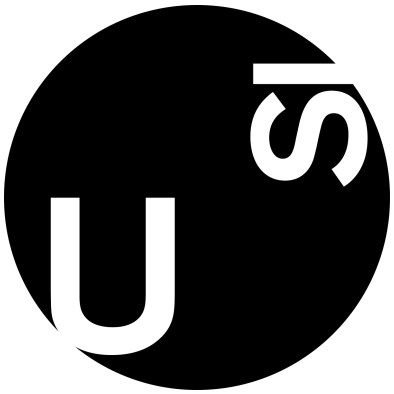

Tabular Deep Concept Reasoning

Master Thesis Proposal

Marian Horn

May 22, 2025

Advisor: Prof. Marc Langheinrich

Co-Advisor: Giovanni De Felice

Faculty of Informatics, USI Lugano

The advisor(s) of this Master's Thesis Proposal confirm that the literature review within this proposal, along with the scope and content of the proposed thesis topic, are sufficient for a master's-level thesis defense in the Faculty of Informatics at Università della Svizzera italiana..

_____

Signature of Advisor(s) & Date

Abstract

The demand for interpretable machine learning models is particularly high in domains where automated decisions have significant real-world consequences, such as health care or finance. One prominent example is credit risk scoring, the process of determining an individual's eligibility for a loan, which requires not only high predictive accuracy but also transparent, accountable reasoning to ensure fairness and compliance with regulations such as the European Union's General Data Protection Regulation. This represents a promising application area for Deep Concept Reasonging (DCR), a novel neural-symbolic approach that enhances the interpretability of machine learning decisions by applying logical rules to concept abstractions derived from the data. DCR is capable of producing both local and global explanations for its predictions.

This thesis proposal outlines a plan for an in-depth investigation of DCR applied to tabular data, including a comparison with other machine learning models and a real-world application to credit risk scoring.

This work begins with a literature review, summarizing the DCR methodology, discussing the demonstrated superiority of tree-based models over deep learning models in the case of tabular data and exploring theoretical foundations of transparency, interpretability and explainability in decision-making systems.

Subsequently, the thesis proposal outlines a sequence of milestones that guide the planned work. It starts with the selection of suitable benchmarking datasets, the identification of relevant machine learning models, and the comparative evaluation of DCR against these models. The primary evaluation criteria are classification accuracy and model interpretability. Based on the results, conclusions will be drawn about the individual strengths and limitations of DCR and the other methods, leading to practical recommendations for when to use which model.

The final part of the thesis will apply DCR to the task of credit risk scoring, where the goal is to produce interpretable classifications of customer creditworthiness based on personal and financial data. This real-world case study is intended to highlight DCR's strengths in delivering both accurate and interpretable outcomes, demonstrating its potential for high-stakes decision-making in regulated environments.

Contents

Chapter 1

Introduction

This thesis proposal outlines the plan for the master's thesis titled *Tabular Deep Concept Reasoning*.

The thesis investigates the application of neural-symbolic learning to tabular data, a format that underpins decision-making and classification in domains ranging from healthcare and finance to retail and logistics. Altough tabular data are among the most common types encountered in practical machine learning applications, achieving high predictive accuracy while maintaining interpretability remains a significant challenge. The thesis focuses on evaluating the Deep Concept Reasoner, a novel neural-symbolic model designed to adress this gap by generating probabilistic programs from abstracted concepts.

The research is structured into several key phases. First, a literature review will examine the theoretical foundation of Deep Concept Reasoning (DCR) and a selection of established models for tabular data, including tree-based methods, ensemble models, and attention-based deep learning architectures. Subsequently, DCR and the selected benchmark models will be implemented and tuned on standard tabular datasets, with particular emphasis on both predictive performance and interpretability. Comparative experiments will be conducted to quantify the trade-offs and strengths of each approach.

Finally, the thesis will apply DCR to credit risk scoring, demonstrating its potential in domains where model transparency is critical. Through this comprehensive evaluation, the project aims to contribute to the understanding of how explainable deep learning models can support both accurate and interpretable decision-making in highly regulated tabular data environments.

