Karlsruhe Institute of Technology

Software Design and Quality

Integrating Explainability into Federated Learning: A Non-functional Requirement Perspective

Master's Thesis of

Nicolas Sebastian Schuler

At the KIT Department of Informatics
KASTEL – Institute of Information Security and Dependability

First examiner: Prof. Dr. Raffaela Mirandola
Second examiner: Prof. Dr.-Ing. Anne Koziolek

04. November 2024 – 05. May 2025

Karlsruher Institut für Technologie
Fakultät für Informatik
Postfach 6980
76128 Karlsruhe

I declare that I have developed and written the enclosed thesis completely by myself. I have not used any other than the aids that I have mentioned. I have marked all parts of the thesis that I have included from referenced literature, either in their original wording or paraphrasing their contents. I have followed the by-laws to implement scientific integrity at KIT.

Karlsruhe, 04. May 2025

. . . . . .

(Nicolas Sebastian Schuler)

Abstract

In recent years, the advancements of Artificial Intelligence (AI)-driven software systems have reached a seemingly unstoppable momentum. From simple applications like to-do-applications to the most complex ones like banking applications, the demand to integrate AI into applications is exceptionally high nowadays. Whilst researchers and practitioners could gain much understanding in developing AI applications, one critical quality attribute seems left behind. This can be emphasized in a simple analogy: although we usually demand professors and teachers to be qualified for their jobs and to be able to explain their reasoning to students, for AI, our bar seems much lower if not absent. Some scholars see this lack of explainability and resulting intransparency as detrimental. Despite much research efforts in Explainable Artificial Intelligence (XAI), a meaningful integration thereof is still very much a subject of ongoing research. Furthermore, for even the most critical applications, e.g., the medical field, traditional Machine Learning (ML) methods cannot satisfy strict privacy requirements. For this reason, the AI paradigm Federated Learning (FL) emerged as a means to train ML models decentralized. Despite the benefits of explainability and FL being present today, the integration of explainability into FL is still seriously lacking.

In this master's thesis, we empirically researched the interaction between FL and explainability. We show that the global FL model outperforms the local one in terms of accuracy and prevalent XAI metrics and that data distribution affects the outcome more than the employed FL algorithm. We investigate which XAI method is the most stable concerning the effect of the predictive multiplicity, propose an adapted version of the XAI method proposed in [50] to optimize explanations for XAI metrics, and lastly also show that further security-related aspects influence XAI methods in a non-negligible way. Furthermore, we will present a concept of explainability derived from a literature search encompassing ideas from multiple disciplines with abduction as the central element in an attempt to better understand what the term explainability means from a human-centric point of view. Finally, with our user survey, we show that while demand for explainability is considered very high, there exists a gap between the usage of AI and XAI methods and a notable difference in how explanations are deemed satisfactory in the first place. Moreover, we show that the benefit of XAI metrics as a means to improve explanations is questionable.

Zusammenfassung

In den letzten Jahren haben Softwaresysteme, getrieben von Künstlicher Intelligenz (KI), ein schier unaufhaltsames Momentum aufgebaut. Von einfachen Anwendungen wie To-do-Listen zu den komplexesten Anwendungen wie in Banken-Systemen ist die Nachfrage, KI in immer mehr Anwendungen zu integrieren, so hoch wie nie zuvor. Während Forscher und Entwickler bei der Entwicklung von KI-Anwendungen bereits viele Erkenntnisse gewinnen konnten, scheint ein entscheidendes Qualitätsmerkmal auf der Strecke zu bleiben. Dies lässt sich mit einer einfachen Analogie verdeutlichen: Obwohl wir von Professoren und Lehrern normalerweise verlangen, dass sie für ihre Arbeit qualifiziert sind und ihre Schlussfolgerungen und Aussagen ihren Studenten erklären und begründen können, scheint die Messlatte für KI-Anwendungen viel niedriger zu sein, wenn nicht gar zu fehlen. Einige Wissenschaftler sehen diesen Mangel an Erklärbarkeit und daraus resultierende Intransparenz als unzumutbar. Trotz vieler Forschungsanstrengungen im Bereich der Erklärbarkeit von künstlicher Intelligenz (XAI), ist deren sinnvolle Integration in Anwendungen noch immer Gegenstand laufender Forschungsvorhaben. Darüber hinaus zeichnen sich noch andere Probleme beispielsweise im Bereich der Medizin darin aus, dass traditionelle Modelle des maschinellen Lernens (ML) die strengen Anforderungen an den Datenschutz nicht erfüllen können. Um diesen Anforderungen Sorge zu tragen, hat sich föderiertes Lernen (FL) als Mittel zum dezentralen Trainieren von ML-Modellen etabliert. Obwohl die Vorteile von Erklärbarkeit und FL heute bereits vorhanden sind, ist die Integration von Erklärbarkeit in FL mangelhaft.

