UNIVERSIDADE D
COIMBRA

Tiago Alves Faria

Financial Interpretability with Intelligent Methods

Understanding intelligent decision-making systems with Knowledge Transfer

Dissertation in the context of the Master in Informatics Engineering, Specialization in Intelligent Systems advised by Prof. Catarina Silva and Prof. Bernardete Ribeiro and presented to the Faculty of Sciences and Technology / Department of Informatics Engineering

Outubro de 2021

Abstract

Deep learning methods, as convolutional and recurrent neural networks, are becoming standard go-to algorithms for a wide range of activity sectors. However, applicability in several critical applications, e.g. public policy, security/safety systems, health diagnosis, and fraud detection, has faced some hurdles due to lack of model interpretability.

Interpretability has been a focus of research since the beginning of Deep Learning because the high accuracy and high abstraction bring the black-box problem, i.e. the accuracy vs interpretability problem. This aspect is also of importance because of trustworthiness issues, i.e. a model that is not trusted is a model that will not be used. These issues often arise in real application scenarios, where end-users are not easily convinced of the reliability of the black-box model. The existence of biased algorithms is a clear example of this problem.

With the increased use of intelligent models as part of recommendation systems and social scorings, social biases present in the data used to train these models have made it a recurrent global problem that needs to be addressed. When algorithms are fed these implicit biases they learn how to be biased too, potentially propagating and escalating the problem in the long run.

Setting the context on financial services, if a bank decides to adopt a Machine Learning (ML) algorithm to classify the creditworthiness of an applicant, and uses bank historical data, when applicants were approved or denied by humans, the algorithm may display patterns of bias against gender or ethnicity. Works on interpretability aim to mitigate problems like this by opening these black-box models. If we can understand the reasoning behind a certain decision not only we will better understand the past decision-making process of the entities using these models, but we will also be able to mitigate future biases as well as increase trust in the decisions supported by the models.

As part of the FinAI [1] project, this work proposes to research and implement interpretability methods, like knowledge transfer, that can be applied in deep learning models used in the financial sector.

Results show that knowledge transfer can be used to improve more interpretable models accuracy and in certain contexts allows for complete substitution of the Deep Neural Net (DNN) model in place of a more interpretable model like a decision-tree. It is also shown than in certain experiments the model to which the knowledge is being transferred too has different capabilities e.g. higher recall while keeping same F-score, which allows for the creation of a model ensemble in order to get the best of both parts.

Keywords

artificial intelligence, interpretability, decision trees, bias, machine learning

[1] https://www.cisuc.uc.pt/en/projects/cost-action-finai-fintech-and-artificial-intelligence-in-finance-towards-a-transparent-financial-industry

This page is intentionally left blank.

Resumo

Métodos de aprendizagem profunda, como redes neurais convolucionais e recorrentes, têm-se tornado nos modelos padrão numa vasta gama de sectores de actividade. No entanto, a aplicabilidade em várias aplicações críticas, por exemplo, sistemas de segurança, diagnósticos de saúde e detecção de fraude, tem enfrentado alguns obstáculos devido à falta de interpretabilidade destes modelos.

A interpretabilidade tem sido um foco de investigação desde o início da concepção de modelos de aprendizagem profunda, isto deve-se ao facto de que a elevada precisão e abstracção destes modelos trazerem o problema da caixa negra o que por sua vez leva a um compromisso entre precisão e interpretabilidade. Este aspecto também é importante devido a problemas de fiabilidade, um modelo que não inspire confiança é um modelo que dificilmente será utilizado. Estas questões surgem frequentemente em cenários reais de aplicação, onde os utilizadores finais não são facilmente convencidos da fiabilidade destes modelos caixa negra. A existência de modelos tendenciosos (biased) é um exemplo claro deste problema.

Com a crescente utilização de modelos inteligentes como parte de sistemas de recomendação e social scoring, os preconceitos sociais presentes nos dados utilizados na formação destes modelos tornaram-se num problema global recorrente que precisa de ser abordado. Quando estes modelos são alimentados com estes preconceitos implícitos durante o a fase de treino, aprendem também a ser preconceituosos, propagando e agravando potencialmente o problema a longo prazo.

Como parte do projeto FinAI [1], este trabalho propõe investigar e implementar métodos para interpretabilidade, como transferência de conhecimento, que possam ser aplicados em modelos de deep-learning utilizados no sector financeiro.

Focando o contexto em serviços financeiros, se um banco decidir adoptar um modelo de aprendizagem máquina para classificar a solvabilidade de um candidato, e utilizar dados históricos do banco, enquanto estes candidatos foram previamente aprovados ou negados por seres humanos, o modelo pode exibir padrões de preconceito contra géneros ou etnias. O trabalho na interpretabilidade visa mitigar problemas como este ao permitir a abertura de modelos de caixa negra. Ao compreender o raciocínio subjacente a uma determinada decisão, não só compreenderemos melhor o processo de tomada de decisão passado das entidades que os utilizam, como também seremos capazes de mitigar os preconceitos futuros, bem como aumentar a confiança nas decisões apoiadas pelos modelos.

Os resultados mostram que a transferência de conhecimento pode ser utilizada para melhorar a precisão de modelos mais interpretáveis como árvores decisão e, em certos contextos, permite que estes substituam os modelos deep-learning. Em alguns casos o modelo para o qual o conhecimento está a ser transferido dadas as suas diferenças durante o treino pode ter capacidades diferentes, por exemplo, um recall, mantendo o mesmo F-score, o que permite a criação de um conjunto de modelos de modo a obter o melhor de ambas as partes.

Palavras-chave

inteligência artificial, interpretabilidade, árvores de decisão, enviesamento

[1] https://www.cisuc.uc.pt/en/projects/cost-action-finai-fintech-and-artificial-intelligence-in-finance-towards-a-transparent-financial-industry

This page is intentionally left blank.

Funding

This work is part of FinAI project (www.cost.eu), an European project towards transparency in the financial Industry with the goals of improving the transparency of AI supported processes, address the disparity between the proliferation in AI models within the financial industry for risk assessment and decision-making, and the limited insight the public has in its consequences and to develop methods to scrutinize the quality of rule-based "smart beta" products across the asset management, banking and insurance industry.

Funding support by COST Action "Fintech and Artificial Intelligence in Finance - Towards a transparent financial industry" (FinAI) CA19130[1] is gracefully acknowledged.

[1] https//fin-ai.eu/

This page is intentionally left blank.

Acknowledgements

Here I would like to take the time to thank those who took their time to contribute throughout my dissertation process.

I want to start by thanking Professor Catarina Silva and Professor Bernardete Ribeiro for their advisory role during this dissertation. To them I share my gratitude for guidance and feedback, and for the constant challenges that prompted the contributions made, which in turn provided me with some experiences that I would never have otherwise.

I also want to show some appreciation towards the anonymous reviewers of the 25^{th} Iberoamerican Congress on Pattern Recognition (CIARP25) and 27^{th} Portuguese Conference on Pattern Recognition (RECPAD21) who accepted my contribution and provided insightful comments.

Most importantly, I would like to thank my friends and family for all the support they gave me throughout this project, without them none of this would have been possible.

This page is intentionally left blank.

Contents

1 Introduction **1**
- 1.1 Motivation . . . 1
- 1.2 FinAI . . . 2
- 1.3 Goals . . . 3
- 1.4 Research Questions . . . 3
- 1.5 Contributions . . . 4
- 1.6 Document overview . . . 4

2 Background and State of the Art **5**
- 2.1 eXplainable Artificial Intelligence (XAI) . . . 7
- 2.2 Laying down harmonized rules on artificial intelligence the newly proposed regulations on AI in the EU . . . 7
- 2.3 The "incompleteness" in the finance sector . . . 9
- 2.4 Evaluating Explanations . . . 10
- 2.5 What makes a good explanation . . . 10
- 2.6 Characteristics of interpretability . . . 11
  - 2.6.1 Methods for interpretability . . . 12
- 2.7 Knowledge Distillation and Knowledge Transfer . . . 15
- 2.8 Conclusion . . . 17

3 Proposed Approach **19**
- 3.1 Problem Definition . . . 19
- 3.2 Methodology . . . 20
- 3.3 The effect of temperature on the students training . . . 21
- 3.4 Research Questions . . . 23
  - 3.4.1 Available Data and data preparation . . . 24
  - 3.4.2 Dealing with the imbalance on the datasets . . . 28
- 3.5 Models Used . . . 29
  - 3.5.1 Deep Neural Nets - The Teacher . . . 29
  - 3.5.2 Feed-forward Neural Network . . . 29
  - 3.5.3 Long Short-Term Memory Neural Network . . . 30
  - 3.5.4 Prosper Loan's Teacher FFNN . . . 30
  - 3.5.5 Interpretable Models - The Students . . . 31
  - 3.5.6 Decision trees . . . 31
  - 3.5.7 On the comprehensibility of decision trees . . . 32

4 Experimental results and analysis **33**
- 4.1 Training of the Neural Nets . . . 33
- 4.2 The training of student models . . . 34
- 4.3 Results . . . 34
- 4.4 German Credit . . . 34

4.5 Prosper Loan 35
4.5.1 The stability of explanations on decision trees as students 37
4.5.2 Fidelity of students as surrogates for explanations 39
4.6 Stock Movement Prediction 40
4.6.1 Alphabet Inc. GOOGL 41
4.6.2 Tesla Inc. 42
4.6.3 Adobe Inc. 43
4.7 Conclusion 43

5 Conclusions and Future Work **45**

A CIARP25 Paper **53**

B RECPAD21 Paper **65**

C Work Planning **71**
C.1 Work Plan for the 1st Semester 71
C.2 Work Plan for the 2nd Semester 71

Acronyms

AI Artificial Intelligence. i, v, ix, 1–3, 5, 7–10, 17

DNN Deep Neural Net. iii, 4, 5, 12, 14, 15, 29, 30, 45

EU European Union. 25

FFNN Feed-Forward Neural Network. 29

GDPR General Data Protection Regulation. 1, 5, 6

ICO Initial Coin Offering. 2

LSTM Long Short Term Memory. 24, 30, 31

ML Machine Learning. iii, 10, 12

XAI eXplainable AI. 7

This page is intentionally left blank.

List of Figures

1.1 Perceived strategic importance of Artificial Intelligence (AI) in the financial sector . . . 1
1.2 *Article 22 of the General Data Protection Regulation* . . . 2

2.1 Average interest over the last years on the subjects of interpretability and explainability. Data source: Google Trends[1] . . . 6
2.2 The different categories of evaluations. Functionally grounded evaluations tend to be supported by previously evaluated explanations at the application and human-grounded level. . . . 10
2.3 Example excerpt taken from [1] of an attention map. . . . 14
2.4 Example of SHAP being used in price prediction for housing. In blue, we have negative Shap values that show everything that pushes the sales value in the negative direction. While the Shap value in red represents everything that pushes it towards a positive direction. Note that this is for a single instance. . . . 15
2.5 Simple example of model compression, here the values of the logits are used as targets for training a student model . . . 16
2.6 Example of Knowledge Distillation, similar to compression but in this case the output of a softmax with temperature is used for training . . . 17
2.7 Example model, the logits are the vector of values that serves as input to the softmax activation layer that will then output a vector of probabilities for each class in a classification problem. . . . 17

3.1 Pre-processing and imbalance treatment of the dataset is done, the scaling values are saved so we can inverse-transform the results after. . . . 20
3.2 The soft labels are extracted from the last layer, before the highest value class is picked as the prediction. In this example we have a 3 class prediction problem. . . . 20
3.3 The student, a decision tree regressor is trained on the soft labels aquired from the teacher, the black-box model. . . . 21
3.4 Steps for knowledge transfer. . . . 21
3.5 F1 score on student and decision-tree trained on hard labels . . . 22
3.6 Student Decision-Tree trained on soft labels . . . 23
3.7 Decision-tree trained on hard labels . . . 23

4.1 Graphical representation of the prediction scores obtained on the different models over the German Credit Dataset . . . 35
4.2 Teacher results on the Prosper Loan Dataset. . . . 36
4.3 Student results on the Prosper Loan Dataset. . . . 36
4.4 Decision path for the first instance. . . . 37
4.5 Decision path for the second instance. . . . 38
4.6 Student's feature importances. . . . 40

4.7 Teacher's feature importances. 40
4.8 Graphical representation of the prediction scores obtained for the Alphabet Inc.(GOOGL) Stock movement prediction. 41
4.9 Graphical representation of the prediction scores obtained for the Tesla Inc.(TSLA) Stock Movement Prediction. 42
4.10 Graphical representation of the prediction scores obtained for the Adobe Inc.(ADBE) Stock Movement Prediction 43

C.1 Gantt chart of the scheduled tasks 71

List of Tables

3.1 Available dataset and their respective number and problem assignment . . . 24
3.2 Time span for each raw dataset retrieved . . . 25
3.3 Stock datasets attributes and their description . . . 25
3.4 Categorical ratio of the predicted outcome over each attribute from the German Credit Dataset . . . 27
3.5 Prosper Loan Dataset Class Distribution . . . 27
3.6 Feed-forward Neural Network parameters for credit score classification on the German Credit Dataset . . . 29
3.7 Long Short-Term Memory neural network parameters for stock movement prediction . . . 30
3.8 Long Short-Term Memory neural network parameters for stock movement prediction . . . 31
3.9 Student Hyperparameters Table . . . 32

4.1 Caption . . . 34
4.2 German Credit Dataset credit default prediction results . . . 34
4.3 Teacher training scores for problem 5. . . . 35
4.4 Teacher testing scores . . . 36
4.5 Results for teacher training on 3 different imbalance treatment techniques. . 36
4.6 Scores for the minority classes for the teacher model. . . . 36
4.7 Alphabet Inc.(GOOGL) stock movement prediction Results . . . 41
4.8 Tesla Inc. stock movement prediction results. . . . 42
4.9 Adobe Inc.(ADBE) stock movement prediction results . . . 43

This page is intentionally left blank.

Chapter 1

Introduction

1.1 Motivation

Nowadays intelligent systems are a core component of many activity sectors, including the financial sector. From stock prediction to asset management and credit scores there is no doubt there is an automated system working on the background.
In the beginning of 2020 a survey [2] conducted on AI on financial services estimated that in the next two years there will be a mass adoption of AI on the finance sector with 77% of the respondents expecting that AI will become essential to their business within 2 years (see Figure 1.1).

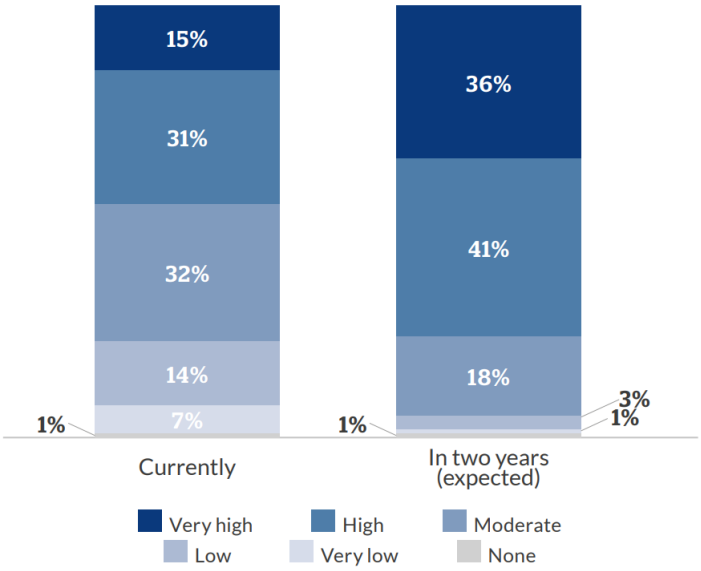
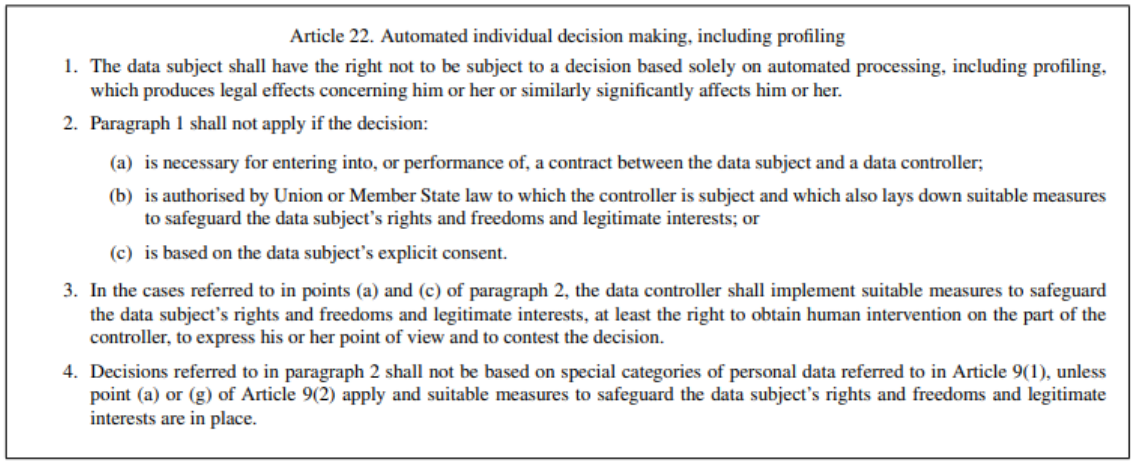


Figure 1.1: Perceived strategic importance of AI in the financial sector adopted from [2]

These systems usually perform well allowing for powerful predictions in very low times. However, even with such unprecedented advancements on prediction power and accuracy on these models, one obstacle stays the same: they often lack transparency.
With the rapid digitalization of many domains of social life and businesses came a sudden urge to acquire data. This led to the need of updating the previous laws on data protection. In April 2016, the European Parliament adopted a set of comprehensive regulations for the collection, storage and use of personal information, the General Data Protection Regulation (GDPR)[3] which now contains *Article 22 - Automated individual decision-making,*

including profiling (see figure 1.2).



Article 22. Automated individual decision making, including profiling

1. The data subject shall have the right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her.
2. Paragraph 1 shall not apply if the decision:
   (a) is necessary for entering into, or performance of, a contract between the data subject and a data controller;
   (b) is authorised by Union or Member State law to which the controller is subject and which also lays down suitable measures to safeguard the data subject's rights and freedoms and legitimate interests; or
   (c) is based on the data subject's explicit consent.
3. In the cases referred to in points (a) and (c) of paragraph 2, the data controller shall implement suitable measures to safeguard the data subject's rights and freedoms and legitimate interests, at least the right to obtain human intervention on the part of the controller, to express his or her point of view and to contest the decision.
4. Decisions referred to in paragraph 2 shall not be based on special categories of personal data referred to in Article 9(1), unless point (a) or (g) of Article 9(2) apply and suitable measures to safeguard the data subject's rights and freedoms and legitimate interests are in place.

Figure 1.2: *Article 22 of the General Data Protection Regulation*

From paragraph (1) : **"The data subject shall have the right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her."**. Most AI systems currently operating in the sector work on deep net architectures that are not easily interpretable, this makes it hard for the organizations relying on these AI systems to detect possible bias which then leads to profiling. The ability to interpret the decisions made by black-box models gives us ways of improving these models, acquire knowledge on possible new strategies as well as improve the relationships with the people involved by gaining trust on the decisions made by these systems which also complies with paragraph (3) **"right to obtain human intervention on the part of the controller, to express his or her point of view and to contest the decision"**. It's in the best interest of companies operating on black-box models to be able to explain why decisions were made given the applicants right to "contest the decision" made by said models.

1.2 FinAI

In early 2018, the European Commission unveiled their action plan for a more competitive and innovative financial market, and initiative on AI with the aim to harness the opportunities presented by technology enabled innovation in financial services.
This work is part of FinAI[1], an European project towards transparency in the financial Industry with the following goals:

- improving the transparency of AI supported processes by developing a data-driven rating methodology for ICO's.
- address the disparity between the proliferation in AI models within the financial industry for risk assessment and decision-making, and the limited insight the public has in its consequences by developing policy papers and methods to increase transparency.

[1]https://www.cisuc.uc.pt/en/projects/cost-action-finai-fintech-and-artificial-intelligence-in-finance-towards-a-transparent-financial-industry

- develop methods to scrutinize the quality of rule-based "smart beta" products across the asset management, banking and insurance industry.

