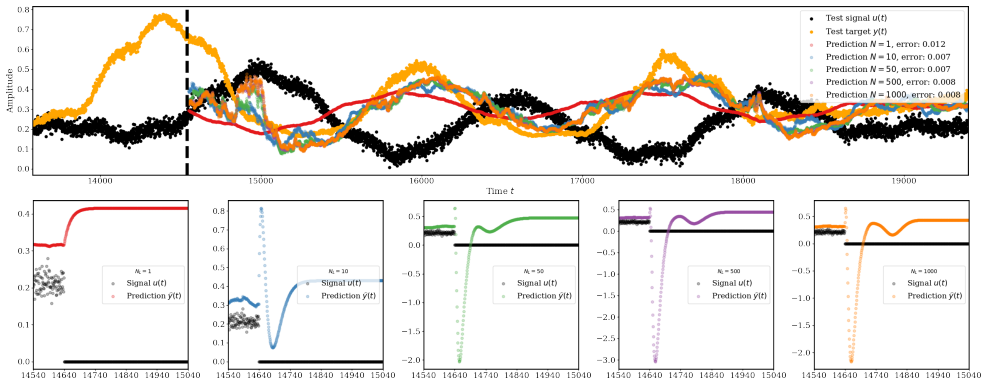
As mentioned before, in this first set of experiments two different studies are carried out. The first one is centered around the potential memory technique, whereas the second one focuses on the temporal patterns method.

3.4.1.1 *Experiment 1: Potential Memory*

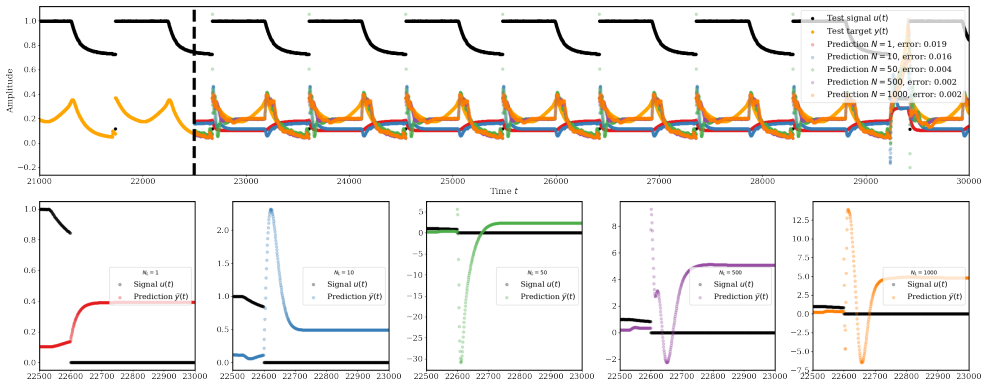
Reservoir size is one of the most important parameters when designing an ESN-based model, mostly because it is a compulsory albeit not sufficient parameter for the model to be able to model a system's behavior. It is compulsory because there is always a minimum amount of memory required to learn a recurrent behavior, yet it is not sufficient because memory does not model behavior by itself, but only allows for its persistence. In most cases, this value is set manually, at most blindly automated by means of wrappers, without any further interpretable information given of the

appropriateness of its finally selected value. This experiment attempts at bringing light into this matter.

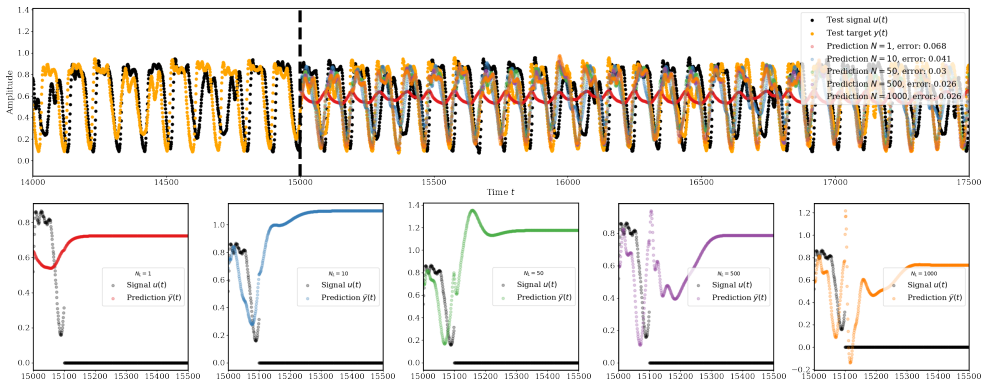
For this purpose different reservoir sizes were explored when fitting an ESN to the three datasets to be modeled. For every dataset, the reservoir size varies while keeping the rest of the parameters fixed. The rest of the parameters are chosen through experimentation. A spectral radius ρ_{max} equal to 0.95 is established, since all system has long-term interactions between input and target, as well as a *leaky rate* of 0.99 to ensure a fast plasticity of the reservoirs.



(a)



(b)



(c)

Figure 3.8: Potential memory analysis of the ESN model trained over the (a) NARMA, (b) battery and (c) traffic datasets for different reservoir sizes $N_L = \{1, 10, 50, 500, 1000\}$ and constant reservoir parameters.

As shown in the images 3.8.a, 3.8.b and 3.8.c, the size of the reservoir can be considered a major issue when fitting the model, until a certain threshold is surpassed. The potential memory technique allows monitoring this circumstance by estimating the model's memory capacity. After surpassing a memory threshold, the potential memory of the reservoirs becomes predictable and does not change any longer. This phenomenon seems to be related with its improvement in error rate, although this second aspect should be considered with caution, since the process of fitting an ESN-based model involves several other parameters that interact with each other. In any case, the technique presented in this first study provides some hints that might help the user understand when a workable reservoir size is selected.

3.4.1.2 *Experiment 2: Temporal Patterns*

When designing a deep ESN model, the next question arising along the process deals with the understanding of whether the model has captured all the dynamics inside the dataset so as to successfully undertake the modeling task. This second experiment analyzes the extent to which a Deep ESN is able to capture the underlying patterns of a system by means of the proposed *temporal patterns*. In this experiment, different parameters are varied to show the usefulness of this technique. This puts in perspective two interesting aspects: first, the technique allows for a quick inspection of whether the model is capturing the patterns of the system. As per the results exposed in what follows, the clarity by which such patterns show up in the plot seems to be related to how well the model is capturing them.

Figures 3.9.a/d/g, 3.9.b/e/h and 3.9.d/f/i summarize the results for different reservoir configurations for the three time series datasets under consideration. Figures are ordered in three different sets {a, b, c}, {d, e, f} and {g, h, i}. The first set correspond to ESN models with equal parameters and different layer configurations (1 and 2 layers). The second set analyzes system configurations with two different ρ_{max} values. The third set regards differences in the value of the α parameter. A simple visual inspection to these plots uncovers that d.2, e.2 and f.2 present clear patterns (in comparison to the other configurations), yielding the smallest errors for each dataset.

From the experimentation, two main findings can be underscored. First, stacking several reservoirs may jeopardize the tuning process for the other parameters of the model. Figures 3.9.a, 3.9.b and 3.9.c shows that adding a new layer without changing any other parameter has decreased the error and even sharpened the patterns captured by the reservoirs. This effect occurs for every dataset, although it is hard to see the differences in the patterns due to the complexity of the models being explained. Figures 3.9.d, 3.9.e and 3.9.f shows the difference between a high and low ρ_{max}, demonstrating the relevance of this parameter to attain a further level of detail in the patterns captured by the reservoirs. Although plots in Figures 3.9.e and 3.9.f are harder to analyze, both expose more detailed patterns, along with the decrease in error for the highest ρ_{max} configuration. Finally, Figures 3.9.g, 3.9.h and 3.9.i examine the differences found when tweaking the α parameter. All three cases present blurred patterns, along with newly emerging artifacts. Since the α parameter is linked to the decay, tuning its value may induce new patterns captured in the reservoirs that may – or may not – be informative for the audience or even for the overall

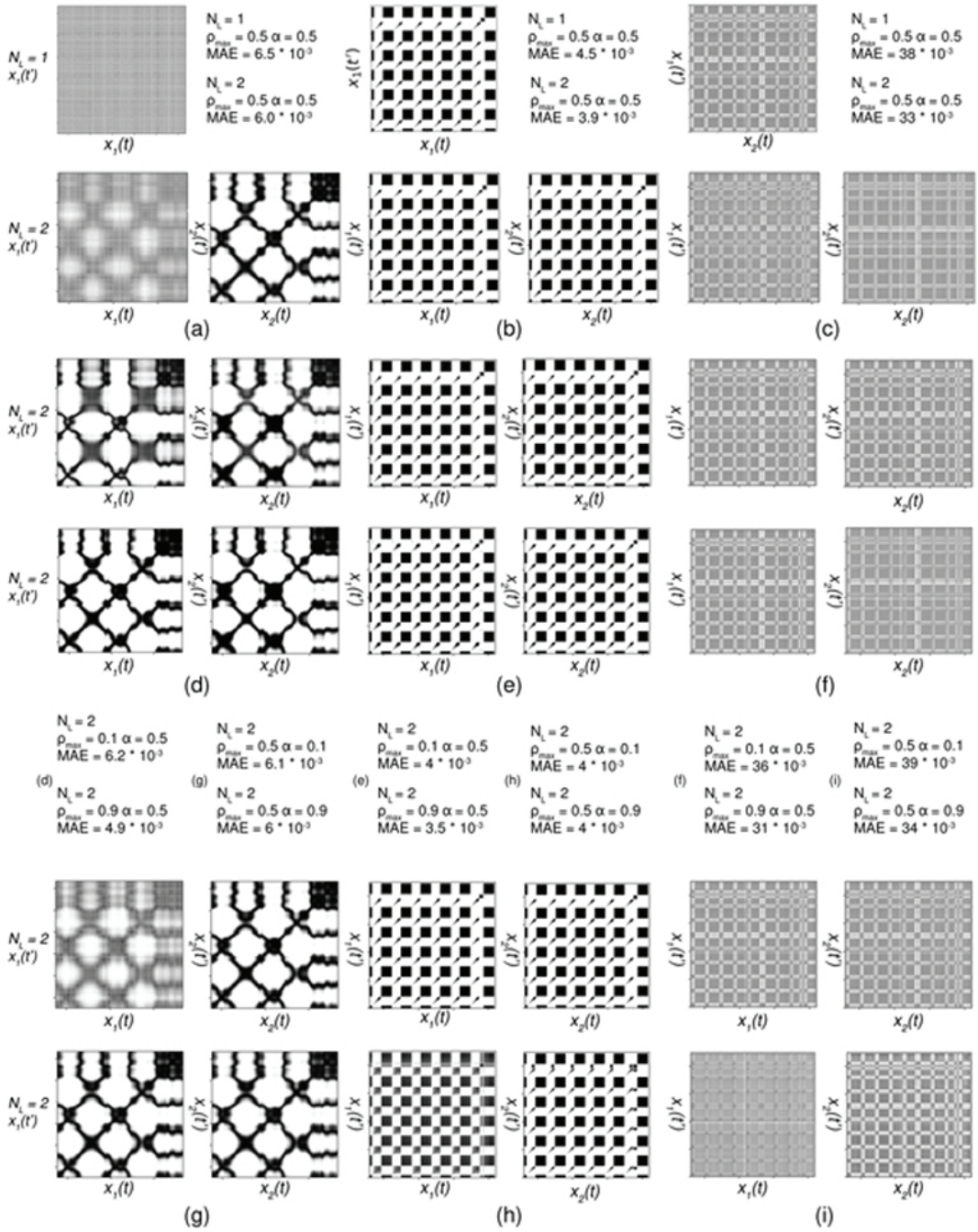


Figure 3.9: Temporal patterns analysis over the (a) and (d) and (g) NARMA; (b) and (e) and (h) battery; and (c) and (f) and (i) traffic datasets, presenting different configuration parameters. It can be observed that the spectral radius impact directly on the detail of the recurrence plots, to the extent of causing a significant performance degradation if not selected properly.

modeling performance of the network. This roughly depends on the dynamics of the system and the task under consideration.

Apart from the usability of showing interpretable information to the user about the capability of the model to capture the patterns within data, these plots enable

another analysis. If we focus on the battery dataset, it is quite clear that the signal is repetitive, stationary and that the system's recurrent patterns are also visible in the recurrence plots. In the case of the traffic data, this phenomenon is much more important. By just staring at the traffic signal it is not easy to realize that the signal also contains certain stationary events (weekends) that produce different behaviors in traffic. By simply staring at the recurrence plots, it is straightforward to realize the periodic structure of the signal, and how these events happen at equally distant time steps throughout the signal (as happens with weekends), with a especially large effect at the top right of the recurrence plot corresponding to a long weekend.

The experiments discussed above should not be limiting in what refers to the different features that can be extracted from recurrence plots. Within these plots, large- and small-scale patterns can be distinguished. From [445] large-scale patterns are considered topological features, whereas small-scale ones are regarded as *textures*. Manifold topological features can be ascertained from recurrence plots, such as homogeneity (stationarity), the presence of periodic and quasi-periodic structures that represent oscillating systems, *drifts* (non-stationarities evolving at a slow speed over time) and *abrupt changes*, that can be of interest to see whether the model is over-fit to rare events in the modeled sequences. Regarding textures, from the recurrence plots one can infer rare and isolated correlated states that can be associated with noise (*single dots*), phase spaces that are visited at different times (recurrent states evinced by *diagonal lines*) and vertical/horizontal lines, which represent an steady state throughout a given time frame. When occurring in recurrence plots, a proper interpretation of these events usually requires intuition and domain knowledge, as we have exemplified in our prior experiments with battery consumption and traffic data.

3.4.2 *Image Classification*

As introduced before, this second experiment analyzes a Deep ESN model for image classification over the MNIST dataset. Once trained, this model is used to examine the output of the third technique comprising the proposed suite of xAI techniques: pixel absence effect.

Figure 3.10 shows the results of computing the pixel absence effect for single pixels, wherein the color represents the importance of canceling each pixel. Specifically, the scale on the upper left corner shows the amount of change the cancelation of a certain pixel has caused on the probability of every class. The blue color indicates the pixels that are important to predict the image as the desired class. The darker the blue color of a certain pixel is, the higher the decrease in probability for the desired class will be. The red color means exactly the opposite. Therefore, the output of this technique must be conceived as an adversarially driven local explanation technique, that pursues to explain which regions of a certain input, upon their cancellation, *push* the output class probability distribution towards or against a certain desired class.

We exemplify the output of the experiment for a given example of digit 8. The reason for this choice is that this digit has several coinciding visual features with other digits. As such, it was expected that a great effect was discerned at the center of the digit itself, favoring the classification for digit 0 since the elimination of the

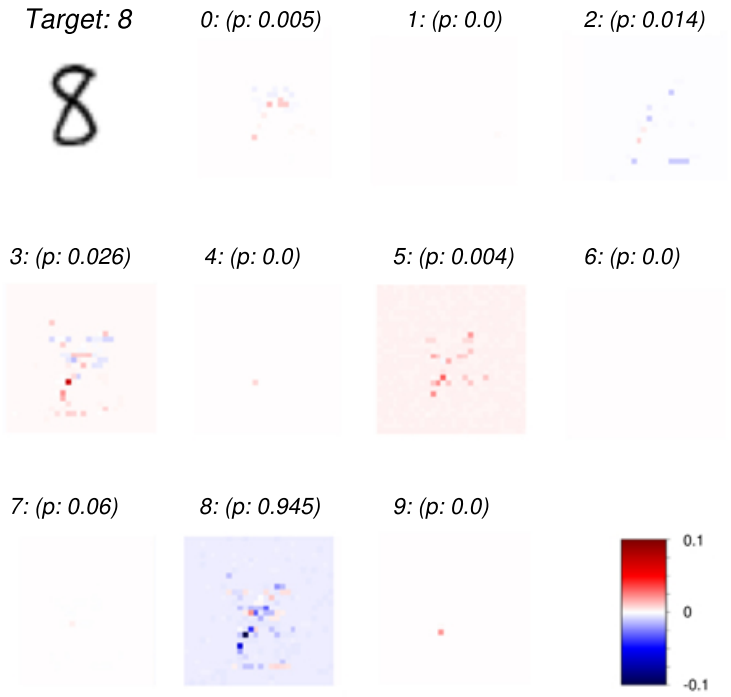


Figure 3.10: Pixel absence effect over a Deep ESN model trained to classify MNIST digits for single pixels.

center pixels in digit 8 should directly render the image as a 0. However, it is important to note that this does not happen for ESN architectures that are sensitive to space transformations. This sensitivity impedes any ability of the model to extract spatially invariant features from the digit image, thereby circumscribing its knowledge within the sequential column-wise patterns found in examples of its training set. This conclusion emphasizes the relevance of finding transformation strategies from the original domain of data to sequences, not only for ensuring a good performance of the learned ESN-based model, but also to be able to elicit explanations that conform to general intuition.

Table 3.3: Accuracy and number of trainable parameters of models tested over each video classification dataset.

Dataset	Deep ESN	Conv2DLSTM	Difference	State of the art	Technique
KTH	56%	22%	+34%	94.39%	Differential gating [464]
WEIZMANN	25%	12%	+13%	98.63%	Bio-inspired hierarchy [465]
IXMAS	40%	36%	+4%	94.16%	Structural information [466]
UCF11	54%	26%	+28%	84.96%	GoogLeNet + RNN [467]
UCF50	75%	63%	+12%	95.24%	Trajectory descriptors [468]
UCF101	45%	42%	+3%	95.10%	Trajectory descriptors [469]
HMDB51	25%	17%	+8%	65.90%	Trajectory descriptors [469]
UCFSPORTS	32%	23%	+9%	96.6%	Grassmann graph embedding [470]
Trainable parameters	Min: 12,000 Max: 200,000	Min: +293,626,418 Max: 1,205,453,326	Min: 305,626,418 Max: +1,005,453,326		

3.4.3 *Video Classification*

As stated in the design of the experimental setup, the discussion ends up with a video classification task approached via ESN-based models. Since to our knowledge this is the first time in the literature that video classification is tackled with ESN architectures, a benchmark is first run to compare its predictive performance to a Conv2DLSTM network, in both cases trained only over unprocessed video data provided in the datasets of Table 3.1. As argued before, it would not be fair to include preprocessing elements that could bias the extracted conclusions about the modeling power of both approaches under comparison.

Table 3.3 summarizes the results of the benchmark, both in terms of test predictive accuracy and the size of the trained models (in terms of trainable parameters). It can be observed that Deep ESN models consistently outperforms their Conv2DLSTM counterparts in all the considered datasets, using significantly less trainable parameters that translate to dramatically reduced training latencies. Clearly, these results are far from state-of-the-art levels of performance attained by deep neural networks benefiting from pretrained weights (from e.g. ImageNet). Nevertheless, they serve as an unprecedented preview of the high potential of Deep ESN models to undertake complex tasks with sequence data beyond those for which they have been utilized so far (mostly, time series and text modeling).

The developed pixel absence effect test was run over the trained Deep ESN model for video classification. The core of the discussion is set on the results for a single video (local explanation) belonging to the `UCF50` dataset, which should ideally determine what zones of the video frame are most important for the current prediction.

The results from this pixel absence analysis depicted in Figure 3.11 uncovers an unintended yet interesting result. Although the image shown in this picture is just one frame of the video, the region of the frame which is declared to be relevant by the analysis is the same throughout the whole video, the difference being the changing colors of its constituent pixel blobs. This effect unveils that the Deep ESN model has learned to focus on the floor to produce its predictions, which suggests that its knowledge is severely subject to an evident bias in the `UCF50` dataset. This bias can be detected visually by simply observing the images depicted in Figure 9.c, which are samples of frames of the same class that also contain a similar floor feature that might cause a bias in the model.

The conclusions drawn above are further elaborated by undertaking a simplified video classification experiment with a dataset synthetically furnished for the purpose. This dataset, denoted as `SquaresVsCrosses`, is composed by videos of two different classes. Examples of one of these classes consist of a 3×3-sized blob of pixels with a square trajectory centered in the middle of the frame, whereas examples of the other class move over the frame by outlining a cross-shape trajectory. To showcase the bias problem, two different experiments are run. In the first one, classes are set spatially disjoint from each other, i.e. the space over the frame in which examples of one class move over time is totally different than the space delimited by trajectories of the examples of the other class. This first experiment should exemplify the problem of spatial bias. The second experiment is run with a dataset in which examples of both classes occupy the same space over the frame, hence, spatial bias is removed.

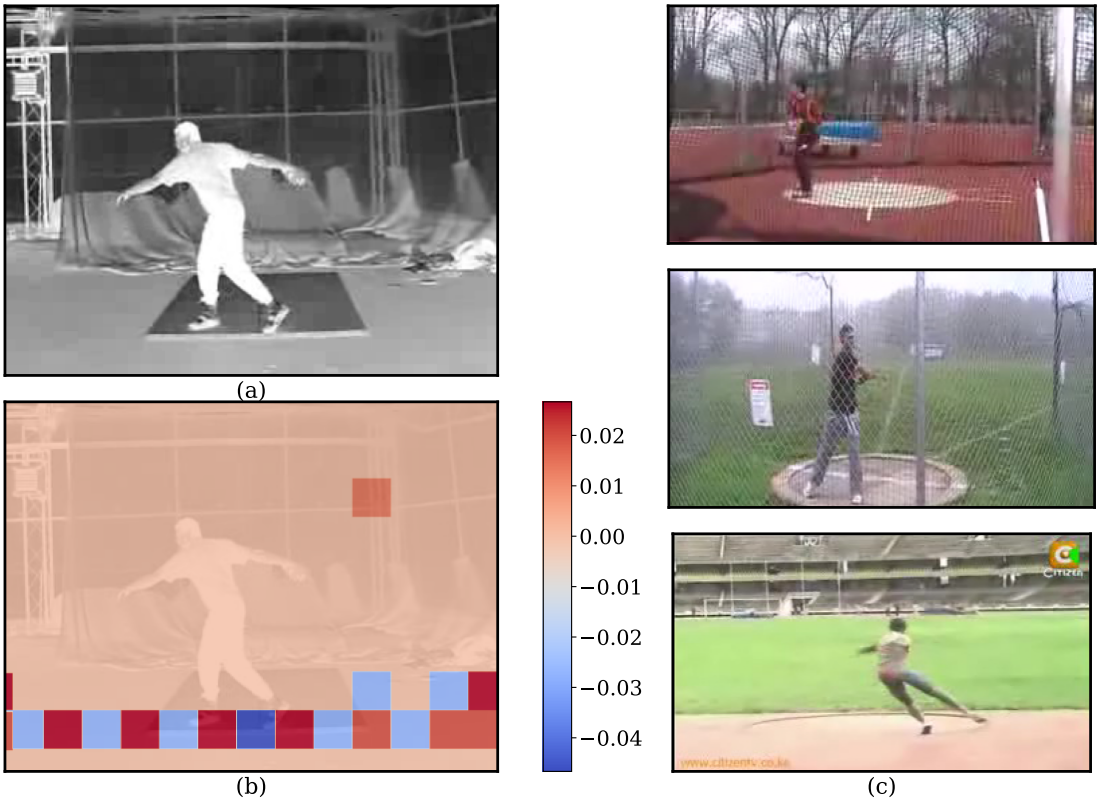


Figure 3.11: Pixel absence analysis of a video frame showing an unintended result that gives importance to sections of the image that apparently should not be important.

Figure 3.12 illustrate the results for these 2 different dataset configurations, where 3.12.a and 3.12.c correspond to the `squares` class, and 3.12.b and 3.12.d to the `crosses` class. In 3.12.a and 3.12.b, examples of both classes are spatially disjoint: `squares` move along the border of the frame, while `crosses` move in the center of the frame. These two first plots reveal that there is no entanglement between the classes. Their pixel absence heat map contains only one color for each class, meaning that all the pixels contained in the image under analysis push towards one class (the true class) and are not present in the other. A bias emerge from the trajectory followed by the blob. Although it should be informative with respect to the class (square or cross), the shape of the blob does not convey any predictive power when compared to the trajectory itself.

Contrarily, the second pair of videos in Figure 3.12 – namely, 3.12.c and 3.12.d – correspond to a dataset where the trajectories of examples of both classes spatially overlap, hence removing the trajectory bias. The apparent difference is that this time, both heat maps show mixed colors when analyzing them with the pixel absence tool. Consequently, some pixels occur more frequently in examples of one class than the other, hence pushing the prediction towards that class when those pixels are activated. Contrarily to the case in which spatial bias was present, the analysis showed pixel importance patterns that are more complex that a simple spatial separation. This experiment concurs with the intuition surfacing from Figure 3.11, and confirms that pixel absence might be a suitable tool for bias detection over the data from which the ESN model has learned. In summary: the use of the pixel absent effect tool can help

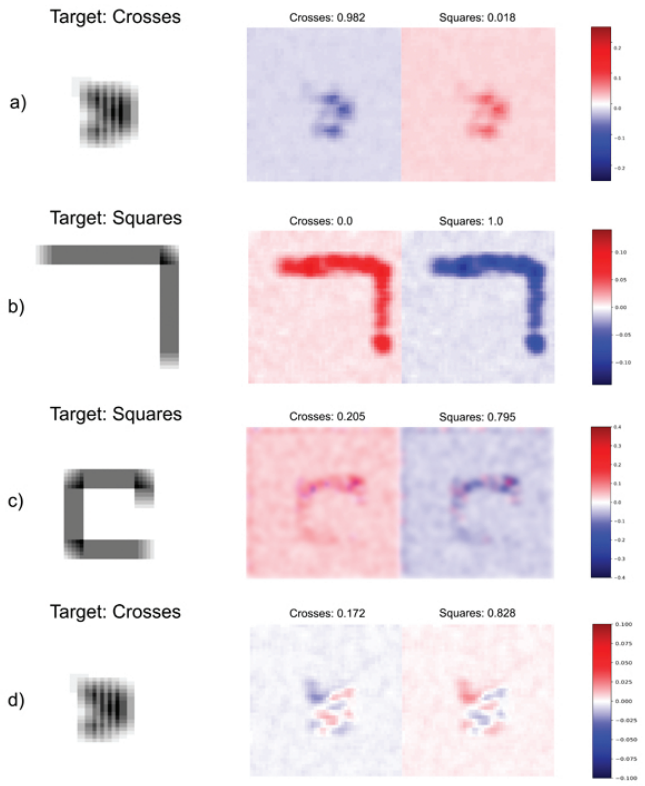


Figure 3.12: Pixel absence analysis of four different videos: (a) and (b) correspond to a model trained with spatially disjoint classes, whereas (c) and (d) correspond to a model with spatially overlapping classes.

the user detect sources of bias systematically, so that countermeasures to avoid their effect in the generalization capability of the model can be triggered.

3.5 SUMMARY

This chapter has elaborated on the need for explaining the knowledge acquired from sequential data with ESN-based models by means of a set of novel post-hoc xAI techniques. Through the lens of the proposed framework, hidden strengths and weaknesses have been discovered for these type of models. To begin with, the modeling power has been assessed over a diversity of data sources, with results well above what were to be expected given their inherent random nature. However, the suite of xAI techniques has also revealed that random reservoirs composed by recurrently connected neural units undergo architectural limitations to model data sources that call for spatially invariant feature learning, such as image and video. Indeed, ESN-based models achieve reasonable levels of predictive accuracy given their low training complexity, specially for video classification without any prior preprocessing, data augmentation and/or pretraining stages along the pipeline. A deeper inspection of their learned knowledge by means of the proposed xAI tools has identified huge data biases across classes in video data, showing that such superior scores might not be extrapolable to videos recorded in other contextual settings.

4

EXPLORING THE TRADE-OFF BETWEEN PLAUSIBILITY, CHANGE INTENSITY AND ADVERSARIAL POWER IN COUNTERFACTUAL EXPLANATIONS USING MULTI-OBJECTIVE OPTIMIZATION

During the last years Deep Neural Networks (also referred to as *Deep Learning*) have started to traverse beyond the theoretical analysis of their properties towards being implemented and deployed in a multitude of real-world applications. This has been specially noted in applications dealing with high-dimensional data, over which Deep Learning has delivered promising results to conquer the broad landscape of Machine Learning modeling approaches. As such, their superior performance has been noted in many scenarios dealing with image, video and/or spatial-temporal data, including agriculture [471], transportation [472] or industry [353], to cite a few. Nowadays the prevalence of Deep Learning in such areas is beyond any doubt.