In dieser Masterarbeit haben wir deshalb die Interaktion zwischen FL und Erklärbarkeit empirisch untersucht. Wir zeigen, dass das globale FL-Modell das Lokale nicht nur in Bezug auf Genauigkeit, sondern auch in gängigen XAI Metriken überlegen ist, wobei wir festellen, dass die Datenverteilung größeren Einfluss auf XAI-Metriken aufweist als die verwendeten FL-Algorithmen. Wir untersuchten, welche XAI-Methoden am stabilsten sind in Bezug auf den Effekt der Vielfachheit von Vorhersagen (engl. *predictive multiplicity*), und schlagen eine Erweiterung für die in [50] vorgeschlagene XAI-Methode vor, um Erklärungen für XAI-Metriken zu optimieren. Wir zeigen auch, dass sicherheitsrelevante Aspekte die XAI-Methoden in nicht unerheblicher Weise beeinflussen und stellen ein Konzept der Erklärbarkeit vor, das aus einer Literaturrecherche abgeleitet wurde, die Ideen aus verschiedenen Disziplinen aufgreift und das Konzept der Abduktion als zentrales Element hat, um besser zu verstehen, was der Begriff Erklärbarkeit aus einer Nutzer-zentrierten Sicht bedeutet. Schlussendlich zeigen wir mit unserer Nutzerstudie, dass obwohl die Nachfrage nach Erklärbarkeit als sehr hoch angesehen wird, große Lücken zwischen der Nutzung von KI und der Anwendung von XAI-Methoden vorherrschen. Darüber hinaus wird erkenntlich, dass es Unterschiede in der Art und Weise gibt, wie Erklärungen als zufriedenstellend

angesehen werden, und dass die Nutzung von XAI-Metriken als Mittel zur Verbesserung von Erklärungen fraglich ist.

Contents

Abstract i

Zusammenfassung iii

1. Introduction 1
- 1.1. Contribution . . . 2
- 1.2. Outline . . . 2

2. Foundations 5
- 2.1. Explainability . . . 5
  - 2.1.1. Design . . . 6
  - 2.1.2. Evaluation . . . 7
- 2.2. Federated Learning . . . 10
  - 2.2.1. Process . . . 11
  - 2.2.2. Algorithms . . . 11
  - 2.2.3. Architectures . . . 15
  - 2.2.4. Evaluation . . . 16
  - 2.2.5. Challenges . . . 17
- 2.3. Explainable Artificial Intelligence (XAI) . . . 17
  - 2.3.1. Local Interpretable Model Agnostic Explanation (LIME) . . . 18
  - 2.3.2. SHapley Additive exPlanations (SHAP) . . . 19
  - 2.3.3. Saliency Map . . . 21
  - 2.3.4. Input X Gradient . . . 21
  - 2.3.5. Integrated Gradients (IG) . . . 22
  - 2.3.6. Grad-CAM . . . 22

3. Related Work 25

4. Explainability and Federated Learning 27
- 4.1. Goals and Questions (1) . . . 27
- 4.2. Experiment Preliminaries . . . 28
  - 4.2.1. Datasets . . . 30
  - 4.2.2. Implementation . . . 30
- 4.3. Experiment 1: Local or Global Model? . . . 31
- 4.4. Experiment 2: Which XAI method is more stable? . . . 35
  - 4.4.1. Methodology . . . 36
  - 4.4.2. Results . . . 36

4.4.3. Remarks . . . 39
4.5. Experiment 3: Does Optimizing Explanations through Aggregation create better Explanations? . . . 39
4.5.1. Proposed Solution . . . 40
4.5.2. Analysis . . . 42
4.5.3. Application for Requirements Engineering . . . 46
4.6. Experiment 4: How much does Differential Privacy harm the explanations? 47
4.7. Experiment 5: What if clients misbehave? . . . 49