1.3 Goals

This dissertation researches one of the many methodologies available in machine learning and AI, knowledge distillation, as a means to interpret higher complexity models by using lower complexity, more interpretable models like decision trees, rule extraction or linear regression as surrogate models.

The following are the goals for this thesis:

- Study the state of the art for interpretability on critical decision making systems on the financial sector.
- Study the available frameworks for model development
- Define case studies for financial applications, on loan acceptance and asset
- Explore knowledge transferring to the extent to which one can replicate a neural net behaviour with less complex models.
- Explore ways of increasing decision tree based models accuracy with deep neural nets.
- Test for points of diminishing returns for knowledge transferring
- Define, implement, and fine tune an interpretability pipeline for interpretation of deep neural nets with the usage of interpretable models as surrogate models by the application of knowledge extraction methods.
- Propose and deploy test setup.

If the surrogate model shows improvement over the same model class trained one can assume the method works for augmenting the predictive power of these interpretable models. If the method shows that the model to which the knowledge is being transferred to is able to somewhat mimic the black-box model even if at the cost of some accuracy, given the appropriate context i.e. for a credit loan it's important that the model has high recall, we can assume the possibility of replacing it or create a new model as an ensemble where teacher and student work together. We study how closely related the decision making of the surrogate model is to the decision making of the black-box model, meaning the fidelity of these models to the model they are trying to mimic by looking at possible explanations and feature importances between the models.

1.4 Research Questions

During the conception of this thesis we try to answer the following questions prompted by the literature and research:

- Can we extend knowledge transferring to a point that allow the models we are transferring knowledge to have the same accuracy or even better than the cumbersome but powerful models we are transferring knowledge from?

- What is the point of saturation of the models we are testing as "student" models?
- Can student models perform better at specific tasks?
- Are decision trees truly interpretable?
- Does a bigger neural net model mean a bigger decision tree is needed?
- How does the student model compare to the teacher in terms of fidelity?
- Are do decision-trees give good stable explanations?

1.5 Contributions

We show that there is an opportunity to create ensembles of DNN and more interpretable models while trying to maintain the predictive performance of DNN's on certain financial contexts such as credit classification. We understand that no matter the complexity of the teacher model, the problem resides on how well the data given to the student models and the model itself performs on the independent regression problem obtained from the teacher, this means that the complexity of the student is not directly related to the complexity of the teacher. Although students try to mimic the behaviour of teachers and not surpass or be worse than them we believe that students can indeed be trained to perform better at certain tasks given that certain contexts, results show that in some cases students can, for example, have better recall for the minority class, this is useful in a credit scoring context. This also answer questions on fidelity, by looking at feature importance between teachers and students we can see that although we can obtain similar results the decision processes are indeed different. And finally by grouping different individuals based on their features and looking at the decision paths for those decisions we understand that decision-trees have good stability for explanations. As a result of this thesis two publications were made:

1. **"Interpreting Decision Patterns in Financial Applications"**: published on May of 2021 and presented at the 25$^{\text{th}}$ edition of the annual Iberoamerican Congress on Pattern Recognition[2]
2. **"Using Knowledge Distillation to Interpret Credit Score Modeling"**: published and presented at the University of Évora for the 27$^{\text{th}}$ Portuguese Conference on Pattern Recognition (RECPAD2021).

Both these papers can be found in the appendices of this document.

1.6 Document overview

This chapter sets the context and high level goals of this dissertation. Chapter 2 describes the background concept and scope of interpretability on the financial sector and provides an overview of the state of the art for interpretability for intelligent systems. Chapter 3 sets the research questions, the proposed approach and methodologies to address them. Chapter 4 describes the results obtained on the different datasets and finally Chapter 5 concludes the document and gives insight into the future work to be done.

[2]https://ciarp25.org/wp-content/uploads/sites/10/2021/05/CIARP25-Papers.pdf

Chapter 2

Background and State of the Art

In the last few years AI has been embraced across the industry and has constantly proven to be a great addition towards increasing performance while reducing operation costs. While AI is not a particularly new thing, there is a clear consensus on the paramount importance featured nowadays by intelligent machines endowed with learning, reasoning and adaptation capabilities. It is these same capabilities that are allowing AI methods to achieve unprecedented levels of performance when learning to solve increasingly complex computational tasks, making them one of the pillars of future development of the human society [4]. As AI has evolved over the years, we see a clear direction towards systems where human intervention is almost non-existent. When these systems take part on important decisions that ultimately affect an individual's life such as in medicine, law, finance and justice there is an emerging need for understanding how such decisions have been led to by these autonomous systems. While models like decision-trees, linear and logistic regression or generalized addictive and linear models are often considered interpretable [5, 6, 7], the last years have witnessed the rise of black-box models such as DNN which can combine efficient learning algorithms with huge parametric spaces making them very complex. Added to the fact that these models are now implemented on the contexts described before the demand for transparency is increasing across the various sectors that make use of AI [8]. In general, humans are reticent to adopt techniques that are not directly interpretable, this creates untrustworthiness in these models and can be seen as an obstacle to the propagation and evolution of said these systems. In recent years we have seen an increasing number of problems that address AI on an ethical level, as an example we have numerous cases of biased AI decision making from sexist recruitment systems to racial bias in healthcare the industry had their fair share of scandals [9, 10], allowing for model interpretability lets stakeholders mitigate these type of problems while keeping them safe from the emerging rigorous legislations that are becoming standard on digital world such as the General Data Protection Regulation (GDPR) on Europe or the newly drafted rules for harmonizing AI systems in Europe.

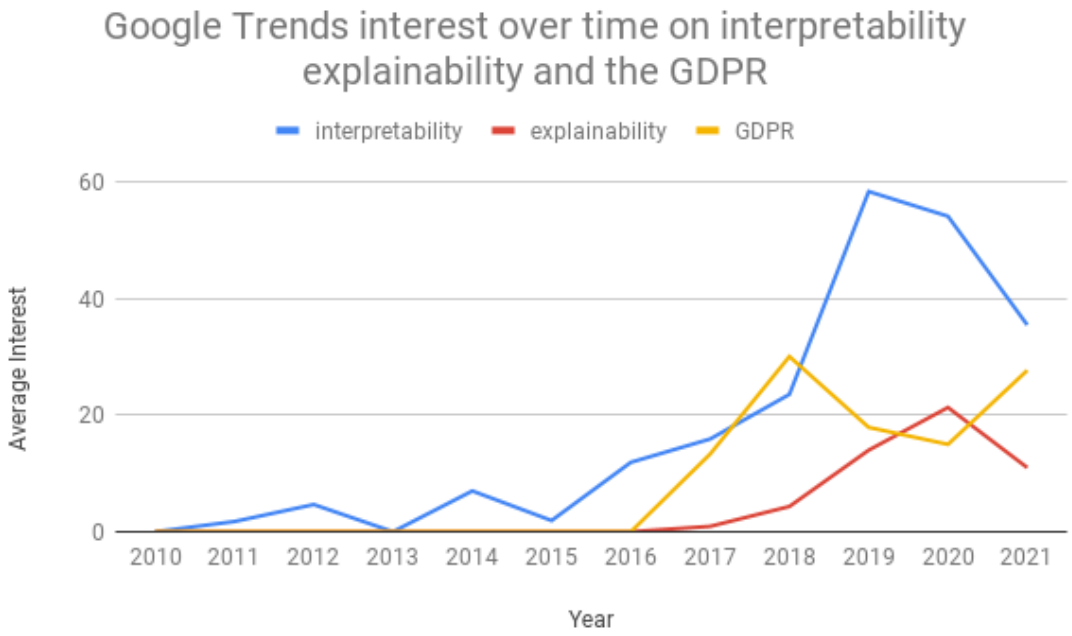


Figure 2.1: Average interest over the last years on the subjects of interpretability and explainability. Data source: Google Trends[1]

If we look at **figure 2.1** we see the interest over time on interpretability by looking at the google trends search terms for interpretability in machine learning and artificial intelligence, we also see that there is an huge increase on the year of 2016, we can associate this with the publishing of GDPR and the implied "right to explanation" [11].

While the ensurance of impartiality in decision-making by detecting and correcting potential bias in the training datasets is indeed one of the main arguments for interpretability, ethical reasons are not the only argument. Interpretability can act as an insurance that only meaningful variables infer the output as well as facilitates robustness by highlighting potential adversarial perturbations that could change the prediction. However, interpretability comes at the cost of performance, it is customary to think that by focusing solely on performance, the systems will be increasingly obscure. The same is true for the opposite, by increasing a models interpretability we might be decreasing it's performance, this is known as the interpretability/performance trade-off [12]. Still, there is an argument on how improving the understanding of a system can lead to the correction of its deficiencies, therefore allowing for better predictions. **eXplainable AI** emerged as a response to the increasing needs for interpretability by proposing to create a set of ML techniques that aims to produce more explainable models while maintaining a high level of learning performance (e.g., prediction accuracy) as well as enabling humans to understand and appropriately trust these models. It is necessary that we understand that interpretability is not a linear concept and that it changes depending on the context it is set in, this means it depends on the class of models we are trying to interpret as well as the data as well as the explainer we are using. In [13] a problem-based classification of these methods and it is important that we understand the differences between these methods before contextualizing them in the finance sector.

[1]https://www.google.com/trends

2.1 eXplainable Artificial Intelligence (XAI)

eXplainable AI (XAI) is a term first introduced by Van Lent et al. [14] to describe the ability of an AI system to explain its behaviour. XAI is a research field that aims to make AI systems results more understandable to humans.

Since then the increasingly adoption of intelligent decision making systems in the industry has shifted AI research towards implementing models and algorithms with emphasis on their predictive power giving less attention to the ability of explaining the decision making of these models. In recent years the social, ethical and legal pressure calls for new AI techniques that are capable of making decisions explainable and understandable.

Explainability has become one of the main barriers AI has been facing in recent years. The inability to explain or to fully understand the reasons ML algorithms perform as well as they do, is a problem that finds its roots in two different causes:

- **Gap between the research community and business** [15] sectors which impedes the full penetration of newer ML models in sectors that have traditionally lagged behind in the digital transformation of their processes such as banking, finances and security as these are highly regulated sectors that don't want to put their assets at risk [11].
- **Search for understanding**. It is clear that fields dealing with huge amounts of reliable data has largely benefited from the adoption of AI and ML techniques. Although we are entering an era in which results and performance metrics are the only interest shown up in research studies, science and society are far from being concerned just by performance. Understanding a model allows for its improvement as well as its practical utility.

According to DARPA [16], XAI aims to "produce more explainable models, while maintaining a high level of learning performance (prediction accuracy), and enable human users to understand, appropriately, trust, and effectively manage the emerging generation of artificially intelligent partners". In order to do this there is a great focus on researching new methods to extract information from complex models or simply focus on the conception of more interpretable models allowing for more transparency and therefore generating trust. FAT*[17] an organization that advocates for fairness, accountability and transparency in machine learning states that explainability "is to ensure that algorithmic decisions as well as any data driving those decisions can be explained to end-users and other stakeholders in non-technical terms".

The term XAI is closely related to interpretability since explaining assumes previous interpretation of the context/problem for the questions asked, thus the need to contextualize it for future references.

2.2 Laying down harmonized rules on artificial intelligence the newly proposed regulations on AI in the EU

On April 2021, a new draft was published on AI regulation in the European union, and is expected to take effect during the next few years. The proposal requires providers and users of high-risk AI systems to comply with rules on data and data governance,

documentation and record-keeping, transparency and provision of information to users, human oversight, and robustness, accuracy and security. AI systems identified as high-risk include AI technology used in:

- **Critical infrastructures** (e.g. transport), that could put the life and health of citizens at risk.
- **Educational or vocational training**, that may determine the access to education and professional course of someone's life (e.g. scoring of exams).
- **Safety components of products** (e.g. AI application in robot-assisted surgery).
- **Employment, workers management and access to self-employment** (e.g. CV-sorting software for recruitment procedures).
- **Essential private and public services** (e.g. credit scoring denying citizens opportunity to obtain a loan).
- **Law enforcement that may interfere with people's fundamental rights** (e.g. evaluation of the reliability of evidence).
- **Migration, asylum and border control management** (e.g. verification of authenticity of travel documents).
- **Administration of justice and democratic processes** (e.g. applying the law to a concrete set of facts).

Companies are to be regulated by external EU regulatory bodies and needed to comply to a set of requirements such as: notify them before their AI systems are put on the market or used; comply with certain data management requirements (related to data quality and representativeness); prepare extensive technical documentation for their AI systems (including demonstrating compliance). Among other requirements, regulated companies and individuals will also have to design their AI systems to meet certain accuracy, robustness, transparency, and cybersecurity standards, enable their outputs to be interpretable by users, and ensure human-in-the-loop capabilities during system use. High-risk systems in particular will be subjected to a set of strict obligations like:

- Adequate risk assessment and mitigation systems
- High quality of the datasets feeding the system to minimise risks and discriminatory outcomes
- Logging of activity to ensure traceability of results
- Detailed documentation providing all information necessary on the system and its purpose for authorities to assess its compliance
- Clear and adequate information to the user
- Appropriate human oversight measures to minimise risk
- High level of robustness, security and accuracy

With the the imposition of rules like the ones mentioned, there is no question that this is the beginning of a new era for interpretable AI where stakeholders across Europe must oblige to the regulations, or suffer the consequences of high value fines and possible full termination of services.

2.3 The "incompleteness" in the finance sector

One might say that it is not really necessary for ML systems to be interpretable, and we can say that is true in some cases. For example, an air craft collision avoidance systems—all computes all its output without the need for human intervention. Explanation is not necessary either because the problem is sufficiently well-studied and validated in real applications that we trust the system's decision, even if it is not perfect.

So when is an explanation required?

In [15] the need for interpretability stems from the "incompleteness" on the problem formalization.

In machine learning **incompleteness** refers to some kind of unquantified bias, for example the effect of including domain knowledge in a model selection process, this is not to be confused with **uncertainty** which can be, e.g., trying to learn from with limited resources, that can be quantified in some way.

From an **ethics** point of view a human may want to guard against certain kinds of discrimination, but fairness may be too abstract to be completely encoded into the system. Even if we can encode protections for specific protected classes into the system, there might be biases that we did not consider a priori e.g. one may not build a race-biased model for credit scoring, but a pattern in data may lead towards the denial of credit of a certain neighbourhood mostly inhabited by people of color that have been historically discriminated against, even if race is not used as a feature during training.

In the finance sector AI tools range from wealth-management activities, access to investment advice, and customer service. However, these tools also pose questions around data security and fair lending.

The financial industry is an highly regulated sector with loan issuers being required by law to make fair decisions. The need for interpretability in the finance sector mainly comes from the need to justify the decisions made. Why should an investor invest money on that stock at that time? What if the prediction ends up wrong? Was the investor aware of the risks he was taken? Why was an applicant's credit rejected?

In the financial context, there are at least six different types of stakeholders: (i) Developers, i.e. those developing or implementing an ML application; (ii) 1st line model checkers, i.e. those directly responsible for making sure model development is of sufficient quality; (iii) management responsible for the application; (iv) 2nd line model checkers, i.e. staff that, as part of a firm's control functions, independently check the quality of model development and deployment; (v) conduct regulators that take an interest in deployed models being in line with conduct rules and (vi) prudential regulators that take an interest in deployed models being in line with prudential requirements.

New regulations on AI 2.3 classify most decision support AI systems in the finance sector as high-risk (has an high impact on an individual's life) and seek to punish institutions that do not meet their requirements stipulated on fairness and transparency. This is creating a new space of **incompleteness** on the the current models applied in finance which are mostly black-box models. The focus required of these regulations are questions that protrude to the conduct regulators these are usually centralized on social contexts namely non-discriminatory decision making. Model interpretability leads to an easier detection of unknown liabilities on existing models, which then acts as a safeguard against new regulations.

The increasing the need for explanations when a prediction doesn't go according to the expected outcome has made it essential that ML/AI enabled decision processes justify their decisions, making interpretability an important requirement for these systems.

2.4 Evaluating Explanations

An important part for the evaluation of explanations and interpretability is having the right framework and a set of defined characteristics for doing so. In [15] a taxonomy is defined on the evaluation of interpretability, this taxonomy splits the evaluation into three categories: **application-grounded**, **human grounded** and **functionally grounded**. The important part to get from this type of taxonomy, would be that while **application-grounded** and human grounded evaluation require humans to be validated **functionally-grounded** evaluation doesn't. On **application-level** evaluation the explanation we are looking at how good an explanation is at the **application-level**, say for example we want to evaluate explanations on credit risk assessment, we can make our model generate such explanations and see how close it would be to an expert on credit risk assessment explaining the same decision. Human level evaluation is the same as application-level evaluations, but in this case, we do not require experts. Instead, we can use common persons to evaluate the explanations, this could simply be making someone pick the best one out of a group of explanations. Both methods are usually costly both in time and money, not only the experiments need to be properly set up in most cases which is time consuming but requiring human labor is also very costly especially in the case of human experts for application-level evaluation. The last category, **function-level evaluation** is human "independent" the example usually is this, we know that decision trees are considered interpretable in most cases thought human level and application level evaluations done prior, so we can use these to create explanations by proxy tasks and focus on the evaluation of the trees themselves i.e. a shorter tree leads to shorter if-then sentences and therefore gets a greater "explainability score" since these are easier to understand.

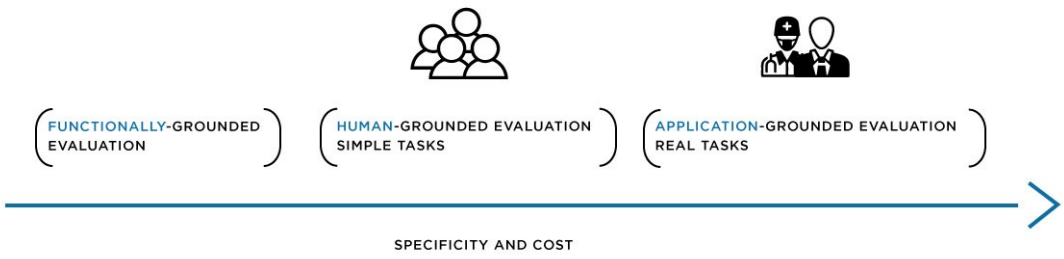


Figure 2.2: The different categories of evaluations. Functionally grounded evaluations tend to be supported by previously evaluated explanations at the application and human-grounded level. [15]

2.5 What makes a good explanation

Good Explanations are contrastive - Explanations are contrasting [18]. Humans usually do not ask why a certain prediction was made, but why this prediction was made instead of another prediction. We tend to think in counterfactual cases, i.e. "How would the

prediction have been if input X had been different?". In the context of loan and credit applications people tend to not care for the reasons to why their application was denied but rather they focus on what should happen for the application to be accepted. Contrastive is the difference between their application and the theoretically accepted application. **The tree explanations are contrastive, since you can always compare the prediction of an instance with other"what if"-scenarios by simply looking at the adjacent nodes to the decision path of a specific instance.**

Good explanations are selective - In general when an individual is looking for an explanation, it doesn't mean he/she is looking to achieve full understanding of every variable that took part in a decision, but the most important parts. Explanations should be short giving only 1 to 3 reasons to why a certain decision was made, even if the world is more complex.

Good explanations focus on the abnormal - In [19] it is understood that humans tend to focus more on abnormal causes to explain events. Tversky defines abnormal by the causes that had a small probability but nevertheless happened. The elimination of these abnormal causes would have greatly changed the outcome (counterfactual explanation). Humans consider these kinds of "abnormal" causes as good explanations. If one of the input features for a prediction was abnormal in any sense and this feature influenced the prediction, the fact that it's a rare occurrence makes it more so important. These rare occurrences should be included in an explanation.

Fidelity - The explanation should predict the event as truthfully as possible. So if we say that a having an high income job helps with having good credit, then that also should apply to all other people. For humans, fidelity of an explanation is not as important as its selectivity, its contrast and its social aspect.

2.6 Characteristics of interpretability

Interpretability is often described in three categories:

- **complexity** - more complex models are usually harder to interpret, one solution is to make models inherently interpretable, this often proves more difficult has often there's a clear trade-off between complexity and interpretability, complex models tend to have higher performance with low interpretability while explainable models are usually less powerful, NeuroDecision^{TM} [20] where Equifax tries to find the middle ground between explainable models and powerful black-box models.