Unfortunately, some concerns arise from the mismatch between research studies dealing with Deep Learning applied to certain modeling tasks (*let the model perform to its best for the task at hand*) and the real-world use of models to improve an already known solution. Most in-field approaches contemplate attempts at improving an already human-created solution to solve a problem (optimizing a process), whereas the most common Deep Learning approaches are better suited to find their own solutions to a more high-level problem (predicting an outcome). Together with this difference, another concern deals with the difficulty to understand and interpret the mechanisms by which Deep Learning works, particularly when the audience that makes decisions on their outputs lack any knowledge about Computer Science. This renders a useless modeling choice for real-world scenarios in which models are used to improve decision making in processes that are managed by humans and/or where decisions affect humans. This is the case observed in medical diagnosis, law or social politics, among others. In other words, actionability of predictions requires a step beyond a proven good generalization performance of the model issuing them.

In order to bridge this gap to model-based decisions, new frameworks for explainability are required. These frameworks aim at giving insights not only to experts in the field of application, but also to those commonly in charge of the use and maintenance of the deployed models. These two audience profiles differ significantly in what refers to their capabilities to understand explanations generated for a given model. These different capabilities entail that approaches to explain Deep Learning models generate explanations better suited for auditing the models by developers, leaving them far from the cognitive requirements of experts that ultimately make decisions in practice.

Recent research is profoundly bothered with bridging this gap. To this end, the broad scope of model explainability has been approached from manifold areas, including robustness by adversarial attacks [473, 474], output confidence estimation [269, 475, 476], visualization of internal representation [145, 439] or attention-based

explanations [477]. Even though the research community is thrilling with new advances in explainability, they do not entirely bridge the aforementioned gap between theoretical developments and their practical adoption. Most explainability solutions [478] consider an audience with profound knowledge of the inner workings of the models, which eases the understanding of explanations, but does not comply with real-world settings often encountered in model-based decision making processes.

Among the alternatives to reach such a universal understanding of model explanations, counterfactual examples is arguably the one that best conforms to human understanding principles when faced with unknown phenomena. Indeed, discerning what would happen should the initial conditions differ in a plausible fashion is a mechanism often adopted by human when attempting at understanding any unknown [479, 480]. Circumscribing the factual boundaries by which a given model works *as usual* can be conceived as a post-hoc explainability method, which grounds on an adversarial analysis of the audited model. From the practical perspective several aspects of the produced counterfactual examples should be considered besides its plausibility, so that the audience of the model can examine the limits of the model from a multi-faceted perspective.

This chapter joins the rest of the work presented before in this thesis at making Deep Learning models more usable in practice via counterfactual-based explanations. To this end, we propose an adversarial strategy to produce counterfactual examples for a Deep Learning classifier. As for the chapter before (3), this contribution to the literature of XAI falls within the scope of Post-Hoc Explainability Techniques. Specifically under Post-Hoc XAI for Deep Learning. This classifier to be audited solves a task defined over a certain dataset (e.g. discriminate male and female images from human faces), so that counterfactuals are generated to explain the boundaries of the model once trained to address the classification task at hand. We further impose that the generated counterfactual examples are *plausible*, i.e., changes made on the input to the classification model have an appearance of credibility without any computer intervention. To ensure plausibility, the proposed method makes use of GANs (Generative Adversarial Networks) in order to learn the underlying probability distribution of each of the features needed to create examples of a target distribution (namely, human faces). Our framework allows for a search among samples of the first distribution to find realistic counterfactuals close to a given test sample that could be misclassified by the model (namely, a face of a male being classified as a female). As a result, our framework makes the user of the model assess its limits with an adversarial analysis of the probability distribution learned by the model, yet maintaining a sufficient level of plausibility for the analysis to be understood by a non-expert user. As a step beyond the state of the art, the proposed framework ensures the production of multi-faceted counterfactual examples by accounting for two additional objectives besides plausibility: 1) the *intensity of the modification* made to an original example to produce its counterfactual version; and 2) its *adversarial power*, which stands for the change in the output of the model that is audited.

In summary, the main contributions of this chapter beyond our preliminary findings reported in [481] can be summarized as follows:

- We present a novel framework to generate multi-faceted counterfactual explanations for targeted classification models. The framework brings together GAN

architectures for generative data modeling and multi-objective optimization for properly balancing among conflicting properties sought for the counterfactuals: plausibility, change intensity and adversarial power.

- The framework is described mathematically, and design rationale for each of its compounding blocks is given and justified.
- Explanations generated by the framework are showcased for several classifiers and GAN models for image and volumetric data, discussing on the trade-off between the properties of the counterfactual set. The explanations generated by this framework could be categorized under *Visual Explanations*, since it is the medium for the explanations. However, the information to be extracted from the visualizations focuses on the understanding the relevance of the input, hence this classification would not be wrong either.
- We argue and show that when inspected from a multi-faceted perspective, counterfactual examples can be a magnificent tool for bias analysis and the discovery of misrepresentations in the data space.

The rest of the chapter is organized as follows: first, Section 4.1 extends the background covered in Chapter 2 by going deeper on the specific background required for connecting the four core aspects of our proposed framework: Deep Learning for image classification, GANs, model explainability and counterfactual explanations. Section 4.2 details the framework proposed in this study, including a mathematical statement of the problem tackled via multi-objective optimization and a discussion on how the output of the framework can be consumed by different audiences. Section 4.3 describes the experimental setup designed to showcase the output of the framework. Section 4.4 presents and discusses the results stemming from the performed experiments. Finally Section 4.5 summarizes the broad lines taken in this chapter.

4.1 RELATED WORK

As anticipated at the beginning of the chapter, the proposed framework resorts to GANs for producing realistic counterfactual examples of classification models. Since the ultimate goal is to favor the understanding of the model classification boundaries by an average user, the framework falls within the XAI (Explainable Artificial Intelligence) umbrella. This section briefly contextualizes and revisits the state of the art of the research areas related to the framework: Deep Learning for image classification and generative modeling (Subsection 4.1.1), XAI and counterfactual analysis (Subsection 4.1.2) and multi-objective optimization (Subsection 4.1.3).

4.1.1 *Deep Learning for Image Classification and Generative Modeling*

When it comes to discrimination tasks over image data, the reportedly superior modeling capabilities of Convolutional Neural Networks (CNNs) are often adopted to capture spatial correlations in image data [11, 482]. This is achieved by virtue of trainable convolutional filters which can be trained via gradient backpropagation or even imported from other networks pretrained for similar tasks, giving rise to image

classification models of the highest performance. The increasing availability of image datasets and the capability of processing them efficiently have yielded hierarchically stacked CNNs that, despite attaining unprecedented levels of accuracy, come at the cost of more complex, less understandable model structures [483]. The more complex the model is, the harder is to pinpoint the reasons for its decisions. The need for auditing these *black-box*es is the core motivation of the study presented in this paper.

Another task for which CNNs are crucial is generative modeling, e.g. the construction of models capable of characterizing the distribution of a given dataset and sampling it to create new, synthetic data instances. When the dataset is composed by images, generative adversarial networks (GANs) are arguably the spearhead in image generative modeling. GANs were first introduced by Goodfellow in [473], bringing the possibility of using neural networks (*function approximators*) to become generators of a desired distribution. Since their inception, GANs have progressively achieved photo-realistic levels of resolution and quality when synthesizing images of different kinds. In general, a GAN architecture consists of two data-based models, which are trained in a mini-max game: one of the players (models) minimizes its error (loss), whereas the other maximizes its gain. In such a setup, multiple models have flourished to date, each governed by its strengths and vulnerabilities [484]. In connection to the scope of this thesis, some of these were conceived with the intention of finding the pitfalls of a certain model and the ways to hack it [473, 474]. Other GAN approaches aim at generating samples of incredibly complex distributions like photo-realistic human faces [485, 486].

As will be later detailed in Section 4.2, the framework proposed in this chapter hybridizes these two uses of CNNs by optimizing the output generated by a GAN to perform a counterfactual analysis of a given classification model to be audited.

4.1.2 *Explainable Artificial Intelligence (XAI) and Counterfactual Explanations*

Model explainability has recently become a topic of capital importance in Machine Learning, giving rise to a plethora of different approaches aimed to explain how decisions are issued by a given model [478]. Most research activity noted in this area is arguably focused on post-hoc XAI tools that produce explanations for single data instances (what is referred to as *local explanations*). The LIME tool presented in [443] is one of this kind, visualizing a model's internal activations when processing a given test sample. A similar approach is followed by LRP (Layer-wise Relevance Propagation) embedded in the SHAP suite [229], which highlights the parts of an input image that push the output of the model towards one label or another. This provides an understandable interface of the reasons why the model produces its decision. More recently, Grad-cam [297] an its successor Grad-cam++ [487] can be considered as the *de facto* standard for the explainability of local decisions, particularly in the field of image classification. These two methods implement a gradient-based inspection of the knowledge captured by a neural network, giving rise to a quantitative measure of the importance of parts of the image for the output of the model. Unfortunately, the dependence of such explanations on the gradient of the model restricts the applicability of these techniques to other techniques beyond neural architectures.

When pursuing model-agnostic local explanations, a common strategy is to analyze the model from a counterfactual perspective. Counterfactual exploration is an innate process for the human being when facing an unknown phenomenon, system or process. The concept behind counterfactual explanations reduces to providing an informed answer to a simple question: *which changes would make the output of the unknown to a certain input vary?*. Such changes constitute a counterfactual example, always related to an input to the process or system under focus. Based on this concept, many contributions have developed to date different XAI approaches to generate counterfactual examples that allow understanding how Machine Learning models behave. Some approaches are based on discovering the ability of a given individual to change the model's outcome. One example is the work in [488], which presents a simple but effective distance-based counterfactual generation approach, that can be used to audit different classifiers (e.g. neural networks and support vector machines). Later, the counterfactual problem is tackled in [489] departing from the premise that a user should be able to change a model's outcome by actionable variables (recourse). This hypothesis is validated over linear classifiers, but also claimed to be extensible to non-linear classifiers by means of local approximations. In a similar fashion, [490, 491] allow the user to guide the generation of counterfactual examples by imposing forbidden changes that cannot be performed along the process. A subset of counterfactual studies are rather focused on the problem of predictive multiplicity [492, 493]. Multiple classifiers may output the same solution while treating the data in different ways, hence the generation of counterfactuals can lead to insights into the question of which of these classifiers is better for the problem at hand. In this area of research [494–497] have developed different schemes to address this problem. Connectedness, proximity, plausibility, stability and robustness are yet other concerns that have pushed the development of techniques for the generation of counterfactuals. In their search for robust interpretability, [498] came up with a method to generate self-explaining models based on explicitness, faithfulness and stability.

Following the extensive analysis carried out in [499], it is of utmost importance to recall the *"master theoretical algorithm"* [500], from which nineteen other algorithms concerning counterfactual explanations can be derived. The nineteen algorithms fall in a categorization of six different counterfactual generation strategies: instance-based, constraint-based, genetic-based, regression-based, game theory-based and case reasoning-based. Instance based approaches are derived from [488, 501], based on feature perturbance measured by a distance metric. The pitfall of these approaches (when pure) resides on their inability to validate instance plausibility. Constraint-based approaches are, in turn, the methods that modulate their counterfactual search by means of a constraint satisfaction problem. The more general scope of these approaches allows for an easier adaptation to the problems at hand. Genetic-based approaches, as the name conveys, guide the search for counterfactuals as a genetic-oriented optimization problem. Regression-based approaches use the weights of a regression model as a proxy to produce counterfactual examples. However, these approaches again fall short at assuring the plausibility and diversity of the produced counterfactual instances. Game-theory based approaches are driven by game-theoretical principles (e.g. Shapley values), but also disregard important properties of its counterfactual outputs. Finally, case reasoning-based approaches seek past solu-

tions (in the model) that are close to a given instance, and adapt them to produce the counterfactual. Once again, such adaptations may produce counterfactual instances that, even if close to a certain input, cannot be claimed to be plausible nor diverse with respect to the input under consideration.

4.1.3 *Multi-objective Optimization*

From the previous section it can be inferred that the generation of counterfactual explanations can be mathematically stated as a multi-objective optimization problem comprising different objectives that can be conflicting with each other. Plausibility – i.e., the likelihood of the counterfactual example to occur in practice – can be thought to conflict with the amplitude of the modifications made to the input of the model. Likewise, intense changes in the output of the audited model (namely, its *adversarial power*), as introduced at the start of this chapter, when fed with the counterfactual example can jeopardize its plausibility. There lies the contribution of the framework proposed in this work: the generation of a portfolio of counterfactual examples to a certain input that optimally balance among these objectives. This portfolio provides richer information for the user to understand the behavior of the audited model, and distinguishes this work from the current research on counterfactual analysis. The conceptual diagram shown in Figure 4.1 illustrates this motivational idea.

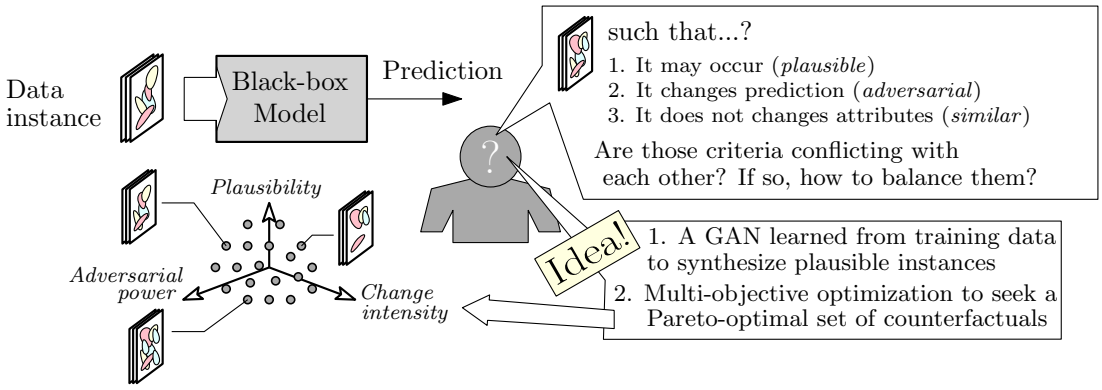


Figure 4.1: Conceptual representation of the rationale behind the confluence of predictive modeling, generative adversarial learning, explainability and multi/objective optimization that lies at the core of the proposed framework.

To this end, the framework presented in this chapter falls between constraint-based, genetic-based and instance-based counterfactual explanations, combining these three categories to render a set of multi-criteria counterfactuals. The usage of a GAN architecture presents the ability of a bounded search within a target distribution, enabling quantitative measures of the plausibility of the generated counterfactual (via the discriminator) and algorithmic means to sample this distribution (via the generator). The usage of a multi-objective optimization algorithm yields the ability to guide the counterfactual generation process as per the desired objectives (plausibility, intensity of the modifications and adversarial power), giving rise to the aforementioned portfolio of multi-criteria counterfactual explanations. Among them, we will resort to multi-objective evolutionary algorithms [502, 503], which efficiently perform the

search for Pareto front approximations of optimization problem comprising multiple objectives without requiring information about their derivatives whatsoever.

4.2 PROPOSED FRAMEWORK

This section covers the proposed framework, including the intuition behind its conceptual design (Subsection 4.2.1), a detailed description of its constituent parts and mathematical components underneath (Subsection 4.2.2), and an outline of the target audiences that can consume the produced counterfactual explanations, supported by hypothetical use cases illustrating this process (Subsection 4.2.3).

4.2.1 *Design Rationale*

The explainability framework explores the weaknesses of a target model by means of counterfactual instances generated by a GAN architecture. One of the key aspects of this framework is that it focuses on discovering the reality-bound weaknesses of the target model in the form of examples that, without exiting the realm of plausibility, are able to confound the target model. For instance, for a classifier mapping human faces to their gender (`male`, `female`), the framework can generate modifications of a given input face that are still considered to be real, but they make the audited model change their predicted gender. The overarching motivation of the framework comes from the human inability to asses the working boundaries of a given model in highly-dimensional spaces. In such complex areas, such as image classification, the domain in which images are bound is complex to be characterized, thereby requiring complex generative modeling approaches capable of modeling it and drawing new samples therefrom. The generator of a GAN architecture serves for this purpose, whereas the discriminator of the GAN allows verifying whether an output produced by the generator is close to the distribution of the dataset at hand, hence giving an idea of the plausibility of the generated instance.

At this point it deserves pausing at the further insights that the GAN-based framework can provide. Modifications of an input image producing a counterfactual can be edited by changing the value of variables that affect the output of the GAN generator. Such variables can represent attributes of the input image that ease the interpretation of the results of the counterfactual study regarding the existence of misrepresentations of the reality captured in the dataset at hand and transferred to the audited models. For instance, in the face-gender classifier exemplified previously, let us consider a GAN model with editable attributes (e.g. an AttGAN [504]), including color hair, face color or facial expressions. A counterfactual study of a `man` face could reveal that for the face to be classified as a `woman`, the color hair attribute of all produced counterfactuals share the same value (*blonde*). Besides the inherent interpretative value of the counterfactuals themselves, our framework can also identify data biases that may have propagated and influenced the generalization capabilities of the audited model.

4.2.2 *Structure and Modules*

Following the diagram shown in Figure 4.2, the design of the proposed framework can be split in three main blocks: audited model (classifier), GAN architecture and multi-objective optimizer. The audited model is fed with the counterfactual example generated by the discriminator model of the GAN architecture, hence its only prerequisite is that the input of the audited model and the output of the discriminator are of the same dimensions. In what follows we will assume that the target model to be audited is a CNN used for image classification. Nevertheless, the framework can be adapted to audit other models and tasks whenever the output of the GAN discriminator and the input of the audited model are equally sized, and the measure of adversarial power is redefined to account for the change induced by the counterfactual in the prediction of the model.

The GAN is the part of the framework in charge of generating the counterfactuals fed to the audited model. Therefore, two requirements are set in this module: 1) the discriminator must be trained for a similar data distribution to that of the audited model; and 2) the generator model must be able to generate samples of such a distribution as per an *attribute vector* $\mathbf{b}$ that controls specific features of the generated instance (image). This attribute vector is tuned by the multi-objective optimization algorithm seeking to maintain plausibility as per the discriminator, changing the output of the audited model and minimizing the intensity of the changes inserted in the original input image.

At this point it is important to emphasize that the audited model is left aside the overall training process of the GAN for several reasons. To begin with, for practicality we assume minimum access to the audited model (black-box analysis). Therefore, the logits of the audited model are exploited with no further information on its inner structure. Furthermore, the goal of the discriminator is to decide whether the generated image follows the distribution of the training set, which must be regarded as a plausibility check. The task undertaken by the audited model can be assorted, for instance, to discriminate among `male` and `female`, `old` and `young` or any other task.

The above three-objective optimization problem can be formulated mathematically as follows: let us denote an image on which the counterfactual analysis is to be made as $\mathbf{x^a} \sim P_X(\mathbf{x})$, which follows a distribution $P_X(\mathbf{x})$ and has an attribute vector $\mathbf{a} \in \mathbb{R}^N$. The generator of the GAN model is denoted as $G(\mathbf{x^a}, \mathbf{b})$, whose inputs are the actual image $\mathbf{x^a}$ and a desired attribute vector $\mathbf{b}$. In conditional generative models the generator is generally composed of an encoder G_{enc} and a decoder G_{dec}. However, for some architectures, the model directly departs from a decoder, given the assumption that the latent code is sampled from a known distribution. Leaving the special cases aside for the sake of a clearer explanation, the image conditionally output by the generator is given by $\mathbf{x^{b\prime}} = G_{dec}(G_{enc}(\mathbf{x^a}), \mathbf{b})$. Ideally, $\mathbf{x^{a\prime}} \approx \mathbf{x^a}$, i.e. the reconstructed image $\mathbf{x^{a\prime}} = G_{dec}(G_{enc}(\mathbf{x^a}), \mathbf{a})$ should resemble $\mathbf{x^a}$ itself. For non-conditional generative architectures, the generated image is given by $\mathbf{x'} = G_{dec}(G_{enc}(\mathbf{x}))$, where the objective is to have $\mathbf{x'} \approx \mathbf{x}$. A discriminator $D(\mathbf{x^{b\prime}})$ along with a classifier $C(\mathbf{x^{b\prime}})$ is placed next along the pipeline to determine 1) whether the synthesized image $\mathbf{x'}$ is vi-

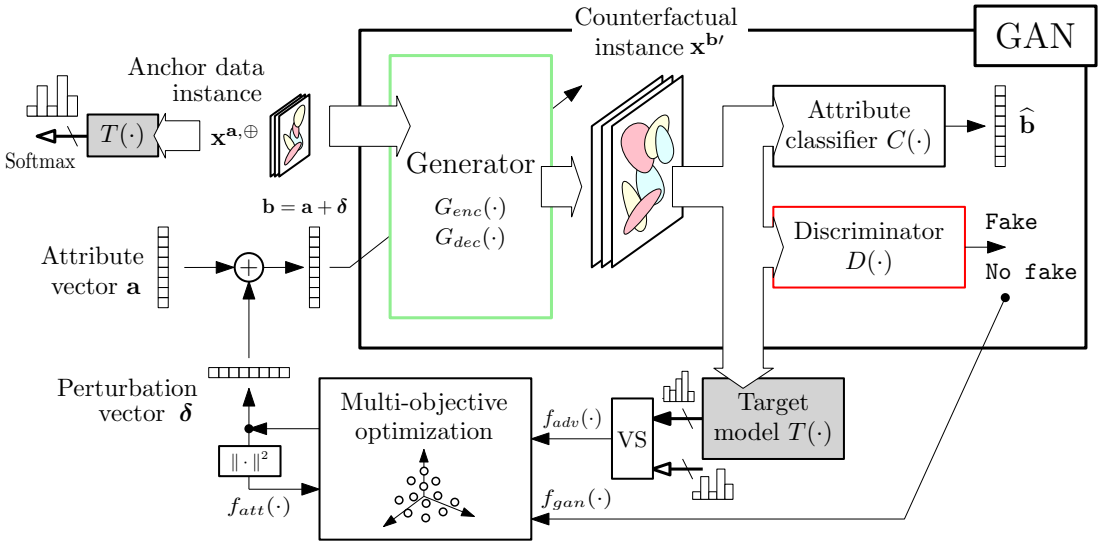


Figure 4.2: Block diagram of the proposed framework, which is capable of producing counterfactual instances for an audited model $T(\cdot)$ based on three criteria: plausibility, adversarial power and change intensity.

sually realistic; and 2) whether the predicted attributes match the input ones. Again, for non-conditional GAN architectures, only the discriminator $D(\mathbf{x}')$ is necessary.

The overall loss function that drives the learning algorithm of the generator and discriminator is defined as a linear combination of the reconstruction and Wasserstein GAN losses. The training loss for encoder $G_{enc}(\mathbf{x}^{\mathbf{a}})$ and decoder $G_{dec}(\mathbf{z}, \mathbf{b})$ are given by:

$$\min_{G_{enc}, G_{dec}} \lambda_1 \mathcal{L}_{rec}(\mathbf{x}^{\mathbf{a}}, \mathbf{x}^{\mathbf{a}\prime}) + \lambda_2 \mathcal{L}^{G}_{att}(\mathbf{b}, \widehat{\mathbf{b}}') + \mathcal{L}^{G}_{adv}(\mathbf{x}^{\mathbf{b}\prime}), \tag{16}$$

where:

$$\mathcal{L}_{rec}(\mathbf{x}^{\mathbf{a}}, \mathbf{x}^{\mathbf{a}\prime}) = \mathbb{E}_{\mathbf{x}^{\mathbf{a}} \sim P_X(\mathbf{x})} \left[\|\mathbf{x}^{\mathbf{a}} - \mathbf{x}^{\mathbf{b}\prime}\|_1 \right], \tag{17}$$

$$\mathcal{L}^{G}_{att}(\mathbf{b}, \widehat{\mathbf{b}}') = \mathbb{E}_{\mathbf{x}^{\mathbf{a}} \sim P_X(\mathbf{x}), \mathbf{b} \sim P_B(\mathbf{b})} \left[\sum_{n=1}^{N=|\mathbf{b}|} H(b_n, \widehat{b_n}') \right], \tag{18}$$

$$\mathcal{L}^{G}_{adv}(\mathbf{x}^{\mathbf{b}\prime}) = -\mathbb{E}_{\mathbf{x}^{\mathbf{a}} \sim P_X(\mathbf{x}), P_B(\mathbf{b})} \left[D(\mathbf{x}^{\mathbf{b}\prime}) \right]. \tag{19}$$

In the above expressions, $\mathbb{E}[\cdot]$ denotes expectation; $P_B(\mathbf{b})$ indicates the distribution of possible attribute vectors $\mathbf{b} = \{b_n\}_{n=1}^{N} \in \mathbb{R}^N[0, 1]$; $H(b_n, \widehat{b_n}')$ is the cross-entropy of binary distributions given by b_n and $\widehat{b_n'} \in \widehat{\mathbf{b}}' = C(\mathbf{x}^{\mathbf{b}'})$; and $D(\mathbf{x}^{\mathbf{b}\prime}) = 0$ if $\mathbf{x}^{\mathbf{b}\prime}$ is predicted to be a fake.

When it comes to the discriminator $D(\cdot)$ and the classifier $C(\cdot)$, their training loss is given by:

$$\min_{D,C} \lambda_3 \mathcal{L}^{C}_{att}(\mathbf{x}^{\mathbf{a}}, \mathbf{a}) + \mathcal{L}^{D}_{adv}(\mathbf{x}^{\mathbf{a}}, \mathbf{b}), \tag{20}$$

with:

$$\mathcal{L}_{att}^{C}(\mathbf{x}^{\mathbf{a}}, \mathbf{a}) = \mathbb{E}_{\mathbf{x}^{\mathbf{a}} \sim P_X(\mathbf{x})} \left[\sum_{n=1}^{|\mathbf{a}|} H(a_n, \widehat{a_n}') \right], \tag{21}$$

$$\mathcal{L}_{adv}^{D}(\mathbf{x}^{\mathbf{a}}, \mathbf{b}) = -\mathbb{E}_{\mathbf{x}^{\mathbf{a}} \sim P_X(\mathbf{x})} \left[D(\mathbf{x}^{\mathbf{a}}) \right] + \mathbb{E}_{\mathbf{x}^{\mathbf{a}} \sim P_X(\mathbf{x}), P_B(\mathbf{b})} \left[D(\mathbf{x}^{\mathbf{b}\prime}) \right], \tag{22}$$

where $\widehat{a_n}' \in C(\mathbf{x}^{\mathbf{a}})$, and coefficients $\{\lambda_i\}_{i=1}^{3}$ permit to balance the importance of the above terms during the training process of the GAN architecture. For more general approaches, such as non-conditional GANs, the training loss is given by:

$$\min_{G_{enc}, G_{dec}} \lambda_1 \mathcal{L}_{rec}(\mathbf{x}, \mathbf{x}') + \mathcal{L}_{adv}^{G}(\mathbf{x}'), \tag{23}$$

where:

$$\mathcal{L}_{rec}(\mathbf{x}, \mathbf{x}') = \mathbb{E}_{\mathbf{x} \sim P_X(\mathbf{x})} \left[\|\mathbf{x} - \mathbf{x}'\|_1 \right], \tag{24}$$

$$\mathcal{L}_{adv}^{G}(\mathbf{x}') = -\mathbb{E}_{\mathbf{x} \sim P_X(\mathbf{x})} \left[D(\mathbf{x}') \right], \tag{25}$$

and again, coefficient $\lambda_1 \in \mathbb{R}[0, 1]$ allows tuning the relative importance of the reconstruction loss when compared to the adversarial loss. Once these losses have been defined, the GAN is trained via back-propagation to minimize the losses in Expressions (16) and (20) when measured over a training dataset.