5. Evaluation **53**
5.1. Goals and Questions (2) . . . 53
5.2. Survey . . . 53
5.2.1. Setup . . . 53
5.2.2. Implementation . . . 54
5.2.3. Execution . . . 54
5.3. Results and Discussion . . . 54
5.4. Threats to Validity . . . 61
5.4.1. Construct Validity . . . 61
5.4.2. Internal Validity . . . 61
5.4.3. External Validity . . . 62
5.4.4. Repeatability . . . 62
5.5. Lessons Learned (1) . . . 63

6. Conceptualizing Explainability **65**
6.1. Goals and Questions (3) . . . 65
6.2. Entangling Explainability . . . 65
6.2.1. Explanations as Proofs . . . 66
6.2.2. Explanations as Causal Reasoning . . . 72
6.2.3. Explanations in Social Science . . . 73
6.2.4. Abduction as the Unifying Element . . . 75
6.3. Approaching a Categorization for Explainability . . . 81
6.4. Remarks . . . 83
6.4.1. What is Missing? . . . 83
6.4.2. Lessons Learned (2) . . . 85

7. Conclusion **87**
7.1. Summary . . . 87
7.2. Future Work . . . 88

Bibliography **91**

A. Appendix **113**

List of Figures

2.1. Assumed Relation between Model Accuracy and Model Explainability [80]. 7
2.2. Visualization of the fl Algorithm 1. 11
2.3. Example Calculation of Shapley Values. 20

4.1. Most Important Flower Classes. 29
4.2. Duration for Metric Calculations. *[Left] Client 0, [Right] Global* 32
4.3. Showing the Influence of Data Partitioning on Accuracy (FedAvg). 33
4.4. Round-to-Round Metric comparing Client [red] and Global [blue] Attribution of Round T against Round $T-1$ (FedAvg/IID). 33
4.5. Comparison Against Different Attributions. 35
4.6. Visualization for Results in Table 4.4. 37
4.7. Example Results for Stability Measurements. 38
4.8. Optimizing Explanations through Aggregation for certain Metrics [50]. . . 40
4.9. Comparing different Aggregated Weight Computation Methods. 43
4.10. Comparing different Aggregation Methods on a per Metric-basis. 44
4.11. Pareto Front Between the Two Objectives Cost and Performance. *Notice: The Perfomance Axis Is Presented Inverse. [Red] Chosen Trade-off Points for Each Class.* 45
4.12. Multi-Objective Optimization on Infidelity and Sensitivity Metrics. 45
4.13. Improvement on Perturbation-based XAI Metrics. 45
4.14. Reusing Aggregated Weights for the next Round. 46
4.15. Calculating Aggregation Weights over all Classes simultaneously. 46
4.16. Results for Saliency FedAvg/IID with and without DP. 50
4.17. Multi-objective Explanation Optimization with and without DP. 51

5.1. Participant's Heterogenity. 55
5.2. Self-expressed Knowledge in SE. 55
5.3. Years of Experiences in SE. 55
5.4. Self-expressed Knowledge in AI and XAI. 56
5.5. Opinions about Explainability in SE. 57
5.6. Essential Parts of an Explanation. 57
5.7. Which Kind of Explanation is Acceptable. 58
5.8. Explanation Acceptance Solely Based on Heatmap. 59
5.9. Explanation Acceptance Based on Heatmap and Textual Information. . . . 59
5.10. Explanation Acceptance Solely Based on Heatmap, Textual Information and Accuracy. 59
5.11. Comparing Optimized Explanations against other XAI methods. 60

6.1. Explainable Artificial Intelligence (XAI) Method applied to ResNet-50. . . . 67
6.2. Simple Rule-based Classifier. . . . 67
6.3. Visualization of Abduction in Contrast to Prediction [96]. . . . 68
6.4. Overview of Common Causal Relations [174]. . . . 72
6.5. Combining Different Views with Abduction. . . . 75
6.6. Image of a Spilled Cup of Liquid. . . . 77
6.7. Hospital Use Case Overview with Federated Learning (FL). . . . 79
6.8. Proposed Explainability Assessment Categorization. . . . 82