This document is organized into two main parts.

The literature review in chapter 2 summarizes three relevant pieces of literature, explaining the methodology of DCR, showing the advantages of tree-based machine learning models over deep learning techniques for tabular data, and discussing the concept of interpretability in machine learning in general.

Furthermore, the milestones in chapter 3 are intended to provide a structured working plan for the analysis and evaluation of DCR on tabular data. They range from the easiest to more challenging tasks, which require more time and effort to complete. All milestones are associated with specific chapters of the thesis, as described in the beginning of chapter 3.

Chapter 2

Literature Review

2.1 Interpretable Neural-Symbolic Concept Reasoning

In 2023, Barbiero et al. published their paper *Interpretable Neural-Symbolic Concept Reasoning*, introducing DCR as a novel interpretable deep learning method. The authors describe the internal workings of the model, evaluate its generalization and interpretability capabilities, and compare it to other interpretable machine learning models [3].

Despite recent advances, deep learning models often lack interpretability and systematic generalization. Even concept-based models, which use concept embedding vectors to improve interpretability, are not completely human-understandable, because these concept embeddings are abstract numerical vectors and do not have clear semantics, as for example, they do not assign a distinct meaning to each embedding dimension. To adress this issue, DCR does not use concept embeddings directly for classification. Instead, it leverages them during training to parameterize two neural networks (NNs) that generate logical rules. These rules are then applied to the concept truth values, which represent on the interval $[0, 1]$ how likely a given concept is present in a particular sample [3].

To construct these rules, the relevance and the role of each concept have to be determined. The relevance $r_{ji} \in \{0, 1\}$ indicates whether concept i is included in the rule for class j, while the role $l_{ji} \in \{\hat{c}_i, \neg\hat{c}_i\}$ specifies wether the truth value $\hat{c}_i$ of concept i contributes positively or negatively to assigning class j. Using this notation, the logic rule to predict the label $\hat{y}_j$ of class j can be expressed as shown in equation 2.1 [3].

$$\hat{y}_j \Leftrightarrow \bigwedge_{i\,:\,r_{ji}=1} l_{ji} \tag{2.1}$$

The concept roles l_{ji} are determined by a feed-forward NN φ_j that takes the concept embedding as input. Similarly, the relevance values are computed by

another feed-forward NN ψ_j, which incorporates a special activation function to promote sparsity and competition between concepts. This activation function includes a parameter τ, which controls the fraction of relevant concepts and thus allows for adjusting the complexity of the generated rules by increasing or decreasing τ [3].

Since role and relevance are determined per sample, the resulting rules are local. To generate global rules over the entire training dataset X, all sample-specific local rules for a given class are booleanized and aggregated via disjunction to obtain the class-specific global rule $\hat{y}_j^C$ [3].

DCR is evaluated on tabular, image and graph-structured data and compared with other machine learning (ML) models that offer a high degree of interpretability, such as logistic regression, decision trees, XGBoost and a locally-interpretable neural model called Relu Net. The evaluation is based on four different criteria. Generalization is measured by the Area Under the Receiver Operating Characteristic Curve (ROC AUC). Interpretability is assessed by comparing the learned rules to the underlying ground-truth rules in the datasets. Additionally, explanation sensitivity to small perturbations in the data is analyzed, as well as the simplicity of counterfactual explanations, measured by the number of concept changes required to flip a prediction [3].

The results confirm that DCR outperforms the compared models across all datasets. While the baseline models perform better when using concept embeddings instead of concept truth values, these additional experiments are less relevant, as models using concept embeddings are no longer interpretable. In contrast, DCR uses the concept embeddings solely to train NNs that generate semantically meaningful rules operating on the concept truth values, thereby maintaining interpretability [3].

Furthermore, it is demonstrated that DCR is capable of learning semantically meaningful rules that reflect the underlying ground-truth logic. This is verified, for example, using a synthetic dataset representing the binary XOR operation. On the mutagenicity dataset, which aims to predict mutagenic effects of molecular structures based on their functional groups, it is shown that DCR is also capable of discovering meaningful rules and relationships and making them comprehensible to humans. This suggests that DCR can be used not only for classification, but also for data exploration and the derivation of new insights into underlying phenomena [3].