Once trained, we exploit the GAN architecture to find counterfactual examples for a given test sample $\mathbf{x}^{\mathbf{a}\oplus,\oplus}$ and an audited model $T(\mathbf{x})$, with classes $\{\texttt{label}_1, \ldots, \texttt{label}_L\}$. Specifically, we model the counterfactual generation process as a perturbation inserted into the attribute vector $\mathbf{a}$ of the test sample, i.e. $\mathbf{b} = \mathbf{a} + \delta$, with $\delta \in \mathbb{R}^N$. This perturbed attribute vector, through G_{enc} and G_{dec}, yields a plausible image $\mathbf{x}^{\mathbf{b},\prime}$ that, when fed to the target model $T(\cdot)$, changes its predicted output. The conflict between adversarial power, plausibility and intensity of the perturbation from which the counterfactual example is produced gives rise to the multi-objective problem formulated as:

$$\min_{\delta \in \mathbb{R}^N} f_{gan}(\mathbf{x}^{\mathbf{a},\oplus}, \delta; G, D), f_{adv}(\mathbf{x}^{\mathbf{a},\oplus}, \delta; G, T), f_{att}(\delta), \tag{26}$$

where:

- $f_{gan}(\mathbf{x}^{\mathbf{a},\oplus}, \delta; G, D)$ quantifies the *unlikeliness* (no plausibility) of the generated counterfactual instance through $G(\cdot)$, which is given by the difference between the output of the discriminator $D(\cdot)$ corresponding to $\mathbf{x}^{\mathbf{a},\oplus}$ and $\mathbf{x}^{\mathbf{b},\prime}$ (Wasserstein distance). The more negative this difference is, the more confident the discriminator is about the plausibility of the generated counterfactual $\mathbf{x}^{\mathbf{b},\prime}$;

- $f_{adv}(\mathbf{x}^{\mathbf{a},\oplus}, \delta; G, T)$ informs about the probability that the generated counterfactual does not confuse the target model $T(\cdot)$, which is given by the negative value of the cross-entropy of the soft-max output of the target model when queried with counterfactual $\mathbf{x}^{\mathbf{b},\prime}$; and

- $f_{att}(\delta)$ measures the intensity of adversarial changes made to the input image $\mathbf{x}^{\mathbf{a},\oplus}$, which is given by $\|\delta\|_2$. As we will later discuss, this measure can be

replaced by other measures of similarity that do not operate over the perturbed attribute vector, but rather over the produced counterfactual image (for instance, structural similarity index measure SSIM between $\mathbf{x}^{\mathbf{a},\oplus}$ and $\mathbf{x}^{\mathbf{b},\prime}$).

To efficiently find a set of input parameter perturbations $\{\delta\}$ balancing among the above three objectives in a Pareto-optimal fashion, we resort to multi-objective optimization algorithms. Specifically, we opt for derivative-free meta-heuristic solvers, which allow efficiently traversing the search space $\mathbb{R}^N$ of decision variables δ and retaining progressively better non-dominated counterfactual instances without requiring any information of the derivatives of the objectives under consideration.

Algorithmus 2 : Generation of multi-criteria counterfactuals

Input : Target model to be audited $T(\mathbf{x})$; GAN architecture (G, D); attribute classifier $C(\mathbf{x})$; annotated training set $\mathcal{D}_{train}$; test image $\mathbf{x}^{\mathbf{a},\oplus}$ for counterfactual study; weights $\{\lambda_i\}_{i=1}^{3}$

Output : Multi-criteria counterfactuals balancing between $f_{gan}(\cdot)$, $f_{adv}(\cdot)$ and $f_{att}(\cdot)$

1 Train GAN architecture via back-propagation over training dataset and loss functions in Expressions (16) and (20)
2 Initialize a population of perturbation vectors $\delta \in \mathbb{R}^N$
3 **while** *stopping criterion not met* **do**
4 Apply search operators to yield offspring perturbation vectors
5 Evaluate $f_{gan}(\cdot)$ (*plausibility*), $f_{adv}(\cdot)$ (*adversarial success*) and $f_{att}(\cdot)$ (*change intensity*) of offspring perturbations
6 Rank perturbations in terms of Pareto optimality
7 Retain the Pareto-best perturbations in the population
8 **end**
9 Select non-dominated perturbations from population
10 Produce counterfactual images by querying the GAN with $\mathbf{x}^{\mathbf{a},\oplus}$ and each selected perturbation vector

Algorithm 2 summarizes the process of generating counterfactuals for target model $T(\cdot)$, comprising both the training phase of the GAN architecture and the meta-heuristic search for counterfactuals subject to the three conflicting objectives. The overall framework departs from the training process of a GAN architecture (line **1**) over a training dataset $\mathcal{D}_{train}$ that collects samples (images) annotated with their attribute vectors $\mathbf{a}$ (only for conditional GANs). Once trained and similarly to the usual workflow of population-based heuristic solvers, the algorithm initializes uniformly at random a population of perturbation vectors (line **2**), which are iteratively evolved and refined (lines **3** to **8**) as per the Pareto optimality of the counterfactual images each of them produces. To this end, evolutionary search operators (crossover and mutation) are applied over the population (line **4**) to produce offspring perturbation vectors, which are then evaluated (line **5**) and ranked depending on their Pareto dominance (line **6**). By keeping in the population those perturbation vectors that score best in terms of Pareto optimality (line **7**) and iterating until a stopping criterion is met, the framework ends up with a population of Pareto-superior perturbation vectors

(line **9**) that can be inspected visually to understand which image components affect most along the direction of each objective (line **10**).

4.2.3 *Target Audiences and Examples of Use Cases*

To round up the presentation of the proposed framework and and the whole thesis, we pause briefly at the target audiences envisioned for its use, as well as a sketch of use cases that could illustrate its use in practical settings. Many examples could be used to exemplify these scopes, among which three specific areas currently under active investigation are chosen: bio-metric authentication, the discovery of new materials and creative industrial applications. These three use cases target two different audiences: developers and final users.

The use of bio-metric authentication is extensive nowadays in a manifold of sectors managing critical assets. However, auditing machine learning models used for bio-metric authentication is not straightforward. They can be audited by adversarial attack testing, but this analysis focuses on subtle (namely, not noticeable) adversarial perturbations made to an input of the audited model. Therefore, they aim at analyzing the robustness of the model against malicious attacks designed not to be easily detectable (e.g. one-pixel attacks), rather than at discerning which plausible inputs can lead to a failure of the authentication system even if not deliberately designed for this purpose. The framework presented in this chapter can be of great value for developers to explore the reality-bound limitations of their methods, helping them determine complementary information requested during the process to increase the robustness of the model against plausible authentication failures.

New material discovery is also a field in which high-dimensional datasets are utilized. The addition of our proposed framework might help experts reduce the amount of non-plausible composites to be synthesized, or to discover diverse alternative materials with differing properties in terms of elasticity, conductivity or thermal expansion, to cite a few. This would in turn ease the practice of material experts by considerably reducing the space of possible materials to be explored and opening new possibilities in their laboratory processes without requiring any technical knowledge in Artificial Intelligence.

Finally, we highlight the possibilities brought by the proposed framework for the creative industry. Such a framework could be coupled with a design software so that it would help in the generation of creative content by proposing new alternatives of already produced products (e.g. new designs of mechanical components, new audio-visual pieces, novel architectural proposals) with varying levels of compliance with respect to plausibility, amount of the change and properties that are specific to the use case at hand. In essence, the framework could be of great use for aiding the creative process hand in hand with the expert.

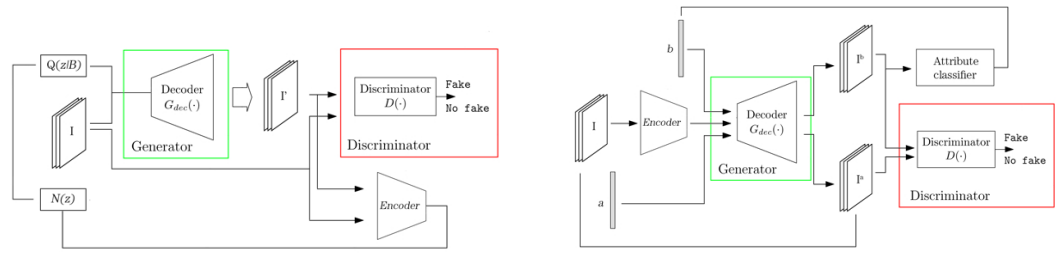
4.3 EXPERIMENTAL SETUP

This section introduces the actual architectures and models that were used to prove the framework. Five GAN architectures are presented, followed by five third party classifier that were audited in the experiments. All GAN architectures are extracted

from the literature as pre-trained cases from the original authors themselves. The classifiers are trained with the test sets of each of the GAN dataset to assure the same data domain is maintained and no knowledge leakage is produced. All the source code for reproducing these experiments will be released at https://github.com/alejandrobarredo/COUNTGEN-Framework, together with a Python library that can be used for applying this framework over custom datasets.

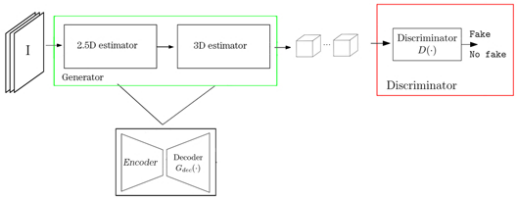
4.3.1 *Considered GAN Architectures*

The architectures utilized fall under three main GAN categories. Although each of them consists of a particular implementation containing its particular caveats. The different GAN approaches are: Conditional GAN, Unconditional GAN and a combination of both.

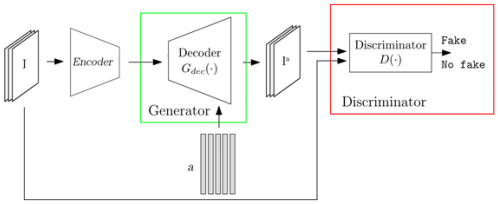


(a) BicycleGAN

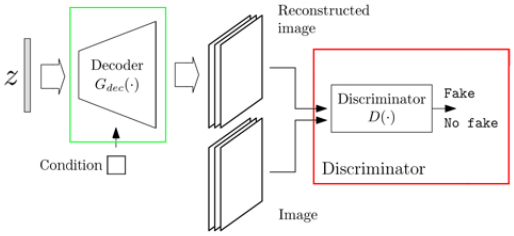
(b) AttGAN



(c) ShapeHDGAN



(d) StyleGAN2



(e) Conditional GAN

Figure 4.3: Block diagram of the proposed systems comprising the (a) BicycleGAN; (b) AttGAN; (c) ShapeHDGAN; (d) StyleGAN2; and (e) Conditional GAN (CGAN).

4.3.1.1 *BicycleGAN*

This first BicycleGAN architecture combines conditional and unconditional GAN architectures for the task of image-to-image translation [505]. To this end, BicycleGAN

generates the output as a distribution of solutions in a conditional generative setting. The mapping is disambiguated through a latent vector which can be sampled at test time. The authors present their solution as an improvement for the known *mode collapse* problem, since it reduces the pitfall of having one-to-many solutions as a result of utilizing a low-dimensional latent vector.

As shown in the diagram of Figure 4.3a, BicycleGAN combines conditional and unconditional GAN architectures to generate their own. The first part, highlighted in green, is that of a cVAE-GAN [506, 507]: the model first encodes the ground truth into a latent space, and then it is reconstructed by means of a generator trained with a Kullback–Leibler divergence loss. The second combined model is a cLR-GAN [508–510]: contrarily to the first part, the cLR-GAN departs from a randomly generated latent vector, while the encoder is trained from recovering it from the output image created in the generator. Finally, the combination of these two different constraints form the BicycleGAN architecture, which enforces the connection between the output and latent code simultaneously for both directions. This resulting architecture is able to generate more diverse and appealing images for every image-to-image translation problem.

The implementation of the network was retrieved from [511] with the pre-trained models utilized for the experiments covered in the next section. The architecture consists of a U-Net [512] generator $G(\cdot)$, which in turn contains an encoder-decoder architecture with symmetric skip connections. The discriminator $D(\cdot)$ is composed as a combination of two PatchGAN [513] of different scales which resolve the fake/real prediction for 70×70 and 140×140 image patches. Finally, for the standalone encoder $G_{enc}(\cdot)$, a ResNet [514] is utilized. Further information about the structure and training process of the BicycleGAN architecture can be found in [505].

4.3.1.2 *AttGAN*

This second architecture presents a conditional GAN capable of editing facial attributes of human faces while preserving the overall detail of the image [504]. In the seminal work presenting this architecture, the training process is performed by conditioning the latent vector to match the vector representing the given facial attributes for the image at hand. The network is devised such that this vector is real-valued, which allows for the inference of facial attributes for a given intensity. During inference, attributes can be changed by modifying the values of the variables in the latent vector.

Figure 4.3b depicts a diagram of the AttGAN model, which is trained by means of two constraining conditions. For one, the model attempts to match the input attributes with the predicted attributes at the end of the architecture. For the other, the model is constrained to match the generated image to that at its input. The latter is governed by a reconstruction loss. The former, forcing the latent vector to match the attributes of the images, is governed by a standard cross-entropy loss. The combination of these two constraints result in a model capable of generating faces with varying attributes and remarkable realism.

The implementation of the network is that available at [515]. The discriminator $D(\cdot)$ is composed of a stack of convolutional layers followed by fully-connected layers. The classifier $C(\cdot)$ shares all the convolutional layers from $D(\cdot)$, and follows the same

structure ended in fully-connected layers. The encoder $G_{enc}(\cdot)$ is composed of several convolutional layers, while the decoder $G_{dec}(\cdot)$ is composed of a stack of transposed convolutional layers. As in BicycleGAN, a symmetrical skip connection is set between the encoder and decoder. Further information about the architecture and training process can be accessed in [505].

4.3.1.3 *ShapeHD*

This third ShapeHDGAN architecture is capable of rendering 3D meshes of objects from single 2D views. This particular task is of great complexity given that the solution landscape is composed of countless shapes that do not pertain to an object and renders them implausible. Most existing approaches fail at generating detailed objects. ShapeHDGAN gives a solution to this problem by virtue of an generative environment with adversarially learned shape priors that serve the purpose of penalizing if the model renders unrealistic meshes.

As shown in Figure 4.3c the model consists of two main components. A 2.5D sketch estimator and a 3D shape estimator that predicts a 3D object from an image. It consists of three stages. In the first stage, the 2.5D estimator – a encoder-decoder structure – predicts the object depth, normals and silhouette from a RGB image. Then, the second stage generates a 3D shape from the previous 2.5D sketch. The last stage is composed by an adversarially trained CNN that tunes the generated shape into a real object.

The implementation was retrieved from [516]. The 2.5D sketch estimator is composed by a ResNet-18 encoder $G_{enc}(\cdot)$ mapping a 256×256 image into 512 feature maps of size 8×8. The $G_{dec}(\cdot)$ model has four stacked transposed convolutional layers. The predicted silhouette permits to mask the depth and normal estimations to be then used as the input of the 3D generator. The 3D shape estimator is also composed of an encoder-decoder architecture. The encoder is an adapted from a ResNet-18 to handle 4 channels and encode them into a 200-dimensional latent vector. The decoder comprises five stacked transposed convolutional layers, which generate a $128 \times 128 \times 128$ voxel at its output. Further details are available at [517].

4.3.1.4 *StyleGAN2*

StyleGAN [14] is a unconditional GAN architecture with one of the most realistic results for unconditional generative image modeling. For this study, we choose the StyleGAN2 implementation, which is a revised variant that improves upon the artifacts of the original StyleGAN model [518] by virtue of small albeit intelligently devised modifications to the generator model of the original StyleGAN model. The implementation was retrieved from [519] with the pre-trained models for the experiments carried out in the following sections.

4.3.1.5 *Conditional GAN*

Finally we decided to add a last model that allows us to explore some variations within. This time, we selected a well-known conditional GAN architecture [520] trained over the MNIST image classification dataset. The conditional GAN departs

from a random noise vector and a single variable that acts as a condition for the generation process. In this way, the generative network learns to switch between the learned distributions for each label by means of a input condition. This feature resembles to that of AttGAN, with the difference that in this one, the models does not start from an encoding.

Figure 4.3e shows the structure of the conditional GAN model. The implementation was retrieved from the public python library GANS2 [521] which includes a set of ready-to-build, plug-and-play GAN architectures.

4.3.2 *Audited Classification Models*

After introducing the GAN models under consideration, we now introduce the models that will be audited by means of our GAN-based counterfactual generation framework. For the experiment utilizing BicycleGAN, a classifier is trained to predict the type of footwear corresponding to the image fed at its input (`Shoe` versus `NoShoe`). For the case considering AttGAN, the classifier to be audited predicts whether the human face input to the model corresponds to a `male` or to a `female`. When the framework considers ShapeHDGAN, the classifier is trained to distinguish between a `chair` and a `Xbox`. For StyleGAN2, the classifier discriminates whether the input image is a `cathedral` or an `office`. Finally, the classifier audited by our framework configured with the cGAN aims to address a multi-class classification problem over the same MNIST dataset, yet ensuring that different data partitions are used for training the cGAN model and the classifier itself.

These third-party models consist of several convolutional layers, ending in a series of fully-connected layers that connect the visual features extracted by the former with the categories defined in the dataset under consideration. Every classifier model was trained with the test data that was not used for training the corresponding GAN architecture, thereby ensuring no information leakage between the generators and the third-party models to be audited. Table 4.1 summarizes the topological configuration of the models for which counterfactuals are generated by our framework, as well as the training parameters set for every case.

The accuracy achieved by the trained classifiers over a 20% holdout of their dataset are reported in Table 4.2, together with the number of classes, total examples to train and validate the audited model, and the class balance ratio. As can be observed in this table, the audited models reach a very high accuracy (over 95% in most cases), so that the adversarial success of the produced counterfactual examples can be rather attributed to the explanatory capabilities of the devised framework than to a bad performance of the audited classifier.

4.3.3 *Multi-objective Optimization Algorithm*

We recall that the optimizer is in charge for tuning the output of the GAN generator to 1) maximize the difference in the result of the audited classifier (adversarial power); 2) minimize the amount of changes induced in the produced counterfactual parameters (change intensity); and 3) maximize the Wasserstein distance between the real and fake examples (plausibility).

Table 4.1: Structure and training parameters of the models audited by the proposed framework.

GAN	Audited classifier $T(\mathbf{x})$	
	Network architecture	Training parameters
BicycleGAN	Conv2d(64, 3×3, ReLu) + Conv2d(32, 3×3, ReLu) + Dense(1, Sigmoid)	Adam, binary cross-entropy loss
AttGAN	Conv2d(16, 3×3, ReLu) + Dropout(0.1) + Conv2d(4, 3×3, ReLu) + Dense(1,Sigmoid)	Adam, binary cross-entropy loss
ShapeHDGAN	Conv2d(32, 3×3, ReLu) + BatchNorm + MaxPooling(2×2) + Conv2d(8, 3×3, ReLu) + BatchNorm + MaxPooling(2×2) + Dense(100,ReLu) + Dense(1, Sigmoid)	SGD(0.01, 0.9), binary cross-entropy loss
StyleGAN2	Conv2d(16, 3×3, ReLu) + Dropout(0.1) + Conv2d(4, 3×3, ReLu) + Dense(1,Sigmoid)	Adam, binary cross-entropy loss
cGAN	Conv2d(32, 3×3, ReLu) + BatchNorm + MaxPooling(2×2) + Dense(100,ReLu) + Dense(10,SoftMax)	SGD(0.01, 0.9), categorical cross-entropy loss

Conv2d(A,B,C): convolutional layer with A filters of size B and activation C.

SGD(l,m): Stochastic Gradient Descent with learning rate l and momentum m.

In all cases the batch size is set to 16 instances, and the number of epochs is 10.

Flattening operations are not displayed for clarity.

Table 4.2: Dataset and accuracy of the different classifiers put to the test

GAN	Dataset	# Examples	Classes	Class Balance	Accuracy	Source
BicycleGAN	`Edges2Shoes`	300	2	45%/55%	94%	[513]
AttGAN	`CelebA`	900	2	49%/51%	98%	[522]
ShapeHDGAN	`ShapeNet`	600	2	49%/51%	96%	[523]
StyleGAN2	`Style`	540	2	49%/51%	98%	[524]
cGAN	`MNIST`	9000	10	10% each	96%	[525]

The search for counterfactual instances optimally balancing among these objectives can be efficiently performed by using a multi-objective evolutionary algorithm. Among the multitude of approaches falling within this family of meta-heuristic solvers, we select NSGA2 [526] with a population size of 100 individuals, 100 offspring produced at every generation, polynomial mutation with probability 1/N (with N denoting the number of decision variables, which vary depending on the experiment and GAN under consideration) and distribution index equal to 20, SBX crossover with probability 0.9 and distribution index 20, and 50 generations (equivalent to 5000 evaluated individuals per run). The use of this optimizer allows for a genetic search guided by non-dominated sorting in the selection phase, yielding a Pareto-dominant set of counterfactual examples that constitute the output of the framework. For its implementation we rely on the jMetalPy library for multi-objective optimization [527].

4.4 RESULTS AND DISCUSSION

We now discuss on the results obtained from the experiments described above, articulating the discussion around the provision of an informed response to three main research questions:

Q1. Is counterfactual generation an optimization problem driven by several objectives?

Q2. Do the properties of the generated counterfactual examples conform to general logic for the tasks and datasets at hand?

Q3. Do multi-criteria counterfactual explanations serve for broader purposes than model explainability?

Answers to each of these research questions will be summarized after an analysis and discussion held over the produced counterfactual examples for each of the audited models detailed in Table 4.1. For every experiment, we draw at random one anchor image $\mathbf{x}^{\mathbf{a},\oplus}$ from the test partition of the audited model and inspect the produced set of counterfactual variants both visually and quantitatively as per the three objectives under consideration. This examination of the results will be arranged similarly across experiments, portraying the output of the framework in a three-dimensional plot comprising the Pareto front approximated by the multi-objective solver. Each of the axes of this plot is driven by one of such objectives: change intensity $f_{att}(\cdot)$, adversarial power $f_{adv}(\cdot)$ and plausibility $f_{gan}(\cdot)$, all defined in Subsection 4.2.2. It is important to note that for easing the visualization of the fronts, plausibility and adversarial power are inverted by displaying $1 - f_{gan}(\cdot)$ and $1 - f_{adv}(\cdot)$, so that $1 - f_{gan}(\cdot) \geqslant 0.5$ denotes the region over which the counterfactual can be considered to be plausible. Similarly, the higher $1 - f_{adv}(\cdot)$ is, the larger the difference between the outputs of the audited model when fed with the anchor image $\mathbf{x}^{\mathbf{a},\oplus}$ and its counterfactual variant will be (larger *adversarial power*).

In the depicted Pareto front approximations for every experiment, several specific counterfactual examples scattered over the front are highlighted with colored markers. These markers refer to the images plotted on the right of the figure, so that it is possible to assess the counterfactual image/voxel corresponding to each of such points. The first image shown in the top row of images shown on the right of the figure represents the reference (anchor) image $\mathbf{x}^{\mathbf{a},\oplus}$, which is the departing point for the counterfactual generation. The first image shown in the bottom row of images is always the image belonging to the opposite class (or a targeted class in the case of the MNIST dataset) whose soft-max output corresponding to its class is lowest (worst predicted example of the other class existing in the dataset). Below every image, a bar diagram can be observed representing the value of the objectives corresponding to the image at hand.

We now proceed with a detailed discussion for every experiment.

4.4.1 *Experiment #1: BicycleGAN-based Counterfactual Generation for Auditing a* `Shoe` *Versus* `NoShoe` *footwear classifier*

The outcomes of this experiment are shown in Figure 4.4. In this figure a coloring pattern is distinguishable over all the counterfactual examples highlighted in the approximated front. The image of the man shoe that serves as the anchor image $\mathbf{x}^{\mathbf{a},\oplus}$ appears to be complete. However, original colors are removed and uniformized all over the image. This fact informs about the influence of the color on the predicted label of the model.

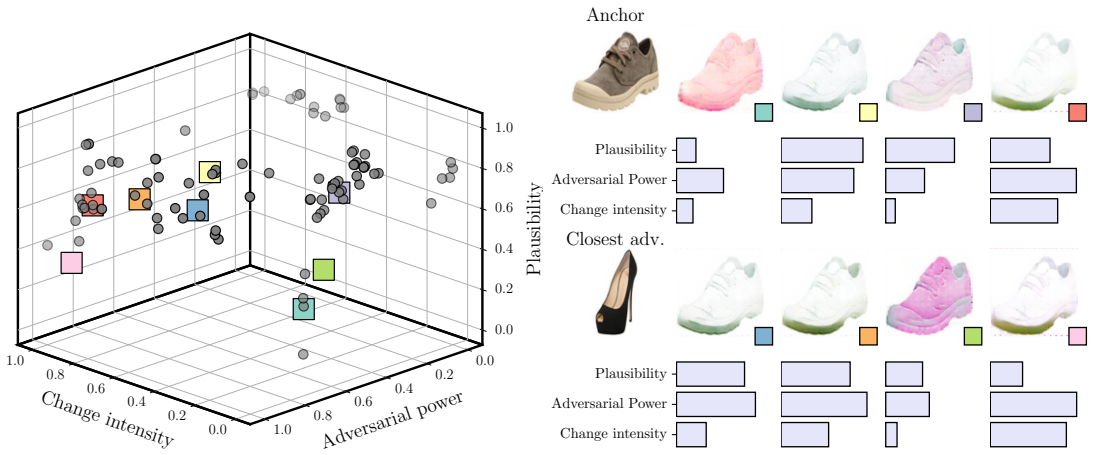


Figure 4.4: Pareto front of the counterfactual examples generated for a `Shoe` example by the proposed framework configured with a BicycleGAN model.

Returning to our intuition exposed in Section 4.2, two concerns must be kept in mind when analyzing these results. First, understanding the constraints of the dataset in use is of paramount importance. The data with which the classifier was trained is composed by different footwear instances. However, this dataset accounts just for a limited subset of the different possible footwear instances available in reality for both classes. This fact will make the predictions of the classifier change sharply between one class and the other when the instance for which it is asked does not conform to the class-dependent distribution of the training dataset. The second concern refers to the spread in the prediction scores. The solution front shows quite a nice spread in the prediction scores at first glance. However, this spread of solutions in the objective space does not entail that the corresponding counterfactual instances are visually diverse. Figure depicts just 8 out of the 100 solutions in the approximated Pareto front, but they suffice to showcase that every generated counterfactual is very similar to each other with the exception of color. This suggests that the classifier is very susceptible to the color feature, and that the shape of the footwear is so relevant for the task that the counterfactual generation process needs to retain this feature to ensure plausibility. This bias is one of the insights provided by the proposed framework in this first experiment.

(a) NoShoes class (b) Shoes class

Figure 4.5: Analysis of the average RGB luminance $\ell(\text{RGB})$ of the Shoe vs NoShoe dataset used to train the target classifier for the BicycleGAN experiment, together with some few examples of every class.

The aforementioned statement is supported by Figure 4.5, which depicts the mean luminance of RGB pixels averaged over all the training examples of every class used for the audited model. Luminance has been computed as:

$$\ell(\text{RGB}) = (0.2126 \cdot \text{R} + 0.7152 \cdot \text{G} + 0.0722 \cdot \text{B})\,/255, \tag{27}$$

where $\ell(\text{RGB}) \in \mathbb{R}[0, 1]$ denotes a measure of luminance (0: dark, 1: light) of a pixel with R (red), G (green) and B (blue) channel values. As it can be observed in the bottom left plot of this figure, shoe instances have a clear bias in terms of footwear shape and image orientation, whereas the central part of the footwear for both classes is darker that the background. This is the reason why our proposed framework operates exclusively on the color feature and maintains the shape of the footwear when attempting at producing a counterfactual example for a shoe, yielding differently (brighter) colored yet identically shaped variants of the anchor.