- **scope** - there are two variations according to the scoop of interpretability:

  - **a)** we can try to understand the whole decision making process that's undergoing inside the model, this way we are trying to global interpretability. Some works that propose globally interpretable models include things like additive models for predicting pneumonia risk [21] or rule sets generated from sparse Bayesian generative model [22].

  - **b)** we try to understand why a model made a specific decision, which we see interpretability at the local level. Perhaps a famous example for these type of

explanations is LIME for Local Interpretable Model-Agnostic Explanation proposed by Ribeiro et al.[23].

- **related to the model** - if the techniques can only be applied to a specific type or class of ML algorithm we say they are model-specific, these methods aren't usually the focus as their low versatility provides limited usage
  . If the techniques can be applied to any type of ML algorithm then we are working with model-agnostic methods. Given their high flexibility there's been a surge in interest in model-agnostic interpretability methods. When we use these techniques we are separating prediction from explanation, making them post-hoc interpretability methods. In this work we focus model-agnostic method that aims to give us global explanations by using knowledge extraction from a DNN into more interpretable models like decision trees.

2.6.1 Methods for interpretability

It's important to understand that the concept of interpretability is not linear and the methodologies for interpretability differ heavily depending on the setup they are being implemented on. In [13] a problem-based taxonomy for classifying methodologies is created, where methods for opening black-box models are categorized on interpretability aim, the structure of the datasets used, the explanator and the black-box model itself. In [13] the interpretability aim is separated into four categories: interpreting the model, interpreting the model output, model inspection and transparency design. When we talk about interpreting a model we try to provide an interpretable model that tries to mimic the behaviour of the model we are trying to interpret. Decision trees are one of the most sought after interpretable models [5, 6] for this purpose and, as such, many methodologies have their basis on using these models as surrogates for interpreting black-box models. This is usually referred to as single tree approximations for neural networks. Single tree approximations for NNs were first presented in 1996 by Craven et al. [24] in the form of Trepan. Trepan queries a given network to induce a decision tree that describes the concept represented by the network, approximating the concepts represented by the networks by maximizing the gain ratio together with and estimation of the current model fidelity.

In [25], Krishnan et al. present a two step method for generating surrogate trees in order to debug complex black-box models. The first step consists in generating a prototype for each target class in the the dataset by using genetic programming to query the trained black-box model while the second step selects the best prototypes for training a decision-tree. This leads to more understandable and smaller models by focusing on small portions of the data set.

In [26] Johansson et al. mimics the behavior of a neural network ensemble by using genetic programming to evolve decision trees that combine the original training data with oracle data (test data labeled by the neural-network) labeled by the neural-network. Results showed that trees evolved using both the oracle data and the original data proved to be significantly more accurate on test data than trees evolved using only the original training data. Another commonly used state of the art understandable model is the **set of rules**. When a set of rules that describes the logic behind the reasoning of a black-box models is returned we achieve interpretability at the **global** level. The problem with these methods is that they are often model-specific and as such, they are not generalizable and can not be employed to solve other instances of black-box problems. In [27] Craven et al. try to explain the behaviour of a neural network by transforming rule extraction into a learning

problem. A training dataset X along with a randomized extension of it are provided as input to the black-box model. If the an instance $i \in X$ with outcome y' is not covered by the set of rules then a conjuntive rule is formed from i,y' considering all the possible antecedents. When we look at single-tree approximations for interpretability we have to look at the advantages and disadvantes of decision-trees as interpretable models. Decision trees are great at capturing interactions between features and have great visualization making for great human-friendly explanations. However, decision-trees fail to deal with linear relationships since any linear relationship between the feature and the output has to be fragmented into steps and approximated by splits. Decision-trees are also unstable, meaning a few changes in the input feature can have a big impact on the predicted outcome, which is usually not desirable. Another problem with decision-trees is that their interpretability is inversely proportional to their depth, meaning the more terminal nodes and the deeper the tree, the more difficult it becomes to understand the decision rules of a tree.

In [28] Johansson et al. explore the accuracy vs. comprehensibility problem by exploiting G-REX, an algorithm used for rule extraction. By using random premutations of the original dataset X labeled by the black-box model as y' they then use X, y' as input for G-REX. G-REX then extracts symbolic rules by exploiting genetic programming as key concept. Other methodologies, and perhaps the most sought after, are agnostic with the respect to the black-box model to be explained, this means they are not limited to only one black-box model making them more versatile. One of the first attempts at agnostic explanations was proposed in [29] by Lou et al.. The authors propose a method that exploits GAMs (Generalized additive models), which are considered very intelligible when only univariate terms are considered. These are able to produce an explanations as the importances of the contributions of each individual feature along with their shape function in the form of regression splines and trees or ensembles of trees. The shape function is the plot of a function that captures the linearities and the non-linearities of a specific feature in relation to the target. This method only works when the data is tabular. A disadvantage of GAMs is that it only relies on assumptions about the data generating process. If those are violated, the interpretation of the weights is no longer valid. The performance of tree-based ensembles like the random forest or gradient tree boosting is in many cases better than the most sophisticated linear models.

PALM shown in [25] which was mentioned previously is also an agnostic method. PALM mimes a black-box model by using a meta-model for partitioning the training dataset and a set of sub-models to approximate and mimic the patterns shown by the black-box model on each partition. The sub-models linked to the leaves of the tree can be arbitrarily complex making PALM black-box agnostic.

Rule extraction has lots of advantages, first of all IF-THEN rules are easy to interpret. A decision rule can also be seen as more compact version of a decision tree while not suffering from the problems decision-trees have such as redundant sub-trees. Another important advantage is that IF-THEN rules usually generate sparse models, which means that not many features are included. They select only the relevant features for the model. This goes in and with the selectiveness of explanations. Decision rules however also some disadvantages. The research and literature for IF-THEN rules focuses on classification and almost completely neglects regression. Decision rules are bad in describing linear relationships between features and output, since they produce step-like prediction functions not allowing for smooth curves. This is related to the fact that rule extraction often requires the features to be categorical, with numerical attributes having to also be transformed into categories.

We've looked at methods for explaining the model, but another important category for methods for interpretability is methodologies that allow for explanations of the outcome of a black-box model, This category approaches provides **local explanations** by using local points of view with respect to the predictions, and has become the most studied in the field of interpretability in the last few years. A common term used in the interpretability is **saliency maps** or **saliency mask**, a saliency mask is a subset of the instance which is mainly responsible for predictions. These are usually used in image and text classification i.e. a saliency image may summarize what a DNN is focusing on an image to make its prediction. It's important to note that these methods are not generalizable and are often tied to a particular type of DNN. In [30] Kelvin Xu et al. introduce an attention based model that can identify the contents of an image. In this work the black-box models consists of an ensemble of two neural-nets. A Convolutional NN (CNN) for feature extraction and a Long short term memory neural network for producing the image caption, that generates a single word for each iteration. The explanation provided is an image that highlights the most important parts for classification as seen by the neural net.

In [1] the authors explore the concept of saliency maps by creating an end-to-end-trainable attention module for convolutional neural network (CNN) used for image classification. The module allows for end-users to visualize the intermediate representations of the input image at different stages in the CNN pipeline.(see **figure 2.3**)

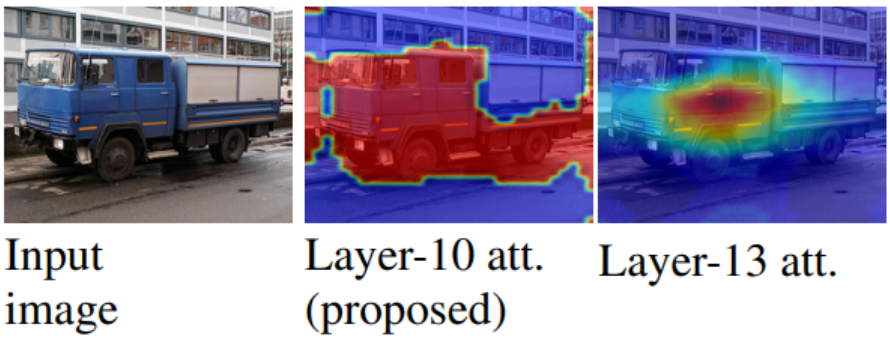


Figure 2.3: Example excerpt taken from [1] of an attention map.

The idea of incorporating hidden layer activations into the visualizations of saliency maps has been named Class Activation Mapping (CAM). A CAM for a particular outcome label indicates the discriminative active region that identifies that label. In [31] the global average pooling in a CNN is uses for generating the CAM. These approaches, although not black-box agnostic to neural networks still require specific model architectures or access to their intermediate layers.

In [23] Ribeiro et al. propose an model agnostic method for explaining outcomes of black-box models in the form o ***Local Interpretable Model-agnostic Explanations*** (LIME). LIME tries to return an understandable explanation for a specific prediction by deriving it locally from the records of the neighbourhood around the record to be explained. Lime returns the importance of the features as explanation. When using Lasso or short trees, the resulting explanations are selective and possibly contrastive making for good explanations. Another advantage of LIME is that this method works on tabular, images and text data. A shortcoming of this method is the required transformation of any type of data in binary format leaving for high dimensionality datasets which mind be an hindrance when using some classes of models.

Another model-agnostic method for interpreting black-box models is presented in [32] where

Lundberg et al. propose a new framework for interpreting black-box models called named SHAP which makes use of shapley values. A shapley value represents the contribution of a feature for a particular outcome by comparing it to the general outcome of that feature. SHAP assigns each feature an importance value for a particular prediction i.e. negative shapley values for a certain feature of value v mean that the feature having value v contributed negatively to the prediction.

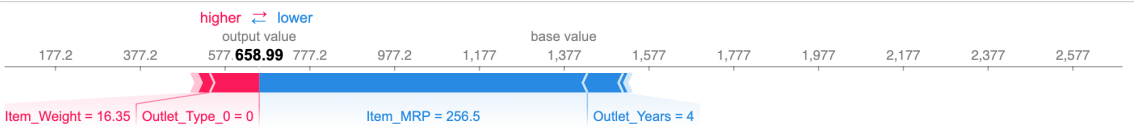


Figure 2.4: Example of SHAP being used in price prediction for housing. In blue, we have negative Shap values that show everything that pushes the sales value in the negative direction. While the Shap value in red represents everything that pushes it towards a positive direction. Note that this is for a single instance.

2.7 Knowledge Distillation and Knowledge Transfer

Knowledge Distillation was first introduced in 2015 [33] and is a generalization of Model Compression [34]. Model compression consists on the transfer of learned knowledge of a slower, larger and better performing model, usually referred to as teacher, onto a smaller model, the student, in an attempt to create faster models while keeping the same prediction power of a more complex one. Caruana et al. [34] achieves this by matching the logits of the smaller model to the logits of a cumbersome model, we refer to logits as the vector of raw (non-normalized) before the last activation layer of a DNN (see figure. 2.7). This means that the smaller model will approximate the behaviour of the more complex one by training on big ammounts of pseudo-data (logits) which in turn should get better results than the same model trained on real. The way this methods works is as follows : a large NN model $\boldsymbol{t}$ is trained using a training supervised dataset $D = \{X, y\}$ and producing output o, for each instance in D passed through $\boldsymbol{t}$ we extract the logit l_i, the vector of logits is then used as a new vector of targets y', that will be used as part of a new transfer dataset $T = \{X, y'\}$. The transfer dataset is used to train a smaller model s as a regression problem.

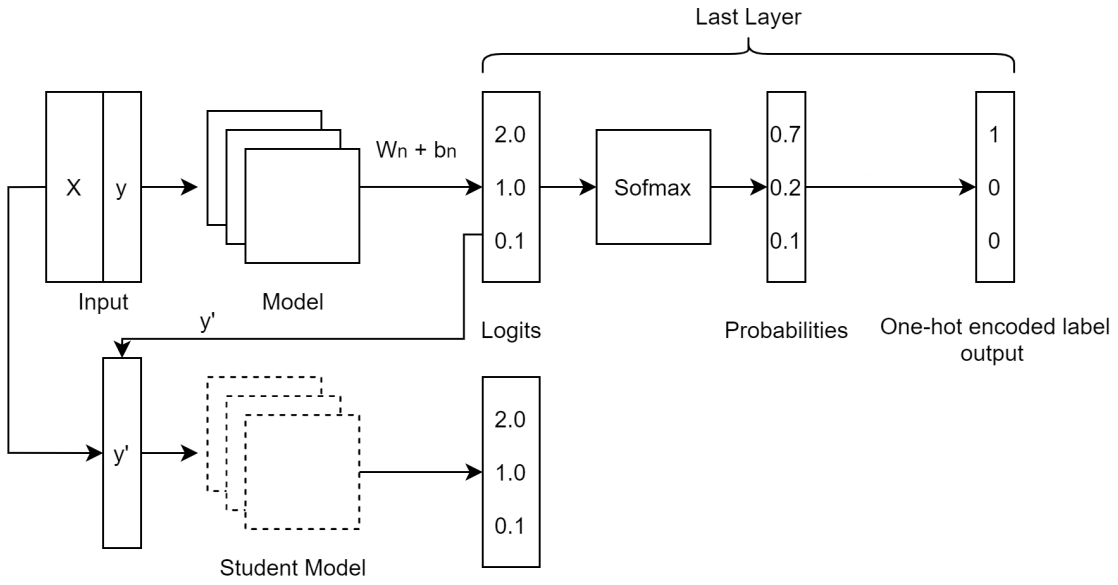


Figure 2.5: Simple example of model compression, here the values of the logits are used as targets for training a student model

Knowledge Distillation is a variant of this approach proposed by Hinton et al. [33], which uses the last layer's soft probabilities instead of logits as targets for training the student model. In this approach Hinton et al. make use of a softmax with teamperature, the difference this and the usual "softmax" is that to calculate the probability for logit l_i is the addition of a new variable T (see equation 2.1 that allows for a softer distribution of each probability vector.

$$l_i = \frac{exp(l_i/T)}{\sum_j exp(l_j/T)} \tag{2.1}$$

Using a higher value for T produces a softer probability distribution over classes, allowing for a better understanding of possible interactions between each class. One of the main claims about using soft targets instead of hard targets is that a lot of helpful information can be carried in soft targets that could not possibly be encoded with a single hard target. In [33] Hinton et al. test this assumption on a speech recognition problem, and results show that this effect is very prevalent when using a considerably less amount of data. They show that training the baseline model with hard targets leads to severe overfitting, whereas the same model trained with soft targets is able to recover almost all the information in the full training set, showing that soft targets are indeed an effective way of communicating the regularities discovered by a model trained on all of the data to another model.

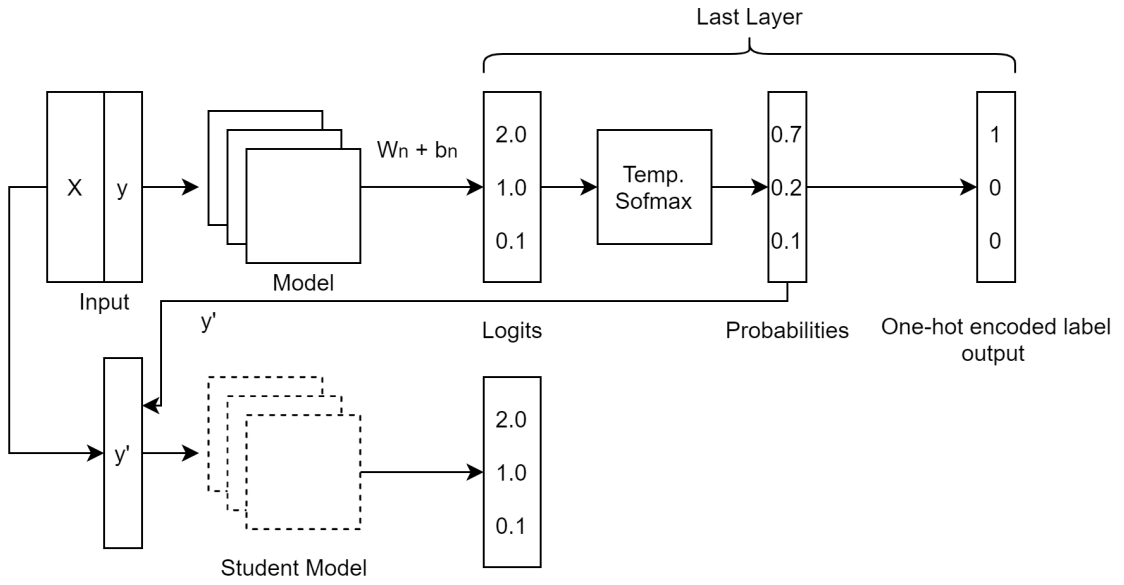


Figure 2.6: Example of Knowledge Distillation, similar to compression but in this case the output of a softmax with temperature is used for training

For training the student models, the most usual approach is by using a function loss like Mean Absolute Error (MAE) or Mean Squared Error (MSE) with the latter being preferred for performance and time reasons. Given an array of size n of true target values y and an array of same size of predicted values y^p, MSE is the mean overseen data of the squared differences between true and predicted values see equation 2.2. When the student is training it's expected that there's no loss of validation as it is mimicking an already trained model, as such the performance of the student is highly dependent on the performance of the teacher.

$$MSE = \frac{\sum_{i=1}^{n}(y_i - y_i^p)^2}{n} \tag{2.2}$$

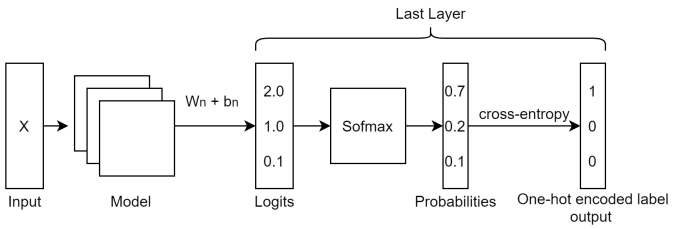


Figure 2.7: Example model, the logits are the vector of values that serves as input to the softmax activation layer that will then output a vector of probabilities for each class in a classification problem.

2.8 Conclusion

Throught the research and literature reviewed we can conclude that interpretability should not be an option anymore, but a standard across the many AI systems used in the various sectors of everyday life. We see that most methodologies have great advantages when it comes to interpretability, but also suffer at the cost of accuracy, this has been the main

obstacle in the adoption of interpretability. And the hope is that while we can try to make black-box models interpretable throught the use of post-hoc methods like LIME, ELI5 or SHAP, the future standard should be to build potent models (in terms of accuracy) that are transparent and inherently interpretable or allow for easier interpretability methods that can be built upon it. Given that most of the research is done in post-hoc methods like the ones mentioned before [23, 32] which is mostly based in probing of features and their values, the aim of this research is to explore knowledge distilaltion as a means of creating inherently transparent models by using black-box models as guidance for training.

Chapter 3

Proposed Approach

3.1 Problem Definition

Higher complexity models like DNNs are obscure by nature, making it hard to understand the decisions made by these models. In general we can classify machine learning decision-support systems into two types [35]:

- **Type A applications** - model predictions are used to support consequential decisions that can have a profound effect on people's lives such as medical diagnosis, loan applications, prison sentencing, safety is paramount;
- **Type B applications** - model predictions are used in settings of lower consequence and large scale, these are our YouTube recommendations, our Facebook ads, the news that show on our feed and every other type of recommendation online, safety is less important.

For this dissertation we focus on Type A applications on the finance sector, respectively loan applications [36]. Over the course of history we have seen discriminatory behaviour towards ethnicity, religion and sex across the industry, remuneration is one particular case as the example shown in [37]. Every decision-making ML algorithm requires data to be trained on, and most of this data is historical. For the financial sector this data tends to be human-derived, not many years ago the outcome of a credit loan was decided by one or a group of a few humans. It is safe to assume that the biased behaviour of humans towards other humans has translated onto this historical data. This gives the algorithms the opportunity to become biased themselves if that means having better accuracy at predicting an outcome.

As part of the work two problems are defined:

1. Are we able to use decision-trees to interpret or replace black-box models in the finance sector?
2. Is the depth of the decision tree that is trying to mimic a black-box model proportional to the complexity of the black box model itself?

In order to answer this questions five datasets were aquired and numbered for representation purposes. In the following section we describe each dataset as well as the problem they are trying to solve.