A.1. Explainability Requirements to Support Abductive Artificial Intelligence (AI) [85]. . . . 113
A.2. Process Model for Abduction [85]. . . . 113
A.3. Flower Strategy Sequence Diagram [17]. . . . 115
A.4. Results for Respecting the Real Cost e.g., Using $\lceil \psi \rceil$. . . . 116
A.5. Improvement via Optimization. . . . 117
A.6. Individual improvement on Perturbation-based XAI Methods. . . . 118
A.7. Individual improvement over all Classes simultaniously. . . . 118
A.8. Measuring Stability FedAvg/IID (1). . . . 118
A.9. Measuring Stability FedAvg/IID (2) 119
A.10. Measuring Stability FedAvg/IID (3). . . . 120
A.11. Measuring Stability FedAvg/Square (1). . . . 121
A.12. Measuring Stability FedAvg/Square (2). . . . 122
A.13. Measuring Stability FedAvg/Square (3). . . . 123
A.14. Measuring Rashomon Effect FedAvg/Square. . . . 124
A.15. Measuring Stability FedAvg/Dirichlet (1). . . . 125
A.16. Measuring Stability FedAvg/Dirichlet (2). . . . 126
A.17. Measuring Stability FedAvg/Dirichlet (3). . . . 127
A.18. Measuring Rashomon Effect FedAvg/Dirichlet. . . . 128
A.19. Results for Experiment 4: 150 FL Rounds. . . . 128

List of Tables

2.1. *Levels of Explainability Readiness* according to [27]. 7
2.2. Metrics for Explanations. 9
2.3. List of Selected FL Algorithms in Flower Version 1.14.0. 12
2.4. Metrics for Evaluating Machine Learning (ML) Models. *Legend: [R] Regression, [C] Classification.* 16

4.1. Comparing Correlation Between Metrics Taken from Local and Global Model Differing Algorithms. 34
4.2. Comparing Correlation Between Metrics Taken from Local and Global Model Differing Data Partitioning. 34
4.3. Some XAI Metrics in the independent and identically distributed (IID) Case. *Global [red], Client [blue].* 35
4.4. Measuring the susceptibility of the Rashom Effect for different XAI Methods. 36
4.5. Results for Saliency FedAvg/IID with and without DP. 49
4.6. Results of Experiment 5 (Shifting Labels): Saliency on FedAvg/IID. 51
4.7. Results of Experiment 5 (Randomizing Labels): Saliency on FedAvg/IID. . . 52

5.1. Fleiss' Kappa Interpretation [118]. 54

6.1. Explainability Elements That Need to Be Identified. 80

A.1. Explainability Properties from different Papers. 114
A.2. Hyperparameters for Series of Experiments 1. 116

1. Introduction

Software systems are increasingly integrated into our daily lives with a seemingly ever-increasing amount of complexity and requirements to fulfill. Additionally, integrating new and exciting Artificial Intelligence (AI) methods and tools into these systems according to stakeholders' desires and needs forces today's software developers to adopt a different kind of thinking than they used to. Especially in areas where stakeholders, including regulatory bodies, demand strict adherence to privacy and transparency requirements, the deployment of Federated Learning (FL) to satisfy both requirements becomes interesting [100]. FL is an AI paradigm proposed to preserve data privacy, allowing multiple parties to train a Machine Learning (ML) model collaboratively without sharing their local data [132]. Such characteristics are essential, where data shall remain local and not be distributed (e.g., patient data in hospitals [180]). However, while recent literature already looks at FL with explainability strictly algorithmically, it remains unclear how to approach it from a requirements perspective, especially regarding the evaluation and comparison of explanations.

Furthermore, current research on explainability as a non-functional requirement is still primarily limited in generating generally applicable concepts to understand explainability [27]. Still lacking the necessary depth, definitions, and means for evaluating practical consideration. This becomes troubling for developers seeking guidance on integrating explainability into their FL system [32, 81].

Therefore – amongst others – this thesis will integrate explainability as a non-functional requirement in the context of FL and empirically evaluate different measurements regarding explanation methods, their explanations, and the FL context. This thesis aims to contribute to a better understanding of explainability in the FL context based on empirically collected and evaluated data. In parallel to the experimental aspect of this thesis, which is a bottom-up approach, we also try to further analyze, conceptualize, and understand explainability from a top-down perspective, based on a literature search encompassing multiple research disciplines, to tackle the human side problem.

1.1. Contribution

For this thesis, we plan to answer the following three research questions:

> **RQ1**: How can explainability (specifically using XAI methods) be approached in FL contexts?
>
> **RQ2**: Can we improve existing explanation methods for the FL context?
>
> **RQ3**: How can we approach the human side of the explainability problem with already existing research knowledge?