In addition, the DCR model has been shown to be robust against small perturbations in the input data, which is essential for maintaining user trust in the model's decisions. Also, in cases of incorrect predictions, the reasons for the error can be identified by closely examining the set of concepts and the corresponding rules. In this way, the error can be traced back to its source — whether a flawed rule, an inappropriate selection of relevant concepts, or

an incorrect generation of the concepts from the original data. Furthermore, DCR facilitates the identification of counterfactual examples, as its prediction confidence decreases rapidly when the most relevant concepts are perturbed. This property enables the determination of minimal concept changes required to flip a prediction [3].

Rarely arising limitation of DCR include the occasional difficulty in interpreting global rules, the potential complexity of rules in some applications requiring a large number of concepts, and the need for concept embeddings or strong concept discovery methods to train the NNs [3].

In summary, the main advantage of DCR lies in its ability to handle tasks that demand both high interpretability and high accuracy, while other models focus on only one of these objectives [3]. A further analysis of this characteristic, particularly in the context of tabular data, will be the focus of the proposed master's thesis.

2.2 Why do tree-based models still outperform deep learning on typical tabular data?

To select suitable ML models for comparison in the proposed thesis, it is essential to analyze the current state-of-the-art in ML on tabular data. Grinsztajn et al. [6] provide a valuable foundation for this by comparing various tree-based models with deep learning (DL) approaches.

First, the authors compiled a collection of datasets suitable for benchmarking, as no standard benchmark specifically designed for tabular data currently exists. Several characteristics that a dataset has to fulfill are considered that may also serve as a useful guide for selecting datasets in the proposed thesis, particularly in milestone 3.2. The criteria include heterogeneous features that do not carry the same signal from different measuring devices, absence of high dimensionality, good documentation, and independently and identically distributed data, thus excluding stream-like or time series datasets. Additionally, artificially generated datasets, datasets with very few features or samples, datasets exhibiting clear determinism, and those that can be classified too easily by e.g. a single tree or logistic regression are excluded. Large training sets are reduced to 10,000 samples, missing data is handled appropriately, and class balancing is ensured. In a final preprocessing step, categorical features with a high number of unique values and numerical features with very few unique values are filtered out [6].

Due to the restricted search budget for hyperparameter tuning, a uniform procedure for this process is defined to reduce the otherwise high variance in model performance. A random search with 400 iterations per dataset is conducted on CPU for tree-based models and on GPU for DL models. This

search is repeated 15 times with different random orderings of the search space to simulate variability and obtain a bootstrap-like estimate of the expected performance [6]. Since similar constraints regarding hyperparameter search budgets may arise in the proposed thesis, this procedure could serve as a good starting point for identifying representative hyperparameters when comparing different models.

To fairly compare model performance across datasets of varying difficulty, a metric is used that normalizes measured accuracy relative to other models, mapping the result to a range between 0 and 1 [6]. This approach may also be suitable for the proposed thesis.

The tree-based models RandomForest, GradientBoostingTrees and XGBoost are benchmarked against the DL models MLP, Resnet, FT-Transformer and SAINT. The results show that tree-based models clearly outperform DL models. The authors identify three reasons for this phenomenon [6].

Firstly, tree-based models are better suited for learning non-smooth functions, whereas NNs are biased toward smooth functions. Tabular data often exhibit irregular or piecewise-constant target functions, which decision trees can fit naturally due to their hierarchical structure, while such functions are more difficult to learn for DL models [6].

Secondly, NNs are more sensitive to uninformative features. They tend to overfit or underperform when too many irrelevant features are present, whereas tree-based models are highly robust in this regard, as they effectively ignore such features during the splitting process [6].

Lastly, DL methods suffer from rotational invariance. While this property can be beneficial in other domains, the order and identity of features are critical in tabular data, where each feature typically carries a distinct semantic meaning. This also explains why embedding layers, which break rotational invariance, often help NNs to achieve better performance on tabular tasks [6].