3.2 Methodology

We propose interpreting the decisions made by a black-box model by using knowledge distillation in order to create interpretable models.
Given dataset $D_i = \{X, y\}$ a black-box model b_i made specifically for dataset i is trained on the full dataset. In order to train model b_i some pre-processing is required, this comes in the form of feature engineering, null value treatment and feature scaling.

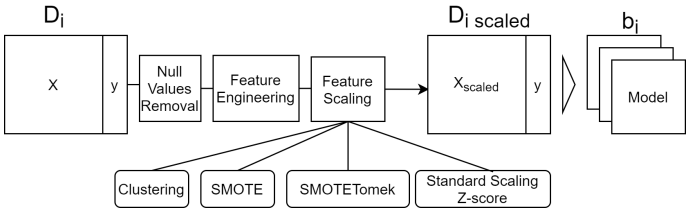


Figure 3.1: Pre-processing and imbalance treatment of the dataset is done, the scaling values are saved so we can inverse-transform the results after.

The next step consists in the acquisition of the soft labels by extracting the logits and probabilities from the last layer of each model b_i by passing the full dataset $D_i scaled$ throught model b_i obtaining the vector of soft labels y'_i. For multi-class problems this vector is a vector of vectors of size equal to the number of classes c, turning the learning problem of the student a multi target output problem.

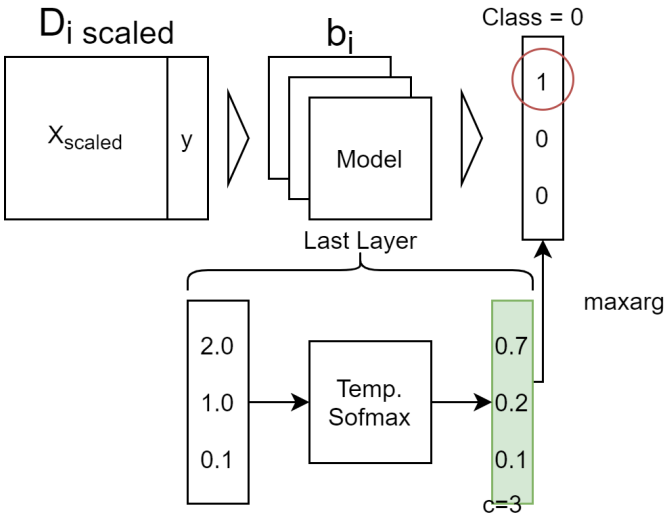


Figure 3.2: The soft labels are extracted from the last layer, before the highest value class is picked as the prediction. In this example we have a 3 class prediction problem.

We now have a transfer set $T_{i_{scaled}} = \{X_{scaled}, y'\}$ that will be used to train a student model s_i on a regression problem, in this case the model is a Decision Tree Regressor, using the soft labels y'_i as targets for training.

The metrics of students and teachers are compared in order to verify the validity of the method, if the metrics are similar, we consider it a success.

After training each student, another model of the same class is trained on the original dataset in order to detect the differences in the trees decision paths. Feature importances are also taken into account. This also serves as validation for the consistency and stability of the trees and the explanations they can provide. In context of biases in machine learning

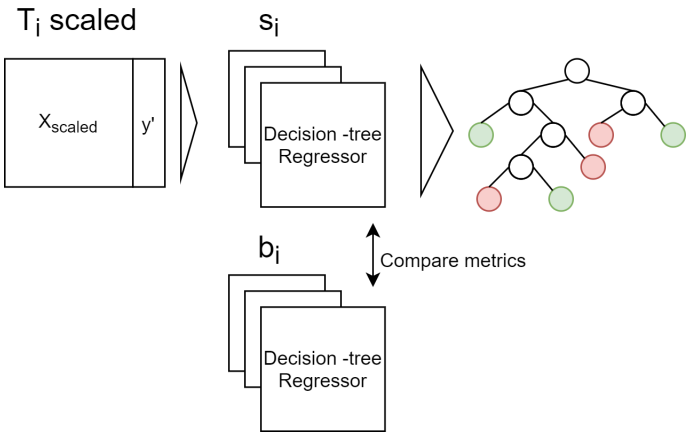


Figure 3.3: The student, a decision tree regressor is trained on the soft labels aquired from the teacher, the black-box model.

algorithms for datasets 1 and 5, the use explanation methods like the SHAP[32] or LIME[23] in both models, allows for the detection of possible signs of discriminatory behaviour by the black-box models.

1. We first start by training a deep neural net that we will call the teacher
2. We take take the logits before they are passed through the last activation layer, this layer is usually a softmax but since all our problems are binary we use a sigmoid activation layer.
3. We then train the picked student model using the logits we took from point 2 as target labels, this first student is always trained using regression techniques
4. To test the student we simply get its output in the form of "logits" and pass it through an activation function, in our case a "sigmoid".
5. The results are then transformed into their binary form
6. A second student model is trained but this time with the actual binary labels
7. The second student is tested
8. Results are compared

Figure 3.4: Steps for knowledge transfer.

3.3 The effect of temperature on the students training

The students are trained with the logits from the last layer of the teacher. Generally the *Softmax* layer outputs the vector q_i of class probabilities after receiving a the logits z_i from

the previous layer. By adding a new variable T so that:

$$q_i = \frac{e^{\frac{z_i}{T}}}{\Sigma_j e^{\frac{z_j}{T}}}$$

When we do this we soften the probability values. Let's say we are classifying a picture and we have 3 classes: dog, cat and rabbit, and the output of the softmax layer is $s = [0.1, 0.8, 0.2]$ after receiving the logits from the last layer. If we instead use a variable temperature that we can increase or decrease as we want we will change the probability distribution for the same class so that its softer, this means that the differences between each class get dilluted as we increase the temperature, taking the increasing the importance of the less probable classes while decreasing the probability of the most probable class, this means that at a certain temperature $s = [0.1, 0.8, 0.2]$ can become $s = [0.2, 0.5, 0.3]$ for example. Which might help smaller classifiers by giving them more information about the other classes. The effects of the temperature were tested on the Prosper Loan dataset as it had the most adequate structure to do so with the **PyTorch** framework. On figures 3.6 and 3.7 we can see how while the student gets increasing levels of recall on the minority class, the decision tree trained on hard labels can have an erratic behaviour while being trained on the same data everytime, keep in mind that although the graphic shows the temperature for the decision tree trained on the hard labels this has no effect on it. In order to show this we obtained the average of the F-score over 30 runs across different temperatures (see figure 3.5). We can see that we get a slight improvement on the F-score for the minor classes on the student until around a temperature of 8 where it drops suddenly, this happens because while the temperature increases we reach class equality as explained above, making the classification fuzzy as we start having less different probabilities between classes.

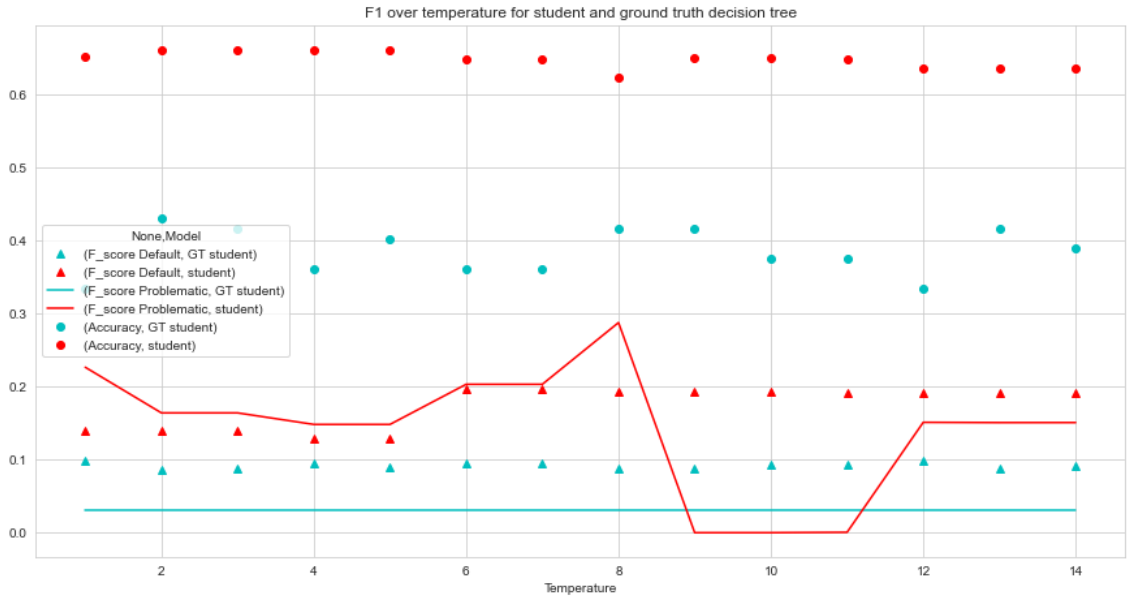


Figure 3.5: F1 score on student and decision-tree trained on hard labels

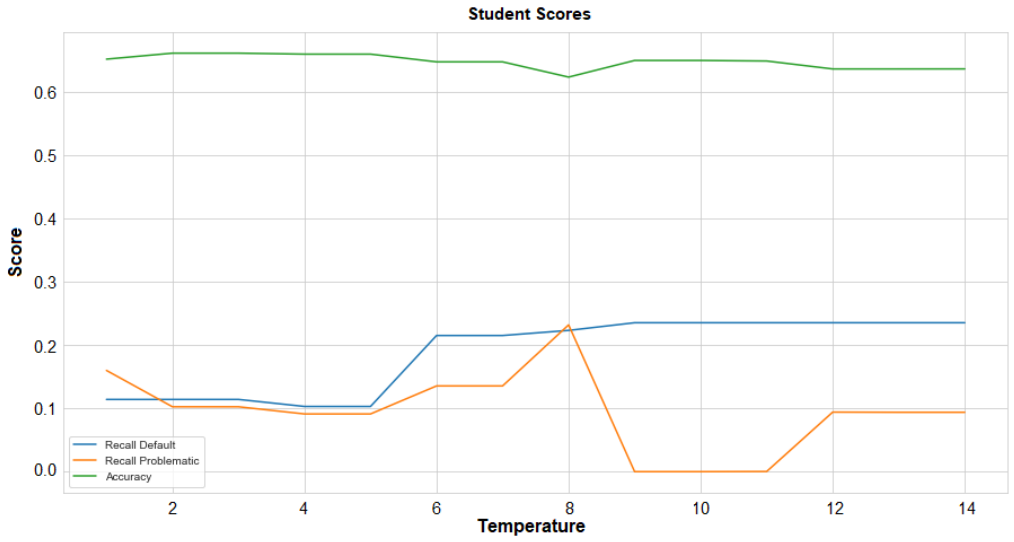


Figure 3.6: Student Decision-Tree trained on soft labels

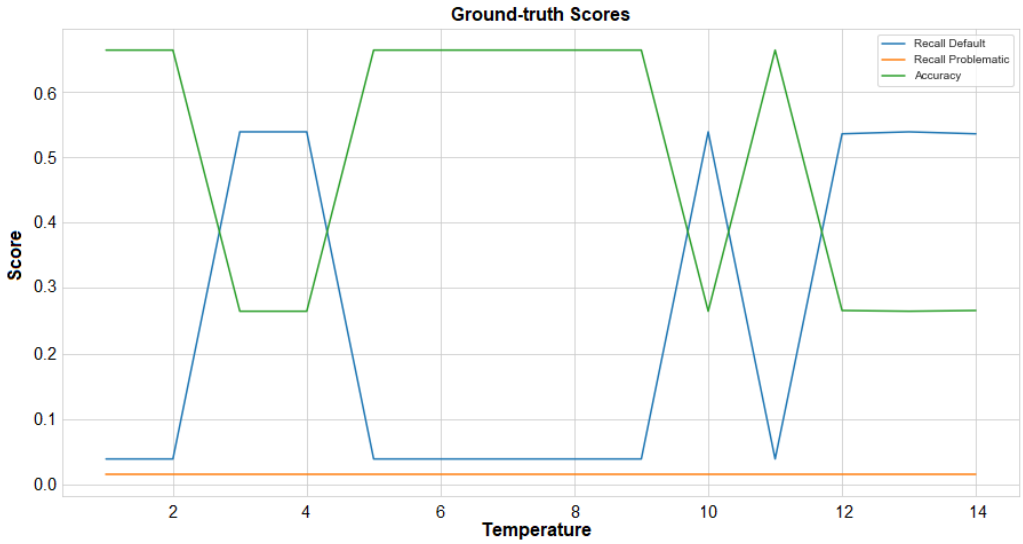


Figure 3.7: Decision-tree trained on hard labels

3.4 Research Questions

When it comes to interpretability in the sector of finance, the tendency is in approaches that let us explain a decision to justify why it is made. In problems like loan applications and credit scores, we aim for highly interpretable models, however if we are to use the method in, let's say stock movement prediction, we need to certify that it is highly accurate, as a bad prediction can cause severe loss of assets, this poses the following questions:

1. Can we extend knowledge transferring to a point that allow the models we are transferring knowledge to have the same accuracy or even better than the cumbersome but powerful models we are transferring knowledge from?
2. What is the point of saturation of the models we are testing as "student" models?
3. Can student models perform better at specific tasks?
4. Are decision trees truly interpretable?

5. Does a bigger neural net model mean a bigger decision tree is needed?

6. How does the student model compare to the teacher in terms of fidelity?

7. Are do decision-trees give good stable explanations?

Assuming the problems stated in **4.1** we are interested in seeing how far knowledge transfer can be pushed, although the core method is very simple, many parameters have yet to be tested. The research on knowledge transferring may also give rise to new methodologies for interpretability not only in the finance sector but other activity sectors as well. If the method was to be extended to "near perfect" transfer of knowledge one could interpret every black-box model in its totality or even replace most black-box systems without loss of information given that the method is model-agnostic.

3.4.1 Available Data and data preparation

For representation purposes each dataset acquired will be given a number represented in table 3.1.

Dataset Name	Dataset Number	Problem
German Credit UCI	1	1
TSLA	2	2
ADBE	3	2
GOOGL	4	2
Prosper Market Place	5	1

Table 3.1: Available dataset and their respective number and problem assignment

As we can see from table 3.1 a total of five available datasets was used. We can split these into two groups based on the problem definition. For the purpose of interpretability in the finance sectors we focus on datasets number 1 and 5 while datasets 2,3 and 4 are used for stress testing of the student models by using them in a more complex problem like stock market forecasting.

The threes stock market datasets are all of the same form and relate to stock market info from three companies, Google [38], Adobe Inc. [39], and Tesla Inc. [40]. These datasets were used in an attempt at setting the complexity of the problem higher for testing the efficacy of the method over a more complex problem that requires an architecture like LSTM.

Stock Market Data

The stock market was extracted for three companies: Google, Tesla Inc. and Adobe Inc.. The length of the data sequence varies for each dataset see table 3.2. All three datasets share the same attributes described below and all prices are represented in USD. The raw attributes are represented in table 3.3

In order to take advantage of the method used the data was transformed into a classification problem of n-days ahead stock movement prediction based on k previous observations. This means that the target for the supervised dataset is a binary value of 1 if the stock has gone up n-days into the future, 0 if the stock price has gone down n-days into the future.

Stock Name	Ticker	Start Date	End Date
Alphabet Inc.	GOOG	2004-08-19	2021-01-05
Tesla Inc.	TSLA	2010-06-29	2021-01-05
Adobe Inc.	ADBE	1986-08-13	2021-01-05

Table 3.2: Time span for each raw dataset retrieved

Attribute Name	**Description**
Opening Price	Market opening price
Closing Price	Market closing price
Adjusted Closing Price	Closing price adjusted to the market
High	Highest price during the day
Low	Lowest price during the day
Volume	Quota of stocks in the market

Table 3.3: Stock datasets attributes and their description

Raw data acquisition of stock market information

All data on the stock from the companies Google, Tesla Inc. and Adobe Inc. was retrieved using the yahoo!Finance API [41]. The Yahoo!Finance API is a range of methods to obtain historical and real time data for a variety of financial markets and products, as shown on Yahoo Finance [41].

German Credit

The German credit dataset is comprised of 1000 instances and classifies people described by a set of attributes as good or bad credit risks.
The data have been contributed as part of a dataset collection created by the Statlog EU project [1] with Prof. Dr. Hans-Joachim Hofmann listed as the data donor.
There are 20 explanatory variables with seven being numerical and 13 being categorical. These are briefly described in Table 3.4

Variable Name	Level	Code	good(%)	bad(%)
checkingAcc	<0 DM	1	13.9	13.5
	0<=...<200 DM	2	16.4	10.5
	>= 200 DM	3	4.9	1.4
	No Checking Account	4	34.8	4.6
credit_Hist	no credits taken all credits paid back duly	1	1.5	2.5
	all credits at this bank paid back duly	2	2.1	2.8
	existing credits paid back duly till now	3	36.1	16.9
	delay in paying off in the past	4	6	2.8

[1] https://cordis.europa.eu/project/rcn/8791/factsheet/en

Table 3.4 continued from previous page

Variable Name	Level	Code	good(%)	bad(%)
	critical account/ other credits existing (not at this bank)	5	24.3	5
purpose	car (new)	1	14.5	8.9
	car (used)	2	8.6	1.7
	others	3	0.7	0.5
	furniture/equipment	4	12.3	5.8
	radio/television	5	21.8	6.2
	domestic appliances	6	0.8	0.4
	repairs	7	1.4	0.8
	education	8	2.8	2.2
	retraining	9	0.8	0.1
	business	10	6.3	3.4
savingsAcc	<100 DM	1	38.6	21.7
	100 <= ... <500 DM	2	6.9	3.4
	500 <= ... <1000 DM	3	5.2	1.1
	>= 1000 DM	4	4.2	0.6
	unknown/ no savings account	5	15.1	3.2
employment_Stat	unemployed	1	3.9	2.3
	<1 year	2	10.2	7
	1<=...<4 years	3	23.5	10.4
	4<=...<7 years	4	13.5	3.9
	>= 7 years	5	18.9	6.4
deptor_stat	none	1	63.5	27.2
	co-applicant	2	2.3	1.8
	guarantor	3	4.2	1
property	real estate	1	22.2	6
	society savings agreement /life insurance	2	16.1	7.1
	car or other,not in attribute 6	3	23	10.2
	unknown / no property	4	8.7	6.7
other_instalment_plans	bank	1	8.2	5.7
	stores	2	2.8	1.9
	none	3	59	22.4
housing	rent	1	10.9	7
	own	2	52.7	18.6
	free	3	6.4	4.4
job_type	unemployed	1	1.5	0.7
	unskilled - resident	2	14.4	5.6
	skilled employee / official	3	44.4	18.6
	self-employed/ highly qualified employee	4	9.7	5.1
telephone	None	1	40.9	18.7
	Yes	2	29.1	11.3
foreign_worker	Yes	1	66.7	29.6
	No	2	3.3	0.4
sex	male	1	20.1	10.9
	female	2	0	0
marital_status	married	1	26.8	13.4

Table 3.4 continued from previous page

Variable Name	Level	Code	good(%)	bad(%)
	single	2	40.2	14.6
	divorced	3	3	2

Table 3.4: Categorical ratio of the predicted outcome over each attribute from the German Credit Dataset

Prosper Marketplace

Prosper Marketplace, Inc. is a San Francisco, California-based company in the peer-to-peer lending industry. This dataset is very similar in structure to the German Credit and was authorized to be used as part of this thesis by Dr. Branka Hadji Misheva. Given that this dataset is a more complete and raw one than the German Credit one which in turn has a more sandbox feel to it, the main purpose was to translate the work done on the German Credit dataset to this one to see if the method would still perform in a similar way given the noise and different dynamics present on this more complete one.The set is comprised of 113937 instances with each consisting of a group of 80 descriptive attributes that characterize the outcome of an individuals loan. After clearing up all null values and dropping some unnecessary columns, the set still consists 106290 instances and a total of 59 attribute columns. The LoanStatus column represents the target for classification that initially consisted of the following 11 classes see table 3.5

Class Name	Number of instances
Current	56566
Completed	33530
Defaulted	3289
Past Due (1-15 days)	806
Past Due (16-30 days)	265
Past Due (31-60 days)	363
Past Due (61-90 days)	313
Past Due (91-120 days)	304
Past Due (>120 days)	16
FinalPaymentInProgress	205
Charged-off.	10632
Cancelled	1

Table 3.5: Prosper Loan Dataset Class Distribution

The instances classified as current were dropped as they had no real value on the training, predicting of any of the models since we don't know what the final outcome was. The rest of the classes were grouped up with all Past Due becoming a new Problematic class; Charged off and Cancelled grouped up with Defaulted, and FinalPaymentInProgress coupled with Completed for the sake of keeping as much data as possible, as so we are left with a 3 class problem with the classes, Defaulted, Problematic and Completed. This leaves us with a dataset comprised of 49724 entries.