4.4.2 *Experiment #2: AttGAN-based Counterfactual Generation for Auditing a Man Versus Woman gender classifier*

The outcome of the devised framework corresponding to this second experiment is shown in Figure 4.6. In this case, the reference image $\mathbf{x}^{\mathbf{a},\oplus}$ is an instance of the Man class from the Celebrity dataset.

We begin by inspecting the shape of the produced Pareto front approximation in the right plot of the figure. It can be observed that diverse counterfactual explanations are found in the trade-off between plausibility and adversarial power, as well as between the intensity of the change and plausibility. By contrast, adversarial power and change intensity seem not to be conflicting with each other. The reason for the uncoupled behavior of these two objectives may reside in the characteristics of the dataset and GAN in use: a small perturbation in the attribute vector imprints already enough changes in the generated counterfactual image to mislead the audit classifier, whereas larger perturbations degrade their plausibility. This is clear as per the range

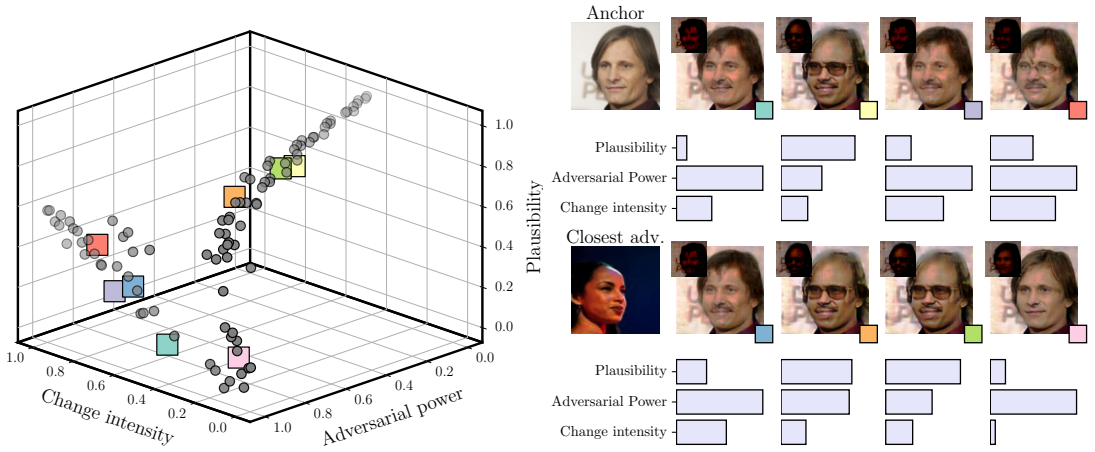


Figure 4.6: Pareto front of the counterfactual examples generated for a `male` example by the proposed framework configured with a AttGAN model. Every nested couple of images on the right represents the original produced counterfactual (small image located in the left corner of every plot) and its contrast adjusted version using histogram equalization [528].

of plausibility values covered by the points in the front with varying change intensity, which are always kept below the plausibility boundary ($1 - \mathrm{f}_{\mathtt{gan}}(\cdot) \leqslant 0.5$).

When qualitatively examining the generated counterfactuals, the plots nested on the right of the figure reveal that once again, the luminance is a deciding factor for adversarially modifying the anchor image. Leaving aside modifications over the color space, it is important to note that the plausibility of counterfactuals seems to be tightly linked to the insertion of glasses or a smiling pose. On the contrary, counterfactuals that produce an intense drift towards the `Female` class in the audited classifier insert long blonde hair into the anchor image. In this experiment, these patterns are related with the constraints imposed by the dataset. However, differently from the previous experiment, the produced counterfactuals are not exiting the data domain over which the model was trained, but are rather exploiting biases existing in the data. Most counterfactuals seen in the front have blonde hair, glasses or a smile pose, whether alone or combined.

In order to explore the reason for such a recurring set of counterfactual features, Figure 4.7 depicts bar diagrams showing the differences in terms of occurrence over the training examples of different combinations of the three attributes, differentiating between counts corresponding to the `male` and `female` classes. It is straightforward to note that the majority of examples featuring any of the combinations of these three attributes belong to the `female` class. Given a face, if it contains those three attributes, it is quite probably a female. This is why the framework produces counterfactuals with these features.

4.4.3 *Experiment #3: ShapeHDGAN-based Counterfactual Generation for Auditing a `Chair` Versus `Xbox` voxel classifier*

This third case of the devised set of experiments comprises an audited classifier that discriminates whether the voxel at its input is a `chair` or a `Xbox`. Therefore, it operates

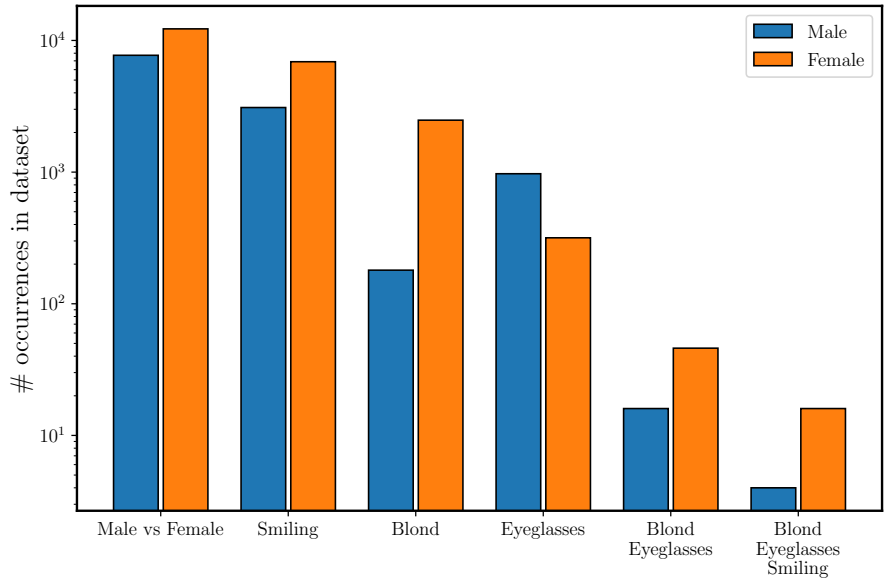


Figure 4.7: Diagram showing the occurrence within the training dataset of the audited model of different feature combinations, split between `male` and `female` examples.

over three-dimensional data, increasing the complexity to qualitatively evaluate the produced counterfactuals with respect to previous experiments.

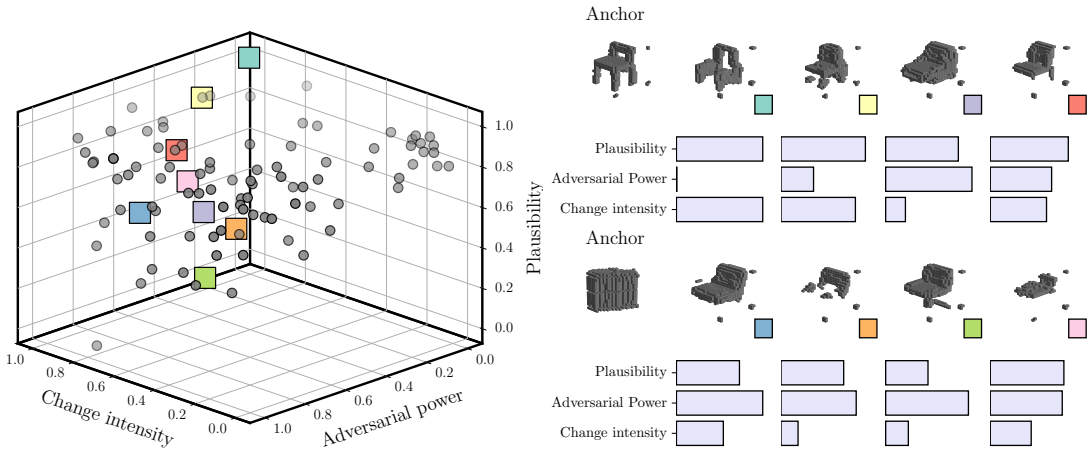


Figure 4.8: Pareto front of the counterfactual examples generated for a `chair` example by the proposed framework configured with a Shape3DGAN model.

The results elicited for a `chair` target instance $\mathbf{x}^{\mathbf{a},\oplus}$ are shown in Figure 4.8. A first inspection of the counterfactual voxels highlighted in the approximated Pareto front suggests that it is hard to analyze what the audited classifier observes in these inputs to get fooled and predict a `Xbox`. It appears that a more dense middle part is capable of misleading the classifier. Voxels being generated by the framework resemble a chair, but possess a clearly more dense middle part. It is quite revealing to see how a chair and an xbox can be of any resemblance. Interestingly, a concern to bring up here is that of scale. These voxels (both the generated ones and the dataset over which the ShapeHDGAN model was trained) are normalized, which in turn means that the size of the object has been lost. Scaling can be an interesting option to improve the

resolution of small objects. In this case, however, it can be the reason to make this classifier prone to error.

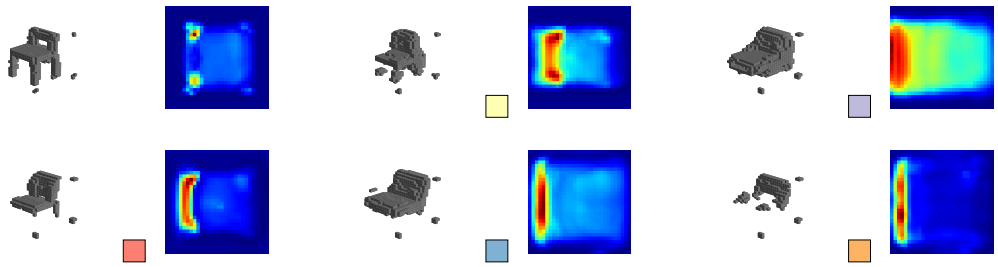


Figure 4.9: Local explanations (heatmaps via Grad-cam++) corresponding to the anchor voxel (leftmost pair of images) and two of the counterfactuals depicted in Figure 4.8.

This last statement can be buttressed by analyzing which structural parts of the counterfactual voxel are of highest importance for the audited classifier to produce its prediction. This can be done by resorting to gradient-based local post-hoc explanation methods such as Grad-cam++ [487]. As can be seen in the examples depicted in this Figure, most of the observational focus of the model is placed on the vertical rectangular part of the `chair`, which conforms to intuition given that the actual shape of a `Xbox` is rectangular. Therefore, counterfactuals for a `chair` instance wherein the vertical part (*backrest*) is reinforced can bias the audited model without jeopardizing their plausibility.

4.4.4 *Experiment #4: StyleGAN2-based Counterfactual Generation for Auditing a classifier of `Cathedral` Versus `Office` classifier*

The results of this fourth experiment (Figure 4.10) unveils a link between the luminance of the overall counterfactual image and its ability to mislead the model. However, in this time the spread of counterfactuals over the adversarial power dimension of the Pareto front is notably lower than in the previous experiments. If the results are compared with those of 4.4, in this case the missing spread observed in the front is validated with what can be visually discerned in the counterfactuals given by the framework.

This is supported by the analysis of the visual differences between the anchor image and the produced counterfactuals shown in Figure 4.11. Specifically, the plot shows the heatmap of mean absolute differences (averaged over the RGB channels) and the SSIM (Structural Similarity Index Measure [529]) among the original anchor image and its counterfactual version. As can be inferred from the visuals in the first row of the figure, the framework is exploiting a burned-out background with minor structural changes in the image. This seems reasonable with the spread found in the front: it shows that these changes in the background of the image can completely fool the model, but there are not changes that would account for a well spread front since the structural differences among both classes are large. Furthermore, `cathedral` instances undergo a misrepresentation bias in the dataset, in the sense that none of the `cathedral` training examples has a totally overcast sky. This suggests that whitening the background of the image may grant a chance for the counterfactual to

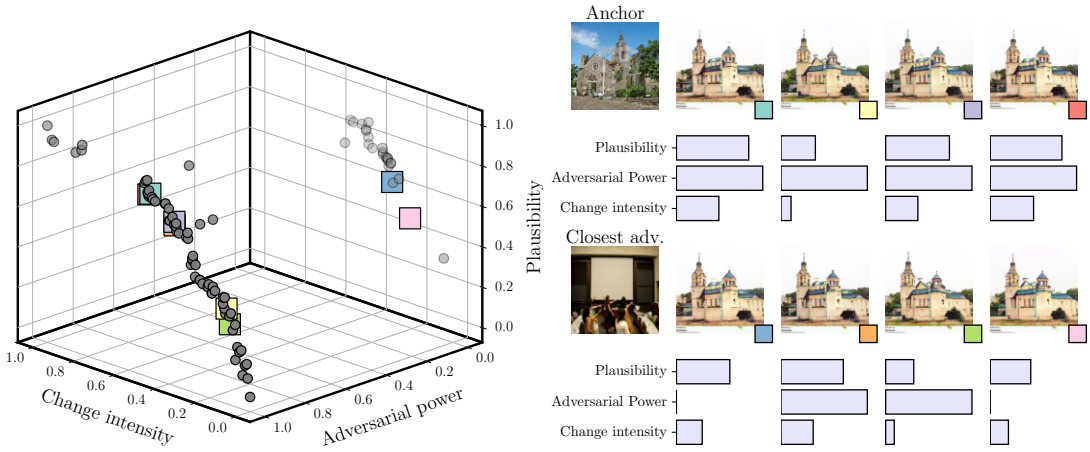


Figure 4.10: Pareto front of the counterfactual examples generated for a `Cathedral` example by the proposed framework configured with a StyleGAN2 model.

mislead the audited classifier, yet without any guarantee for success given the scarce similarity between images belonging to both classes.

4.4.5 *Experiment #5: CGAN-based Counterfactual Generation for Auditing a `MNIST` Classifier*

In correspondence to Q3, this last experiment is devised to elucidate whether the output of the proposed framework can be used for any other purpose than model explainability. To this end, we run and assess visually the counterfactuals generated for the digit classification dataset defined over the well-known `MNIST` classifier. The characterization of every class defined in it is done by a naïve conditional GAN.

Figure 4.12 portrays the output of the framework when generating counterfactuals for an anchor image $\mathbf{x}^{\mathbf{a},\oplus}$ corresponding to digit 4. From what can be observed in the samples extracted from the front, visual information corresponding to digits 4 and 8 appear to be interfering with the capability of the audited model to discriminate among them. This intuition is buttressed by the fact that the closest element is a sample corresponding to digit 8, as displayed in the first bottom image of the plot. Indeed, once again misrepresented visual artifacts in the dataset are opening a path to generate plausible counterfactuals, since most instances generated by the framework are digits with incomplete shapes. This may come from the fact that the MNIST dataset is mostly composed by digits that are correctly written.

We prove the converse to this statement by running again the experiment with an additionally inserted class in the dataset that contains digits of every class over which a part has been erased. This narrows the opportunities for the framework to generate counterfactuals by erasing selected shape fragments of the anchor digit. This is confirmed in Figure 4.13, which depicts the output of the framework in this alternative setting: the counterfactual instances generated can be declared to be plausible with respect to this extended dataset, yet a visual inspection of their corresponding digits concludes that they do not resemble a numerical digit. In summary: the output of

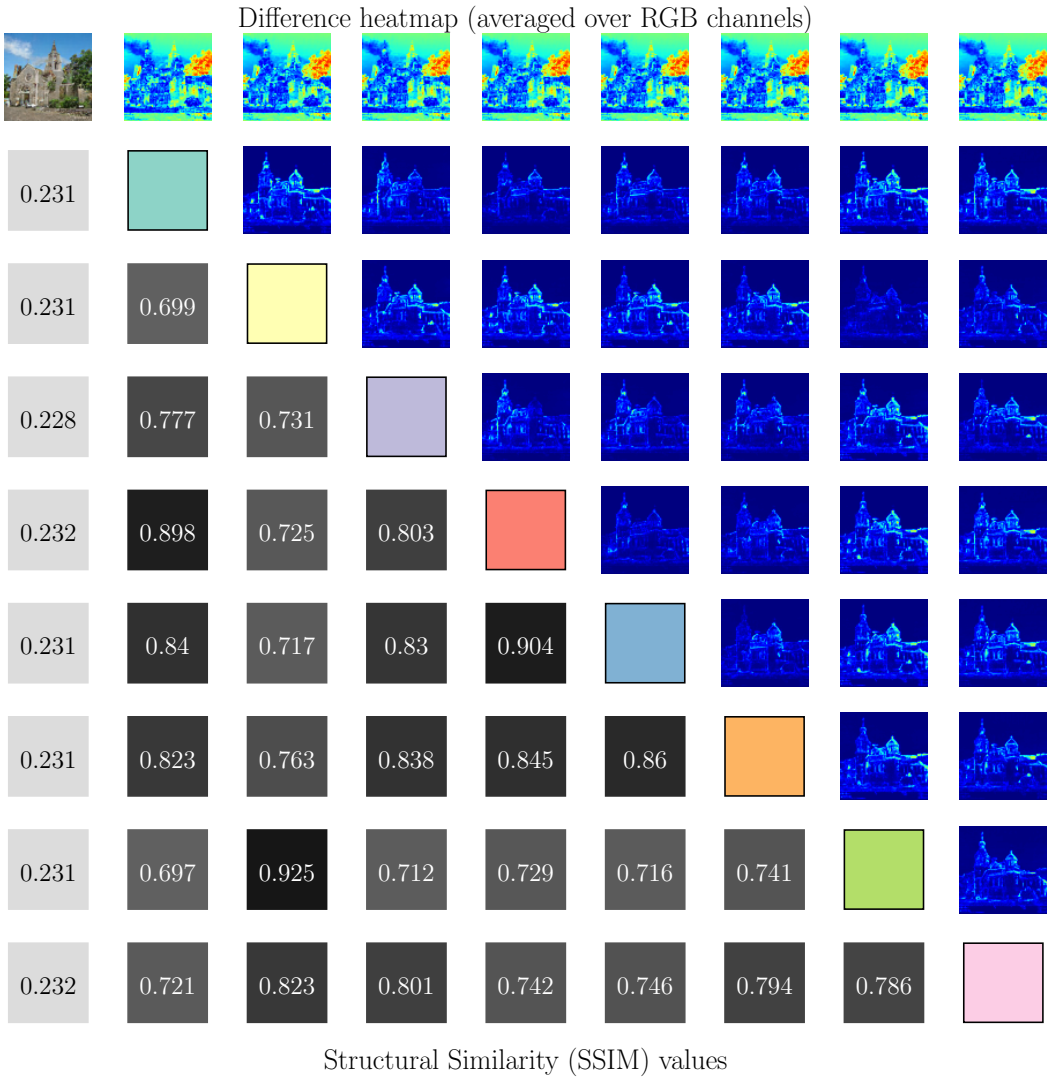


Figure 4.11: Comparison between the original counterfactuals and the anchor image following the colored markers in Figure 4.6: the upper triangular part of the matrix is composed by heatmaps depicting the mean absolute difference of the RGB pixels of every pair of images in comparison, whereas the lower triangular part denotes the SSIM value quantitatively reflecting the similarity between the images.

our framework can tell which domain over the image (color, shape) can be leveraged to make the audited model more robust against input artifacts.

4.5 SUMMARY

This chapter has proposed a novel framework that leverages the generative strength of GAN architectures and the efficient exploration capabilities of multi-objective optimization algorithm to traverse search spaces of large dimensionality. The devised framework combines these two branches of Artificial Intelligence to produce multi-criteria counterfactual explanations for a given input example and a *black-box* model

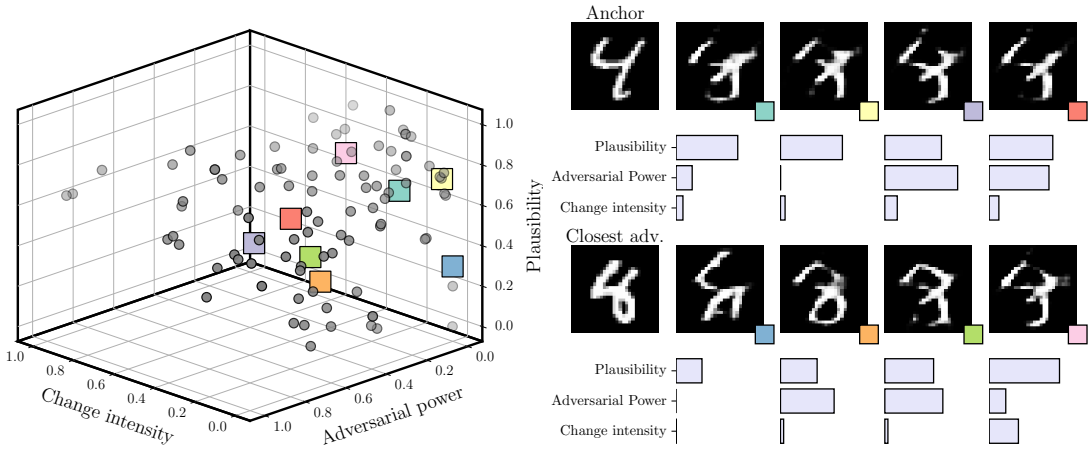
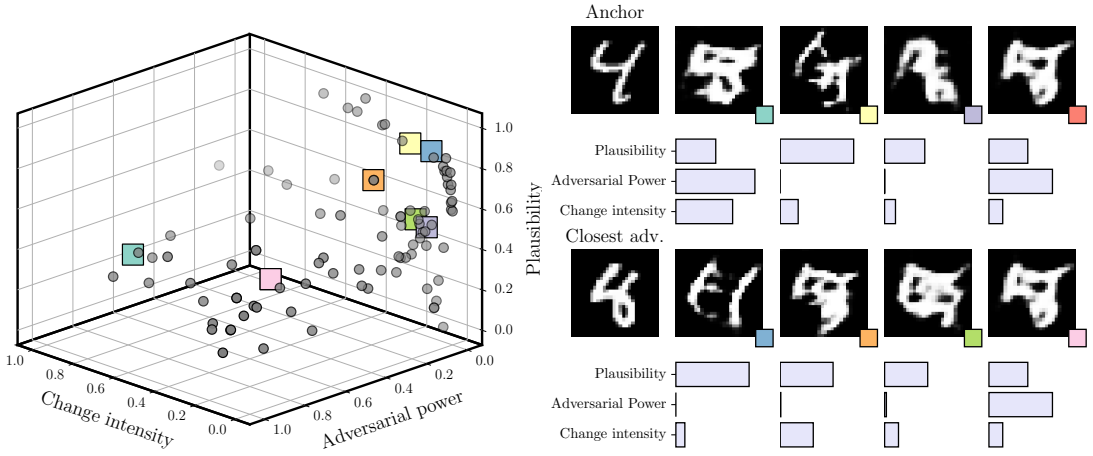


Figure 4.12: Pareto front of the counterfactual examples generated for an MNIST digit classification model by the proposed framework configured with a CGAN model.

(a)



(b)

Figure 4.13: (a) Sample of the unfinished digits generated for supplementing the MNIST dataset as an additional class. These digits are designed to cover the gap found in the initial phase of experiment #5, which showed a weakness of the MNIST dataset concerning non-finished digits. (b) Output of the framework when auditing the same model trained over the augmented dataset. In this case, images of the produced counterfactual instances do not conform to what human thinking expects to be a digit.

to be audited. Specifically, a GAN model is used to furnish a generative model that characterizes the distribution of input examples which, together with its discriminator module and its conditional dependence on an attribute vector, synthesizes examples that can be considered *plausible*. The trained GAN is therefore used as a proxy evaluator of the plausibility of new data instances that change the output of the audited model (counterfactuals). Our designed framework seeks the set of coun-

terfactual examples that best balance between *plausibility*, and *adversarial power*, incorporating a third objective (*change intensity*) that may be also in conflict, depending on the dataset at hand.

Five experiments have been run and discussed to answer three research questions aimed to understand the contribution of the framework to the explainability and understanding of the model being audited. The conclusions drawn with respect to such questions are given below:

- *Q1. Is counterfactual generation an optimization problem driven by several objectives?*

  As evinced by the Pareto front approximations obtained for the five experiments, counterfactual explanations are clearly governed by multiple objectives of relevant importance for the search. Depending on the dataset, some of such objectives could not be conflicting with each other. Nevertheless, the task of finding good counterfactual explanations must be approached as a search comprising different goals for the sake of a more enriched interface for the user of the audited model.

- *Q2. Do the properties of the generated counterfactual examples conform to general logic for the tasks and datasets at hand?*

  Definitely: our discussion on the results obtained for every experiment we have qualitatively inspected images and voxel volumes corresponding to the produced counterfactual instances. Artifacts observed in such adversarial images not only can be explained departing from common sense as per the task addressed by the audited model (e.g. color variations or emphasized structural parts of the voxels), but also exploit differences and similarities found among the data classes feeding the model at hand.

- *Q3. Do multi-criteria counterfactual explanations serve for broader purposes than model explainability?*

  Indeed, counterfactual analysis may contribute to the discovery of hidden biases resulting from misrepresentations in the training dataset of the audited model. Our discussions have empirically identified that counterfactual explanations can reflect such misrepresentations which, depending on the context, can be understood as a hidden compositional (attribute-class) bias or a potential vulnerability for adversarial attacks.

On a closing note, the framework presented in this chapter has showcased that counterfactual explanations must be tackled as a multi-faceted challenge due to the diversity of audiences and profiles for which they are generated. Understanding how a *black-box* behaves within the prediction boundaries of its feature spaces empowers non-expert users and improves their trust in the model's output. However, an advance use of this explanatory interface should regard other aspects to respond the *so much for how much?* question in counterfactual analysis. This is in essence the ultimate purpose of the framework proposed in this chapter.

5

CONCLUDING REMARKS

As covered in this thesis, XAI is a growing field of paramount importance for the society. The ever-growing complexity of AI models, coupled with the increasing demand for their application in a plethora of real-life environments, have produced the birth of this field to which this thesis has contributed. It has attempted to unify a field previously disperse, by means of a new definition of XAI followed by a recollection of the most important aspects related to the pursuit of XAI. It has also contributed to the body of research conducted under the umbrella of XAI by presenting two different frameworks that not also attempt at improving the knowledge about the functioning of methods, but they include specifications for multiple audiences within the loop. This last part, as discussed in the second chapter, is paramount since explanations have conceptually no sense if they are not coupled with an objective audience. Specifically, the contributions of this thesis can be devised in four separated blocks:

- **A new definition of Explainable Artificial Intelligence** After the reviewing more than 500 works, this thesis has contributed to designing a new definition of Explainable Artificial Intelligence that attempts at taking into consideration all the works reviewed.

  > *Given an audience, an **explainable** Artificial Intelligence is one that produces details or reasons to make its functioning clear or easy to understand.*

  This new definition combines the previous definition coined by Gunning in [7] and extends it with the concept of audience to make it more complete and accountable for the concerns of the literature.

- **A new taxonomy** This thesis has introduced a new taxonomy that collapses of the works reviewed under the field of XAI in ML in two branches (*Transparent Models* and *Post-Hoc Explanations*) that subsequently develop:
  - *Transparent Models*: consider the more classical ML models that are readily interpretable, namely: Logistic/Linear Regression, Decision Trees, K-Nearest Neighbors, Rule-base Learners, General Additive Models and Bayesian Models.
  - *Post-Hoc Explanations*: account for all the techniques that are created with the intent of explaining a non transparent model. These can be divided in two:
    * Model-Agnostic: techniques suitable for any model. They do not take into consideration the internals of the models themselves, in turn, they analyze they behaviour and come up with information about the model. This information is again categorized between: *Explanation by simplification*, *Feature relevance explanations*,*Local explanations* and *Visual explanations*

    * Model-Specific: techniques that take into consideration the internals of the models, hence, are only suitable for specific models. Among these techniques categories for five specific models can be found: *Ensembles and Multiple Classifier Systems*, *Support Vector Machines*, *Multi-Layer Neural Networks*, *Convolutional Neural Networks* and *Recurrent Neural Networks*

  Given the growth rate discrepancies of Deep Learning and Non-Deep Machine Learning techniques, a purely Deep Learning XAI taxonomy is presented. This technique is divided in four different categories that account for Post-Hoc techniques within DL:*Explanation of Deep Network Processing*, *Explanation of Deep Network Representation*, *Explanation Producing Systems* and *Hybrid Transparent and black-box Methods*.