2.3 Explainable Machine Learning for Scientific Insights and Discoveries

Roscher et al. [10] provide a comprehensive overview of the various aspects of interpretability in ML and explainable AI. Their work can serve both as background for the introductory section, as described in milestone 3.1 of the proposed thesis, and as a methodological reference for assessing and evaluating model interpretability, as outlined in milestone 3.7.

To generate novel scientific insights and identify causal relationships from observational data processed by ML models, it is crucial to understand the

model's decision-making process. Therefore, the concepts of explainability, interpretability and transparency are discussed and compared in detail [10].

2.3.1 Transparency

Transparency refers to the understandability of the entire ML process, including model transparency, design transparency and algorithmic transparency. These aspects can be illustrated using kernel methods and neural networks as example [10].

Model transparency describes how understandable the overall model structure is, particularly the mapping from inputs to outputs, and is refered to as simulatability. Kernel methods and NNs are generally considered model-transparent because their mathematical formulation is known and can be explicitely written down. However, in the case of NNs, understanding what specific parameters represent can be difficult or impossible [10].

Design transparency concerns the justifiability of design choices. In kernel methods, design choices can often be made transparently based on domain knowledge. In contrast NNs typically lack design transparency, as many architectural decisions such as the number of layers or the choice of activation functions are determined heuristically [10].

Algorithmic transparency is present when the training process is deterministic and reproducible. For example, kernel methods frequently involve convex optimization and therefore result in unique, reproducible solutions. In contrast, NNs typically include stochastic optimization and random initialization, which lead to different solutions across training runs. As a result they are not algorithmically transparent [10].

2.3.2 Interpretablity

In contrast to transparency, the interpretability of a model does not focus only on reproducibility but on the ability to make human-understandable sense of the model's internal workings or decision logic. It emphasizes *why* the model makes a particular decision, rather than merely *how* it does so. Interpretability necessarily involves the data, as it is crucial to consider how the model behaves on specific inputs. While transparency can support interpretability, it is not a strict requirement. Interpretability can also be achieved for black-box models through post-hoc methods [10].

There are several techniques for achieving interpretability in black-box models. Proxy models approximate the predictions of a complex model using simpler, interpretable models such as decision trees or linear models. Also, heatmaps or saliency masks can be used to analyze feature importance. These are commonly applied in image classification tasks to help explain model

decisions. A similar technique for NNs is layer-wise relevance propagation, which uses gradients to attribute output relevance to input features [10].

2.3.3 Explainability

Explainability involves the interpretation of a set of samples in conjunction with contextual domain knowledge and the specific analysis goal. It not only addresses why a decision is made, but also why and how decisions differ across various samples [10].

There are four main purposes of explainability in ML. The first is the justification of individual decisions, which is essential for ensuring fairness, detecting bias, and meeting regulatory requirements in certain application domains. Second, gaining control over a model through explainability supports monitoring, flaw detection, and vulnerability assessment to ensure safety and reliability. Third, improvement of a model becomes more feasible with high explainability, as it provides developers with insight into model's behavior and guides optimization. Finally, scientific discovery can be enabled by explainable models that uncover new knowledge, pattern or causality [1].

2.3.4 Methods to Achieve Explainability

The authors distinguish four categories of approaches for using explainable ML to gain scientific insights. Group 1 explains model outputs post-hoc using domain knowledge, sometimes integrating scientific constraints directly into the model design. Group 2 focuses on predicting interpretable scientific properties directly, either by employing transparent models or by validating predictions against known physical laws. Group 3 uses interpretation tools such as feature importance or attention mechanisms to understand a model's behavior. These methods often incorporate domain knowledge to guide or validate the interpretations. Group 4 builds interpretable models from the ground up, with the goal of extracting scientific rules and theories directly from the model itself [10]. DCR, whose analysis and evaluation form the focus of the proposed thesis, also belongs to this last group.