3.4.2 Dealing with the imbalance on the datasets

For all datasets referring to credit scoring, there was a problem with dealing with imbalance, this frequent on such dataset types since the number of defaulted cases is far inferior from the rest of the classes. To treat this a few techniques were tried out:

Over-sampling

Oversampling was achieved through the usage of SMOTE (Synthetic Minority Oversampling Technique), as the name implies SMOTE first selects a minority class instance a at random and finds its k nearest minority class neighbors. The synthetic instance is then created by choosing one of the k nearest neighbors b at random and connecting a and b to form a line segment in the feature space. The synthetic instances are generated as a convex combination of the two chosen instances a and b.

Mix of Over-sampling with Under-Sampling

Another tested technique was using both oversampling and undersampling. This method couples the ability to generate synthetic data for minority class of SMOTE with the Tomek Links (see definition 3.4.1) ability to remove the data that are identified as Tomek links from the majority class (that is, samples of data from the majority class that are closest with the minority class data). The process goes as follows:

1. Choose random data from the minority class.
2. Calculate the distance between the random data and its k nearest neighbors.
3. Multiply the difference with a random number between 0 and 1, then add the result to the minority class as a synthetic sample.
4. Repeat step number 2–3 until the desired proportion of minority class is met. (End of SMOTE)
5. Choose random data from the majority class.
6. If the random data's nearest neighbor is the data from the minority class (i.e. create the Tomek Link), then remove the Tomek Link.

Definition 3.4.1 (Tomek Link)**.** Let $d(x_i, x_j)$ denote the Euclidean distance between x_i and x_j, where x_i denotes sample that belongs to the minority class and x_j denotes sample that belongs to the majority class. If there is no sample x_k satisfies the following condition:

1. $d(x_i, x_k) < d(x_i, x_j)$, or
2. $d(x_j, x_k) < d(x_i, x_j)$

then the pair of (x_i, x_j) is a **Tomek Link**.

Clustering the abundant class

This approach consists of clustering the abundant class and using these clusters medoids representative of each group. For each cluster, only the medoid (centre of cluster) is kept. The model is then trained using all instances of the rare class and the medoids only. The problem with this method is that we are decreasing the dataset size considerably, as so it becomes unfeasible even if we get good results on the teacher.

3.5 Models Used

3.5.1 Deep Neural Nets - The Teacher

The teacher term refers to the more complex model from which the knowledge is being transferred, we call it teacher as it had to learn its predictions from scratch. Although DNNs are not the most common practice for tabular data, and there might be better and more interpretable solutions that could solve the problems at hadn, DNNs are still black-box models, and were used solely for the purpose of illustrating means to interpret such models. DNNs still outperform simpler models on complex problems, however the datasets that were provided didn't seem to formulate a complex enough problem to where DNNs could clearly outperform decision-trees.

3.5.2 Feed-forward Neural Network

For the German Credit dataset a simple Feed-forward neural network with 2 hidden layers and 1 output was used, provided the simplicity of the task. Feed-forwark networks are one of the most simple archiqtecture for DNNs, this proved to be enough for a problem as complex as the one provided by the german credit dataset.

Feed-forward Neural Network Hyperparameters

Model Parameters	
Parameter	Value
Number of Hidden Layers	2
Type	FFNN
Units in each layer	(256,128)
Bidirectional	False
Training Parameters	
Parameter	Value
Loss	binary_crossentropy
Optimizer	"adam"
EPOCHS	50

Table 3.6: Feed-forward Neural Network parameters for credit score classification on the German Credit Dataset

3.5.3 Long Short-Term Memory Neural Network

For the stock movement prediction dataset an LSTM arquitecture was applied. The reason for using this type of architecture is because LSTMs are useful to deal with sequential data or data with temporal relationships, since there can be lags of unknown duration between important events in a time series. LSTM's have internal memory, meaning they can store data and relate it with the current data, this can be very useful in the problem of stock movement as we want to predict future movements based on previous observations giving us far better results than other traditional algorithms. The neural network only ended up with 2 hidden layers and was trained on each dataset, for 120 epochs divided in batchs of size 64, for the loss function binary cross entropy:

$$H_p(q) = \frac{1}{N}\sum_{i=1}^{N} y_i \cdot log(p(y_i)) \cdot (1 - y_i) \cdot log(1 - p(y_i)) \tag{3.1}$$

Where y is a binary label (1 or 0) and $p(y)$ is the probability of y being whichever is the chosen label (1 or 0) for all the N instances of points.

LSTM Hyperparameters

Model Parameters	
Parameter	Value
Number of Hidden Layers	2
Cell Type	LSTM
LSTM Units	256
Bidirectional	False
Training Parameters	
Parameter	Value
Learning Rate	0.001
Loss	binary_crossentropy
Optimizer	"adam"
Batch Size	64
EPOCHS	120

Table 3.7: Long Short-Term Memory neural network parameters for stock movement prediction

3.5.4 Prosper Loan's Teacher FFNN

For the Prosper Loan dataset a different DNN was used since its a similar problem to the German Credit dataset one but working on a 3 class classification. When working with neural networks, the main parameters where optimization occurs is namely activation functions, number of hidden layers and their respective number of neurons, learning rate (lr). Each dataset was split into 70% training and 30%testing while keeping the original class ratio for the test set.

Class	Values
L1	[512,256,128,64]
L2	[512,256,128,64]
L3	[512,256,128,64]
L4	[512,256,128,64]
lr	[0.1,0.01,0.001]

Where Ln represents the number of units in the n^{th} hidden layer and lr represents the learning rate. Each layer was followed by batch normalization with ReLU activation (see equation 4.1 and a dropout with 20% probability. Each model seemed to stabilize at 50 epoch's after hyper parameter tunning so the number of epochs was fixed at 50. The end results is a deep-neural net with 4 hidden sequential layers of sizes 512,256,128,64 and a learning rate of 0.001.

$$ReLU(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & x < 0 \end{cases} \tag{3.2}$$

Prosper FFNN Hyperparameters

Model Parameters	
Parameter	Value
Number of Hidden Layers	4
Cell Type	FFNN
LSTM Units	[512,256,128,64]
Learning Rate	0.001
Bidirectional	False
Training Parameters	
Parameter	Value
Loss	binary_crossentropy
Optimizer	"adam"
Batch Size	254
EPOCHS	50

Table 3.8: Long Short-Term Memory neural network parameters for stock movement prediction

3.5.5 Interpretable Models - The Students

The term student here refers to the model to which knowledge is being transferred to, we call it student, as it is making use of soft labels that were acquired by training another model that we call the teacher. The decision-tree structure was used as a student model, due to its visualization and the fact that it is often considered an interpretable model.

3.5.6 Decision trees

Decision trees are highly interpretable models and the focus for the intermediate report. The interpretation is simple: We start at the root node, at each node we have an if condition that determines which following node we are descending to, when we reach a leaf

node we get the prediction outcome. We then get all the nodes we've been to and connect those with 'AND' giving us a chained IF condition.
Example: If feature *W* is greater than threshold c AND feature **Y** is greater than threshold **X** AND ... then the predicted outcome is **Z**. The tree structures are great for capturing interactions between features in the data. The data ends up in distinct groups that are often easier to understand than points on a multi-dimensional hyperplane.

The tree structure also has a natural visualization, making interpretation pretty simple.

Student Hyperparameters

All six gradient boosted tree models followed the same hyperparameters, each individual case should get improvement from hyperparameter tuning.

Hyperparameter	On XGBoost model	Value
The number of trees (n_estimators)	n_estimators	100
Max depth(max_depth)	max_depth	3
Learning Rate	learning_rate	0.1
Minimum Child Weight	min_child_weight	1
Subsample ratio of the training instance.	subsample	1
Subsample ratio of columns when constructing each tree	colsample_bytree	1
Subsample ratio of columns for each level	colsample_bylevel	1
Gamma	gamma	0

Table 3.9: Student Hyperparameters Table

3.5.7 On the comprehensibility of decision trees

There is a point to be made on whether decision trees are interpretable or not, for example: one may say that high depth trees are not interpretable given their exponentially increasing complexity. However, through the literature we see that in most cases where the interpretability of decision-trees falters, i.e., node redundancy and tree depth can be addressed by post-hoc processes. Many user-based evaluation experiments were conducted on the interpretability of decision trees already. In [42], 100 non-expert users were asked to compare the understandability of decision trees and rule lists induced from two small datasets from the UCI ML repository – namely, Contact Lenses and Labor. Decision trees were in general deemed by users to be more understandable than rule lists for both datasets. In another experiment, decision trees and tables were compared in the context of a computer game where users were asked to interpret decision-trees and decision-tables to make investment decisions that maximized their profit in the game [6]. Among a group of 67 non-expert users, decision trees were overall found to be more comprehensible than decision tables. The greater comprehensibility of decision trees was attributed to their ability in graphically revealing the patterns in the data and the ease with which users can follow a tree path until its leaf node. We understand that in general the tree structure is great for visual representation and extraction of decision-rules as such is a good model to be used in knowledge distillation as a surrogate model for neural-nets.

Chapter 4

Experimental results and analysis

In this chapter we provide an overview of the available data data and an in-depth analysis of the results we obtained from each dataset as well as the experimental setup and processeswe obtained them by.

4.1 Training of the Neural Nets

All neural nets were trained using grid-search across a number of parameters described in the following sections. For problems 1 and 5 each neural network was ran throught the same set of parameters for grid searching, as the problems were extremely similar in terms of features, with the most significant difference being the number of classes for the classification task. While problem 5 is a 3 class classification problem problem one is a binary classification problem. When working with neural networks, the main parameters where optimization occurs is namely activation functions, number of hidden layers and their respective number of neurons, learning rate (lr). Each dataset was split into 70% training and 30%testing while keeping the original class ratio for the test set.

Class	Values
L1	[512,256,128,64]
L2	[512,256,128,64]
L3	[512,256,128,64]
L4	[512,256,128,64]
lr	[0.1,0.01,0.001]

Where Ln represents the number of units in the n^{th} hidden layer and lr represents the learning rate. Each layer was followed by batch normalization with ReLU activation (see equation 4.1 and a dropout with 20% probability. Each model seemed to stabilize at 50 epoch's after hyper parameter tunning so the number of epochs was fixed at 50. The end results is a deep-neural net with 4 hidden sequential layers of sizes 512,256,128,64 and a learning rate of 0.001.

$$ReLU(x) = \begin{cases} x, & \text{if } x \geq 0 \\ 0, & x < 0 \end{cases} \tag{4.1}$$

4.2 The training of student models

Student models represented a regression problem. This means the class of models used for this problems had to be regressors. As we were looking for a decision-tree structure the models picked were gradient boosted trees for regression and decision tree regressors.

Model	Problem
Gradient Boosted Trees	1,2,3,4
Decision Tree Regressor	5

Table 4.1: Caption

4.3 Results

4.4 German Credit

The German Credit dataset, although very simple both in size and structure, was a prepared dataset, as such it didn't present many difficulties to work with besides it's small size. This dataset served as the foundation for the the Prosper dataset and most of the methodology used on this one was translated into it. The results showed that there was potential for knowledge distillation to be used for interpreting black-box models. As we can see from the results on table 4.2 the student can always come close to the teachers predictions by simply working on a regression problem. From here we investigate more advanced topics on the prosper loan dataset.

Model	Accuracy	Precision	Recall	F1 Score
Teacher	0.768	0.606061	0.555556	0.579710
Student	0.772	0.615385	0.555556	0.583942
Student On Groundtruth	0.772	0.636364	0.486111	0.551181

Table 4.2: German Credit Dataset credit default prediction results

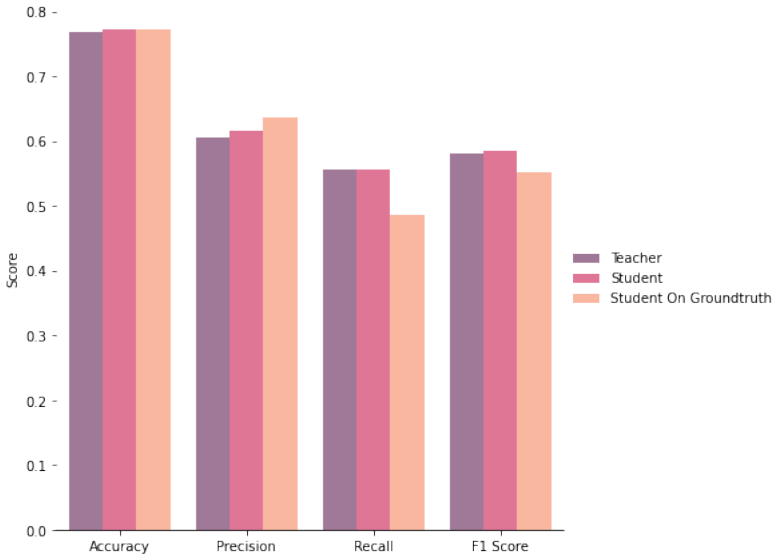


Figure 4.1: Graphical representation of the prediction scores obtained on the different models over the German Credit Dataset

4.5 Prosper Loan

All models were trained on the optimal configuration given the search space defined in the section above, we see that in theory we get the best results using SMOTE and SMOTE-Tomek which work very similarly. However we get a better score for minority classes using the medoids clustering technique at the cost of accuracy (see table 4.6).

Although the neural net shows great accuracy during training see table 4.3, it seems that it is not great at generalizing given the results on table 4.4. This is due to the very imbalanced nature of the dataset. As we can see from the support column on table 4.4.

Class	Precision	Recall	F1-score	Support
Default	0.98	0.90	0.94	23189
Good	0.74	0.97	0.84	22282
Problematic	0.95	0.74	0.83	22516

accuracy			0.87	67987
macro avg	0.89	0.87	0.87	67987
weighted avg	0.89	0.87	0.87	67987
weighted avg	0.63	0.68	0.62	14918

Table 4.3: Teacher training scores for problem 5.

Class	Precision	Recall	F1-score	Support
Default	0.19	0.07	0.11	990
Good	0.71	0.94	0.81	10134
Problematic	0.51	0.16	0.24	3794
accuracy			0.68	14918
macro avg	0.47	0.39	0.39	14918
weighted avg	0.63	0.68	0.62	14918

Table 4.4: Teacher testing scores

	Precision	Recall	f1-score	Accuracy	Support	Sampling
Weighted Avg	0.62	0.68	0.60	0.68	14918	SMOTETomek
	0.62	0.68	0.61	0.68	14918	SMOTE
	0.55	0.58	0.55	0.58	8607	Clustering

Table 4.5: Results for teacher training on 3 different imbalance treatment techniques.

Class	Precision	Recall	f1-score	Support	Sampling
Default	0.19	0.09	0.12	14918	SMOTETomek
	0.20	0.08	0.11	990	SMOTE
	0.61	0.18	0.28	2278	Clustering
Problematic	0.50	0.08	0.14	8607	SMOTETomek
	0.50	0.13	0.20	3794	SMOTE
	0.68	0.68	0.68	8934	Clustering

Table 4.6: Scores for the minority classes for the teacher model.

For the results obtained on the student, we see that on a 3 class problem the teacher helps guiding the student's predictions. On figures 4.2 and 4.3 we see how closely the student model, follows the teacher predictions, fidelity of its explanations is talked about on the following sections.

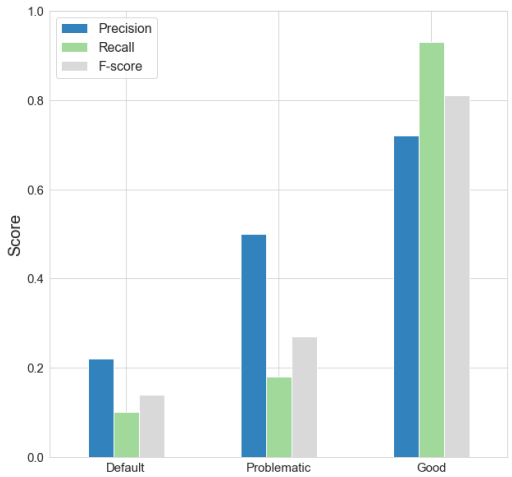


Figure 4.2: Teacher results on the Prosper Loan Dataset.

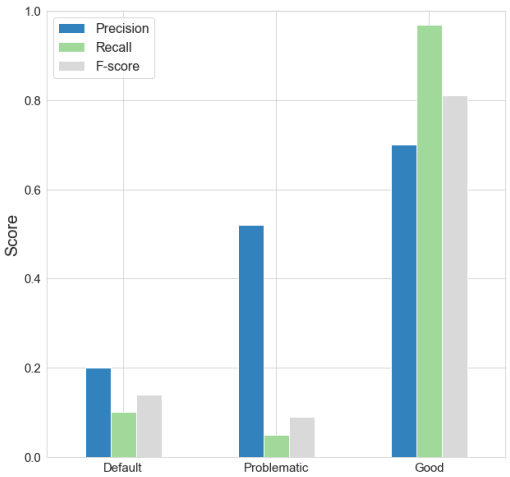


Figure 4.3: Student results on the Prosper Loan Dataset.

4.5.1 The stability of explanations on decision trees as students

When we talk about stability in interpretability and explanations we are talking about an internal evaluation of the model we are interpreting. While fidelity compares surrogate models, or explanators, to the model they are trying to explain in terms of how close is the explanation given by the explanator to the truth that the black-box model represents, stability refers to how two very similar instances might be explained by the same model. High stability means that slight variations in the features of an instance do not substantially change the explanation this does not refer to the prediction, when we look at a decision-tree, we can explain each decision in its totality since the model allows for traceability. As long as thresholds are maintained, the decision path for a decision-tree will still remain the same. The problem resides on what we define as close, or similar and this is hard to define. Looking at problem 5 using the prosper loan dataset 3.5. We use standardized Euclidean distance to select a group of similar observations at random, from those observations we picked two, at random, whose outcome was the same, in this case **Defaulted**. The standardized Euclidean distance between two n-vectors u and v is:

$$StdEuclideanDist = \sqrt{\sum \frac{(u_i - v_i)^2}{V[x_i]}} \tag{4.2}$$

where V is the variance vector, $V[i]$ is the variance computed over all the i^{th} components of the points. For these two instances we trace the path using the code in listing 4.1. From figures **??** and 4.5 we can see that both instances follow the same exact path. Although this path might not always be exactly the same, the results show that for the most part nodes at higher levels in the tree, which create the bigger splits are present in the path for similar predictions with most differences occurring at the deeper levels (last to second-to-last) of a decision tree with 14 levels of depth, as long as the predicted class by the model is the same.

```
Rules used to predict sample number 775(True Outcome:Good||Predicted Outcome(Defaulted)):

decision node 0 : Employed_1 was less than [[0.5]] with a value of [[0.]]
decision node 1 : PublicRecordsLast10Years was less than [[0.9981714]] with a value of [[0.]]
decision node 2 : TotalProsperLoans was less than [[2.36902097]] with a value of [[0.]]
decision node 3 : InvestmentFromFriendsCount was less than [[0.50000001]] with a value of [[0.]]
decision node 4 : InquiriesLast6Months was less than [[29.44195014]] with a value of [[1.]]
decision node 5 : OpenRevolvingAccounts was less than [[15.5000006]] with a value of [[1.]]
decision node 6 : DelinquenciesLast7Years was less than [[27.01995751]] with a value of [[0.]]
decision node 7 : PublicRecordsLast10Years was less than [[0.99651877]] with a value of [[0.]]
decision node 8 : CurrentDelinquencies was less than [[9.5414045]] with a value of [[0.]]
decision node 9 : Investors was less than [[531.49999762]] with a value of [[15.]]
decision node 10 : Investors was less than [[447.50000101]] with a value of [[15.]]
decision node 11 : Reno_1 was less than [[0.5]] with a value of [[0.]]
decision node 12 : Investors was less than [[29.50000011]] with a value of [[15.]]
decision node 13 : BorrowerRate was less than [[0.31490613]] with a value of [[0.1405]]
```

Figure 4.4: Decision path for the first instance.

```
Rules used to predict sample number 1694(True Outcome:Good||Predicted Outcome(Defaulted)):

decision node 0 : Employed_1 was less than [[0.5]] with a value of [[0.]]
decision node 1 : PublicRecordsLast10Years was less than [[0.9981714]] with a value of [[0.]]
decision node 2 : TotalProsperLoans was less than [[2.36902097]] with a value of [[0.]]
decision node 3 : InvestmentFromFriendsCount was less than [[0.50000001]] with a value of [[0.]]
decision node 4 : InquiriesLast6Months was less than [[29.44195014]] with a value of [[0.]]
decision node 5 : OpenRevolvingAccounts was less than [[15.5000006]] with a value of [[0.]]
decision node 6 : DelinquenciesLast7Years was less than [[27.01995751]] with a value of [[5.]]
decision node 7 : PublicRecordsLast10Years was less than [[0.99651877]] with a value of [[0.]]
decision node 8 : CurrentDelinquencies was less than [[9.5414045]] with a value of [[1.]]
decision node 9 : Investors was less than [[531.49999762]] with a value of [[25.]]
decision node 10 : Investors was less than [[447.50000101]] with a value of [[25.]]
decision node 11 : Reno_1 was less than [[0.5]] with a value of [[0.]]
decision node 12 : Investors was less than [[29.50000011]] with a value of [[25.]]
decision node 13 : BorrowerRate was less than [[0.31490613]] with a value of [[0.182]]
```

Figure 4.5: Decision path for the second instance.