- **A new framework for Echo State Networks** Extracted from the previous chapter of this thesis, a gap for randomized neural networks was devised. No techniques had been proposed for this field specifically. Randomized Neural Networks, and ESNs specifically have presented incredible performances compared with other state of the art techniques. However, the essence of their weights been randomly assigned has always risen concerns among practitioners and users at the time of application. This framework introduces three techniques to audit ESNs: *Potential Memory*, *Temporal Patterns* and *Pixel-Absence Relevance*.
  - *Potential Memory*: Deals with the matter of reservoir size by analyzing the dynamic response of the network. It allows to have an input of whether the choosen size is appropriate for the problem or not.
  - *Temporal Patterns*: Opens a window into the reservoirs dynamics and abilities to capture the data. This technique hints the user on whether the network is capturing the dynamics of the data introduced.
  - *Pixel-Absence Relevance*: This technique analyses the joint contributions of each of the inputs of the ESN in order to asses their relative importance. Being specifically devised for image and video inputs due to its simpler representation for the human user.

- **A new framework for model analysis by means of counterfactual generation** Another concern raised during the analysis of the literature is the single-objective focus followed for counterfactual analysis coupled with fail to look for counterfactuals that do not just intent to confound the models, but it is more focused on getting information about it. In this manner, this framework presents the idea of treating a counterfactual analysis as a multi-objective problem in which three main objectives are considered: *Adversarial power*, *Plausibility* and *Change Intensity*. This framework combines the strengths of GANs, Convolutional networks and Genetic optimizers to give birth to a technique able to detect vulnerabilities (*bias*, *data misrepresentations* and *working boundaries*) in the target model and data used to train it.

This thesis summarizes the research carried during the last three years. Other than simply contributing to the body of literature of the field of XAI, it brought up two

main concerns that ought to be taken care of by the community. XAI should not only be focused on what is mainstream nowadays. Forfeiting what is not yet in the spearhead of the field narrows the sight for new discoveries and paths that ought to be promising. With the inclusion of a framework for ESNs and a new paradigm for Counterfactual generation, this thesis attempts to lessen this deficiency. Furthermore, this dissertation has clearly underlined the importance of keeping the audience in the focus of XAI. After all, XAI's birth is not due to technical matters, but to humans necessities and keeping the humans within the loop means not loosing the essence of it.

5.1 LIST OF PUBLICATIONS

- Journal publications

  [1] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Benjamins Richard Molina Daniel, et al. "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI." In: *Information Fusion* 58 (2020), pp. 82–115.
  JRC 12.975 3/139 **Q1** *Information Systems*.

  [2] Alejandro Barredo Arrieta, Sergio Gil-Lopez, Ibai Laña, Miren Nekane Bilbao, and Javier Del Ser. "On the post-hoc explainability of deep echo state networks for time series forecasting, image and video classification." In: *Neural Computing and Applications* (2021), pp. 1–21.
  JRC 5.606 31/139 **Q1** *Effective and Efficient Deep Learning*.

  [3] Alejandro Barredo Arrieta, Javier Del Ser, Natalia Díaz-Rodríguez, Andreas Holzinger, and Francisco Herrera. "Exploring the Trade-off between Plausibility, Change Intensity and Adversarial Power in Counterfactual Explanations using Multi-objective Optimization." In: *Knowledge-Based Systems* (2021).
  JRC 8.038 16/139 **Q1** *Artificial Intelligence [UNDER REVIEW]*.

- Conference publications

  [1] Alejandro Barredo-Arrieta and Javier Del Ser. "Plausible counterfactuals: Auditing deep learning classifiers with realistic adversarial examples." In: *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2020, pp. 1–7.

  [2] Alejandro Barredo-Arrieta, Ibai Laña, and Javier Del Ser. "What lies beneath: A note on the explainability of black-box machine learning models for road traffic forecasting." In: *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE. 2019, pp. 2232–2237.

5.2 FUTURE RESEARCH LINES

The findings resulted from this thesis have open the way of several research paths. The different paths devised could be categorized in three branches. One considering

the more theoretical approach and mainly based on the work presented in Chapter 2. Another branch concerned around the further exploration of Reservoir Computing and Randomized Neural Networks 3. And a final branch related with the contents of Chapter 4 surrounding the matters of counterfactual generation.

- *Theoretical lines* Our reflections about the future of XAI, conveyed in the discussions held throughout this thesis (mainly Chapter 2), agree on the compelling need for a proper understanding of the potentiality and caveats opened up by XAI techniques. It is our vision that model interpretability must be addressed jointly with requirements and constraints related to data privacy, model confidentiality, fairness and accountability. A responsible implementation and use of AI methods in organizations and institutions worldwide will be only guaranteed if all these AI principles are studied jointly. To work with such a prospect, research along the lines of: a) Generating objective methodologies to asses explainability, privacy awareness, fairness and accountability b) Methodologies that improve upon our abilities to design models well balanced among performance and interpretability.

- *Randomized Neural Networks* Several research paths are planned for the future departing from the findings reported in Chapter 3. To begin with, different ways will be investigated to leverage the design flexibility of the reservoirs towards inducing expert knowledge contained in a transparent model in the initialization parameters of a Deep ESN. In this regard, elements from model distillation will be explored to convey such expert knowledge, possibly by driving the reservoir initialization process not entirely at random. Furthermore, explanations generated by the proposed tools can be used for other machine learning models that are also partly governed by random processes (e.g. random vector functional link networks, or random convolutional kernels). Another interesting research direction is to derive new strategies to transform spatially correlated data into sequences, as results reported in Chapter 3 have found out that an inherent loss of information is held as a result of transforming spatially correlated data into sequences. Finally, a close look will be taken at the interplay between explainability and epistemic uncertainty, the latter especially present in reservoir computing models. As in other randomization based machine learning approaches for sequential data, it is a matter of describing memoirs from the past in a statistically consistent, understandable fashion.

- *Counterfactual generation* Finally, Chapter 4 has brought up some ideas for new research lines that seem promising. The framework proposed in this thesis presented an equally balanced multi-objective search for the generation of counterfactuals. It may be interesting to explore the different vulnerabilities such a framework could find by imposing unbalanced objective constraints. The other promising research line is related to the actual objectives utilized for this framework. Adding new objectives could serve further interests in a model-auditing setup. For example, instead of looking for adversarial power, looking for what makes the model uncertain could enlighten the features that are less characteristic of each class.

BIBLIOGRAPHY

[1] Lewis Carroll and John Tenniel. *Alice's adventures in Wonderland*. Maecenas Press, 1969.

[2] Daniel Crevier. *AI: the tumultuous history of the search for artificial intelligence*. Basic Books, Inc., 1993.

[3] Stanford University. *Artificial Intelligence Index Report 2021*. 2021. URL: https://aiindex.stanford.edu/report/ (visited on 09/22/2021).

[4] Davide Castelvecchi. "Can we open the black box of AI?" In: *Nature News* 538.7623 (2016), p. 20.

[5] Zachary C. Lipton. "The Mythos of Model Interpretability." In: *Queue* 16.3 (2018), 30:31–30:57.

[6] Cynthia Rudin. "Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead." In: (2018). eprint: arXiv:1811.10154.

[7] David Gunning. *Explainable artificial intelligence (xAI)*. Tech. rep. Defense Advanced Research Projects Agency (DARPA), 2017.

[8] Sebastian Risi Rafael Bidarra Gregory Michael Youngblood Jichen Zhu Antonios Liapis. "Explainable AI for Designers: A Human-Centered Perspective on Mixed-Initiative Co-Creation." In: *2018 IEEE Conference on Computational Intelligence and Games (CIG)* (2018), pp. 1–8.

[9] Herman T Tavani. *Ethics and technology*. Wiley, 2016.

[10] Thilo Hagendorff. "The ethics of AI ethics: An evaluation of guidelines." In: *Minds and Machines* 30.1 (2020), pp. 99–120.

[11] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. "Imagenet large scale visual recognition challenge." In: *International Journal of Computer Vision* 115.3 (2015), pp. 211–252.

[12] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. "Mastering the game of Go with deep neural networks and tree search." In: *nature* 529.7587 (2016), pp. 484–489.

[13] Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. "Google's neural machine translation system: Bridging the gap between human and machine translation." In: *arXiv preprint arXiv:1609.08144* (2016).

[14] Tero Karras, Samuli Laine, and Timo Aila. "A style-based generator architecture for generative adversarial networks." In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 4401–4410.

[15] Erico Tjoa and Cuntai Guan. *A Survey on Explainable Artificial Intelligence (XAI): Towards Medical XAI*. 2019. eprint: 1907.07374.

[16] Leilani H. Gilpin, David Bau, Ben Z. Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. *Explaining Explanations: An Overview of Interpretability of Machine Learning*. 2018. eprint: 1806.00069.

[17] Filip Karlo Došilović, Mario Brčić, and Nikica Hlupić. "Explainable artificial intelligence: A survey." In: *41st International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO)*. 2018, pp. 210–215.

[18] Amina Adadi and Mohammed Berrada. "Peeking inside the black-box: A survey on Explainable Artificial Intelligence (XAI)." In: *IEEE Access* 6 (2018), pp. 52138–52160.

[19] Or Biran and Courtenay Cotton. "Explanation and justification in machine learning: A survey." In: *IJCAI-17 workshop on explainable AI (XAI)*. Vol. 8. 2017, p. 1.

[20] Shane T Shane T. Mueller, Robert R Hoffman, William Clancey, and Gary Klein. *Explanation in Human-AI Systems: A Literature Meta-Review Synopsis of Key Ideas and Publications and Bibliography for Explainable AI*. Tech. rep. Defense Advanced Research Projects Agency (DARPA) XAI Program, 2019.

[21] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. "A Survey of Methods for Explaining Black Box Models." In: *ACM Computing Surveys* 51.5 (2018), 93:1–93:42.

[22] Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. "Methods for Interpreting and Understanding Deep Neural Networks." In: *Digital Signal Processing* 73 (Feb. 2018), pp. 1–15. DOI: 10.1016/j.dsp.2017.10.011.

[23] A. Fernandez, F. Herrera, O. Cordon, M. Jose del Jesus, and F. Marcelloni. "Evolutionary Fuzzy Systems for Explainable Artificial Intelligence: Why, When, What for, and Where to?" In: *IEEE Computational Intelligence Magazine* 14.1 (2019), pp. 69–81.

[24] Michael Gleicher. "A framework for considering comprehensibility in modeling." In: *Big data* 4.2 (2016), pp. 75–88.

[25] Mark W Craven. *Extracting comprehensible models from trained neural networks*. Tech. rep. University of Wisconsin-Madison Department of Computer Sciences, 1996.

[26] Ryszard S Michalski. "A theory and methodology of inductive learning." In: *Machine learning*. Springer, 1983, pp. 83–134.

[27] José Díez, Kareem Khalifa, and Bert Leuridan. "General theories of explanation: buyer beware." In: *Synthese* 190.3 (2013), pp. 379–396.

[28] Derek Doran, Sarah Schulz, and Tarek R Besold. *What does explainable AI really mean? A new conceptualization of perspectives*. 2017. eprint: 1710.00794.

[29] Finale Doshi-Velez and Been Kim. *Towards a rigorous science of interpretable machine learning*. 2017. eprint: 1702.08608.

[30] Alfredo Vellido, José David Martín-Guerrero, and Paulo JG Lisboa. "Making machine learning models interpretable." In: *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*. Vol. 12. Citeseer. 2012, pp. 163–172.

[31] Elizabeth Walter. *Cambridge advanced learner's dictionary*. Cambridge University Press, 2008.

[32] Philippe Besnard and Anthony Hunter. *Elements of Argumentation*. The MIT Press, 2008. ISBN: 0262026430, 9780262026437.

[33] Francesca Rossi. *AI Ethics for Enterprise AI*. IBM, 2019. URL: https://economics.harvard.edu/files/economics/files/rossi-francesca_4-22-19_ai-ethics-for-enterprise-ai_ec3118-hbs.pdf.

[34] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should I trust you?: Explaining the predictions of any classifier." In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2016, pp. 1135–1144.

[35] Maria Fox, Derek Long, and Daniele Magazzeni. *Explainable planning*. 2017. eprint: 1709.10256.

[36] H Chad Lane, Mark G Core, Michael Van Lent, Steve Solomon, and Dave Gomboc. *Explainable artificial intelligence for training and tutoring*. Tech. rep. University of Southern California, 2005.

[37] W James Murdoch, Chandan Singh, Karl Kumbier, Reza Abbasi-Asl, and Bin Yu. *Interpretable machine learning: definitions, methods, and applications*. 2019. eprint: 1901.04592.

[38] Jacob Haspiel, Na Du, Jill Meyerson, Lionel P Robert Jr, Dawn Tilbury, X Jessie Yang, and Anuj K Pradhan. "Explanations and Expectations: Trust Building in Automated Vehicles." In: *Companion of the ACM/IEEE International Conference on Human-Robot Interaction*. ACM. 2018, pp. 119–120.

[39] Ajay Chander, Ramya Srinivasan, Suhas Chelian, Jun Wang, and Kanji Uchino. "Working with Beliefs: AI Transparency in the Enterprise." In: *Workshops of the ACM Conference on Intelligent User Interfaces*. 2018.

[40] Alan B Tickle, Robert Andrews, Mostefa Golea, and Joachim Diederich. "The truth will come to light: Directions and challenges in extracting the knowledge embedded within trained artificial neural networks." In: *IEEE Transactions on Neural Networks* 9.6 (1998), pp. 1057–1068.

[41] Christos Louizos, Uri Shalit, Joris M Mooij, David Sontag, Richard Zemel, and Max Welling. "Causal effect inference with deep latent-variable models." In: *Advances in Neural Information Processing Systems*. 2017, pp. 6446–6456.

[42] Olivier Goudet, Diviyan Kalainathan, Philippe Caillou, Isabelle Guyon, David Lopez-Paz, and Michele Sebag. "Learning functional causal models with generative neural networks." In: *Explainable and Interpretable Models in Computer Vision and Machine Learning*. Springer, 2018, pp. 39–80.

[43] Susan Athey and Guido W Imbens. "Machine learning methods for estimating heterogeneous causal effects." In: *stat* 1050.5 (2015).

[44] David Lopez-Paz, Robert Nishihara, Soumith Chintala, Bernhard Scholkopf, and Léon Bottou. "Discovering causal signals in images." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 6979–6987.

[45] Chelsea Barabas, Karthik Dinakar, Joichi Ito, Madars Virza, and Jonathan Zittrain. *Interventions over predictions: Reframing the ethical debate for actuarial risk assessment*. 2017. eprint: 1712.08238.

[46] Rich Caruana, Yin Lou, Johannes Gehrke, Paul Koch, Marc Sturm, and Noemie Elhadad. "Intelligible Models for HealthCare: Predicting Pneumonia Risk and Hospital 30-day Readmission." In: *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '15. 2015, pp. 1721–1730.

[47] Andreas Theodorou, Robert H Wortham, and Joanna J Bryson. "Designing and implementing transparency for real time inspection of autonomous robots." In: *Connection Science* 29.3 (2017), pp. 230–241.

[48] Andreas Holzinger, Chris Biemann, Constantinos S Pattichis, and Douglas B Kell. *What do we need to build explainable AI systems for the medical domain?* 2017. eprint: 1712.09923.

[49] Wojciech Samek, Thomas Wiegand, and Klaus-Robert Müller. *Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models*. 2017. eprint: 1708.08296.

[50] Vera Francesca Piech Chris Wadsworth Christina. *Achieving fairness through adversarial learning: an application to recidivism prediction*. 2018. eprint: 1807.00199.

[51] Xiaoyong Yuan, Pan He, Qile Zhu, and Xiaolin Li. "Adversarial examples: Attacks and defenses for deep learning." In: *IEEE Transactions on Neural Networks and Learning Systems* 30.9 (2019), pp. 2805–2824.

[52] Benjamin Letham, Cynthia Rudin, Tyler H McCormick, David Madigan, et al. "Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model." In: *The Annals of Applied Statistics* 9.3 (2015), pp. 1350–1371.

[53] Maaike Harbers, Karel van den Bosch, and John-Jules Meyer. "Design and evaluation of explainable BDI agents." In: *IEEE/WIC/ACM International Conference on Web Intelligence and Intelligent Agent Technology*. Vol. 2. IEEE. 2010, pp. 125–132.

[54] MS Hane Aung, PJ G Lisboa, Terence A Etchells, Antonia C Testa, Ben Van Calster, Sabine Van Huffel, Lil Valentin, and Dirk Timmerman. "Comparing analytical decision support models through boolean rule extraction: A case study of ovarian tumour malignancy." In: *International Symposium on Neural Networks*. Springer. 2007, pp. 1177–1186.

[55] Adrian Weller. *Challenges for transparency*. 2017. eprint: 1708.01870.

[56] Alex A Freitas. "Comprehensible classification models: a position paper." In: *ACM SIGKDD explorations newsletter* 15.1 (2014), pp. 1–10.

[57] Vitaly Schetinin, Jonathan E Fieldsend, Derek Partridge, Timothy J Coats, Wojtek J Krzanowski, Richard M Everson, Trevor C Bailey, and Adolfo Hernandez. "Confident interpretation of Bayesian decision tree ensembles for clinical applications." In: *IEEE Transactions on Information Technology in Biomedicine* 11.3 (2007), pp. 312–319.

[58] David Martens, Jan Vanthienen, Wouter Verbeke, and Bart Baesens. "Performance of classification models from a user perspective." In: *Decision Support Systems* 51.4 (2011), pp. 782–793.

[59] Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. "Interpretable deep models for ICU outcome prediction." In: *AMIA Annual Symposium Proceedings*. Vol. 2016. American Medical Informatics Association. 2016, p. 371.

[60] Nahla Barakat and Joachim Diederich. "Eclectic Rule-Extraction from Support Vector Machines." In: *International Journal of Computer, Electrical, Automation, Control and Information Engineering* 2.5 (2008), pp. 1672–1675.

[61] Francisco J Chiyah Garcia, David A Robb, Xingkun Liu, Atanas Laskov, Pedro Patron, and Helen Hastie. *Explain yourself: A natural language interface for scrutable autonomous robots*. 2018. eprint: 1803.02088.

[62] Pat Langley, Ben Meadows, Mohan Sridharan, and Dongkyu Choi. "Explainable agency for intelligent autonomous systems." In: *AAAI Conference on Artificial Intelligence*. 2017, pp. 4762–4763.

[63] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. "Explaining nonlinear classification decisions with deep taylor decomposition." In: *Pattern Recognition* 65 (2017), pp. 211–222.

[64] Schütt Kristof T Alber Maximilian Kindermans Pieter-Jan and Erhan Dumitru Kim Been Dähne Sven Müller Klaus-Robert. *Learning how to explain neural networks: Patternnet and patternattribution*. 2017. eprint: 1705.05598.

[65] Gabrielle Ras, Marcel van Gerven, and Pim Haselager. "Explanation methods in deep learning: Users, values, concerns and challenges." In: *Explainable and Interpretable Models in Computer Vision and Machine Learning*. Springer, 2018, pp. 19–36.

[66] Sebastian Bach, Alexander Binder, Klaus-Robert Müller, and Wojciech Samek. "Controlling explanatory heatmap resolution and semantics via decomposition depth." In: *IEEE International Conference on Image Processing (ICIP)*. IEEE. 2016, pp. 2271–2275.

[67] Gajendra Jung Katuwal and Robert Chen. *Machine learning model interpretability for precision medicine*. 2016. eprint: 1610.09045.

[68] Mark A Neerincx, Jasper van der Waa, Frank Kaptein, and Jurriaan van Diggelen. "Using perceptual and cognitive explanations for enhanced human-agent team performance." In: *International Conference on Engineering Psychology and Cognitive Ergonomics*. Springer. 2018, pp. 204–214.

[69] Julian D Olden and Donald A Jackson. "Illuminating the "black box": a randomization approach for understanding variable contributions in artificial neural networks." In: *Ecological modelling* 154.1-2 (2002), pp. 135–150.

[70] Josua Krause, Adam Perer, and Kenney Ng. "Interacting with predictions: Visual inspection of black-box machine learning models." In: *CHI Conference on Human Factors in Computing Systems*. ACM. 2016, pp. 5686–5697.

[71] Lars Rosenbaum, Georg Hinselmann, Andreas Jahn, and Andreas Zell. "Interpreting linear support vector machine models with heat map molecule coloring." In: *Journal of Cheminformatics* 3.1 (2011), p. 11.

[72] Jie Tan, Matthew Ung, Chao Cheng, and Casey S Greene. "Unsupervised feature construction and knowledge extraction from genome-wide assays of breast cancer with denoising autoencoders." In: *Pacific Symposium on Biocomputing Co-Chairs*. World Scientific. 2014, pp. 132–143.

[73] Samantha Krening, Brent Harrison, Karen M Feigh, Charles Lee Isabell, Mark Riedl, and Andrea Thomaz. "Learning from explanations using sentiment and advice in RL." In: *IEEE Transactions on Cognitive and Developmental Systems* 9.1 (2017), pp. 44–55.

[74] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. *Model-agnostic interpretability of machine learning*. 2016. eprint: 1606.05386.

[75] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation." In: *PloS one* 10.7 (2015), e0130140.

[76] Terence A Etchells and Paulo JG Lisboa. "Orthogonal search-based rule extraction (OSRE) for trained neural networks: a practical and efficient approach." In: *IEEE Transactions on Neural Networks* 17.2 (2006), pp. 374–384.

[77] Sreedharan Sarath Kulkarni Anagha Chakraborti Tathagata Zhuo Hankz Hankui Kambhampati Subbarao Zhang Yu. "Plan explicability and predictability for robot task planning." In: *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2017, pp. 1313–1320.

[78] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. "A simple neural network module for relational reasoning." In: *Advances in Neural Information Processing Systems*. 2017, pp. 4967–4976.

[79] Chao-Ying Joanne Peng, Tak-Shing Harry So, Frances K Stage, and Edward P St John. "The use and interpretation of logistic regression in higher education journals: 1988–1999." In: *Research in Higher Education* 43.3 (2002), pp. 259–293.

[80] B Üstün, WJ Melssen, and LMC Buydens. "Visualisation and interpretation of support vector regression models." In: *Analytica Chimica Acta* 595.1-2 (2007), pp. 299–309.

[81] Quanshi Zhang, Yu Yang, Haotian Ma, and Ying Nian Wu. "Interpreting CNNs via decision trees." In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 6261–6270.

[82] Mike Wu, Michael C Hughes, Sonali Parbhoo, Maurizio Zazzi, Volker Roth, and Finale Doshi-Velez. "Beyond sparsity: Tree regularization of deep models for interpretability." In: *AAAI Conference on Artificial Intelligence*. 2018, pp. 1670–1678.

[83] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. *Distilling the knowledge in a neural network*. 2015. eprint: 1503.02531.

[84] Nicholas Frosst and Geoffrey Hinton. *Distilling a neural network into a soft decision tree*. 2017. eprint: 1711.09784.

[85] M Gethsiyal Augasta and Thangairulappan Kathirvalavakumar. "Reverse engineering the neural networks for rule extraction in classification problems." In: *Neural Processing Letters* 35.2 (2012), pp. 131–150.

[86] Zhi-Hua Zhou, Yuan Jiang, and Shi-Fu Chen. "Extracting symbolic rules from trained neural network ensembles." In: *AI Communications* 16.1 (2003), pp. 3–15.

[87] Hui Fen Tan, Giles Hooker, and Martin T Wells. *Tree space prototypes: Another look at making tree ensembles interpretable*. 2016. eprint: 1611.07115.

[88] Ruth C Fong and Andrea Vedaldi. "Interpretable explanations of black boxes by meaningful perturbation." In: *IEEE International Conference on Computer Vision*. 2017, pp. 3429–3437.

[89] Tim Miller, Piers Howe, and Liz Sonenberg. "Explainable AI: Beware of inmates running the asylum." In: *International Joint Conference on Artificial Intelligence, Workshop on Explainable AI (XAI)*. Vol. 36. 2017, pp. 36–40.

[90] Randy Goebel, Ajay Chander, Katharina Holzinger, Freddy Lecue, Zeynep Akata, Simone Stumpf, Peter Kieseberg, and Andreas Holzinger. "Explainable AI: the new 42?" In: *International Cross-Domain Conference for Machine Learning and Knowledge Extraction*. Springer. 2018, pp. 295–303.

[91] Vaishak Belle. "Logic meets Probability: Towards Explainable AI Systems for Uncertain Worlds." In: *International Joint Conference on Artificial Intelligence*. 2017, pp. 5116–5120.

[92] Lilian Edwards and Michael Veale. "Slave to the algorithm: Why a right to an explanation is probably not the remedy you are looking for." In: *Duke L. & Tech. Rev.* 16 (2017), p. 18.

[93] Yin Lou, Rich Caruana, Johannes Gehrke, and Giles Hooker. "Accurate intelligible models with pairwise interactions." In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2013, pp. 623–631.

[94] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. "Show, attend and tell: Neural image caption generation with visual attention." In: *International Conference on Machine Learning*. 2015, pp. 2048–2057.

[95] Johan Huysmans, Karel Dejaeger, Christophe Mues, Jan Vanthienen, and Bart Baesens. "An empirical evaluation of the comprehensibility of decision table, tree and rule based predictive models." In: *Decision Support Systems* 51.1 (2011), pp. 141–154.

[96] Nahla H Barakat and Andrew P Bradley. "Rule extraction from support vector machines: A sequential covering approach." In: *IEEE Transactions on Knowledge and Data Engineering* 19.6 (2007), pp. 729–741.

[97] F Chaves Adriana da Costa, Marley Maria BR Vellasco, and Ricardo Tanscheit. "Fuzzy rule extraction from support vector machines." In: *International Conference on Hybrid Intelligent Systems*. IEEE. 2005, pp. 335–340.

[98] David Martens, Bart Baesens, Tony Van Gestel, and Jan Vanthienen. "Comprehensible credit scoring models using rule extraction from support vector machines." In: *European Journal of Operational Research* 183.3 (2007), pp. 1466–1476.

[99] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. "Learning deep features for discriminative localization." In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2016, pp. 2921–2929.

[100] R Krishnan, G Sivakumar, and P Bhattacharya. "Extracting decision trees from trained neural networks." In: *Pattern Recognition* 32.12 (1999), pp. 1999–2009.

[101] Xiuju Fu, ChongJin Ong, Sathiya Keerthi, Gih Guang Hung, and Liping Goh. "Extracting the knowledge embedded in support vector machines." In: *IEEE International Joint Conference on Neural Networks*. Vol. 1. IEEE. 2004, pp. 291–296.

[102] Ben Green. ""Fair" Risk Assessments: A Precarious Approach for Criminal Justice Reform." In: *5th Workshop on Fairness, Accountability, and Transparency in Machine Learning*. 2018.

[103] Alexandra Chouldechova. "Fair prediction with disparate impact: A study of bias in recidivism prediction instruments." In: *Big Data* 5.2 (2017), pp. 153–163.

[104] Reingold Ome Rothblum Guy Kim Michael. "Fairness through computationally-bounded awareness." In: *Advances in Neural Information Processing Systems*. 2018, pp. 4842–4852.

[105] Bernard Haasdonk. "Feature space interpretation of SVMs with indefinite kernels." In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27.4 (2005), pp. 482–492.

[106] Anna Palczewska, Jan Palczewski, Richard Marchese Robinson, and Daniel Neagu. "Interpreting random forest classification models using a feature contribution method." In: *Integration of Reusable Systems*. Springer, 2014, pp. 193–218.

[107] Soeren H Welling, Hanne HF Refsgaard, Per B Brockhoff, and Line H Clemmensen. *Forest floor visualizations of random forests*. 2016. eprint: 1605.09196.

[108] Glenn Fung, Sathyakama Sandilya, and R Bharat Rao. "Rule extraction from linear support vector machines." In: *ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*. ACM. 2005, pp. 32–40.