Chapter 3

Milestones

Based on the presented literature, as well as additional sources, a detailed analysis of DCR with a focus on various aspects is to be conducted. The main milestones for this evaluation are described in this chapter. They serve as the guiding thread of the work and are associated with specific chapters of the thesis.

The introductory chapter will motivate the enhancement of interpretability in ML, particularly in the context of applications on tabular data. This provides the main reason for choosing DCR over other blackbox machine learning models. The content of this introductory chapter will be informed by the completion of milestone 3.1.

The second chapter, which adresses background and related work, will explore the various aspects of DCR and analyze its internal functionality, in accordance with milestone 3.3. Milestone 3.5 will also contribute to this section by selecting and describing suitable state-of-the-art machine learning models specialized for handling tabular data. A third component of this chapter may include a literature review on the topic of classification on tabular data, discussing format-specific challenges, limitations, and edge cases.

The third chapter, which covers the methodology, will be based on milestone 3.2, which aims to identify comparable and meaningful benchmarking datasets. It will also incorporate milestones 3.4 and 3.6, which involve optimizing the hyperparameters of DCR and the comparative ML models to suit the conditions of the selected datasets, as well as defining the experimental setup for benchmarking all methods.

The evaluation of the benchmarking results will be presented in the fourth chapter, corresponding to milestone 3.7. This chapter will primarily consist of a comparative analysis focusing on the key aspects accuracy and interpretability, based on the results obtained in the previous chapter.

The objective of the fifth chapter is to draw conclusions from the evaluation results. Here, the accuracy and interpretability scores of the different models across various applications and datasets will form the basis for deriving recommendations on which methods are most appropriate under which conditions. These conclusions will be developed as part of milestone 3.8.

According to the final milestone 3.9, DCR will be applied to the problem of credit risk scoring. This chapter will show the practical utility of DCR in addressing a real world problem. In addition to achieving interpretable and accurate classifications, one of the objectives will be to make the DCR model discover logical rules, that make the scoring decisions understandable to humans.

The final chapter will compile and summarize all findings of the thesis. Particular emphasis will be placed on the conclusions regarding DCR and ML on tabular data as well as the interpretable credit risk scoring. Finally, an outlook about possible future work will be provided.

The following pages present and describe the aforementioned milestones in detail. This includes justification why these steps are necessary, their intended purpose, and initial ideas on how to approach them.

3.1 Importance of Interpretability in Machine Learning on Tabular Data

This introductory aspect is crucial for motivating the later use of DCR instead of more commonly used ML methods. The importance of interpretability varies depending on the application and the point of view. Reasons for increasing interpretability include trust and reliability in models decisions, validating the fairness of model decisions, and facilitating debugging when interpretable intermediate results are available. Relevant literature for this section may include works as *Stop Explaining Black Box Machine Learning Models for High Stakes Decisions and Use Interpretable Models Instead* [11] or *Interpretable Machine Learning* [7].

3.2 Selection and Description of Benchmarking Tabular Datasets

When selecting a set of benchmarking datasets, several characteristics beyond domain relevance and generalizability have to be considered. Based on the criteria outlined by Grinsztajn et al. [6], the following conditions can guide this selection. Datasets should be based on real-world data rather than synthetic or artificially generated data. They should not be excessively high-dimensional and have to be well documented and contain a sufficient

number of features and samples. Additionally, datasets should not be trivially easy to classify, for instance, they should not be solvable by a single decision tree. Class balancing and the treatment of missing values must also be addressed. Furthermore, a seperate evaluation of categorical and numerical datasets may be considered, as the performance of many ML methods varies significantly between these datatypes. For models that do not support categorical data natively, preprocessing using one-hot encoding will be necessary to encode categorical features into binary variables. To simplify preprocessing, categorical data should have low cardinality, meaning each feature has a limited number of unique values. Conversely, numerical data should exhibit high cardinality, meaning a higher number of distinct values to be meaningful as continous variable [6].