Listing 4.1: Decision pathing code for individual predictions

```
def tree_path(tree,data,sample_id,scaler,preds):

    node_indicator = tree.decision_path(X_test_scaled)
    leaf_id = tree.apply(data)
    feature = tree.tree_.feature
    threshold = tree.tree_.threshold
    sample_id = sample_id

    # obtain ids of the nodes `sample_id` goes through
    node_index = node_indicator.indices[node_indicator.indptr[
        sample_id]:                node_indicator.indptr[sample_id +
         1]]
    cat_names = data.columns.values


    node_dict = {}

    print('Rules used to predict sample number {id}(True Outcome
        :{outcome}||Predicted Outcome({preds})):\n '.format(id=
        sample_id,outcome=dict(enumerate(categorical_code))[y_test
        [sample_id]],preds=dict(enumerate(categorical_code))[preds
        [sample_id]]))
    for node_id in node_index:
        #take values for feature rescaling
        sc = MinMaxScaler()
        sc.min_, sc.scale_ = scaler.min_[feature[node_id]],
            scaler.scale_[feature[node_id]]

        # continue to the next node if it is a leaf node

        if leaf_id[sample_id] == node_id:
            continue

        # check if value of the split feature for sample 0 is
            below threshold
```

```
        if (X_test_scaled[sample_id, feature[node_id]] <=
            threshold[node_id]):
            threshold_sign = "less than"
        else:
            threshold_sign = "more than"


        rescaled_feat = sc.inverse_transform(X_test_scaled[
            sample_id, feature[node_id]].reshape(1,-1))

        print("decision node {node} : {feature} was "
              "{inequality} {threshold} with a value of {value}".
                  format(
                    node=node_id,
                    sample=sample_id,
                    feature=cat_names[feature[node_id]],
                    value=rescaled_feat,
                    inequality=threshold_sign,
                    threshold=sc.inverse_transform(threshold[
                        node_id].reshape(1,-1))))

        if cat_names[feature[node_id]] in node_dict:
            node_dict[cat_names[feature[node_id]]].append(sc.
                inverse_transform(threshold[node_id].reshape(1,-1)
                ))
        else:
            node_dict[cat_names[feature[node_id]]] = [
                rescaled_feat]
    return node_dict
```

From the experiments conducted as well as the literature is easy to realize that one major problem of the decision tree is that the decision-tree is an unstable model. This happens because it captures feature interactions very well and by having variability in the data used for training we will have different threshold values at each nodes. This splits however create very good groupings that allow for stable interpretation of the decision paths. Another problem that the decision-tree structure has is the fact that sometimes a rule might repeat itself, this adds little value and increases the tree complexity. However, during explanation, any redundant nodes can be presented as the same node for a certain feature, which leads to easier interpretation. As a conclusion we deem decision trees stable models when looking from a perspective of interpretability.

4.5.2 Fidelity of students as surrogates for explanations

On this section we talk about the fidelity of students explanations. Decision-trees are white box models and as so we can see the whole process of the decision so if we are talking about the fidelity of an explanation given by a decision tree, we can safely understand that there is no variability for the explanations given by it, if a certain feature affects an instance in one

way it will also affect other instances the same way as the explanation it gives is a ruleset. Once the threshold is active it will always show in the rules that shaped the decision path. However we are interested in seeing if the same rules would apply to the model that the student is trying to mimic. To do this we can simply test the feature importance on the same instance in both models. It is expected that a decision-tree trained on a regression problem doesn't look for the same things in the data as a neural-net does. This can be seen in figures 4.7 and 4.6 where we took the same correctly predicted as default instance and compared how much each feature contributed for that prediction. By looking at the right side of the figures we see that the student got the class probabilities close to those of the teacher, however, by looking at the right side we see that both models are looking at very different things. This is acceptable if we are looking into using deep neural nets as helpers for interpretable models. However it means that we cannot interpret the decision process of the teacher by looking at the student since the student is mimicking the result and not the process.

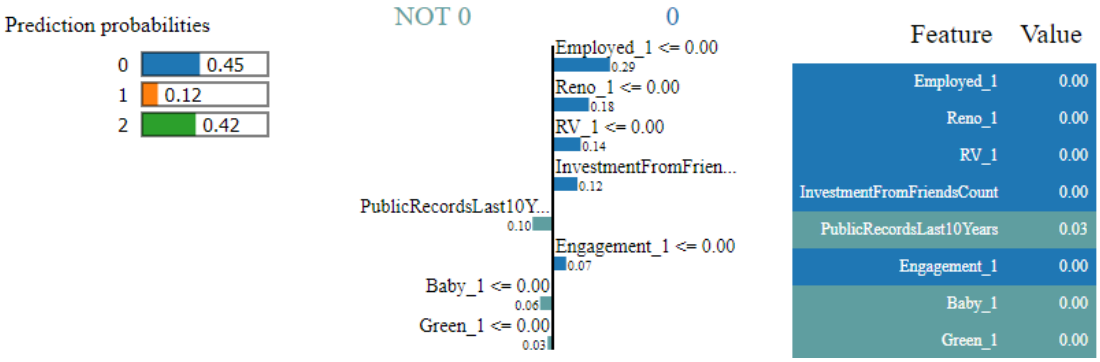


Figure 4.6: Student's feature importances.

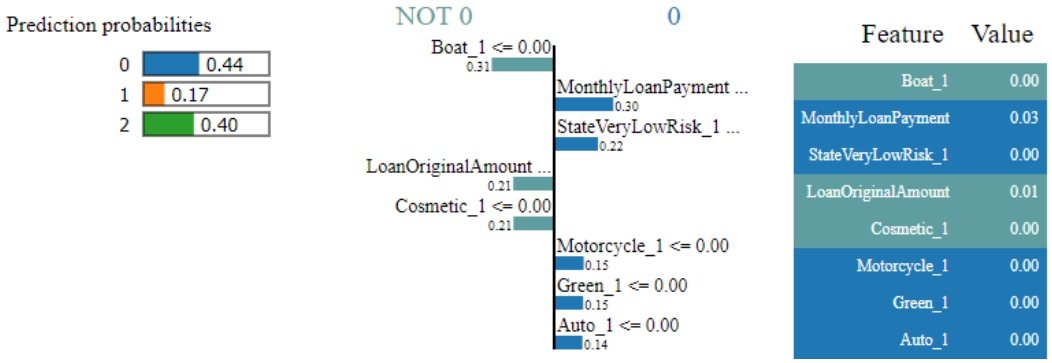


Figure 4.7: Teacher's feature importances.

4.6 Stock Movement Prediction

The usage of this dataset, was more of a stress testing case: would the students still perform when using a more complex architecture such as a LSTM as a teacher, given the increased amount of data given to the student. The results presented on all three stock market datasets are very similar, given that the problem for the student was still a binary regression problem we deem that the complexity of the teacher is not important for the student given that the student only works on the teachers outputs, and is simply trying to solve a regression problem. This means that we can use deep neural nets to solve very

complex problems and transform them into regression problems that more interpretable models can solve. One interesting behaviour was an higher recall than the teacher on both Tesla and Adobe datasets see tables 4.8 and 4.9. This might be due to the fact that we are training a different model on a regression problem which allows for certain features to have better correlation with the target, which might provide better classifications in some cases, but also to due to the fact that a different training will always allow variability in the results of teachers and students.

4.6.1 Alphabet Inc. GOOGL

Model	Accuracy	Precision	Recall	F1 Score
Teacher	0.643118	0.653484	0.760965	0.703141
Student	0.546894	0.560694	0.850877	0.675958
Student On Groundtruth	0.561510	0.576923	0.789474	0.666667

Table 4.7: Alphabet Inc.(GOOGL) stock movement prediction Results

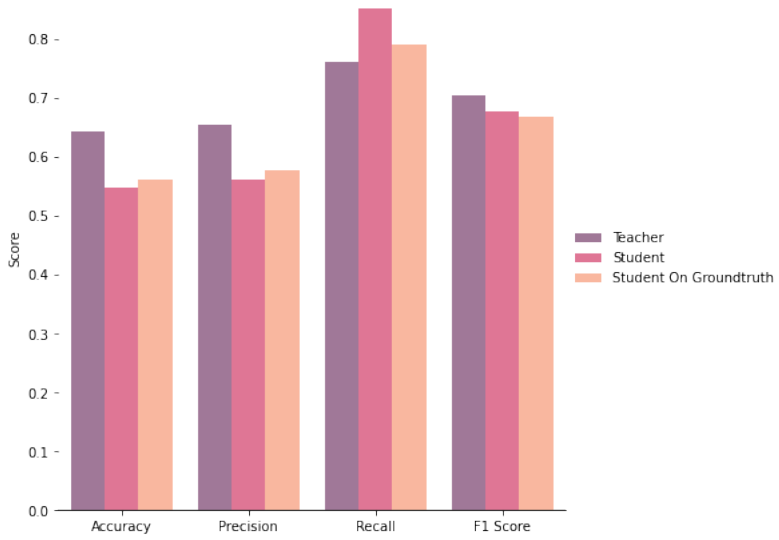


Figure 4.8: Graphical representation of the prediction scores obtained for the Alphabet Inc.(GOOGL) Stock movement prediction.

4.6.2 Tesla Inc.

Model	Accuracy	Precision	Recall	F1 Score
Teacher	0.689524	0.710247	0.712766	0.711504
Student	0.550476	0.561828	0.741135	0.639144
Student On Groundtruth	0.542857	0.561828	0.659574	0.607843

Table 4.8: Tesla Inc. stock movement prediction results.

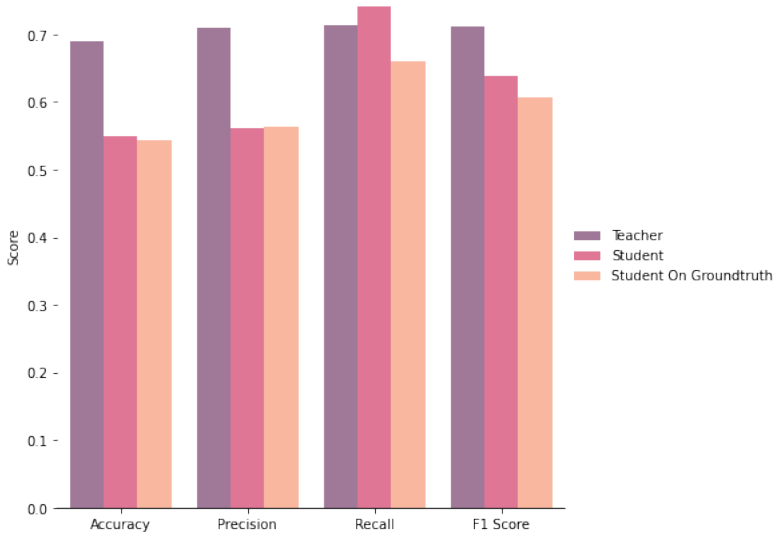


Figure 4.9: Graphical representation of the prediction scores obtained for the Tesla Inc.(TSLA) Stock Movement Prediction.

4.6.3 Adobe Inc.

Model	Accuracy	Precision	Recall	F1 Score
Teacher	0.652023	0.665964	0.689204	0.677385
Student	0.536416	0.550750	0.680480	0.608780
Student On Groundtruth	0.535838	0.547421	0.717557	0.621048

Table 4.9: Adobe Inc.(ADBE) stock movement prediction results

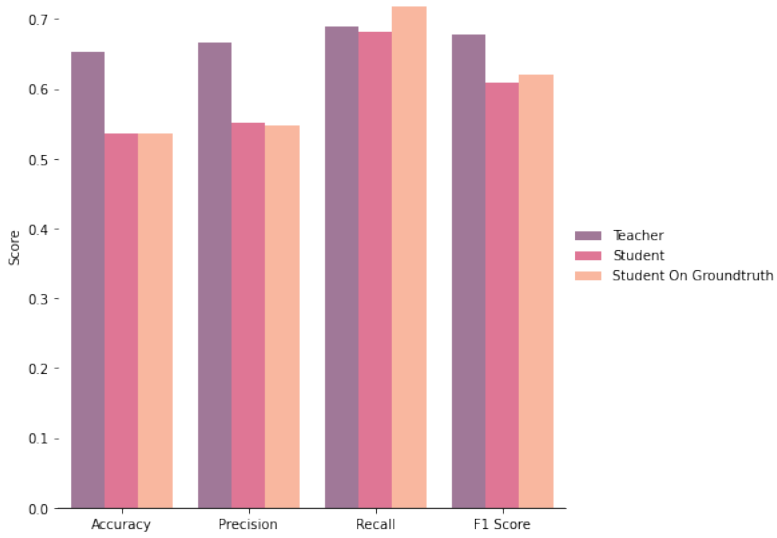


Figure 4.10: Graphical representation of the prediction scores obtained for the Adobe Inc.(ADBE) Stock Movement Prediction

4.7 Conclusion

Results show that in certain contexts black-box models can help less complex but more interpretable models achieve better classification performances. The process of knowledge distillation can be seen as "putting the problem in other words", in this case, by allowing the black-box models to solve a problem and turning into a regression problem where the categorical classes are translated into numerical ones that can be more easily separated or related to each other by regressors we allow the student models to solve easier versions of the same problem.

We study the behaviour of decision-trees in the scope of interpretability and conclude that these models give stable explanations, this means that small differences between variables are not able to drastically change the explanation itself.

Finally we understand that we cannot directly interpret black-box models by using knowledge distillation, this is for the simple reason that student models are being separately trained on a different representation of the same problem, which allows for differences during trained given the necessity for different loss functions, as well as the different architecture of the model itself.

This page is intentionally left blank.

Chapter 5

Conclusions and Future Work

In this thesis, we tackled current problems on interpretability of the decision-making systems in the financial sector, proposing to use knowledge transfer to get inside information on a black-model decision process by conveying learned behaviour from the latter onto more interpretable models.

By using five different datasets, two in the context of a real world problem, namely credit scoring and the other three in the context of stock market price forecasting we tested the methodology on a more complex DNN architecture. A pipeline consisting of a deep neural net, and two decision trees, one trained on logits (student) and the other trained on the real data was created for each of the above problems and their evaluation assessment performed.

Results of the predictions obtained allow us to conclude that there is in fact space for knowledge distillation in interpretability. Not for direct interpretation of such black-box models per se but as a means to achieve interpretable models. We conclude that we cannot interpret a decision process of a black-box model by mimicking its predictions with a decision-tree, given that we are only using the predictions of the models we are trying to interpret to guide the predictions of the interpretable model. This means however that we can create better interpretable models by supporting them with complex black-box models like DNNs.

Decision-trees have some disadvantages, for example, linear relationships between an input feature and the outcome have to be approximated by splits, creating a step function which is not efficient. Nevertheless, this does not necessarily affect their interpretability, even if the same feature is segmented throughout the decision path explanations we can reduce it to the range at which a feature had the most impact for , i.e, features at higher levels provide the more substantial splits.

Additionally, decision-trees are also very unstable, which means that subtle changes in data may create a completely different tree. This happens because each node will depend on the previous one, if a different feature is selected for the first node it will create a chain reaction creating an entirely different tree. However, their instability is only related to the training process, by looking at how similar individuals are classified we deem decision-trees stable when it comes to explanation, not only they give similar explanations to similar individuals but also give good explanations that can explain the abnormal by looking at the nodes that can differ, which are mostly located at the end of the decision path.

During the conception of this work, we had some problems particularly with the frameworks used. The use of ordinary classification problems created a new challenge as XGBoost

doesn't have support for multi-target training. Although there is a work around by using wrappers to turn multi-classification problems or multi-regression problems the classifier loses the relation between classes, which is the core of the method. This led to a decrease in the performance for the Prosper Loan Marketplace Credit classification problem, which was a multi-class problem that required the usage of less powerful decision-tree regressors.

Future work can be sought considering that knowledge distillation was introduced firstly as a form of model compression, it mostly applies on neural nets, as so, it is easier to create an infrastructure for neural nets only and obtain better results as we have better tools to do so.

Additionally, the distillation onto different class models is still a novelty that hasn't been fully studied into, although research is starting to delve into where soft-decision-trees which embed a form of decision-tree that is more like a neural net instead of relying on the more classical decision-tree approach. These pointed open problems pave the way for the conception from scratch (and implementation) of models that could more easily accommodate the process of knowledge transfer, that allow for multi-output regression as well as the use of different loss-functions that could help classification. Furthermore, these models could also possibly consider the values represented at the middle layers of the neural nets that would allow for interpretation of the deeper layers of such models. It would be interesting to explore such models in the future.

References

[1] Saumya Jetley, Nicholas A. Lord, Namhoon Lee, and Philip H. S. Torr. Learn to pay attention, 2018.

[2] Lukas Ryll, Mary Barton, Bryan Zhang, Jesse McWaters, Emmanuel Schizas, Rui Hao, Keith Bear, Massimo Preziuso, Elizabeth Seger, Robert Wardrop, P. Rau, Pradeep Debata, Philip Rowan, Nicola Adams, Mia Gray, and Nikos Yerolemou. Transforming paradigms: A global ai in financial services survey. *SSRN Electronic Journal*, 02 2020. doi: 10.2139/ssrn.3532038.

[3] Parliament and council of the european union (2016). general data protection regulation.

[4] D.M. West. *The Future of Work: Robots, AI, and Automation*. Brookings Institution Press, 2018. ISBN 9780815732938. URL `https://books.google.pt/books?id=W_zHtAEACAAJ`.

[5] Hiva Alahyari and Niklas Lavesson. User-oriented assessment of classification model understandability. volume 227, 05 2011. doi: 10.3233/978-1-60750-754-3-11.

[6] Girish H. Subramanian, John Nosek, Sankaran P. Raghunathan, and Santosh S. Kanitkar. A comparison of the decision table and tree. *Commun. ACM*, 35(1): 89–94, January 1992. ISSN 0001-0782. doi: 10.1145/129617.129621. URL `https://doi.org/10.1145/129617.129621`.

[7] Alex A. Freitas. Comprehensible classification models: A position paper. *SIGKDD Explor. Newsl.*, 15(1):1–10, March 2014. ISSN 1931-0145. doi: 10.1145/2594473.2594475. URL `https://doi.org/10.1145/2594473.2594475`.

[8] Alun D. Preece, Dan Harborne, Dave Braines, Richard Tomsett, and Supriyo Chakraborty. Stakeholders in explainable AI. *CoRR*, abs/1810.00184, 2018. URL `http://arxiv.org/abs/1810.00184`.

[9] Ziad Obermeyer, Brian Powers, Christine Vogeli, and Sendhil Mullainathan. Dissecting racial bias in an algorithm used to manage the health of populations. *Science*, 366 (6464):447–453, 2019. doi: 10.1126/science.aax2342.