[109] Ying Zhang, HongYe Su, Tao Jia, and Jian Chu. "Rule extraction from trained support vector machines." In: *Pacific-Asia Conference on Knowledge Discovery and Data Mining*. Springer. 2005, pp. 61–70.

[110] Drew Linsley, Dan Shiebler, Sven Eberhardt, and Thomas Serre. *Global-and-local attention networks for visual recognition*. 2018. eprint: 1805.08819.

[111] Shang-Ming Zhou and John Q Gan. "Low-level interpretability and high-level interpretability: a unified view of data-driven interpretable fuzzy system modelling." In: *Fuzzy Sets and Systems* 159.23 (2008), pp. 3091–3131.

[112] Jenna Burrell. "How the machine 'thinks': Understanding opacity in machine learning algorithms." In: *Big Data & Society* 3.1 (2016), p. 2053951715622512.

[113] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. *Not just a black box: Learning important features through propagating activation differences*. 2016. eprint: 1605.01713.

[114] Yinpeng Dong, Hang Su, Jun Zhu, and Bo Zhang. "Improving interpretability of deep neural networks with semantic information." In: *IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 4306–4314.

[115] Greg Ridgeway, David Madigan, Thomas Richardson, and John O'Kane. "Interpretable Boosted Naïve Bayes Classification." In: *ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. 1998, pp. 101–104.

[116] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. "Interpretable convolutional neural networks." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8827–8836.

[117] Sungyong Seo, Jing Huang, Hao Yang, and Yan Liu. "Interpretable convolutional neural networks with dual local and global attention for review rating prediction." In: *Proceedings of the Eleventh ACM Conference on Recommender Systems*. ACM. 2017, pp. 297–305.

[118] Klaus Larsen, Jørgen Holm Petersen, Esben Budtz-Jørgensen, and Lars Endahl. "Interpreting parameters in the logistic regression model with random effects." In: *Biometrics* 56.3 (2000), pp. 909–914.

[119] Bilwaj Gaonkar, Russell T Shinohara, Christos Davatzikos, Alzheimers Disease Neuroimaging Initiative, et al. "Interpreting support vector machine models for multivariate group wise analysis in neuroimaging." In: *Medical image analysis* 24.1 (2015), pp. 190–204.

[120] Kai Xu, Dae Hoon Park, Chang Yi, and Charles Sutton. *Interpreting Deep Classifier by Visual Distillation of Dark Knowledge*. 2018. eprint: 1803.04042.

[121] Houtao Deng. *Interpreting tree ensembles with intrees*. 2014. eprint: 1408.5456.

[122] Pedro Domingos. "Knowledge discovery via multiple models." In: *Intelligent Data Analysis* 2.1-4 (1998), pp. 187–202.

[123] Caruana Rich Hooker Giles Lou Yin Tan Sarah. "Distill-and-Compare: Auditing Black-Box Models Using Transparent Model Distillation." In: *AAAI/ACM Conference on AI, Ethics, and Society*. ACM. 2018, pp. 303–310.

[124] Richard A Berk and Justin Bleich. "Statistical procedures for forecasting criminal behavior: A comparative assessment." In: *Criminology & Public Policy* 12.3 (2013), pp. 513–544.

[125] Satoshi Hara and Kohei Hayashi. *Making tree ensembles interpretable*. 2016. eprint: 1606.05390.

[126] Andreas Henelius, Kai Puolamäki, and Antti Ukkonen. *Interpreting classifiers through attribute interactions in datasets*. 2017. eprint: 1707.07576.

[127] Helen Hastie, Francisco Javier Chiyah Garcia, David A Robb, Pedro Patron, and Atanas Laskov. "MIRIAM: a multimodal chat-based interface for autonomous systems." In: *ACM International Conference on Multimodal Interaction*. ACM. 2017, pp. 495–496.

[128] David Bau, Bolei Zhou, Aditya Khosla, Aude Oliva, and Antonio Torralba. "Network dissection: Quantifying interpretability of deep visual representations." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 6541–6549.

[129] Haydemar Núñez, Cecilio Angulo, and Andreu Català. "Rule extraction from support vector machines." In: *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning (ESANN)*. 2002, pp. 107–112.

[130] Haydemar Núñez, Cecilio Angulo, and Andreu Català. "Rule-based learning systems for support vector machines." In: *Neural Processing Letters* 24.1 (2006), pp. 1–18.

[131] Michael Kearns, Seth Neel, Aaron Roth, and Zhiwei Steven Wu. *Preventing fairness gerrymandering: Auditing and learning for subgroup fairness*. 2017. eprint: 1711.05144.

[132] Emrah Akyol, Cedric Langbort, and Tamer Basar. *Price of transparency in strategic machine learning*. 2016. eprint: 1610.08210.

[133] Dumitru Erhan, Aaron Courville, and Yoshua Bengio. "Understanding representations learned in deep architectures." In: *Department dInformatique et Recherche Operationnelle, University of Montreal, QC, Canada, Tech. Rep* 1355 (2010), p. 1.

[134] Ye Zhang and Byron Wallace. *A sensitivity analysis of (and practitioners' guide to) convolutional neural networks for sentence classification*. 2015. eprint: 1510.03820.

[135] J. Ross Quinlan. "Simplifying decision trees." In: *International journal of man-machine studies* 27.3 (1987), pp. 221–234.

[136] Yichen Zhou and Giles Hooker. *Interpreting Models via Single Tree Approximation*. 2016. eprint: 1610.09036.

[137] A Navia-Vázquez and Emilio Parrado-Hernández. "Support vector machine interpretation." In: *Neurocomputing* 69.13-15 (2006), pp. 1754–1759.

[138] Kailkhura Bhavya Sattigeri Prasanna Ramamurthy Karthikeyan Natesan Thiagarajan Jayaraman J. *TreeView: Peeking into deep neural networks via feature-space partitioning*. 2016. eprint: 1611.07429.

[139] Matthew D Zeiler and Rob Fergus. "Visualizing and understanding convolutional networks." In: *European conference on computer vision*. Springer. 2014, pp. 818–833.

[140] Aravindh Mahendran and Andrea Vedaldi. "Understanding deep image representations by inverting them." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 5188–5196.

[141] Jorg Wagner, Jan Mathias Kohler, Tobias Gindele, Leon Hetzel, Jakob Thaddaus Wiedemer, and Sven Behnke. "Interpretable and Fine-Grained Visual Explanations for Convolutional Neural Networks." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 9097–9107.

[142] Atsushi Kanehira and Tatsuya Harada. "Learning to explain with complemental examples." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 8603–8611.

[143] Daniel W Apley. *Visualizing the effects of predictor variables in black box supervised learning models*. 2016. eprint: 1612.08468.

[144] Mateusz Staniak and Przemyslaw Biecek. "Explanations of Model Predictions with live and breakDown Packages." In: *The R Journal* 10.2 (2018), pp. 395–409.

[145] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Robert Fergus. "Deconvolutional networks." In: *CVPR*. Vol. 10. 2010, p. 7.

[146] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. *Striving for simplicity: The all convolutional net*. 2014. eprint: 1412.6806.

[147] Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, and Rory Sayres. *Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (TCAV)*. 2017. eprint: 1711.11279.

[148] Antonio Polino, Razvan Pascanu, and Dan Alistarh. *Model compression via distillation and quantization*. 2018. eprint: 1802.05668.

[149] W James Murdoch and Arthur Szlam. *Automatic rule extraction from long short term memory networks*. 2017. eprint: 1702.02540.

[150] Mark W Craven and Jude W Shavlik. "Using sampling and queries to extract rules from trained neural networks." In: *Machine learning proceedings 1994*. Elsevier, 1994, pp. 37–45.

[151] A Duygu Arbatli and H Levent Akin. "Rule extraction from trained neural networks using genetic algorithms." In: *Nonlinear Analysis: Theory, Methods & Applications* 30.3 (1997), pp. 1639–1648.

[152] Ulf Johansson and Lars Niklasson. "Evolving decision trees using oracle guides." In: *2009 IEEE Symposium on Computational Intelligence and Data Mining*. IEEE. 2009, pp. 238–244.

[153] Tao Lei, Regina Barzilay, and Tommi Jaakkola. *Rationalizing neural predictions*. 2016. eprint: 1606.04155.

[154] Alec Radford, Rafal Jozefowicz, and Ilya Sutskever. *Learning to generate reviews and discovering sentiment*. 2017. eprint: 1704.01444.

[155] Ramprasaath R Selvaraju, Abhishek Das, Ramakrishna Vedantam, Michael Cogswell, Devi Parikh, and Dhruv Batra. *Grad-CAM: Why did you say that?* 2016.

[156] Ravid Shwartz-Ziv and Naftali Tishby. *Opening the black box of deep neural networks via information*. 2017. eprint: 1703.00810.

[157] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. *Understanding neural networks through deep visualization*. 2015. eprint: 1506.06579.

[158] Phillip E Pope, Soheil Kolouri, Mohammad Rostami, Charles E Martin, and Heiko Hoffmann. "Explainability Methods for Graph Convolutional Neural Networks." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2019, pp. 10772–10781.

[159] Pratik Gajane and Mykola Pechenizkiy. *On formalizing fairness in prediction with machine learning*. 2017. eprint: 1710.03184.

[160] Cynthia Dwork and Christina Ilvento. *Composition of fairsystems*. 2018. eprint: 1806.06122.

[161] Solon Barocas, Moritz Hardt, and Arvind Narayanan. *Fairness and Machine Learning*. http://www.fairmlbook.org. fairmlbook.org, 2019.

[162] Been Kim, Elena Glassman, Brittney Johnson, and Julie Shah. *iBCM: Interactive Bayesian case model empowering humans via intuitive interaction*. Tech. rep. MIT-CSAIL-TR-2015-010, 2015.

[163] Hui-Xin Wang, Laura Fratiglioni, Giovanni B Frisoni, Matti Viitanen, and Bengt Winblad. "Smoking and the occurence of Alzheimer's disease: Cross-sectional and longitudinal data in a population-based study." In: *American journal of epidemiology* 149.7 (1999), pp. 640–644.

[164] Pramila Rani, Changchun Liu, Nilanjan Sarkar, and Eric Vanman. "An empirical study of machine learning techniques for affect recognition in human–robot interaction." In: *Pattern Analysis and Applications* 9.1 (2006), pp. 58–69.

[165] Judea Pearl. *Causality*. Cambridge university press, 2009.

[166] Max Kuhn and Kjell Johnson. *Applied predictive modeling*. Vol. 26. Springer, 2013.

[167] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*. Vol. 112. Springer, 2013.

[168] Christian Szegedy, Wojciech Zaremba, Ilya Sutskever, Joan Bruna, Dumitru Erhan, Ian Goodfellow, and Rob Fergus. *Intriguing properties of neural networks*. 2013. eprint: 1312.6199.

[169] David Ruppert. *Robust statistics: The approach based on influence functions*. Taylor & Francis, 1987.

[170] Sumanta Basu, Karl Kumbier, James B Brown, and Bin Yu. "Iterative random forests to discover predictive and stable high-order interactions." In: *Proceedings of the National Academy of Sciences* 115.8 (2018), pp. 1943–1948.

[171] Bin Yu et al. "Stability." In: *Bernoulli* 19.4 (2013), pp. 1484–1500.

[172] Bryce Goodman and Seth Flaxman. "European Union regulations on algorithmic decision-making and a "right to explanation"." In: *AI Magazine* 38.3 (2017), pp. 50–57.

[173] Kaylee Burns, Lisa Anne Hendricks, Kate Saenko, Trevor Darrell, and Anna Rohrbach. *Women also Snowboard: Overcoming Bias in Captioning Models*. 2018. eprint: 1803.09797.

[174] Adrien Bennetot, Jean-Luc Laurent, Raja Chatila, and Natalia Díaz-Rodríguez. "Towards Explainable Neural-Symbolic Visual Reasoning." In: *NeSy Workshop IJCAI 2019, Macau, China*. 2019.

[175] Robert Tibshirani. "Regression shrinkage and selection via the lasso." In: *Journal of the Royal Statistical Society: Series B (Methodological)* 58.1 (1996), pp. 267–288.

[176] Yin Lou, Rich Caruana, and Johannes Gehrke. "Intelligible models for classification and regression." In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2012, pp. 150–158.

[177] Kenji Kawaguchi. "Deep learning without poor local minima." In: *Advances in neural information processing systems*. 2016, pp. 586–594.

[178] Anupam Datta, Shayak Sen, and Yair Zick. "Algorithmic transparency via quantitative input influence: Theory and experiments with learning systems." In: *2016 IEEE symposium on security and privacy (SP)*. IEEE. 2016, pp. 598–617.

[179] Zoran Bursac, C Heath Gauss, David Keith Williams, and David W Hosmer. "Purposeful selection of variables in logistic regression." In: *Source code for biology and medicine* 3.1 (2008), p. 17.

[180] James Jaccard. *Interaction effects in logistic regression: Quantitative applications in the social sciences*. Sage Thousand Oaks, CA, 2001.

[181] David W Hosmer Jr, Stanley Lemeshow, and Rodney X Sturdivant. *Applied logistic regression*. Vol. 398. John Wiley & Sons, 2013.

[182] Chao-Ying Joanne Peng, Kuk Lida Lee, and Gary M Ingersoll. "An introduction to logistic regression analysis and reporting." In: *The journal of educational research* 96.1 (2002), pp. 3–14.

[183] Ulrich Hoffrage and Gerd Gigerenzer. "Using natural frequencies to improve diagnostic inferences." In: *Academic medicine* 73.5 (1998), pp. 538–540.

[184] Carina Mood. "Logistic regression: Why we cannot do what we think we can do, and what we can do about it." In: *European sociological review* 26.1 (2010), pp. 67–82.

[185] Rivest Ronald L Laurent Hyafil. "Constructing optimal binary decision trees is NP-complete." In: *Information processing letters* 5.1 (1976), pp. 15–17.

[186] Paul E Utgoff. "Incremental induction of decision trees." In: *Machine learning* 4.2 (1989), pp. 161–186.

[187] J. Ross Quinlan. "Induction of decision trees." In: *Machine learning* 1.1 (1986), pp. 81–106.

[188] Lior Rokach and Oded Z Maimon. *Data mining with decision trees: theory and applications*. Vol. 69. World scientific, 2014.

[189] Steven Rovnyak, Stein Kretsinger, James Thorp, and Donald Brown. "Decision trees for real-time transient stability prediction." In: *IEEE Transactions on Power Systems* 9.3 (1994), pp. 1417–1426.

[190] HA Nefeslioglu, E Sezer, C Gokceoglu, AS Bozkir, and TY Duman. "Assessment of landslide susceptibility by decision trees in the metropolitan area of Istanbul, Turkey." In: *Mathematical Problems in Engineering* 2010 (2010), Article ID 901095.

[191] Sadegh Bafandeh Imandoust and Mohammad Bolandraftar. "Application of k-nearest neighbor (knn) approach for predicting economic events: Theoretical background." In: *International Journal of Engineering Research and Applications* 3.5 (2013), pp. 605–610.

[192] Leping Li, David M Umbach, Paul Terry, and Jack A Taylor. "Application of the GA/KNN method to SELDI proteomics data." In: *Bioinformatics* 20.10 (2004), pp. 1638–1640.

[193] Gongde Guo, Hui Wang, David Bell, Yaxin Bi, and Kieran Greer. "An KNN model-based approach and its application in text categorization." In: *International Conference on Intelligent Text Processing and Computational Linguistics*. Springer. 2004, pp. 559–570.

[194] Shengyi Jiang, Guansong Pang, Meiling Wu, and Limin Kuang. "An improved K-nearest-neighbor algorithm for text categorization." In: *Expert Systems with Applications* 39.1 (2012), pp. 1503–1509.

[195] Ulf Johansson, Rikard König, and Lars Niklasson. "The Truth is In There-Rule Extraction from Opaque Models Using Genetic Programming." In: *FLAIRS Conference*. Miami Beach, FL. 2004, pp. 658–663.

[196] J Ross Quinlan. "Generating production rules from decision trees." In: *ijcai*. Vol. 87. Citeseer. 1987, pp. 304–307.

[197] Pat Langley and Herbert A Simon. "Applications of machine learning and rule induction." In: *Communications of the ACM* 38.11 (1995), pp. 54–64.

[198] Daniel Berg. "Bankruptcy prediction by generalized additive models." In: *Applied Stochastic Models in Business and Industry* 23.2 (2007), pp. 129–143.

[199] Raffaella Calabrese et al. "Estimating bank loans loss given default by generalized additive models." In: *UCD Geary Institute Discussion Paper Series, WP2012/24* (2012).

[200] Pakize Taylan, G-W Weber, and Amir Beck. "New approaches to regression by generalized additive models and continuous optimization for modern applications in finance, science and technology." In: *Optimization* 56.5-6 (2007), pp. 675–698.

[201] Hiroto Murase, Hiroshi Nagashima, Shiroh Yonezaki, Ryuichi Matsukura, and Toshihide Kitakado. "Application of a generalized additive model (GAM) to reveal relationships between environmental factors and distributions of pelagic fish and krill: a case study in Sendai Bay, Japan." In: *ICES Journal of Marine Science* 66.6 (2009), pp. 1417–1424.

[202] N Tomić and S Božić. "A modified geosite assessment model (M-GAM) and its application on the Lazar Canyon area (Serbia)." In: *International journal of environmental research* 8.4 (2014), pp. 1041–1052.

[203] Antoine Guisan, Thomas C Edwards Jr, and Trevor Hastie. "Generalized linear and generalized additive models in studies of species distributions: setting the scene." In: *Ecological Modelling* 157.2-3 (2002), pp. 89–100.

[204] Peter Rothery and David B Roy. "Application of generalized additive models to butterfly transect count data." In: *Journal of Applied Statistics* 28.7 (2001), pp. 897–909.

[205] Amandine Pierrot and Yannig Goude. "Short-Term Electricity Load Forecasting With Generalized Additive Models." In: *16th Intelligent System Applications to Power Systems Conference, ISAP 2011*. IEEE. 2011, pp. 410–415.

[206] Thomas L. Griffiths, Charles Kemp, and Joshua B. Tenenbaum. "Bayesian models of cognition." In: (Apr. 2008). DOI: 10.1184/R1/6613682.v1. URL: https://kilthub.cmu.edu/articles/Bayesian_models_of_cognition/6613682.

[207] Brian H Neelon, A James O'Malley, and Sharon-Lise T Normand. "A Bayesian model for repeated measures zero-inflated count data with application to outpatient psychiatric service use." In: *Statistical modelling* 10.4 (2010), pp. 421–439.

[208] MK McAllister and GP Kirkwood. "Bayesian stock assessment: a review and example application using the logistic model." In: *ICES Journal of Marine Science* 55.6 (1998), pp. 1031–1060.

[209] Gabriel Synnaeve and Pierre Bessiere. "A Bayesian model for opening prediction in RTS games with application to StarCraft." In: *Computational Intelligence and Games (CIG), 2011 IEEE Conference on*. IEEE. 2011, pp. 281–288.

[210] Seung-Ki Min, Daniel Simonis, and Andreas Hense. "Probabilistic climate change predictions applying Bayesian model averaging." In: *Philosophical transactions of the royal society of london a: mathematical, physical and engineering sciences* 365.1857 (2007), pp. 2103–2116.

[211] Gary Koop, Dale J Poirier, and Justin L Tobias. *Bayesian econometric methods*. Cambridge University Press, 2007.

[212] Anthony R Cassandra, Leslie Pack Kaelbling, and James A Kurien. "Acting under uncertainty: Discrete Bayesian models for mobile-robot navigation." In: *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS'96*. Vol. 2. IEEE. 1996, pp. 963–972.

[213] Hugh A Chipman, Edward I George, and Robert E McCulloch. "Bayesian CART model search." In: *Journal of the American Statistical Association* 93.443 (1998), pp. 935–948.

[214] Been Kim, Cynthia Rudin, and Julie A Shah. "The bayesian case model: A generative approach for case-based reasoning and prototype classification." In: *Advances in Neural Information Processing Systems*. 2014, pp. 1952–1960.

[215] Khanna Rajiv Koyejo Oluwasanmi O Kim Been. "Examples are not enough, learn to criticize! criticism for interpretability." In: *Advances in Neural Information Processing Systems*. 2016, pp. 2280–2288.

[216] Ulf Johansson, Lars Niklasson, and Rikard König. "Accuracy vs. comprehensibility in data mining models." In: *Proceedings of the seventh international conference on information fusion*. Vol. 1. 2004, pp. 295–300.

[217] Rikard Konig, Ulf Johansson, and Lars Niklasson. "G-REX: A versatile framework for evolutionary data mining." In: *2008 IEEE International Conference on Data Mining Workshops*. IEEE. 2008, pp. 971–974.

[218] Himabindu Lakkaraju, Ece Kamar, Rich Caruana, and Jure Leskovec. *Interpretable & explorable approximations of black box models*. 2017. eprint: 1707.01154.

[219] Saumitra Mishra, Bob L Sturm, and Simon Dixon. "Local Interpretable Model-Agnostic Explanations for Music Content Analysis." In: *ISMIR*. 2017, pp. 537–543.

[220] Guolong Su, Dennis Wei, Kush R Varshney, and Dmitry M Malioutov. *Interpretable two-level Boolean rule learning for classification*. 2015. eprint: 1511.07361.

[221] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. *Nothing else matters: Model-agnostic explanations by identifying prediction invariance*. 2016. eprint: 1611.05817.

[222] Mark William Craven. "Extracting Comprehensible Models from Trained Neural Networks." AAI9700774. PhD thesis. 1996. ISBN: 0-591-14495-6.

[223] Osbert Bastani, Carolyn Kim, and Hamsa Bastani. *Interpretability via model extraction*. 2017. eprint: 1706.09773.

[224] Giles Hooker. "Discovering additive structure in black box functions." In: *Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM. 2004, pp. 575–580.

[225] Philip Adler, Casey Falk, Sorelle A Friedler, Tionney Nix, Gabriel Rybeck, Carlos Scheidegger, Brandon Smith, and Suresh Venkatasubramanian. "Auditing black-box models for indirect influence." In: *Knowledge and Information Systems* 54.1 (2018), pp. 95–122.

[226] Pang Wei Koh and Percy Liang. "Understanding black-box predictions via influence functions." In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 1885–1894.

[227] Paulo Cortez and Mark J Embrechts. "Opening black box data mining models using sensitivity analysis." In: *2011 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*. IEEE. 2011, pp. 341–348.

[228] Paulo Cortez and Mark J Embrechts. "Using sensitivity analysis and visualization techniques to open black box data mining models." In: *Information Sciences* 225 (2013), pp. 1–17.

[229] Lee Su-In Lundberg Scott M. "A unified approach to interpreting model predictions." In: *Advances in Neural Information Processing Systems*. 2017, pp. 4765–4774.

[230] Igor Kononenko et al. "An efficient explanation of individual classifications using game theory." In: *Journal of Machine Learning Research* 11.Jan (2010), pp. 1–18.

[231] Hugh Chen, Scott Lundberg, and Su-In Lee. *Explaining Models by Propagating Shapley Values of Local Components*. 2019. eprint: arXiv:1911.11888.

[232] Piotr Dabkowski and Yarin Gal. "Real time image saliency for black box classifiers." In: *Advances in Neural Information Processing Systems*. 2017, pp. 6967–6976.

[233] Andreas Henelius, Kai Puolamäki, Henrik Boström, Lars Asker, and Panagiotis Papapetrou. "A peek into the black box: exploring classifiers by randomization." In: *Data mining and knowledge discovery* 28.5-6 (2014), pp. 1503–1529.

[234] d'Alessandro Brian Provost-Foster Martens David Moeyersoms Julie. *Explaining classification models built on high-dimensional sparse data*. 2016. eprint: 1607.06280.

[235] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert MÃžller. "How to explain individual classification decisions." In: *Journal of Machine Learning Research* 11.Jun (2010), pp. 1803–1831.

[236] Julius Adebayo and Lalana Kagal. *Iterative orthogonal feature projection for diagnosing bias in black-box models*. 2016. eprint: 1611.04967.

[237] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. *Local rule-based explanations of black box decision systems*. 2018. eprint: 1805.10820.

[238] Sanjay Krishnan and Eugene Wu. "Palm: Machine learning explanations for iterative debugging." In: *Proceedings of the 2nd Workshop on Human-In-the-Loop Data Analytics*. ACM. 2017, p. 4.

[239] Marko Robnik-Šikonja and Igor Kononenko. "Explaining classifications for individual instances." In: *IEEE Transactions on Knowledge and Data Engineering* 20.5 (2008), pp. 589–600.

[240] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Anchors: High-precision model-agnostic explanations." In: *AAAI Conference on Artificial Intelligence*. 2018, pp. 1527–1535.

[241] David Martens and Foster Provost. "Explaining Data-driven Document Classifications." In: *MIS Quarterly* 38.1 (2014), pp. 73–100.

[242] Daizhuo Chen, Samuel P Fraiberger, Robert Moakler, and Foster Provost. "Enhancing transparency and control when drawing data-driven inferences about individuals." In: *Big data* 5.3 (2017), pp. 197–212.

[243] Kapelner Adam Bleich-Justin Pitkin Emil Goldstein Alex. "Peeking inside the black box: Visualizing statistical learning with plots of individual conditional expectation." In: *Journal of Computational and Graphical Statistics* 24.1 (2015), pp. 44–65.

[244] Giuseppe Casalicchio, Christoph Molnar, and Bernd Bischl. "Visualizing the feature importance for black box models." In: *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer. 2018, pp. 655–670.

[245] Gabriele Tolomei, Fabrizio Silvestri, Andrew Haines, and Mounia Lalmas. "Interpretable predictions of tree-based ensembles via actionable feature tweaking." In: *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2017, pp. 465–474.

[246] Lidia Auret and Chris Aldrich. "Interpretation of nonlinear relationships between process variables by use of random forests." In: *Minerals Engineering* 35 (2012), pp. 27–42.

[247] Nazneen Fatema Rajani and Raymond Mooney. "Stacking with auxiliary features for visual question answering." In: *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. 2018, pp. 2217–2226.

[248] Nazneen Fatema Rajani and Raymond J Mooney. "Ensembling Visual Explanations." In: *Explainable and Interpretable Models in Computer Vision and Machine Learning*. Springer, 2018, pp. 155–172.

[249] Haydemar Núñez, Cecilio Angulo, and Andreu Català. "Rule-based learning systems for support vector machines." In: *Neural Processing Letters* 24.1 (2006), pp. 1–18.

[250] Zhenyu Chen, Jianping Li, and Liwei Wei. "A multiple kernel support vector machine scheme for feature selection and rule extraction from gene expression data of cancer tissue." In: *Artificial Intelligence in Medicine* 41.2 (2007), pp. 161–175.

[251] Haydemar Núñez, Cecilio Angulo, and Andreu Català. "Support vector machines with symbolic interpretation." In: *VII Brazilian Symposium on Neural Networks, 2002. SBRN 2002. Proceedings.* IEEE. 2002, pp. 142–147.

[252] Peter Sollich. "Bayesian methods for support vector machines: Evidence and predictive class probabilities." In: *Machine learning* 46.1-3 (2002), pp. 21–52.

[253] Peter Sollich. "Probabilistic methods for support vector machines." In: *Advances in neural information processing systems*. 2000, pp. 349–355.

[254] Will Landecker, Michael D Thomure, Luís MA Bettencourt, Melanie Mitchell, Garrett T Kenyon, and Steven P Brumby. "Interpreting individual classifications of hierarchical networks." In: *2013 IEEE Symposium on Computational Intelligence and Data Mining (CIDM)*. IEEE. 2013, pp. 32–38.

[255] Aleks Jakulin, Martin Možina, Janez Demšar, Ivan Bratko, and Blaž Zupan. "Nomograms for visualizing support vector machines." In: *Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*. ACM. 2005, pp. 108–117.

[256] LiMin Fu. "Rule generation from neural networks." In: *IEEE Transactions on Systems, Man, and Cybernetics* 24.8 (1994), pp. 1114–1124.