3.3 Review of Method, Advantages and Opportunities of DCR

Based on Barbiero et al. [3] and section 2.1 the concept of DCR has to be reviewed in detail to establish a solid foundation for the subsequent evaluation and analysis. An additional examination of the Python implementation of DCR, available in the pytorch_explain GitHub repository, could provide a deeper understanding of the method's inner workings. A theoretical comparison to other concept-based ML models may also reveal further insights about the advantages or limitations of DCR, which may also contribute to milestone 3.8.

3.4 Application and Optimization of DCR on Tabular Benchmarking Datasets

3.4.1 Tuning of DCR for the Benchmarks

DCR has several hyperparameters that influence classification performance and therefore have to be tuned. These include the learning rate and the number of epochs for training the NNs. Additionally, the parameter τ can be adjusted to control the complexity of generated rules. This represents a potential tradeoff between interpretability of the model, if the rules rely on only a few concepts, against the accuracy, which generally improves when rules are less sparse [3].

In the case of multi-task training, meaning training the concept encoder jointly with DCR in an end-to-end fashion of training, further hyperparameters must be considered. These include the balance between the concept prediction loss and the task prediction loss, as well as the number of concepts and the size of concept embeddings.

3.4.2 Benchmarking of DCR

After determining the most promising hyperparameters, DCR will be applied to all datasets. Seperate runs for categorical, numerical and mixed data types are possible. Depending on the outcomes of milestone 3.4.1, additional runs with different hyperparameters may reveal interesting insights for example with regard to the trade-off between accuracy and interpretability. For all runs, the accuracy and confusion matrix have to be recorded for later analysis. In addition, the logical rules, both global and local, have to be saved to facilitate the evaluation of the interpretability.

3.5 Review and Analysis State-Of-The-Art Machine Leaning Models for Tabular Data

The selection of ML models for comparative analysis is crucial for the final evaluation, as it establishes the reference framework for assessing DCR. Therefore, a diverse set of methods should be composed, representing different methodological approaches and underlying techniques. According to Grinsztajn et al. [6], tree-based approaches are particularly well-suited for tabular data. As a result, the primary focus will be on tree-based models, although alternative methods will also be considered. In addition, all selected models should offer at least some degree of interpretability through their internal structure, as black-box methods are difficult to compare meaningfully with DCR in terms of interpretability. The following selection can serve as an initial starting point for identifying suitable models for comparison.

3.5.1 Catboost

Catboost is a gradient boosting method that can handle numerical features but is specifically designed to process categorical data efficiently, especially compared to other decision tree boosting methods. It applies ordered target statistics to encode categorical features and constructs a series of oblivious decision trees. Each new tree is build with the goal of spliting the data in a way that reduces the combined loss function of all the previous trees. This is achieved by using the gradient of this loss function. As catboost is one of the state-of-the-art decision tree boosting algorithms and performs exceptionally well on categorical data, it is a strong candidate for comparison with DCR [9].

3.5.2 Random Forest

In contrast to boosting methods, Random Forest (RF) relys on bagging and thus represents a different approach to combining decision trees into a more powerful model. It constructs a set of independent decision trees in

parallel, where each split is based on a random subset of features. The final classification is obtained by averaging the predictions of all individula trees. Although RF was introduced in 2001, it remains a widely used and robust baseline for decision tree bagging and is still considered state-of-the-art in many tabular data applications [4, 6].

3.5.3 Neural Oblivious Decision Ensembles

Neural Oblivious Decision Ensembles (NODE) is a DL model specifically designed for tabular data, aiming to close the performance gap between tree-based and DL models by combining the strengths of both approaches [8]. A Python implementation is available on GitHub.

NODE employs Oblivious Decision Trees, which are made differentiable through the use of the entmax transformation instead of hard splits. This allows gradient-based optimization via backpropagation. The model also uses multi-layer stacking, where outputs of all preceding layers are concatenated and passed to the next tree, enabling hierarchical representation learning. NODE has demonstated the ability to outperform tree-based models such as CatBoost or XGBoost. A certain degree of interpretability can be achieved by analyzing feature importance and the deepest decision trees, which have the greatest influence on the final prediction [8].