[10] Jeffrey Dastin. Amazon scraps secret AI recruiting tool that showed bias against women, 10 2018. URL `https://www.reuters.com/article/us-amazon-com-jobs-automation-insight-idUSKCN1MK08G`.

[11] Bryce Goodman and Seth Flaxman. European Union Regulations on Algorithmic Decision-Making and a "Right to Explanation". *AI Magazine*, 38(3):50–57, 2017. doi: 10.1609/aimag.v38i3.2741.

[12] Gintare Karolina Dziugaite, Shai Ben-David, and Daniel M. Roy. Enforcing interpretability and its statistical impacts: Trade-offs between accuracy and interpretability. *CoRR*, abs/2010.13764, 2020. URL https://arxiv.org/abs/2010.13764.

[13] Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. A Survey of Methods for Explaining Black Box Models. *ACM Computing Surveys*, 51(5):1–42, 2019. doi: 10.1145/3236009.

[14] M. Lent, W. Fisher, and M. Mancuso. An explainable artificial intelligence system for small-unit tactical behavior. In *AAAI*, 2004.

[15] Finale Doshi-Velez and Been Kim. Towards a rigorous science of interpretable machine learning, 2017.

[16] Matt Turek. Explainable artificial intelligence (xai). URL https://www.darpa.mil/program/explainable-artificial-intelligence.

[17] Principles for accountable algorithms and a social impact statement for algorithms. URL https://www.fatml.org/resources/principles-for-accountable-algorithms.

[18] Peter Lipton. Contrastive explanation. *Royal Institute of Philosophy Supplement*, 27: 247–266, 1990. doi: 10.1017/S1358246100005130.

[19] A. Tversky and D. Kahneman. The framing of decisions and the psychology of choice. *Science*, 211 4481:453–8, 1981.

[20] Equifax launches neurodecision® technology. URL https://investor.equifax.com/news-and-events/press-releases/2018/03-26-2018-143044126.

[21] 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, page 1721–1730, New York, NY, USA, 2015. Association for Computing Machinery. ISBN 9781450336642. doi: 10.1145/2783258.2788613. URL https://doi.org/10.1145/2783258.2788613.

[22] Benjamin Letham, Cynthia Rudin, Tyler H. McCormick, and David Madigan. Interpretable classifiers using rules and bayesian analysis: Building a better stroke prediction model. *The Annals of Applied Statistics*, 9(3):1350–1371, Sep 2015. ISSN 1932-6157. doi: 10.1214/15-aoas848. URL http://dx.doi.org/10.1214/15-AOAS848.

[23] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '16, page 1135–1144, New York, NY, USA, 2016. Association for Computing Machinery. ISBN 9781450342322. doi: 10.1145/2939672.2939778. URL https://doi.org/10.1145/2939672.2939778.

[24] Mark W. Craven and Jude W. Shavlik. Extracting tree-structured representations of trained networks. In *Proceedings of the 8th International Conference on Neural Information Processing Systems*, NIPS'95, page 24–30, Cambridge, MA, USA, 1995. MIT Press.

[25] 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*,

HILDA'17, New York, NY, USA, 2017. Association for Computing Machinery. ISBN 9781450350297. doi: 10.1145/3077257.3077271. URL https://doi.org/10.1145/3077257.3077271.

[26] Ulf Johansson and Lars Niklasson. Evolving decision trees using oracle guides. *2009 IEEE Symposium on Computational Intelligence and Data Mining*, 2009. doi: 10.1109/cidm.2009.4938655.

[27] Mark W. Craven and Jude W. Shavlik. Using Sampling and Queries to Extract Rules from Trained Neural Networks. *Machine Learning Proceedings 1994*, pages 37–45, 1994. doi: 10.1016/b978-1-55860-335-6.50013-1.

[28] Ulf Johansson, Lars Niklasson, and Rikard König. Accuracy vs. comprehensibility in data mining models. 2004.

[29] Yin Lou, Rich Caruana, and Johannes Gehrke. Intelligible models for classification and regression. *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining - KDD '12*, 2012. doi: 10.1145/2339530.2339556.

[30] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, Ruslan Salakhutdinov, Richard S. Zemel, and Yoshua Bengio. Show, attend and tell: Neural image caption generation with visual attention. *CoRR*, abs/1502.03044, 2015. URL http://arxiv.org/abs/1502.03044.

[31] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization, 2015.

[32] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc., 2017. URL https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf.

[33] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network, 2015.

[34] Cristian Bucila, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. volume 2006, pages 535–541, 08 2006. doi: 10.1145/1150402.1150464.

[35] Kush R. Varshney. Engineering safety in machine learning, 2016.

[36] Dheeru Dua and Casey Graff. UCI machine learning repository, 2017. URL http://archive.ics.uci.edu/ml.

[37] Credit card race, age, gender statistics, Dec 2019. URL https://www.creditcards.com/credit-card-news/race-age-gender-statistics/.

[38] Alphabet, inc. (goog) stock price, news, quote & history, Jan 2021. URL https://finance.yahoo.com/quote/GOOG/.

[39] Adobe inc. (adbe) stock price, news, quote & history, Jan 2021. URL https://finance.yahoo.com/quote/ADBE.

[40] Tesla, inc. (tsla) stock price, news, quote & history, Jan 2021. URL https://finance.yahoo.com/quote/TSLA/.

[41] Yahoo finance - stock market live, quotes, business & finance news. URL https://finance.yahoo.com/.

[42] Hiva Alahyari and Niklas Lavesson. User-oriented assessment of classification model understandability. volume 227, 05 2011. doi: 10.3233/978-1-60750-754-3-11.

Appendices

Appendix A

CIARP25 Paper

The camera ready version of the submitted and accepted article for the 25^{th} Iberoamerican Congress on Pattern Recognition begins on the next page.

This page is intentionally left blank.

Interpreting Decision Patterns in Financial Applications

Tiago Faria, Catarina Silva, and Bernardete Ribeiro

University of Coimbra, Centre for Informatics and Systems of the University of Coimbra, Department of Informatics Engineering, Portugal
{tiagofaria,catarina,bribeiro}@dei.uc.pt

Abstract. Decisions in financial applications that directly impact citizens are often based on black-box intelligent methods. Given the growing interest in making these decisions more transparent, and the emergent legislation on interpretability and privacy, new solutions to give some insight on such black-boxes, presenting explanations on the decision patterns are being sought. In this paper we propose a method that transfers knowledge from black-box models to more interpretable models to understand the decision patterns in financial applications. Results on credit risk and stock market data show that it is possible to use white-box methods that work on black-box results to show the potential interpretation of the decision patterns.

Keywords: pattern recognition · distillation · interpretability · decision trees

1 Introduction

The rapid digitalization of our world has led us to great advances in services and activities we are involved in. Artificial Intelligence (AI) is now a big part of our lives even though not all of us are aware of it. From simple things like selecting the content we see online to partaking in critical decision making in our lives, AI has a strong presence across most activity sectors.

The increasing awareness that these systems do in fact exist and the notion that there is not always a human supervising them has surfaced the need for explanations.

From deciding if you get a loan to what is about to be shown on your Facebook feed AI is here, and it's not leaving. One of the sectors particularly impacted is the financial sector. As of the beginning of 2020 it was estimated that in the next two years there will be a mass adoption of AI in the financial sector with an impressive 77% expecting that AI will become essential to their business within the next two years [10], as can be gleaned from Figure 1. Intelligent systems are being applied in the financial sector in areas like:

- **Customer service**, e.g. operational cost savings from using chatbots in banking will reach $7.3 billion globally by 2023 [1].

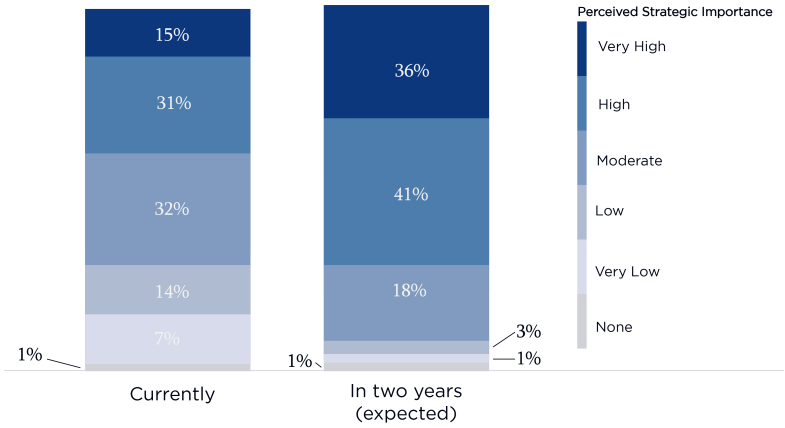


Fig. 1. Perceived strategic importance of AI services in the finance sector for 2022 [10].

- **Banking operations**, e.g., credit scoring, the use of AI technology enables more accurate scoring and allows for improved access to credit by reducing the risks and the number of false positives and false negatives.
- **Security purposes**, e.g., AI is providing great assistance in the detection of fraud and other suspicious activities that are linked to financial crime generally.

The financial industry is highly regulated and in the case of loan issuers, laws around the world, e.g. the European Union General Data Protection Regulation (EU GDPR), start to determine that in a not far away future, financial institutions must effectively show that the decisions they take are fair. The systems implemented in the financial sector are usually black-box models, highly capable of achieving their goal with high performance. The problem with black-box models is, although they are usually very capable, their decision processes are not clear and also prone to bias. Thus, one significant challenge of using AI-based systems that, for instance predict credit scores, is that there is no underlying interpretability infrastructure that can provide *reason code* to borrowers, e.g., when a credit is denied.

In this work we propose a method that transfers knowledge from deep models to decision-tree models to understand the decision patterns in financial applications. Results obtained in two distinct financial applications: credit risk and stock market show that it is possible to use white-box methods that work on black-box results to show the potential interpretation of the decision patterns.

The rest of the paper is organized as follows. Section 2 introduces relevant background on interpretable AI in finance and describes previous works in this research area. Section 3 details the proposed approach and Section 4 presents the experimental setup. Section 5 discusses the results and finally Section 6 highlights the conclusions and proposes lines of future research.

2 Background - Interpretable AI in Finance

Machine learning critical decision-making is a relatively recent topic. As humans get assisted or even replaced by intelligent models, existing legislation becomes obsolete and data regulation is often ineffective. Hence, new regulation like the European Union General Data Protection Regulation (EU GDPR), appeared which includes article 22 (see Figure 2) on automated decision making establishing the need for interpretability in the sector.

Although still in debate, it has been said that the GDPR has introduced the *right to explanation* based on the paragraph 1 of article 22 ” “shall implement suitable measures to safeguard. . . at least the right to obtain human intervention on the part of the controller, to express his or her point of view and to contest the decision” otherwise a person has “the right not to be subject to a decision based solely on automated processing”.

As safeguard for the companies implementing these models, as well as for the subjects that are targeted, interpretability starts to become essential in the transition to fully digital automated services.

Article 22. Automated individual decision making, including profiling

1. The data subject shall have the right not to be subject to a decision based solely on automated processing, including profiling, which produces legal effects concerning him or her or similarly significantly affects him or her.
2. Paragraph 1 shall not apply if the decision:
   (a) is necessary for entering into, or performance of, a contract between the data subject and a data controller;
   (b) is authorised by Union or Member State law to which the controller is subject and which also lays down suitable measures to safeguard the data subject's rights and freedoms and legitimate interests; or
   (c) is based on the data subject's explicit consent.
3. In the cases referred to in points (a) and (c) of paragraph 2, the data controller shall implement suitable measures to safeguard the data subject's rights and freedoms and legitimate interests, at least the right to obtain human intervention on the part of the controller, to express his or her point of view and to contest the decision.
4. Decisions referred to in paragraph 2 shall not be based on special categories of personal data referred to in Article 9(1), unless point (a) or (g) of Article 9(2) apply and suitable measures to safeguard the data subject's rights and freedoms and legitimate interests are in place.

Fig. 2. Article 22 of the EU GDPR.

2.1 Interpretability approaches

A lot of work has been done in the field of interpretability in the recent years. When it comes to the classification of techniques we can typically classify them in three categories:

– **Scope** - if the aim is to achieve global explanations in order to get an understanding of the decision making process of a model as whole we are talking about global interpretability techniques. On the other hand, if we are trying to understand how a model came up with a certain outcome for a specific

observation we are talking about local explanations and therefore global interpretability. Works in this area include Baehrens et al.[3] method for explaining local decisions taken by arbitrary nonlinear classification algorithm, using the local gradients that characterize how a data point has to variate in order to change its predicted label. Another very famous work is Ribeiro's et al. [9] **LIME** for Local Interpretable Model-Agnostic Explanation. A model that can approximate a black-box model locally in the neighborhood of any prediction of interest.

- **Relation to the model**, methods dependent on the model they are being applied to, intrinsic methods are of these class like Caruana's et al. [5] the drawback of this practice is that it is limited in to a certain class of models, this is why there's a preference for model-agnostic methods like **knowledge distillation**.

- **Complexity**, the most basic way of having an interpretable model would be making it inherently and intrinsically interpretable, a common challenge, which often hinders the usability of such methods. This is the tradeoff between interpretability and accuracy, more interpretable models tend to be less accurate and vice-versa [11].

Some of these works have also given birth to toolboxes for interpretability like ELI5[2] which aims to give local explanations and has a strong connection to LIME[9] which has also been turned into a toolbox or **Shap** which makes use of shapley values from game theory to see how features are impacting a model outcome by giving them respective weights across their whole range of values.

2.2 Interpretability models

Surrogate Models

Surrogate models can be classified in two types, global surrogate models and local surrogate models.

A global surrogate model is a model that is trained to mimic a black-box model giving us a global overview of what the black-box model is trying to achieve. This model is usually interpretable and can be used to draw conclusions about the way the mimicked model is trying to make its predictions.

Local surrogate models on the other hand are interpretable models that are used to explain individual predictions of black-box machine learning models. **LIME** [9] makes use of this type of surrogate models. Local surrogate models forget all the data and focus on a specific observation and how small perturbations on its features affect the outcome on the black-box model.

Knowledge extraction methods

As explained before one way of interpreting black-models would be making then interpretable in the first place, but that's not feasible as most models already

in-place, usually deep nets, are black-box. That would mean we would have to replace those models with completely new interpretable ones, structured to solve very specific problems that might not be as accurate as the previous ones. This would make these models not only expensive to run but also limited to the problems they are solving.

Knowledge extraction techniques try to extract explanations about the internal representation of complex models like deep neural nets. One of these methods is **model distillation**.

Model Distillation

Distillation is the process of transferring **dark knowledge** [8] from a deep neural net (usually denominated "the teacher") to less complex models ("the students"), these can be smaller deep nets or an interpretable model like a decision tree. Dark knowledge, also referred as hidden knowledge or latent knowledge in some literature, can be understood as information that is not seen with the "naked eye". In machine learning it refers to all the information contained in the hidden layers of a neural network model: the weights and ways each neuron connects to each other, inputs and inputs of each one or the way they jointly activate for a certain observation.
In the case of model distillation one is particularly interested in the last layer of a model, this can be seen as the layer where a decision has matured and is ready to be output.
Let's say we have a model m for classification of 3 classes which has a **softmax** layer l_s as the last layer, and, for a given observation o we know that the model outputs $m(o) =$ "$class1$" . If we are interested in dark knowledge we need to look deeper into the model. We'll find that the result was given by $argmax(output(l_s))$ and $output(l_s) = [0.6, 0.3, 0.1]$.
We can understand why the model output was "class 1": it presented the highest probability, we can also understand that our model learned that for the specific observation it would be 3 times more likely to be classified in "class 2" than in "class 3". This type of information a.k.a as **dark knowledge** is the rationale on which model distillation operates. It can be particularly useful when classes are strongly related to each other and it has been proven that model distillation can produce smaller models that can be as accurate as more complex ones [4] through the usage of **dark knowledge**.
Model compression [4] is also referred many times in the literature as one of the first examples of model distillation originally proposed to reduce the computational cost of a model at run-time by reducing its complexity which was later explored for interpretability. Tan et al. [12] proposed that model distillation can be used to distill complex models into transparent models like generalized additive models and splines. Che et al. [6] introduced in their paper a knowledge-distillation approach called Interpretable Mimic Learning, to learn interpretable phenotype features for making robust prediction while mimicking the performance of deep learning models. A recent work by Xu et al. [13] presented **DarkSight**, a visualization method for interpreting the predictions of

a black-box classifier on a data set in a way inspired by the notion of **dark knowledge**. This method combines ideas from knowledge distillation, dimension reduction, and visualization of deep neural nets.
The premise in all of the methods mentioned above is to use the capabilities of deep neural nets and translate the processes they learned during training to another model. We are interested not only on the ability to make the same class of models more efficient [4], but also in the ability of possibly changing their class[6] to a more interpretable one.

3 Proposed approach

Although knowledge distillation is not a new topic, we believe that there's more to do with it when it comes to interpretability, the interaction between classes can be a good resource to explain how a model came up with a certain decision. Some work in distilling knowledge to interpretable models has been done by Che et al. [6], but the models used were GAM's and splines, which don't have a great visualization, decision-trees are very easy to visualize and better at capturing feature interaction.

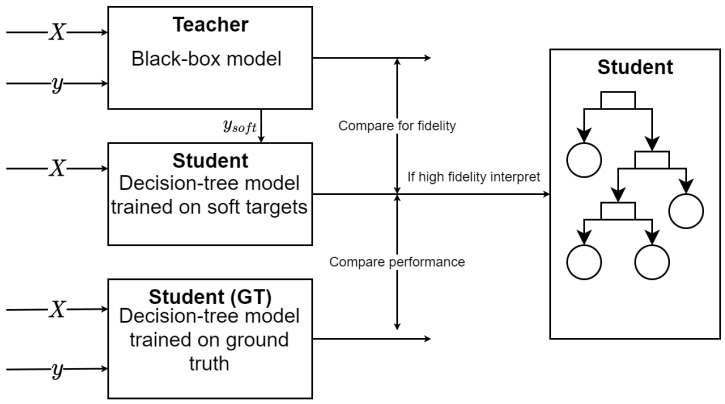


Fig. 3. Knowledge distillation process for the proposed approach

Methods to interpret trees can be more intuitively easy to come up with and explore. We believe that the tree structure is ideal for capturing interaction between features in data, visualization of decision-trees is also human-friendly making them better for explanation and interpretation. Figure 3 depicts the knowledge distillation process for the proposed approach.

We propose distilling knowledge from a deep neural net to a decision-tree by matching logits (scores before the last softmax layer), we do this by using these logits as targets to train a decision-tree for regression. This decision-tree should

in theory mimic the way the deep neural net makes its decisions. It should capture not only the good parts but the bad parts. This tree can then be evaluated for interpretability. For each defined problem: 1)credit risk management, 2) stock movement prediction we define a supervised training dataset $D_{train} = \{X, y\}$. For each dataset we train a deep neural net model, which we will call **"Teacher"**. We then extract the **logits**(values of the last layer before the softmax), y_{soft}, and use them as soft targets to train a decision-tree based model, which we call **"Student"** using XGBoost toolkit. The results of these models are to be compared in order to check how closely the mimic model is following the deep net model, we do this by checking their performance scores on the assumption that for the same observations, a similar evaluation metric on both models indicates similar decision-making. If this holds true we can interpret a decision-tree assuming its decision process is similar to the neural net. Finally, we train a third model on ground-truth labels which we will call **"Student (GT)"**. This model serves as benchmark to validate the usage of a deep neural net on a problem in the first place. If the **"Student (GT)"** model proves to be more precise, than a neural net we have a problem that its not complex enough to justify the usage of deep neural nets, and so the usage of the distillation on said model.

4 Experimental setup

4.1 Dataset description

The German credit dataset was taken from UCI ML Repository [7] and is comprised of 1000 instances and classifies people described by a set of attributes as good or bad credit risks.The data have been contributed as part of a dataset collection created by the Statlog EU project[1] with Prof. Dr. Hans-Joachim Hofmann listed as the data donor.
There are 20 explanatory variables with seven being numerical and 13 being categorical, with 30% observations accounting for the positive class (having bad credit). Both stock price historical data was acquired using the Yahoo!finance API, raw data consists of a time series with the columns open Price, closing Price, adjusted closing price, volume, highest and lowest price of the day.