Based on the research questions, we aim to investigate FL in conjunction with existing explainability methods thoroughly. While we can not cover every subtopic that FL offers, we can at least evaluate the most common and simplest one, the image classification task, with the CIFAR-10 data set.

Therefore, to answer **RQ1**, we will conduct several experiments to understand how the FL context can influence common Explainable Artificial Intelligence (XAI) metrics. Then, we move to another type of question that is usually just glanced at or ignored in this context but sets this thesis distinctively apart from other research, analyzing the impact of the effect of predictive multiplicity. We will measure to which degree the explanations change by repeatedly executing the same experiment. Lastly, for this research question, we will look at how explanations are affected by security-related aspects, e.g., introducing Differential Privacy (DP) and FL clients who misbehave.

We then reach **RQ2**, in which we consider optimization techniques to improve explanations measurably. Our results are evaluated based on prevalent XAI metrics in the research literature, and a user survey was conducted to verify that the proposed method holds real-world merit.

Finally, for **RQ3**, we will investigate a concept of explainability derived from a literature search that encompasses ideas from multiple disciplines to approach the human side of the explanation problem. This research question became increasingly important while finding answers for **RQ1** and **RQ2** because it was evident that the human component can not be neglected when it comes to achieving explainability.

1.2. Outline

The thesis is structured as follows: In Chapter 2, we discuss fundamentals related to explainability, FL, and XAI. Chapter 3 focuses on existing work on the subject matters and relates them to the content of this thesis. In Chapter 4, we explore explainability in the context of FL through various experiments. Some of our results from Chapter 4 are then validated in our user survey in Chapter 5. In parallel, in Chapter 6, we investigate explainability based on a multidisciplinary view and try to conceptualize explainability

with abduction from a top-down perspective. Finally, the thesis will be summarized, and opportunities for further research will be presented in Chapter 7.

2. Foundations

This chapter introduces the most essential concepts for the proposed thesis. Readers are advised to see the presented topics as related to each other. The division into sections is made for clarity purposes.

2.1. Explainability

No definitive, agreed-upon definition for explainability as a non-functional requirement exists at the time of writing. However, multiple attempts have been made [27, 33, 37, 51, 52, 59, 109, 159, 163]. One particular helpful definition is presented by Kohl et al. [109] as stated in the following:

> *Definition 1 (Explanation For)*: E is an explanation of explanandum X[1] with respect to aspect Y for target group G, in context C, if and only if the processing of E in context C by any representative R of G makes R understand X with respect to Y.
>
> (Kohl et al. [109])

The authors define explanations as *enabling understanding* in a *context* and *target-aware* fashion. They argue that by coupling the definition of an explanation to the concept and mechanisms of *human understanding*, the research community can benefit from the results already gathered in psychology and cognitive science. The additional constraints placed by the *context* and *target awareness* are stated to be necessary because it is naturally recognizable that not every explanation is appropriate in every *context* or by every *targeted* explainee [24, 167].

They then proceed with defining an explainable system:

> *Definition 2 (Explainable System)*: A system S is explainable by means M with respect to aspect Y of an explanandum X, for target group G in context C, if and only if M is able to produce an E in context C such that E is an explanation of X with respect to Y, for G in C.
>
> (Kohl et al. [109])

[1] Sometimes the *explananda* is also called "phenomena" [163].

Here, the means M is someone or something (which can be separated from the system)[2] that provides the explanation. The definition of the non-functional requirement is then provided as follows:

> *Definition 3 (Explainability Requirement)*: A system S must be explainable for target group G in context C with respect to aspect Y of explanandum X.
>
> (Kohl et al. [109])

The requirement is defined as a specific quality of the system and, therefore, non-functional [77, 182]. The non-functional nature of this requirement can also be inferred because no specific means M, as stated in *Definition 2*, has been specified. Leaving the *operationalization* of the explainability requirement as an open question that is additionally constrained by the *context* and *target awareness* mentioned above.

2.1.1. Design

Explainability is not autotelic; a set of common overall goals can be identified. To this end, the authors Ali et al. [5] provide a list of six goals[3] that an explainable system shall satisfy:

- **Empower** individuals to combat any harmful consequences that arise from the explanandum X[4].
- **Assist** individuals to make informed choices after receiving the explanation E.
- **Expose** the rationals behind the explanandum X or possibly a lack of it.
- **Enabling Integration** of algorithms into systems S in compliance with human values.
- **Enhance** satisfaction and confidence in system S for target group G.
- **Enforce** legal requirements (e.g., the Right of Explanation [163]).