3.5.4 TabNet

Another DL architecture designed for tabular data is TabNet. It leverages instance-wise attention and sparse feature selection to match the performance of tree-based methods while still using neural networks. The integrated feature selection mechanism enables interpretability through feature importance analysis and supports visual exploration of instance-wise decision behavior. These properties makes TabNet a potential candidate for comparison with DCR [2].

3.6 Benchmarking of Machine Learning Models

3.6.1 Hyperparameter Tuning for Each Model

The goal of this milestone is to optimize the ML models for the benchmarking datasets. This process is specific to the hyperparameters and architecture of each model.

Due to limited resources for hyperparameter tuning, a similar approach to the one described in chapter 2.2 may be adopted. Given constraints in time and computational resources, it is likely that globally optimal hyperparameters cannot be identified. Instead, a series of simplified searches with

varying initialization or search spaces can be used to find a set of sufficiently good hyperparameter configurations for each model. These will be used in milestone 3.6.2 to perform multiple benchmarking runs [6].

3.6.2 Applying ML Models on the Benchmarking Datasets

Using the set of hyperparameters obtained in the previous milestone, each model will be executed multiple times in different configurations. After measuring the classification accuracy and generating confusion matrices for further analysis, the mean and the variance of each model's performance across different hyperparameters settings will be calculated and visualized.

3.7 Comparative Analysis of DCR and Benchmark Methods

The comparison between DCR and the other tabular ML models will focus on the two main aspects accuracy and interpretability.

3.7.1 Accuracy

Accuracy is the most fundamental metric for assessing how well a model performs on a given dataset, making its analysis essential. Since each model will be evaluated using a set of different hyperparameter configurations, the comparison will not be limited to a single accuracy value per model. Instead, best-case, worst-case, and average-case performance will be considered. Furthermore, accuracy can be analyzed both across all datasets collectively and within each specific dataset. This allows for evaluating each model's abilitiy to handle different data types, such as numerical or categorical features. Additional dataset characteristics, such as the number of features, the proportion of uninformative features, or the distribution of values within features, could reveal interesting insights.

3.7.2 Interpretability

In contrast to accuracy, interpretability is more difficult to quantify, as there is no universally accepted metric for its evaluation. Nevertheless, it is a central focus of the proposed thesis on interpretable ML.

Depending on a model's architecture, interpretability may be assessed through feature importance, decision making for specific samples, or detailed examination of learned weights or decision trees. The overall simplicity of a model, as well as its sensitivity to perturbations, may also provide valuable insights into its interpretability.

3.8 Interpretation of Benchmarking Results

Based on the results obtained in milestone 3.7, further conclusions can be drawn regarding the suitability of specific models for particular applications. Assuming that no single model outperforms all others in terms of both accuracy and interpretability across all datasets, it will be possible to derive recommendations on which model to use under which conditions. Ideally, these recommendations will be supported not only by the empirical findings from milestone 3.7 but also by theoretical considerations related to each model's internal structure and functionality. This includes analyzing the inherent strengths and weaknesses that contribute to the observed performance patterns.

3.9 Application of DCR to Credit Risk Score Classification

As the final task of the proposed thesis, the utility of DCR will be demonstrated through a real-world application on tabular data. Specifically, DCR will be applied to the task of classifying credit risk scores based on characteristics of credit applicants.

3.9.1 Relevance of Interpretability

A primary motivation for requiring interpretability in credit risk scoring is compliance with legal regulations, such as the European Union's General Data Protection Regulation (GDPR). This regulation prohibits to use automated decision-making without meaningful explanation if the decision has a significant impact on an individual's life [5]. This condition clearly applies to credit risk scoring, as the approval or denial of a credit can have substantial consequences for an individual.

In addition to legal compliance, interpretability is essential for building trust with customers, especially in case of credit denial, because transparency helps users understand the rationale behind decisions. Moreover, fairness and the detection or mitigation of bias can only be proven if the decision-making process is interpretable.