[257] Shavlik Jude W. Towell Geoffrey G. "Extracting refined rules from knowledge-based neural networks." In: *Machine Learning* 13.1 (1993), pp. 71–101.

[258] Sebastian Thrun. "Extracting Rules from Artificial Neural Networks with Distributed Representations." In: *Proceedings of the 7th International Conference on Neural Information Processing Systems*. NIPS'94. 1994, pp. 505–512.

[259] Rudy Setiono and Wee Kheng Leow. "FERNN: An Algorithm for Fast Extraction of Rules from Neural Networks." In: *Applied Intelligence* 12.1 (2000), pp. 15–25.

[260] I. A. Taha and J. Ghosh. "Symbolic interpretation of artificial neural networks." In: *IEEE Transactions on Knowledge and Data Engineering* 11.3 (1999), pp. 448–463.

[261] H. Tsukimoto. "Extracting rules from trained neural networks." In: *IEEE Transactions on Neural Networks* 11.2 (2000), pp. 377–389.

[262] Jan Ruben Zilke, Eneldo Loza Mencía, and Frederik Janssen. "DeepRED–Rule extraction from deep neural networks." In: *International Conference on Discovery Science*. Springer. 2016, pp. 457–473.

[263] G. P. J. Schmitz, C. Aldrich, and F. S. Gouws. "ANN-DT: an algorithm for extraction of decision trees from artificial neural networks." In: *IEEE Transactions on Neural Networks* 10.6 (1999), pp. 1392–1401.

[264] Makoto Sato and Hiroshi Tsukimoto. "Rule extraction from neural networks via decision tree induction." In: *IJCNN'01. International Joint Conference on Neural Networks. Proceedings (Cat. No. 01CH37222)*. Vol. 3. IEEE. 2001, pp. 1870–1875.

[265] Raphael Féraud and Fabrice Clérot. "A methodology to explain neural network classification." In: *Neural networks* 15.2 (2002), pp. 237–246.

[266] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. *Learning Important Features Through Propagating Activation Differences*. 2017. eprint: 1704.02685.

[267] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. "Axiomatic attribution for deep networks." In: *International Conference on Machine Learning*. Vol. 70. JMLR. org. 2017, pp. 3319–3328.

[268] Julius Adebayo, Justin Gilmer, Ian Goodfellow, and Been Kim. *Local explanation methods for deep neural networks lack sensitivity to parameter values*. 2018. eprint: 1810.03307.

[269] Nicolas Papernot and Patrick McDaniel. *Deep k-nearest neighbors: Towards confident, interpretable and robust deep learning*. 2018. eprint: 1803.04765.

[270] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. *Visualizing and understanding neural models in NLP*. 2015. eprint: 1506.01066.

[271] Shawn Tan, Khe Chai Sim, and Mark Gales. "Improving the interpretability of deep neural networks with stimulated learning." In: *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE. 2015, pp. 617–623.

[272] Laura Rieger, Chandan Singh, W. James Murdoch, and Bin Yu. *Interpretations are useful: penalizing explanations to align neural networks with prior knowledge*. 2019. eprint: arXiv:1909.13584.

[273] Anh Nguyen, Alexey Dosovitskiy, Jason Yosinski, Thomas Brox, and Jeff Clune. "Synthesizing the preferred inputs for neurons in neural networks via deep generator networks." In: *Advances in Neural Information Processing Systems*. 2016, pp. 3387–3395.

[274] Yixuan Li, Jason Yosinski, Jeff Clune, Hod Lipson, and John E Hopcroft. "Convergent Learning: Do different neural networks learn the same representations?" In: *ICLR*. 2016.

[275] Mengchen Liu, Jiaxin Shi, Zhen Li, Chongxuan Li, Jun Zhu, and Shixia Liu. "Towards better analysis of deep convolutional neural networks." In: *IEEE transactions on visualization and computer graphics* 23.1 (2016), pp. 91–100.

[276] Yash Goyal, Akrit Mohapatra, Devi Parikh, and Dhruv Batra. *Towards transparent AI systems: Interpreting visual question answering models*. 2016. eprint: 1608.08974.

[277] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. *Deep inside convolutional networks: Visualising image classification models and saliency maps*. 2013. eprint: 1312.6034.

[278] Anh Nguyen, Jason Yosinski, and Jeff Clune. "Deep neural networks are easily fooled: High confidence predictions for unrecognizable images." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 427–436.

[279] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. "Long-term recurrent convolutional networks for visual recognition and description." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 2625–2634.

[280] Min Lin, Qiang Chen, and Shuicheng Yan. *Network in network*. 2013. eprint: 1312.4400.

[281] Lisa Anne Hendricks, Zeynep Akata, Marcus Rohrbach, Jeff Donahue, Bernt Schiele, and Trevor Darrell. *Generating Visual Explanations*. 2016. eprint: 1603.08507.

[282] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. "Residual attention network for image classification." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2017, pp. 3156–3164.

[283] Tianjun Xiao, Yichong Xu, Kuiyuan Yang, Jiaxing Zhang, Yuxin Peng, and Zheng Zhang. "The application of two-level attention models in deep convolutional neural network for fine-grained image classification." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015, pp. 842–850.

[284] Quanshi Zhang, Ruiming Cao, Ying Nian Wu, and Song-Chun Zhu. *Growing Interpretable Part Graphs on ConvNets via Multi-Shot Learning*. 2016. eprint: 1611.04246.

[285] Leila Arras, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. *Explaining recurrent neural network predictions in sentiment analysis*. 2017. eprint: 1706.07206.

[286] Andrej Karpathy, Justin Johnson, and Li Fei-Fei. *Visualizing and understanding recurrent networks*. 2015. eprint: 1506.02078.

[287] Jérémie Clos, Nirmalie Wiratunga, and Stewart Massie. "Towards explainable text classification by jointly learning lexicon and modifier terms." In: *IJCAI-17 Workshop on Explainable AI (XAI)*. 2017, p. 19.

[288] Scott Wisdom, Thomas Powers, James Pitton, and Les Atlas. *Interpretable recurrent neural networks using sequential sparse recovery*. 2016. eprint: 1611.07252.

[289] Viktoriya Krakovna and Finale Doshi-Velez. *Increasing the interpretability of recurrent neural networks using hidden markov models*. 2016. eprint: 1606.05320.

[290] Edward Choi, Mohammad Taha Bahadori, Jimeng Sun, Joshua Kulas, Andy Schuetz, and Walter Stewart. "Retain: An interpretable predictive model for healthcare using reverse time attention mechanism." In: *Advances in Neural Information Processing Systems*. 2016, pp. 3504–3512.

[291] Leo Breiman. *Classification and regression trees*. Routledge, 2017.

[292] Ana Lucic, Hinda Haned, and Maarten de Rijke. *Explaining Predictions from Tree-based Boosting Ensembles*. 2019. eprint: arXiv:1907.02582.

[293] Scott M. Lundberg, Gabriel G. Erion, and Su-In Lee. *Consistent Individualized Feature Attribution for Tree Ensembles*. 2018. eprint: arXiv:1802.03888.

[294] Cristian Buciluǎ, Rich Caruana, and Alexandru Niculescu-Mizil. "Model compression." In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. ACM. 2006, pp. 535–541.

[295] René Traoré, Hugo Caselles-Dupré, Timothée Lesort, Te Sun, Guanghang Cai, Natalia Díaz Rodríguez, and David Filliat. *DisCoRL: Continual Reinforcement Learning via Policy Distillation*. 2019. eprint: 1907.05855.

[296] Matthew D Zeiler, Graham W Taylor, Rob Fergus, et al. "Adaptive deconvolutional networks for mid and high level feature learning." In: *ICCV*. Vol. 1. 2. 2011, p. 6.

[297] Cogswell Michael Das Abhishek Vedantam Ramakrishna Parikh Devi Batra Dhruv Selvaraju Ramprasaath R. "Grad-cam: Visual explanations from deep networks via gradient-based localization." In: *Proceedings of the IEEE International Conference on Computer Vision*. 2017, pp. 618–626.

[298] Chris Olah, Alexander Mordvintsev, and Ludwig Schubert. "Feature Visualization." In: *Distill* (2017). https://distill.pub/2017/feature-visualization. DOI: 10.23915/distill.00007.

[299] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. "Sanity checks for saliency maps." In: *Advances in Neural Information Processing Systems*. 2018, pp. 9505–9515.

[300] Christopher Olah, Arvind Satyanarayan, Ian Johnson, Shan Carter, Ludwig Schubert, Katherine Ye, and Alexander Mordvintsev. "The Building Blocks of Interpretability." In: *Distill* (2018). URL: https://distill.pub/2018/building-blocks/.

[301] Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. *Distilling knowledge from deep networks with applications to healthcare domain*. 2015. eprint: 1512.03542.

[302] Ivan Donadello, Luciano Serafini, and Artur D'Avila Garcez. "Logic tensor networks for semantic image interpretation." In: *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI* (2017), pp. 1596–1602.

[303] Ivan Donadello. "Semantic image interpretation-integration of numerical data and logical knowledge for cognitive vision." PhD thesis. University of Trento, 2018.

[304] Artur S. d'Avila Garcez, Marco Gori, Luís C. Lamb, Luciano Serafini, Michael Spranger, and Son N. Tran. *Neural-Symbolic Computing: An Effective Methodology for Principled Integration of Machine Learning and Reasoning*. 2019. eprint: 1905.06088.

[305] Robin Manhaeve, Sebastijan Dumancic, Angelika Kimmig, Thomas Demeester, and Luc De Raedt. "DeepProbLog: Neural Probabilistic Logic Programming." In: *Advances in Neural Information Processing Systems 31*. 2018, pp. 3749–3759.

[306] Ivan Donadello, Mauro Dragoni, and Claudio Eccher. "Persuasive Explanation of Reasoning Inferences on Dietary Data." In: *First Workshop on Semantic Explainability @ ISWC 2019*. 2019.

[307] R. G. Krishnan, U. Shalit, and D. Sontag. *Deep Kalman Filters*. 2015. eprint: 1511.05121.

[308] M. Karl, M. Soelch, J. Bayer, and P. van der Smagt. *Deep Variational Bayes Filters: Unsupervised Learning of State Space Models from Raw Data*. 2016. eprint: 1605.06432.

[309] Matthew J Johnson, David K Duvenaud, Alex Wiltschko, Ryan P Adams, and Sandeep R Datta. "Composing graphical models with neural networks for structured representations and fast inference." In: *Advances in Neural Information Processing Systems 29*. 2016, pp. 2946–2954.

[310] Shuai Zheng, Sadeep Jayasumana, Bernardino Romera-Paredes, Vibhav Vineet, Zhizhong Su, Dalong Du, Chang Huang, and Philip HS Torr. "Conditional random fields as recurrent neural networks." In: *Proceedings of the IEEE international conference on computer vision*. 2015, pp. 1529–1537.

[311] Nina Narodytska, Alexey Ignatiev, Filipe Pereira, and Joao Marques-Silva. "Learning Optimal Decision Trees with SAT." In: *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence, IJCAI-18*. 2018, pp. 1362–1368.

[312] Octavio Loyola-González. "Black-Box vs. White-Box: Understanding Their Advantages and Weaknesses From a Practical Point of View." In: *IEEE Access* 7 (2019), pp. 154096–154113.

[313] Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, and Sebastian Riedel. *Language Models as Knowledge Bases?* 2019. arXiv: 1909.01066 [cs.CL].

[314] Kurt Bollacker, Natalia Díaz-Rodríguez, and Xian Li. "Extending Knowledge Graphs with Subjective Influence Networks for Personalized Fashion." In: *Designing Cognitive Cities*. Ed. by Edy Portmann, Marco E. Tabacchi, Rudolf Seising, and Astrid Habenstein. Springer International Publishing, 2019, pp. 203–233.

[315] Wenling Shang, Alexander Trott, Stephan Zheng, Caiming Xiong, and Richard Socher. *Learning World Graphs to Accelerate Hierarchical Reinforcement Learning*. 2019. eprint: 1907.00664.

[316] Mark Zolotas and Yiannis Demiris. "Towards Explainable Shared Control using Augmented Reality." In: Oct. 2019.

[317] Marta Garnelo, Kai Arulkumaran, and Murray Shanahan. *Towards deep symbolic reinforcement learning*. 2016. eprint: 1609.05518.

[318] Vito Bellini, Angelo Schiavone, Tommaso Di Noia, Azzurra Ragone, and Eugenio Di Sciascio. "Knowledge-aware Autoencoders for Explainable Recommender Systems." In: *Proceedings of the 3rd Workshop on Deep Learning for Recommender Systems*. DLRS 2018. 2018, pp. 24–31.

[319] Cheng-Zhi Anna Huang, Ashish Vaswani, Jakob Uszkoreit, Noam Shazeer, Curtis Hawthorne, Andrew M Dai, Matthew D Hoffman, and Douglas Eck. *Music Transformer: Generating Music with Long-Term Structure*. 2018. eprint: 1809.04281.

[320] Marcella Cornia, Lorenzo Baraldi, and Rita Cucchiara. *SMArT: Training Shallow Memory-aware Transformers for Robotic Explainability*. 2019. arXiv: 1910.02974 [cs.CV].

[321] A. Aamodt and E. Plaza. "Case-based reasoning: Foundational issues." In: *Methodological Variations, and System Approaches* 7.1 (1994), pp. 39–59.

[322] Rich Caruana. "Case-Based Explanation for Artificial Neural Nets." In: *Artificial Neural Networks in Medicine and Biology, Proceedings of the ANNIMAB-1 Conference*. 2000, pp. 303–308.

[323] Mark T. Keane and Eoin M. Kenny. *The Twin-System Approach as One Generic Solution for XAI: An Overview of ANN-CBR Twins for Explaining Deep Learning*. 2019. eprint: 1905.08069.

[324] Tameru Hailesilassie. *Rule extraction algorithm for deep neural networks: A review*. 2016. eprint: 1610.05267.

[325] J. M. Benitez, J. L. Castro, and I. Requena. "Are Artificial Neural Networks Black Boxes?" In: *IEEE Trans. Neural Networks* 8.5 (1997), pp. 1156–1164.

[326] Ulf Johansson, Rikard König, and L. Niklasson. "Automatically balancing accuracy and comprehensibility in predictive modeling." In: *Proceedings of the 8th International Conference on Information Fusion*. Vol. 2. Aug. 2005, 7 pp.

[327] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. *SmoothGrad: removing noise by adding noise*. 2017. eprint: 1706.03825.

[328] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. *Towards better understanding of gradient-based attribution methods for Deep Neural Networks*. 2017. eprint: 1711.06104.

[329] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. *How transferable are features in deep neural networks?* 2014. eprint: 1411.1792.

[330] Azizpour Hossein Sullivan Josephine Carlsson Stefan Sharif Razavian Ali. *CNN Features off-the-shelf: an Astounding Baseline for Recognition*. 2014. eprint: 1403.6382.

[331] Shuyang Du, Haoli Guo, and Andrew Simpson. *Self-driving car steering angle prediction based on image recognition*. Tech. rep. Technical Report, Stanford University, 2017.

[332] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. *Object Detectors Emerge in Deep Scene CNNs*. 2014. eprint: 1412.6856.

[333] Yongfeng Zhang and Xu Chen. *Explainable Recommendation: A Survey and New Perspectives*. 2018. eprint: 1804.11192.

[334] Jonathan Frankle and Michael Carbin. *The Lottery Ticket Hypothesis: Finding Sparse, Trainable Neural Networks*. 2018. eprint: 1803.03635.

[335] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. *Attention Is All You Need*. 2017. eprint: 1706.03762.

[336] Jiasen Lu, Jianwei Yang, Dhruv Batra, and Devi Parikh. "Hierarchical Question-image Co-attention for Visual Question Answering." In: *Proceedings of the 30th International Conference on Neural Information Processing Systems*. NIPS'16. 2016, pp. 289–297.

[337] Abhishek Das, Harsh Agrawal, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. *Human Attention in Visual Question Answering: Do Humans and Deep Networks Look at the Same Regions?* 2016. eprint: 1606.03556.

[338] Dong Huk Park, Lisa Anne Hendricks, Zeynep Akata, Anna Rohrbach, Bernt Schiele, Trevor Darrell, and Marcus Rohrbach. *Multimodal Explanations: Justifying Decisions and Pointing to the Evidence*. 2018. eprint: 1802.08129.

[339] Andrew Slavin Ross, Michael C. Hughes, and Finale Doshi-Velez. *Right for the Right Reasons: Training Differentiable Models by Constraining their Explanations*. 2017. eprint: 1703.03717.

[340] I. T. Jolliffe. "Principal Component Analysis and Factor Analysis." In: *Principal Component Analysis*. Springer New York, 1986, pp. 115–128.

[341] A Hyvärinen and Erkki Oja. "Oja, E.: Independent Component Analysis: Algorithms and Applications. Neural Networks 13(4-5), 411-430." In: *Neural networks* 13 (2000), pp. 411–430.

[342] Michael W. Berry, Murray Browne, Amy Nicole Langville, Victor Paúl Pauca, and Robert J. Plemmons. "Algorithms and applications for approximate nonnegative matrix factorization." In: *Computational Statistics & Data Analysis* 52 (2007), pp. 155–173.

[343] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 2013. eprint: 1312.6114.

[344] Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew M Botvinick, Shakir Mohamed, and Alexander Lerchner. "beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework." In: *ICLR*. 2017.

[345] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. *InfoGAN: Interpretable Representation Learning by Information Maximizing Generative Adversarial Nets*. 2016. eprint: 1606.03657.

[346] Quanshi Zhang, Yu Yang, Yuchen Liu, Ying Nian Wu, and Song-Chun Zhu. *Unsupervised Learning of Neural Networks to Explain Neural Networks*. 2018. eprint: 1805.07468.

[347] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. *Dynamic Routing Between Capsules*. 2017. eprint: 1710.09829.

[348] Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Dhruv Batra, and Devi Parikh. *VQA: Visual Question Answering*. 2015. eprint: 1505.00468.

[349] Akira Fukui, Dong Huk Park, Daylen Yang, Anna Rohrbach, Trevor Darrell, and Marcus Rohrbach. *Multimodal Compact Bilinear Pooling for Visual Question Answering and Visual Grounding*. 2016. eprint: 1606.01847.

[350] Diane Bouchacourt and Ludovic Denoyer. *EDUCE: Explaining model Decisions through Unsupervised Concepts Extraction*. 2019. eprint: 1905.11852.

[351] Christoph Hofer, Marcus Denker, and Stéphane Ducasse. "Design and Implementation of a Backward-In-Time Debugger." In: *NODe 2006*. Vol. P-88. Lecture Notes in Informatics. 2006, pp. 17–32.

[352] Cynthia Rudin. *Please stop explaining black box models for high stakes decisions*. 2018. eprint: 1811.10154.

[353] Alberto Diez-Olivan, Javier Del Ser, Diego Galar, and Basilio Sierra. "Data fusion and machine learning for industrial prognosis: Trends and perspectives towards Industry 4.0." In: *Information Fusion* 50 (2019), pp. 92–111.

[354] Robert R. Hoffman, Shane T. Mueller, Gary Klein, and Jordan Litman. *Metrics for Explainable AI: Challenges and Prospects*. 2018. eprint: arXiv:1812.04608.

[355] Sina Mohseni, Niloofar Zarei, and Eric D. Ragan. *A Multidisciplinary Survey and Framework for Design and Evaluation of Explainable AI Systems*. 2018. eprint: arXiv:1811.11839.

[356] Alun Preece, Dan Harborne, Dave Braines, Richard Tomsett, and Supriyo Chakraborty. *Stakeholders in Explainable AI*. 2018. eprint: 1810.00184.

[357] Tim Miller. "Explanation in artificial intelligence: Insights from the social sciences." In: *Artif. Intell.* 267 (2019), pp. 1–38.

[358] Ruth M. J. Byrne. "Counterfactuals in Explainable Artificial Intelligence (XAI): Evidence from Human Reasoning." In: *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI-19*. 2019, pp. 6276–6282.

[359] Marta Garnelo and Murray Shanahan. "Reconciling deep learning with symbolic artificial intelligence: representing objects and relations." In: *Current Opinion in Behavioral Sciences* 29 (2019), pp. 17–23.

[360] Giuseppe Marra, Francesco Giannini, Michelangelo Diligenti, and Marco Gori. *Integrating Learning and Reasoning with Deep Logic Models*. 2019. eprint: 1901.04195.

[361] Kate Kelley, Belinda Clark, Vivienne Brown, and John Sitzia. "Good practice in the conduct and reporting of survey research." In: *International Journal for Quality in Health Care* 15.3 (2003), pp. 261–266.

[362] Sandra Wachter, Brent Mittelstadt, and Luciano Floridi. "Why a right to explanation of automated decision-making does not exist in the general data protection regulation." In: *International Data Privacy Law* 7.2 (2017), pp. 76–99.

[363] Tribhuvanesh Orekondy, Bernt Schiele, and Mario Fritz. *Knockoff Nets: Stealing Functionality of Black-Box Models*. 2018. eprint: 1812.02766.

[364] Seong Joon Oh, Bernt Schiele, and Mario Fritz. "Towards reverse-engineering black-box neural networks." In: *Explainable AI: Interpreting, Explaining and Visualizing Deep Learning*. Springer, 2019, pp. 121–144.

[365] Ian J Goodfellow, Jonathon Shlens, and Christian Szegedy. *Explaining and harnessing adversarial examples*. 2014. eprint: 1412.6572.

[366] Earlence Fernandes Bo Li Amir Rahmati Chaowei Xiao Atul Prakash Tadayoshi Kohno Dawn Song Kevin Eykholt Ivan Evtimov. *Robust Physical-World Attacks on Deep Learning Models*. 2017. eprint: 1707.08945.

[367] Ian J. Goodfellow, Nicolas Papernot, and Patrick D. McDaniel. *cleverhans vo.1: an adversarial machine learning library*. 2016. eprint: 1610.00768.

[368] Huang Xiao, Battista Biggio, Blaine Nelson, Han Xiao, Claudia Eckert, and Fabio Roli. "Support Vector Machines Under Adversarial Label Contamination." In: *Neurocomputing* 160.C (2015), pp. 53–62.

[369] Battista Biggio, Igino Corona, Davide Maiorca, Blaine Nelson, Nedim Šrndić, Pavel Laskov, Giorgio Giacinto, and Fabio Roli. "Evasion Attacks Against Machine Learning at Test Time." In: *Proceedings of the 2013th European Conference on Machine Learning and Knowledge Discovery in Databases - Volume Part III*. ECMLPKDD'13. 2013, pp. 387–402.

[370] Battista Biggio, Ignazio Pillai, Samuel Rota Bulò, Davide Ariu, Marcello Pelillo, and Fabio Roli. *Is Data Clustering in Adversarial Settings Secure?* 2018. eprint: 1811.09982.

[371] Zhaoqing Pan, Weijie Yu, Xiaokai Yi, Asifullah Khan, Feng Yuan, and Yuhui Zheng. "Recent progress on generative adversarial networks (GANs): A survey." In: *IEEE Access* 7 (2019), pp. 36322–36333.

[372] David Charte, Francisco Charte, Salvador García, María J del Jesus, and Francisco Herrera. "A practical tutorial on autoencoders for nonlinear feature fusion: Taxonomy, models, software and guidelines." In: *Information Fusion* 44 (2018), pp. 78–96.

[373] Christian F Baumgartner, Lisa M Koch, Kerem Can Tezcan, Jia Xi Ang, and Ender Konukoglu. "Visual feature attribution using wasserstein gans." In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2018, pp. 8309–8319.

[374] Carlo Biffi, Ozan Oktay, Giacomo Tarroni, Wenjia Bai, Antonio De Marvao, Georgia Doumou, Martin Rajchl, Reem Bedair, Sanjay Prasad, Stuart Cook, et al. "Learning interpretable anatomical features through deep generative models: Application to cardiac remodeling." In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2018, pp. 464–471.

[375] Shusen Liu, Bhavya Kailkhura, Donald Loveland, and Yong Han. *Generative Counterfactual Introspection for Explainable Deep Learning*. 2019. eprint: arXiv:1907.03077.

[376] Kush R Varshney and Homa Alemzadeh. "On the safety of machine learning: Cyber-physical systems, decision sciences, and data products." In: *Big data* 5.3 (2017), pp. 246–255.

[377] Gary M Weiss. "Mining with rarity: a unifying framework." In: *ACM Sigkdd Explorations Newsletter* 6.1 (2004), pp. 7–19.

[378] Joshua Attenberg, Panos Ipeirotis, and Foster Provost. "Beat the machine: Challenging humans to find a predictive model's "unknown unknowns"." In: *Journal of Data and Information Quality (JDIQ)* 6.1 (2015), p. 1.

[379] Le Zhang and Ponnuthurai N Suganthan. "A survey of randomized algorithms for training neural networks." In: *Information Sciences* 364 (2016), pp. 146–155.

[380] Claudio Gallicchio and Simone Scardapane. "Deep randomized neural networks." In: *Recent Trends in Learning From Data* (2020), pp. 43–68.

[381] Navreet Kaur, Manuel Gonzales, Cristian Garcia Alcaraz, Laura E Barnes, Kristen J Wells, and Jiaqi Gong. "Theory-Guided Randomized Neural Networks for Decoding Medication-Taking Behavior." In: *2021 IEEE EMBS International Conference on Biomedical and Health Informatics (BHI)*. IEEE. 2021, pp. 1–4.

[382] Davide Bacciu, Michele Colombo, Davide Morelli, and David Plans. "Randomized neural networks for preference learning with physiological data." In: *Neurocomputing* 298 (2018), pp. 9–20.

[383] Claudio Gallicchio, Alessio Micheli, and Luca Pedrelli. "Deep reservoir computing: A critical experimental analysis." In: *Neurocomputing* 268 (2017), pp. 87–99.

[384] Weipeng Cao, Xizhao Wang, Zhong Ming, and Jinzhu Gao. "A review on neural networks with random weights." In: *Neurocomputing* 275 (2018), pp. 278–287.

[385] Simone Scardapane and Dianhui Wang. "Randomness in neural networks: an overview." In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 7.2 (2017), e1200.

[386] Gina Neff, Anissa Tanweer, Brittany Fiore-Gartland, and Laura Osburn. "Critique and contribute: A practice-based framework for improving critical data studies and data science." In: *Big data* 5.2 (2017), pp. 85–97.

[387] Andrew Iliadis and Federica Russo. "Critical data studies: An introduction." In: *Big Data & Society* 3.2 (2016), p. 2053951716674238.

[388] Atluri Gowtham Faghmous James H Steinbach Michael Banerjee Arindam Ganguly Auroop Shekhar Shashi Samatova Nagiza Kumar Vipin Karpatne Anuj. "Theory-guided data science: A new paradigm for scientific discovery from data." In: *IEEE Transactions on Knowledge and Data Engineering* 29.10 (2017), pp. 2318–2331.

[389] Geoffroy Hautier, Christopher C Fischer, Anubhav Jain, Tim Mueller, and Gerbrand Ceder. "Finding nature's missing ternary oxide compounds using machine learning and density functional theory." In: *Chemistry of Materials* 22.12 (2010), pp. 3762–3767.

[390] Christopher C Fischer, Kevin J Tibbetts, Dane Morgan, and Gerbrand Ceder. "Predicting crystal structure by merging data mining with quantum mechanics." In: *Nature materials* 5.8 (2006), p. 641.

[391] Stefano Curtarolo, Gus LW Hart, Marco Buongiorno Nardelli, Natalio Mingo, Stefano Sanvito, and Ohad Levy. "The high-throughput highway to computational materials design." In: *Nature materials* 12.3 (2013), p. 191.

[392] Wang Linwei Shi Pengcheng Wong Ken CL. "Active model with orthotropic hyperelastic material for cardiac image analysis." In: *International Conference on Functional Imaging and Modeling of the Heart*. Springer. 2009, pp. 229–238.