4.2 Evaluation metrics

Tests were done using basic metrics for model evaluation that tell us how well a model is performing, which are then to be compared between the "teacher" and "student" models. In order to evaluate the decision task, a contingency matrix can be defined to represent the possible outcomes of the classification, as shown in Table 1. In cases where the weight of false positives and false negatives have different cost or in unbalanced datasets its better to use metrics that difference into account, as such for our problem we look to F1-score as being a more important metric than accuracy. If the F1-score of the student models is somewhat

[1] https://cordis.europa.eu/project/rcn/8791/factsheet/en

Table 1. Contingency table for binary classification

	Class Positive	Class Negative
Assigned Positive	a	b
	(True Positives)	(False Positives)
Assigned Negative	c	d
	(False Negatives)	(True Negatives)

similar or better than the teacher model we can presume that its reliable to use this models as surrogate. In specific cases where the weight of the false negatives is greater, such is the case of credit risk classification, we give more importance to the recall score, while trying to maintain a good F1-score.

4.3 Models

Two neural network architectures were used for the teacher model. A feed-forward neural network for the credit risk classification dataset with 2 layers of 256 and 128 hidden units respectively and a long short-term memory architecture for the stock movement prediction problem with 2 layers of 256 hidden units. The selected interpretable model a gradient boosted regression tree from XBGBoost's python library with the default parameters.

5 Experimental results and analysis

The models were evaluated based on their respective accuracy, precision, recall and F1-score, paying special attention to the F1-score and recall in the case of credit risk classification. For the German credit dataset, we have indication that the student is capturing the teacher's decisions very close by checking the that the scores are similar across all four metrics. We pay special attention to the recall metric, that is in fact exactly the same in the student and teacher models (see Table 2). This is particularly good in this context since the weight of having **false negatives** is far greater than the weight of **false positives**. It is more important to not misclassify people with bad credit as having good credit than the inverse. We believe that the higher complexity of a neural net helps in better classifying a minority class. In the german credit dataset we have a minority class that represents only 30% of the total observations. Not only that but if we look at the F1-score across all tree models, we get the best performance on the student model, which tells us that training a model with the support of a neural net's **dark knowledge** might be beneficial to get better performance on less complex models.

Table 2. German Credit Dataset credit default prediction results

Model	Accuracy	Precision	Recall	F1-Score
Teacher	76.80%	60.61%	55.56%	57.97%
Student	77.20%	61.54%	55.56%	58.40%
Student (GT)	77.20%	63.64%	48.61%	55.12%

In the context of stock prediction we find a similar behaviour. Stock price history datasets are in constant change, being updated everyday. At time of acquisition the class balance was at around between both sets ranged from 45% to 50% for the minority class, meaning these were relatively balanced. As the problem is more complex than the credit risk classification and has a much larger scale, the teacher outperformed both the student and student (GT) models. Since in stock prediction its as important to know when a stock price is going up as as well as when its going down we look to the F1-score for comparison (see tables 3, 4). We still see a slight improvement on the student when compared with the student GT, which enforces the belief that in general, less complex models can benefit from model distillation. We also see a tendency for high recall scores that should represent a better classification of minorities which requires further investigation.

Table 3. Alphabet Inc.(GOOGL) stock movement prediction Results

Model	Accuracy	Precision	Recall	F1-Score
Teacher	64.31%	65.35%	76.10%	70.31%
Student	54.69%	56.07%	85.09%	67.60%
Student (GT)	56.15%	57.69%	78.95%	66.67%

Table 4. Tesla Inc.(TSLA) stock movement prediction Results

Model	Accuracy	Precision	Recall	F1-Score
Teacher	68.95%	71.03%	71.27%	71.15%
Student	55.05%	56.18%	74%	63.92%
Student (GT)	54.29%	56.18%	65.96%	60.78%

6 Conclusions and future work

The results obtained are indicative that the method can be used to improve solutions in particular contexts, as is with the credit risk case. We believe that by optimizing models and the process of training can be optimized with hyper-parameter tuning. Another exploration that can be interesting is working on the

inner layers of neural net which have more complex interactions.
The focus of the future work will be to optimize and tune the transfer process to better adapt it to decision-trees aswell as define new metrics for fidelity, in order to better define the decision process we pretend to check how the decisions represented by the tree are different from the model by looking at specific cases and checking how differences in features change the outcome in both models.

References

1. Bank cost savings via chatbots to reach $7.3 billion by 2023, as automated customer experience evolves, `https://www.juniperresearch.com/press/press-releases/bank-cost-savings-via-chatbots-reach-7-3bn-2023`
2. Eli5, a library for debugging ml classifiers/regressors and explaining their decisions, `https://eli5.readthedocs.io/en/latest/overview.html`
3. Baehrens, D., Schroeter, T., Harmeling, S., Kawanabe, M., Hansen, K., Müller, K.R.: How to explain individual classification decisions. Journal of Machine Learning Research **11**(61), 1803–1831 (2010)
4. Buciluǎ, C., Caruana, R., Niculescu-Mizil, A.: Model compression. In: Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. p. 535–541. KDD '06, Association for Computing Machinery, New York, NY, USA (2006)
5. Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., Elhadad, N.: Intelligible models for HealthCare. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM (aug 2015)
6. Che, Z., Purushotham, S., Khemani, R., Liu, Y.: Distilling knowledge from deep networks with applications to healthcare domain (2015)
7. Dua, D., Graff, C.: UCI machine learning repository (2017), `http://archive.ics.uci.edu/ml`
8. Hinton, G., Vinyals, O., Dean, J.: Distilling the knowledge in a neural network. In: NIPS Deep Learning and Representation Learning Workshop (2015)
9. Ribeiro, M.T., Singh, S., Guestrin, C.: Why should i trust you? : Explaining the predictions of any classifier (2016)
10. Ryll, L., Barton, M., Zhang, B., McWaters, J., Schizas, E., Hao, R., Bear, K., Preziuso, M., Seger, E., Wardrop, R., Rau, P., Debata, P., Rowan, P., Adams, N., Gray, M., Yerolemou, N.: Transforming paradigms: A global ai in financial services survey. SSRN Electronic Journal (02 2020). https://doi.org/10.2139/ssrn.3532038
11. Sarkar, S., Weyde, T., Garcez, A., Slabaugh, G., Dragicevic, S., Percy, C.: Accuracy and interpretability trade-offs in machine learning applied to safer gambling. In: CoCo@NIPS (2016)
12. Tan, S., Caruana, R., Hooker, G., Lou, Y.: Distill-and-compare. Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society (Dec 2018)
13. Xu, K., Park, D.H., Yi, C., Sutton, C.: Interpreting Deep Classifier by Visual Distillation of Dark Knowledge. arXiv e-prints arXiv:1803.04042 (Mar 2018)

Appendix B

RECPAD21 Paper

The camera ready version of the submitted and accepted article for the 27^{th} Portuguese Conference on Pattern Recognition (RECPAD2021) begins on the next page.

This page is intentionally left blank.

Using Knowledge Distillation to Interpret Credit Score Modeling

Tiago Faria
tiagofaria@student.dei.uc.pt
Catarina Silva
catarina@dei.uc.pt
Bernardete Ribeiro
bribeiro@dei.uc.pt

Universidade de Coimbra
CISUC - Centro de Informática e Sistemas
FCTUC-DEI - Departamento de Engenharia Informática
Coimbra, Portugal

Abstract

In the last decade many accurate decision support systems have been constructed as black boxes. However, applicability in several critical applications, e.g. public policy, security/safety systems, health diagnosis and fraud detection, has been faced with some hurdles due to lack of model interpretability. In this work we present knowledge distillation as a stepping stone to achieve model interpretability by interpretable models mimic more complex ones such as deep neural nets. We show that there's a possibility for less complex but interpretable models to mimic deep neural nets, by giving transforming classification problems into a regression problem.

1 Introduction

The financial industry is highly regulated and in the case of loan issuers, laws around the world, e.g. the European Union General Data Protection Regulation (EU GDPR), start to determine that in a not far away future, financial institutions must effectively show that the decisions they take are fair. The systems implemented in the financial sector are usually black-box models, highly capable of achieving their goal with high performance. The problem with black-box models is, although they are usually very capable, their decision processes are not clear and also prone to bias. Thus, one significant challenge of using AI-based systems that, for instance predict credit scores, is that there is no underlying interpretability infrastructure that can provide reason code to borrowers, e.g.,when a credit is denied.In this work we propose a method that uses knowledge transfer from deep models to decision-tree models in an attempt to understand the decision patterns in financial applications.

2 Background

Machine learning critical decision-making is a relatively recent topic. As humans get assisted or even replaced by intelligent models, existing legislation becomes obsolete and data regulation is often ineffective. Hence, new regulations like the European Union General Data Protection Regulation (EU GDPR), which includes article 22 on automated decision making are establishing the need for interpretability in the sector. Although still in debate, the GDPRs article 22 clauses on automated individual decision-making have introduced the right to explanation [1] for all individuals to obtain "meaningful explanations of the logic involved" while being targets of automated decision-making algorithms. As safeguard for the companies implementing these models, as well as for the subjects that are targeted, interpretability starts to become essential in the transition to fully digital automated services. In fact, some companies are starting to learn the problems of black-box models in their services [2]. Knowledge Distillation was first introduced in 2015 [3] and is a generalization of **Model Compression** [4]. **Model Compression** consists on the transfer of learned knowledge of as lower, larger and better performing model onto a smaller, faster. Caruana et al. [4] achieves this by matching the logits of the smaller model to the logits of a cumbersome model. This means that the smaller model will approximate the behaviour of the more complex one by training on big ammounts of pseudo-data (logits) which in turn will get better results than the same model trained on real data given there's more information stored on logits than there is on hard labels. **Knowledge Distillation** is a variant of this approach proposed by Hinton et al. [3] which uses the last layer's soft probabilities instead of logits as targets for training a smaller deep neural net student model.

3 Proposed Approach

Although knowledge distillation is not a new topic, we believe that there's more to do with it when it comes to interpretability, the interaction between classes can be a good resource to explain how a model came up with a certain decision. Some work in distilling knowledge to interpretable models has been done by Che et al.[5], but the models used were GAM's and splines, which don't have a great visualization, decision-trees are very easy to visualize and better at capturing feature interaction.

We propose distilling knowledge from a deep neural net to a decision-tree by training a deep neural net model using a dataset **{X,y}** which is often called **Teacher** (this could also be a previously trained model), we then use the Teacher's softmax layer output **y'** as targets for a decision-tree regressor which we call the Student. While the teacher has learned classification, the student will simply try to match the teacher. In theory if we can achieve a perfect score in the student, we get a surrogate model that is easily interpretable.

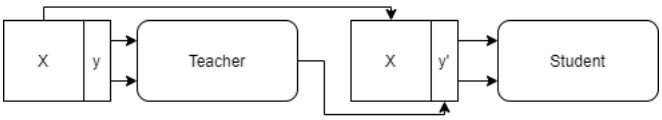


Figure 1: Knowledge distillation process

Methods to interpret trees can be more intuitively easy to come up with and explore. We believe that the tree structure is ideal for capturing interaction between features in data, visualization of decision-trees is also human-friendly making them better for explanation and interpretation. Figure 1 depicts the knowledge distillation process for the proposed approach.

We propose distilling knowledge from a deep neural net to a decision-tree by matching logits (scores before the last softmax layer), we do this by using these logits as targets to train a decision-tree for regression. This decision-tree should in theory mimic the way the deep neural net makes its decisions. It should capture not only the good parts but the bad parts. This tree can then be evaluated for interpretability.

4 Experimental Setup

4.1 Dataset

The data used on this project was kindly provided and given permission to work on by Jörg Osterrieder and Branka Misheva as part of **COST**. The dataset is comprised of 113937 instances with each consisting of a group of 80 descriptive attributes that characterize the outcome of an individuals loan given by Prosper. Prosper Marketplace, Inc. is an american company in the peer-to-peer lending industry. Data was cleared of all null values and unnecessary columns and consisting of 106290 instances and a total of 59 attribute columns at the end of the clearing process. LoanStatus represents the target for classification that initially consisted of 11 classes.

The instances classified as Current were dropped as they had no real value on the training, predicting of any of the models since we don't know what the final outcome was. The rest of the classes were grouped up with all Past Due becoming a new Problematic class; Charged off and Cancelled grouped up with Defaulted, and FinalPaymentInProgress coupled with Completed for the sake of keeping as much data as possible, as so we are left with a 3 class problem with the classes, Defaulted, Problematic and Completed. This leaves us with a dataset comprised of 49724 entries.

Table 1: Initial classes and their distribution

Class Name	Number of Instances
Current	56566
Completed	33530
Defaulted	3289
Past Due (aggregated 1-120+ days)	2067
FinalPaymentInProgress	205
Charged-off.	10632
Cancelled	1

4.2 Methodology

A deep neural net was trained in classifying the 3 classes, using a 70% of the total dataset for training, leaving 30% for testing, after hyper-parameter optimization, the best neural network was chosen to be the teacher. After training we pass the full training dataset **X** once again through the same neural network obtaining a list of probabilities vectors **y'** size $n \cdot c$ where n is the number of instances of the training dataset and c is the number of classes in the classification problem, in our case $c = 3$. After we obtain the new set **{X,y'}** we use it to train a decision tree regressor, which we call the student. We than compare the student and teacher for similarity in the predictive power by looking at the respective scores for the classes. A second decision-tree is trained in order to validate the differences between training a model using knowledge distillation and training a model on ground-truth labels.

4.3 Evaluation Metrics

Tests were done using basic metrics for model evaluation that tell us how well a model is performing, which are then to be compared between the "teacher" and "student" models. In cases where the weight of false positives and false negatives have different cost or in unbalanced datasets its better to use metrics that difference into account, as such for our problem we look to F1-score as being a more important metric than accuracy. If the values of precision and recall across all three classes classified by the student model is somewhat similar or better than the teacher model we can presume that its reliable to replace the teacher with this model.

5 Results and Analysis

After both models were trained we used the test portion of the dataset to assess the validity of the method, obtaining the following results.

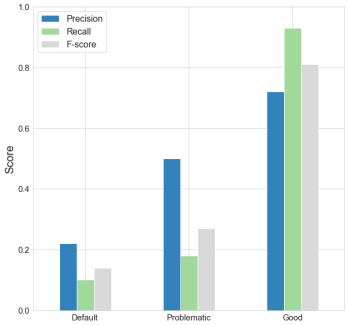


Figure 2: Teacher results

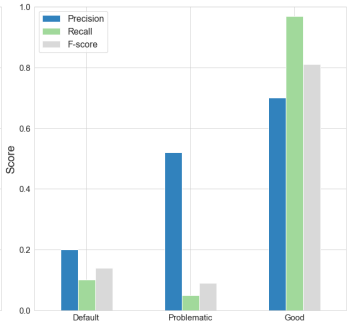


Figure 3: Student results

As we can see, on figures 2 and 3 we see extremely similar scores on both student and teacher, as it was said before, the student is meant to copy it, this means that it will also try to capture the worse parts as we can see from the problematic part. While if we look at the results for the ground-truth (see figure 4) model we see that these look quite different. This happens because we've given more information to the the model trought the labels by transforming a classification problem into a regression problem, giving the model more "in-between" values that it can guide himself with making it easier to achieve higher accuracies, this can be seen as a form of pre-processing. On the other hand, a model trained with hard-labels sees every instance of the same class as exactly the same, not taking into account the possible similarities or relationship with other classes, this makes the results differ based on the structure and parameters of the model itself.

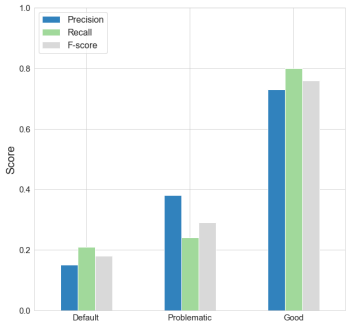


Figure 4: Model trained on ground-truth results

Table 2: Scores for the three models

Class	Model	Precision	Recall	F_1	Acc
Default	Teacher	0.22	0.10	0.14	0.68
	Student	0.20	0.10	0.14	0.68
	Ground-truth	0.15	0.21	0.18	0.62
Problematic	Teacher	0.50	0.18	0.27	0.68
	Student	0.52	0.05	0.09	0.68
	Ground-truth	0.38	0.24	0.29	0.62
Good	Teacher	0.72	0.93	0.81	0.68
	Student	0.70	0.97	0.81	0.68
	Ground-truth	0.73	0.80	0.76	0.62

6 Conclusions and Future Work

In this work we show the potential of using knowledge distillation to improve a less complex model's accuracy, in our experiment we achieve extremely similar scores on both teacher and student. This leads us to think that if we improve the teacher's prediction accuracy for the minority class, will have an interpretable model, in our case a decision-tree that performs better than the same model trained on ground-truth labels.

To improve performance on the teacher as it is a very complex problem with a very particular emphasis on the fact that it is very imbalanced, future work would be on improving the pipeline to get better results on the teacher, for example by creating an ensemble of neural models, or by distilling a teacher onto another neural net which has shown to be effective.

Acknowledgements

This work acknowledges research support by COST Action "Fintech and Artificial Intelligence in Finance - Towards a transparent financial industry" (FinAI) CA19130 (`https//fin-ai.eu/`).

References

[1] Bryce Goodman and Seth Flaxman. European Union Regulations on Algorithmic Decision-Making and a "Right to Explanation". *AI Magazine*, 38(3):50–57, 2017. doi: 10.1609/aimag.v38i3.2741.

[2] Taylor Telford. Apple Card algorithm sparks gender bias allegations against Goldman Sachs, 11 2019. URL `https://www.washingtonpost.com/business/2019/11/11/apple-card-algorithm-sparks-gender-bias-\-allegations-against-goldman-sachs/`.

[3] Geoffrey Hinton, Oriol Vinyals, and Jeffrey Dean. Distilling the knowledge in a neural network. In *NIPS Deep Learning and Representation Learning Workshop*, 2015.

[4] Cristian Bucila, Rich Caruana, and Alexandru Niculescu-Mizil. Model compression. volume 2006, pages 535–541, 08 2006. doi: 10.1145/1150402.1150464.

[5] Zhengping Che, Sanjay Purushotham, Robinder Khemani, and Yan Liu. Distilling knowledge from deep networks with applications to healthcare domain, 2015.

Appendix C

This page is intentionally left blank.

Appendix C

Work Planning

This chapter provides an overview of the scheduled plan and high level tasks for the successful completion of the proposed work. Figure 5.1 displays a gant chart of the tasks. Planning is divided in two to accommodate the semester split and current intermediate report.

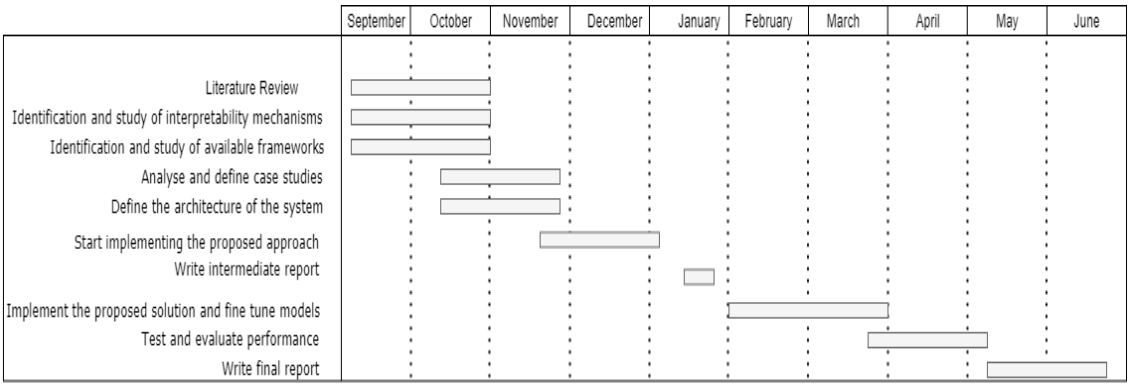


Figure C.1: Gantt chart of the scheduled tasks

C.1 Work Plan for the 1st Semester

- Literature review
- Identification and study of interpretability mechanisms
- Identification and study of available frameworks
- Analyse and define case studies
- Define the architecture of the system
- Start implementing the proposed approach
- Write intermediate report

C.2 Work Plan for the 2nd Semester

- Implement the proposed solution and fine tune models

- Test and evaluate performance
- Write final report