3.9.2 DCR for Credit Risc Scores

To evaluate the applicability of DCR in the domain of credit risc score, the Home Credit Default Risk Dataset can be used. It contains 122 features that describe customer-related information such as gender, loan, or number of children but also the requested type of credit contract. This dataset was originally published as part of a competition on machine learning hosted

by the Home Credit Group to identify suitable ML models for estimating a customer's ability to repay loans. The dataset will require preprocessing to impute missing values and may need to be subsampled. Following preprocessing, the hyperparameters of DCR will be tuned, similar to the procedure of milestone 3.4.1, before applying the model to the test set.

3.9.3 Conclusions

In addition to evaluating the model's accuracy and comparing it to the performance benchmarks from the original competition, the main focus will be on interpreting the determined logical rules. On the one hand, local rules for individual samples can provide insights into the specific reasons for credit approval or denial based on customer data. On the other hand, the global rules may serve as general guidlines for banks and lending institutions for credit decision-making. A final discussion of the results will highlight limitations, advantages, and disadvantages of applying DCR for credit risk scoring, and in the best case proving its practical utility in this field.

List of Abbreviations

DCR Deep Concept Reasoning

DL deep learning

GDPR General Data Protection Regulation

ML machine learning

NN neural network

NODE Neural Oblivious Decision Ensembles

RF Random Forest

Bibliography

[1] Amina Adadi and Mohammed Berrada. Peeking inside the black-box: A survey on explainable artificial intelligence (xai). *IEEE Access*, 6:52138–52160, 2018.

[2] Sercan Ö Arik and Tomas Pfister. Tabnet: Attentive interpretable tabular learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 6679–6687, 2021.

[3] Pietro Barbiero, Gabriele Ciravegna, Francesco Giannini, Mateo Espinosa Zarlenga, Lucie Charlotte Magister, Alberto Tonda, Pietro Lió, Frederic Precioso, Mateja Jamnik, and Giuseppe Marra. Interpretable neural-symbolic concept reasoning. In *International Conference on Machine Learning*, pages 1801–1825. PMLR, 2023.

[4] Leo Breiman. Random forests. *Machine learning*, 45:5–32, 2001.

[5] Bryce Goodman and Seth Flaxman. European union regulations on algorithmic decision-making and a "right to explanation". *AI Magazine*, 38(3):50–57, Oct. 2017.

[6] Léo Grinsztajn, Edouard Oyallon, and Gaël Varoquaux. Why do tree-based models still outperform deep learning on typical tabular data? *Advances in neural information processing systems*, 35:507–520, 2022.

[7] Christoph Molnar. *Interpretable Machine Learning*. 3 edition, 2025.

[8] Sergei Popov, Stanislav Morozov, and Artem Babenko. Neural oblivious decision ensembles for deep learning on tabular data. *arXiv preprint arXiv:1909.06312*, 2019.

[9] Liudmila Prokhorenkova, Gleb Gusev, Aleksandr Vorobev, Anna Veronika Dorogush, and Andrey Gulin. Catboost: unbi-

ased boosting with categorical features. *Advances in neural information processing systems*, 31, 2018.

[10] Ribana Roscher, Bastian Bohn, Marco F. Duarte, and Jochen Garcke. Explainable machine learning for scientific insights and discoveries. *IEEE Access*, 8:42200–42216, 2020.

[11] Cynthia Rudin. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature machine intelligence*, 1(5):206–215, 2019.

The signed declaration of originality is a component of every semester paper, Bachelor's thesis, Master's thesis and any other degree paper undertaken during the course of studies, including the respective electronic versions.

Lecturers may also require a declaration of originality for other written papers compiled for their courses.

I hereby confirm that I am the sole author of the written work here enclosed and that I have compiled it in my own words. Parts excepted are corrections of form and content by the supervisor.

Title of work (in block letters):

Authored by (in block letters):
For papers written by groups the names of all authors are required.

Name(s):	**First name(s):**

With my signature I confirm that

- I have documented all methods, data and processes truthfully.
- I have not manipulated any data.
- I have mentioned all persons who were significant facilitators of the work.

I am aware that the work may be screened electronically for plagiarism.

Place, date	**Signature(s)**

For papers written by groups the names of all authors are required. Their signatures collectively guarantee the entire content of the written paper.