[393] Jingjia Xu, John L Sapp, Azar Rahimi Dehaghani, Fei Gao, Milan Horacek, and Linwei Wang. "Robust transmural electrophysiological imaging: Integrating sparse and dynamic physiological models into ecg-based inference." In: *International Conference on Medical Image Computing and Computer-Assisted Intervention*. Springer. 2015, pp. 519–527.

[394] Timothée Lesort, Mathieu Seurin, Xinrui Li, Natalia Díaz-Rodríguez, and David Filliat. *Unsupervised state representation learning with robotic priors: a robustness benchmark*. 2017. eprint: arXiv:1709.05185.

[395] Joel Z Leibo, Qianli Liao, Fabio Anselmi, Winrich A Freiwald, and Tomaso Poggio. "View-tolerant face recognition and Hebbian learning imply mirror-symmetric neural tuning to head orientation." In: *Current Biology* 27.1 (2017), pp. 62–67.

[396] Franziska Schrodt, Jens Kattge, Hanhuai Shan, Farideh Fazayeli, Julia Joswig, Arindam Banerjee, Markus Reichstein, Gerhard Bönisch, Sandra Díaz, John Dickie, et al. "BHPMF–a hierarchical Bayesian approach to gap-filling and trait prediction for macroecology and functional biogeography." In: *Global Ecology and Biogeography* 24.12 (2015), pp. 1510–1521.

[397] David Leslie. "Understanding artificial intelligence ethics and safety." In: (2019). DOI: 10.5281/zenodo.3240529. eprint: arXiv:1906.05684.

[398] Herbert Jaeger. "Adaptive nonlinear system identification with echo state networks." In: *Advances in neural information processing systems*. 2003, pp. 609–616.

[399] Mantas Lukoševičius and Herbert Jaeger. "Reservoir computing approaches to recurrent neural network training." In: *Computer Science Review* 3.3 (2009), pp. 127–149.

[400] Herbert Jaeger and Harald Haas. "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication." In: *science* 304.5667 (2004), pp. 78–80.

[401] Qiuyi Wu, Ernest Fokoue, and Dhireesha Kudithipudi. "On the statistical challenges of echo state networks and some potential remedies." In: *arXiv preprint arXiv:1802.07369* (2018).

[402] Herbert Jaeger. "Reservoir riddles: Suggestions for echo state network research." In: *Proceedings. 2005 IEEE International Joint Conference on Neural Networks, 2005.* Vol. 3. IEEE. 2005, pp. 1460–1462.

[403] Luca Anthony Thiede and Ulrich Parlitz. "Gradient based hyperparameter optimization in echo state networks." In: *Neural Networks* 115 (2019), pp. 23–29.

[404] Muhammed Maruf Öztürk, İbrahim Arda Cankaya, and Deniz İpekci. "Optimizing Echo State Network through a novel Fisher Maximization based Stochastic Gradient Descent." In: *Neurocomputing* (2020).

[405] Alejandro Barredo Arrieta, Natalia Díaz-Rodríguez, Javier Del Ser, Adrien Bennetot, Siham Tabik, Alberto Barbado, Salvador García, Sergio Gil-López, Daniel Molina, Richard Benjamins, et al. "Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI." In: *Information Fusion* 58 (2020), pp. 82–115.

[406] Wolfgang Maass, Thomas Natschläger, and Henry Markram. "Real-time computing without stable states: A new framework for neural computation based on perturbations." In: *Neural computation* 14.11 (2002), pp. 2531–2560.

[407] Herbert Jaeger. "The "echo state" approach to analysing and training recurrent neural networks-with an erratum note." In: *Bonn, Germany: German National Research Center for Information Technology GMD Technical Report* 148.34 (2001), p. 13.

[408] Peter F Dominey. "Complex sensory-motor sequence learning based on recurrent state representation and reinforcement learning." In: *Biological cybernetics* 73.3 (1995), pp. 265–274.

[409] Jochen J Steil. "Backpropagation-decorrelation: online recurrent learning with O (N) complexity." In: *2004 IEEE International Joint Conference on Neural Networks (IEEE Cat. No. 04CH37541)*. Vol. 2. IEEE. 2004, pp. 843–848.

[410] Javier Del Ser, Ibai Lana, Eric L Manibardo, Izaskun Oregi, Eneko Osaba, Jesus L Lobo, Miren Nekane Bilbao, and Eleni I Vlahogianni. "Deep echo state networks for short-term traffic forecasting: Performance comparison and statistical assessment." In: *IEEE International Conference on Intelligent Transportation Systems (ITSC)*. IEEE. 2020, pp. 1–6.

[411] Filippo Palumbo, Claudio Gallicchio, Rita Pucci, and Alessio Micheli. "Human activity recognition using multisensor data fusion based on reservoir computing." In: *Journal of Ambient Intelligence and Smart Environments* 8.2 (2016), pp. 87–107.

[412] Emanuele Crisostomi, Claudio Gallicchio, Alessio Micheli, Marco Raugi, and Mauro Tucci. "Prediction of the Italian electricity price for smart grid applications." In: *Neurocomputing* 170 (2015), pp. 286–295.

[413] Herbert Jaeger, Mantas Lukoševičius, Dan Popovici, and Udo Siewert. "Optimization and applications of echo state networks with leaky-integrator neurons." In: *Neural networks* 20.3 (2007), pp. 335–352.

[414] Claudio Gallicchio and Alessio Micheli. "Richness of deep echo state network dynamics." In: *International Work-Conference on Artificial Neural Networks*. 2019, pp. 480–491.

[415] Claudio Gallicchio and Alessio Micheli. "Echo state property of deep reservoir computing networks." In: *Cognitive Computation* 9.3 (2017), pp. 337–350.

[416] Herbert Jaeger. *Tutorial on training recurrent neural networks, covering BPPT, RTRL, EKF and the" echo state network" approach*. Vol. 5.
GMD-Forschungszentrum Informationstechnik Bonn, 2002.

[417] Claudio Gallicchio, Alessio Micheli, and Luca Pedrelli. "Design of deep echo state networks." In: *Neural Networks* 108 (2018), pp. 33–47.

[418] Kai Liu and Jie Zhang. "Nonlinear process modelling using echo state networks optimised by covariance matrix adaption evolutionary strategy." In: *Computers & Chemical Engineering* 135 (2020), p. 106730.

[419] Leila Arras, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. "Explaining Recurrent Neural Network Predictions in Sentiment Analysis." In: *Proceedings of the 8th Workshop on Computational Approaches to Subjectivity, Sentiment and Social Media Analysis*. 2017, pp. 159–168.

[420] Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. "Visualizing and Understanding Neural Models in NLP." In: *Proceedings of NAACL-HLT*. 2016, pp. 681–691.

[421] Misha Denil, Alban Demiraj, and Nando De Freitas. "Extraction of salient sentences from labelled documents." In: *arXiv preprint arXiv:1412.6815* (2014).

[422] Jiwei Li, Will Monroe, and Dan Jurafsky. "Understanding neural networks through representation erasure." In: *arXiv preprint arXiv:1612.08220* (2016).

[423] Akos Kádár, Grzegorz Chrupała, and Afra Alishahi. "Representation of linguistic form and function in recurrent neural networks." In: *Computational Linguistics* 43.4 (2017), pp. 761–780.

[424] W James Murdoch, Peter J Liu, and Bin Yu. "Beyond word importance: Contextual decomposition to extract interactions from LSTMs." In: *arXiv preprint arXiv:1801.05453* (2018).

[425] Mahmoud Hassaballah and Ali Ismail Awad. *Deep learning in computer vision: principles and applications*. CRC Press, 2020.

[426] Thomas Rojat, Raphaël Puget, David Filliat, Javier Del Ser, Rodolphe Gelin, and Natalia Díaz-Rodríguez. "Explainable Artificial Intelligence (XAI) on Time Series Data: A Survey." In: *arXiv preprint arXiv:2104.00950* (2021).

[427] Jessica Lin, Eamonn Keogh, Li Wei, and Stefano Lonardi. "Experiencing SAX: a novel symbolic representation of time series." In: *Data Mining and knowledge discovery* 15.2 (2007), pp. 107–144.

[428] Jessica Lin, Eamonn Keogh, Stefano Lonardi, and Bill Chiu. "A symbolic representation of time series, with implications for streaming algorithms." In: *Proceedings of the 8th ACM SIGMOD workshop on Research issues in data mining and knowledge discovery*. 2003, pp. 2–11.

[429] Eamonn Keogh, Kaushik Chakrabarti, Michael Pazzani, and Sharad Mehrotra. "Dimensionality reduction for fast similarity search in large time series databases." In: *Knowledge and information Systems* 3.3 (2001), pp. 263–286.

[430] Lotfi A Zadeh. "Fuzzy logic." In: *Computer* 21.4 (1988), pp. 83–93.

[431] Francisco Herrera, Enrique Herrera-Viedma, and Luis Martinez. "A fusion approach for managing multi-granularity linguistic term sets in decision making." In: *Fuzzy sets and systems* 114.1 (2000), pp. 43–58.

[432] Francisco Herrera, Sergio Alonso, Francisco Chiclana, and Enrique Herrera-Viedma. "Computing with words in decision making: foundations, trends and prospects." In: *Fuzzy optimization and decision making* 8.4 (2009), pp. 337–364.

[433] Corrado Mencar and José M Alonso. "Paving the way to explainable artificial intelligence with fuzzy modeling." In: *International Workshop on Fuzzy Logic and Applications*. Springer. 2018, pp. 215–227.

[434] Wojciech Samek, Grégoire Montavon, Andrea Vedaldi, Lars Kai Hansen, and Klaus-Robert Müller. *Explainable AI: interpreting, explaining and visualizing deep learning*. Vol. 11700. Springer Nature, 2019.

[435] Yin-Wen Chang and Chih-Jen Lin. "Feature ranking using linear SVM." In: *Causation and prediction challenge*. PMLR. 2008, pp. 53–64.

[436] Scott M Lundberg, Gabriel G Erion, and Su-In Lee. "Consistent individualized feature attribution for tree ensembles." In: *arXiv preprint arXiv:1802.03888* (2018).

[437] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. "Smoothgrad: removing noise by adding noise." In: *arXiv preprint arXiv:1706.03825* (2017).

[438] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. "Explaining nonlinear classification decisions with deep taylor decomposition." In: *Pattern Recognition* 65 (2017), pp. 211–222.

[439] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. "Deep inside convolutional networks: Visualising image classification models and saliency maps." In: *arXiv preprint arXiv:1312.6034* (2013).

[440] Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. "Towards better understanding of gradient-based attribution methods for deep neural networks." In: *arXiv preprint arXiv:1711.06104* (2017).

[441] David Baehrens, Timon Schroeter, Stefan Harmeling, Motoaki Kawanabe, Katja Hansen, and Klaus-Robert Müller. "How to explain individual classification decisions." In: *The Journal of Machine Learning Research* 11 (2010), pp. 1803–1831.

[442] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. "Learning important features through propagating activation differences." In: *International Conference on Machine Learning*. PMLR. 2017, pp. 3145–3153.

[443] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "Why should i trust you?: Explaining the predictions of any classifier." In: *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 2016, pp. 1135–1144.

[444] Norbert Marwan, M Carmen Romano, Marco Thiel, and Jürgen Kurths. "Recurrence plots for the analysis of complex systems." In: *Physics reports* 438.5-6 (2007), pp. 237–329.

[445] Jean-Pierre Eckmann, S Oliffson Kamphorst, David Ruelle, et al. "Recurrence plots of dynamical systems." In: *World Scientific Series on Nonlinear Science Series A* 16 (1995), pp. 441–446.

[446] Claudio Gallicchio and Alessio Micheli. "Deep Reservoir Computing: A Critical Analysis." In: *ESANN*. 2016.

[447] Nils Schaetti, Michel Salomon, and Raphaël Couturier. "Echo state networks-based reservoir computing for mnist handwritten digits recognition." In: *IEEE International Conference on Computational Science and Engineering (CSE)*. IEEE. 2016, pp. 484–491.

[448] Alexander Woodward and Takashi Ikegami. "A reservoir computing approach to image classification using coupled echo state and back-propagation neural networks." In: *International conference image and vision computing, Auckland, New Zealand*. 2011, pp. 543–458.

[449] Abdelkerim Souahlia, Ammar Belatreche, Abdelkader Benyettou, and Kevin Curran. "An experimental evaluation of echo state network for colour image segmentation." In: *2016 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2016, pp. 1143–1150.

[450] Zhiqiang Tong and Gouhei Tanaka. "Reservoir computing with untrained convolutional neural networks for image recognition." In: *International Conference on Pattern Recognition (ICPR)*. IEEE. 2018, pp. 1289–1294.

[451] Xingjian Shi, Zhourong Chen, Hao Wang, Dit-Yan Yeung, Wai-Kin Wong, and Wang-chun Woo. "Convolutional LSTM network: A machine learning approach for precipitation nowcasting." In: *arXiv preprint arXiv:1506.04214* (2015).

[452] Ibai Laña, Javier Del Ser, Ales Padró, Manuel Vélez, and Carlos Casanova-Mateo. "The role of local urban traffic and meteorological conditions in air pollution: A data-based case study in Madrid, Spain." In: *Atmospheric Environment* 145 (2016), pp. 424–438.

[453] Christian Schuldt, Ivan Laptev, and Barbara Caputo. "Recognizing human actions: a local SVM approach." In: *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* Vol. 3. IEEE. 2004, pp. 32–36.

[454] Moshe Blank, Lena Gorelick, Eli Shechtman, Michal Irani, and Ronen Basri. "Actions as space-time shapes." In: *Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1*. Vol. 2. IEEE. 2005, pp. 1395–1402.

[455] Daniel Weinland, Remi Ronfard, and Edmond Boyer. "Free viewpoint action recognition using motion history volumes." In: *Computer vision and image understanding* 104.2-3 (2006), pp. 249–257.

[456] Jingen Liu, Jiebo Luo, and Mubarak Shah. "Recognizing realistic actions from videos "in the wild"." In: *2009 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2009, pp. 1996–2003.

[457] Kishore K Reddy and Mubarak Shah. "Recognizing 50 human action categories of web videos." In: *Machine vision and applications* 24.5 (2013), pp. 971–981.

[458] Zamir Amir Roshan Shah Mubarak Soomro Khurram. "UCF101: A dataset of 101 human actions classes from videos in the wild." In: *arXiv:1212.0402* (2012).

[459] Mikel D Rodriguez, Javed Ahmed, and Mubarak Shah. "Action mach a spatio-temporal maximum average correlation height filter for action recognition." In: *IEEE conference on computer vision and pattern recognition*. IEEE. 2008, pp. 1–8.

[460] Hildegard Kuehne, Hueihan Jhuang, Estíbaliz Garrote, Tomaso Poggio, and Thomas Serre. "HMDB: a large video database for human motion recognition." In: *2011 International conference on computer vision*. IEEE. 2011, pp. 2556–2563.

[461] Yann LeCun. "The MNIST database of handwritten digits." In: (1998). URL: http://yann.lecun.com/exdb/mnist/.

[462] Dong Han, Liefeng Bo, and C. Sminchisescu. "Selection and context for action recognition." In: *2009 IEEE 12th International Conference on Computer Vision*. 2009, pp. 1933–1940. DOI: 10.1109/ICCV.2009.5459427.

[463] Deepti Ghadiyaram, Du Tran, and Dhruv Mahajan. "Large-scale weakly-supervised pre-training for video action recognition." In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 2019, pp. 12046–12055.

[464] Moez Baccouche, Franck Mamalet, Christian Wolf, Christophe Garcia, and Atilla Baskurt. "Sequential deep learning for human action recognition." In: *International workshop on human behavior understanding*. Springer. 2011, pp. 29–39.

[465] Na Shu, Q Tang, and Haihua Liu. "A bio-inspired approach modeling spiking neural networks of visual cortex for human action recognition." In: *2014 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2014, pp. 3450–3457.

[466] Jingen Liu and Mubarak Shah. "Learning human actions via information maximization." In: *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE. 2008, pp. 1–8.

[467] Shikhar Sharma, Ryan Kiros, and Ruslan Salakhutdinov. "Action recognition using visual attention." In: *arXiv preprint arXiv:1511.04119* (2015).

[468] Yemin Shi, Wei Zeng, Tiejun Huang, and Yaowei Wang. "Learning deep trajectory descriptor for action recognition in videos using deep neural networks." In: *2015 IEEE international conference on multimedia and expo (ICME)*. IEEE. 2015, pp. 1–6.

[469] Limin Wang, Yu Qiao, and Xiaoou Tang. "Action recognition with trajectory-pooled deep-convolutional descriptors." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2015, pp. 4305–4314.

[470] Mehrtash T Harandi, Conrad Sanderson, Sareh Shirazi, and Brian C Lovell. "Kernel analysis on Grassmann manifolds for action recognition." In: *Pattern Recognition Letters* 34.15 (2013), pp. 1906–1915.

[471] Andreas Kamilaris and Francesc X Prenafeta-Boldú. "Deep learning in agriculture: A survey." In: *Computers and Electronics in Agriculture* 147 (2018), pp. 70–90.

[472] Javier Del Ser, Eneko Osaba, Javier J Sanchez-Medina, and Iztok Fister. "Bioinspired computational intelligence and transportation systems: a long road ahead." In: *IEEE Transactions on Intelligent Transportation Systems* (2019), , to appear, DOI: 10.1109/TITS.2019.2897377.

[473] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. "Generative adversarial nets." In: *Adv. in Neural Information Processing Systems*. 2014, pp. 2672–2680.

[474] Martin Arjovsky, Soumith Chintala, and Léon Bottou. "Wasserstein generative adversarial networks." In: *International Conference on Machine Learning*. 2017, pp. 214–223.

[475] Yarin Gal and Zoubin Ghahramani. "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning." In: *International Conference on Machine Learning*. Vol. 48. 2016, pp. 1050–1059.

[476] Akshayvarun Subramanya, Suraj Srinivas, and R. Venkatesh Babu. "Confidence estimation in Deep Neural networks via density modelling." In: *arXiv preprint arXiv:1707.07013* (2017).

[477] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation." In: *PloS one* 10.7 (2015), e0130140.

[478] Alejandro Barredo Arrieta et al. "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI." In: *Information Fusion* 58 (2020), pp. 82–115.

[479] Ruth MJ Byrne. "Mental models and counterfactual thoughts about what might have been." In: *Trends in cognitive sciences* 6.10 (2002), pp. 426–431.

[480] Nicole Van Hoeck, Patrick D Watson, and Aron K Barbey. "Cognitive neuroscience of human counterfactual reasoning." In: *Frontiers in human neuroscience* 9 (2015), p. 420.

[481] Alejandro Barredo-Arrieta and Javier Del Ser. "Plausible counterfactuals: Auditing deep learning classifiers with realistic adversarial examples." In: *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE. 2020, pp. 1–7.

[482] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. "Imagenet classification with deep convolutional neural networks." In: *Adv. in Neural Information Processing Systems*. 2012, pp. 1097–1105.

[483] Zachary C Lipton. "The mythos of model interpretability." In: *Queue* 16.3 (2018), pp. 31–57.

[484] A Hindupur. "The GAN Zoo: A list of all named GANs." In: (2017).

[485] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris N Metaxas. "Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks." In: *IEEE International Conference on Computer Vision*. 2017, pp. 5907–5915.

[486] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. "Gp-gan: Towards realistic high-resolution image blending." In: *ACM International Conference on Multimedia*. 2019, pp. 2487–2495.

[487] Aditya Chattopadhay, Anirban Sarkar, Prantik Howlader, and Vineeth N Balasubramanian. "Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks." In: *2018 IEEE winter conference on applications of computer vision (WACV)*. IEEE. 2018, pp. 839–847.

[488] Sandra Wachter, Brent Mittelstadt, and Chris Russell. "Counterfactual explanations without opening the black box: Automated decisions and the GDPR." In: *Harv. JL & Tech.* 31 (2017), p. 841.

[489] Berk Ustun, Alexander Spangher, and Yang Liu. "Actionable recourse in linear classification." In: *Proceedings of the Conference on Fairness, Accountability, and Transparency*. 2019, pp. 10–19.

[490] Amir-Hossein Karimi, Gilles Barthe, Borja Balle, and Isabel Valera. "Model-agnostic counterfactual explanations for consequential decisions." In: *International Conference on Artificial Intelligence and Statistics*. PMLR. 2020, pp. 895–905.

[491] Rafael Poyiadzi, Kacper Sokol, Raul Santos-Rodriguez, Tijl De Bie, and Peter Flach. "FACE: Feasible and actionable counterfactual explanations." In: *Proceedings of the AAAI/ACM Conference on AI, Ethics, and Society*. 2020, pp. 344–350.

[492] Leo Breiman. "Statistical modeling: The two cultures (with comments and a rejoinder by the author)." In: *Statistical science* 16.3 (2001), pp. 199–231.

[493] Charles Marx, Flavio Calmon, and Berk Ustun. "Predictive multiplicity in classification." In: *International Conference on Machine Learning*. PMLR. 2020, pp. 6765–6774.

[494] Martin Pawelczyk, Klaus Broelemann, and Gjergji Kasneci. "On counterfactual explanations under predictive multiplicity." In: *Conference on Uncertainty in Artificial Intelligence*. PMLR. 2020, pp. 809–818.

[495] Kaivalya Rawal, Ece Kamar, and Himabindu Lakkaraju. "Can I Still Trust You?: Understanding the Impact of Distribution Shifts on Algorithmic Recourses." In: *arXiv preprint arXiv:2012.11788* (2020).

[496] Yatong Chen, Jialu Wang, and Yang Liu. "Strategic Classification with a Light Touch: Learning Classifiers that Incentivize Constructive Adaptation." In: *arXiv preprint arXiv:2011.00355* (2020).

[497] Atoosa Kasirzadeh and Andrew Smart. "The use and misuse of counterfactuals in ethical machine learning." In: *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*. 2021, pp. 228–236.

[498] David Alvarez-Melis and Tommi S Jaakkola. "Towards robust interpretability with self-explaining neural networks." In: *arXiv preprint arXiv:1806.07538* (2018).

[499] Yu-Liang Chou, Catarina Moreira, Peter Bruza, Chun Ouyang, and Joaquim Jorge. "Counterfactuals and Causability in Explainable Artificial Intelligence: Theory, Algorithms, and Applications." In: *arXiv preprint arXiv:2103.04244* (2021).

[500] Waldo Hasperué. "The master algorithm: how the quest for the ultimate learning machine will remake our world." In: *Journal of Computer Science and Technology* 15.02 (2015), pp. 157–158.

[501] David Lewis. *Counterfactuals*. John Wiley & Sons, 2013.

[502] Carlos A Coello Coello, Gary B Lamont, David A Van Veldhuizen, et al. *Evolutionary algorithms for solving multi-objective problems*. Vol. 5. Springer.

[503] Aimin Zhou, Bo-Yang Qu, Hui Li, Shi-Zheng Zhao, Ponnuthurai Nagaratnam Suganthan, and Qingfu Zhang. "Multiobjective evolutionary algorithms: A survey of the state of the art." In: *Swarm and evolutionary computation* 1.1 (2011), pp. 32–49.

[504] Zhenliang He, Wangmeng Zuo, Meina Kan, Shiguang Shan, and Xilin Chen. "Attgan: Facial attribute editing by only changing what you want." In: *IEEE Transactions on Image Processing* 28.11 (2019), pp. 5464–5478.

[505] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. "Toward multimodal image-to-image translation." In: *arXiv preprint arXiv:1711.11586* (2017).

[506] Diederik P Kingma and Max Welling. "Auto-encoding variational bayes." In: *arXiv preprint arXiv:1312.6114* (2013).

[507] Anders Boesen Lindbo Larsen, Søren Kaae Sønderby, Hugo Larochelle, and Ole Winther. "Autoencoding beyond pixels using a learned similarity metric." In: *International conference on machine learning*. PMLR. 2016, pp. 1558–1566.

[508] Jeff Donahue, Philipp Krähenbühl, and Trevor Darrell. "Adversarial feature learning." In: *arXiv preprint arXiv:1605.09782* (2016).

[509] Vincent Dumoulin, Ishmael Belghazi, Ben Poole, Olivier Mastropietro, Alex Lamb, Martin Arjovsky, and Aaron Courville. "Adversarially learned inference." In: *arXiv preprint arXiv:1606.00704* (2016).

[510] Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. "Infogan: Interpretable representation learning by information maximizing generative adversarial nets." In: *arXiv preprint arXiv:1606.03657* (2016).

[511] Deepak Pathak Trevor Darrell Alexei A. Efros Oliver Wang Eli Shechtman. Jun-Yan Zhu Richard Zhang. *Toward Multimodal Image-to-Image Translation*. 2020. URL: https://github.com/junyanz/BicycleGAN (visited on 06/23/2021).

[512] Fischer Philipp Brox Thomas Ronneberger Olaf. "U-net: Convolutional networks for biomedical image segmentation." In: *International Conference on Medical image computing and computer-assisted intervention*. Springer. 2015, pp. 234–241.

[513] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. "Image-to-image translation with conditional adversarial networks." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2017, pp. 1125–1134.

[514] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. "Deep residual learning for image recognition." In: *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2016, pp. 770–778.

[515] Meina Kan Shiguang Shan Xilin Chen Zhenliang He Wangmeng Zuo. *AttGAN: Facial Attribute Editing by Only Changing What You Want*. 2020. URL: https://github.com/LynnHo/AttGAN-Tensorflow (visited on 06/23/2021).

[516] Linqi Zhou Amir Arsalan Soltani Zhoutong Zhang Xiuming Zhang. *GenReShapeHD*. 2018. URL: https://github.com/xiumingzhang/GenRe-ShapeHD (visited on 08/05/2021).

[517] Jiajun Wu, Chengkai Zhang, Xiuming Zhang, Zhoutong Zhang, William T Freeman, and Joshua B Tenenbaum. "Learning 3D Shape Priors for Shape Completion and Reconstruction." In: *European Conference on Computer Vision (ECCV)*. 2018.

[518] Carl Bergstrom and Jevin West. *Which face is real?* 2019. URL: http://www.whichfaceisreal.com/learn.html (visited on 11/15/2019).

[519] Miika Aittala Janne Hellsten Jaakko Lehtinen Timo Aila Tero Karras Samuli Laine. *Analyzing and Improving the Image Quality of StyleGAN*. 2021. URL: https://github.com/NVlabs/stylegan2 (visited on 06/23/2021).

[520] Mehdi Mirza and Simon Osindero. "Conditional generative adversarial nets." In: *arXiv preprint arXiv:1411.1784* (2014).

[521] *GANs 2.0: Generative Adversarial Networks in TensorFlow 2.0*. 2021. URL: https://github.com/tlatkowski/gans-2.0 (visited on 06/23/2021).

[522] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. "Deep Learning Face Attributes in the Wild." In: *Proceedings of International Conference on Computer Vision (ICCV)*. 2015.

[523] Angel X. Chang et al. "ShapeNet: An Information-Rich 3D Model Repository." In: *CoRR* abs/1512.03012 (2015). arXiv: 1512.03012. URL: http://arxiv.org/abs/1512.03012.

[524] Fisher Yu, Yinda Zhang, Shuran Song, Ari Seff, and Jianxiong Xiao. "LSUN: Construction of a Large-scale Image Dataset using Deep Learning with Humans in the Loop." In: *arXiv preprint arXiv:1506.03365* (2015).

[525] Yann LeCun and Corinna Cortes. "MNIST handwritten digit database." In: (2010). URL: http://yann.lecun.com/exdb/mnist/.

[526] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and TAMT Meyarivan. "A fast and elitist multiobjective genetic algorithm: NSGA-II." In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197.

[527] Antonio Benítez-Hidalgo, Antonio J Nebro, José García-Nieto, Izaskun Oregi, and Javier Del Ser. "jMetalPy: A Python framework for multi-objective optimization with metaheuristics." In: *Swarm and Evolutionary Computation* 51 (2019), p. 100598.

[528] Rafael C Gonzalez, Richard E Woods, et al. *Digital image processing*. 2002.

[529] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. "Image quality assessment: from error visibility to structural similarity." In: *IEEE Transactions on Image Processing* 13.4 (2004), pp. 600–612.