The definition of these goals informs the design process for explainable systems. However, it is not suited for the categorization thereof. For the categorization of explainable systems, the authors in [16, 27, 163] proposed to separate explainable systems into *levels of explainability readiness*. See Table 2.1 for reference.

The levels are categorized according to a rising amount of self-awareness and the capability of providing explanations. At face value, such a categorization is more targeted for cyber-physical, self-adaptive systems like robots that exert behavior to be explained and recognize the need for explanation intelligently (at least level 2 onward systems). Furthermore, level 5 is explicitly highlighted. The concept of having multiple communicating feedback loops

[2] If the explanation is part of the system S, the system is called *self-explainable* [163].

[3] The original phrasing has been changed to match the presented explainability definitions.

[4] Some scholars argue that explanations should be *risk-focused* in nature [170], meaning to be able to assess and weight the explanation provided by a system, against any risks in regard to the extent that users are affected by the system in question.

Explainability	Description
Level 1	No explainability
Level 2	Recognition of the need of explainability
Level 3	Local aspect explainability
Level 4	Global aspect explainability
Level 5	Communicated explainability

Table 2.1.: *Levels of Explainability Readiness* according to [27].

operating in a decentralized manner, and by that means carrying out the explainability, can be mapped to Federated Learning (FL) which will be presented in the next section.

Another important consideration for the design of explainable systems is the often assumed[5] tradeoff between the Machine Learning (ML) model's performance characteristics and the explainability of the system [80, 135, 169]. Figure 2.1 shows the common inverse relation assumption between the two system characteristics. Simpler ML models (e.g., linear regression, decision trees) are generally considered inherently more interpretable and more explainable. In contrast, complex ML models tend to be "black-boxes" in nature (e.g., Deep Neural Networks (DNNs), ensemble methods).

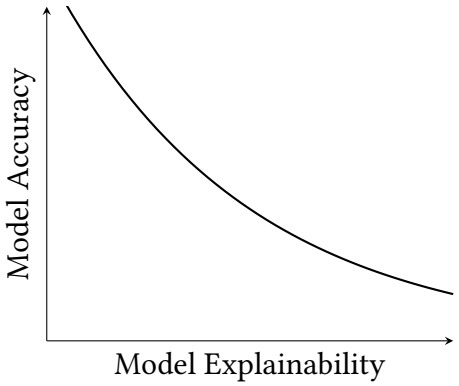


Figure 2.1.: Assumed Relation between Model Accuracy and Model Explainability [80].

2.1.2. Evaluation

Evaluation methods for explainability can be divided into three distinct sets of experiments: (i) Application-grounded (end user experiments), (ii) Human-grounded (layperson experiments), and (iii) Functional-grounded (objective measurements) [13, 29, 57, 142, 162]. The list above is sorted in descending order of evaluation cost. Application-grounded evaluation needs human participants who act as domain experts. This type of evaluation is the most expensive but also the most beneficial in assessing a system's explainability. By relaxing the

[5] No empirical validation has been done to the author's best knowledge, and research suggests some doubts [188].

domain expert requirement to laypersons as subjects for the experiments, Human-grounded evaluations are more affordable than Application-grounded ones. However, the validity is generally decreased compared to the domain-specific feedback provided by the Application-grounded evaluation. This decrease is also partly due to task simplification because domain knowledge can not be assumed from participants. Lastly, in Functional-grounded experiments, no human participation is required. Instead, the explainability of the system in question is assessed through various metrics that can be computed. On the one hand, this approach is the least costly in terms of human resources needed for the evaluation, but on the other, it can come at the risk of providing low validity. The reason for this is that the computed metrics might not be able to assess the true explainability of the system. Real explainability can only be assessed by humans (see Section 2.1), and computed metrics are assumed proxies that can be ill-defined. However, while the explainability assessment of the system itself is difficult with functional-grounded experiments, evaluating and comparing explanations E provided by a means M is still possible.

To quantify the fulfillment of an Explanation requirement, the authors Bersani et al. [16] propose the following formula:

$$Q_E(M) \geq \epsilon \tag{3}$$

The means M of an explanation E is evaluated by a quality function Q_E and must be greater or equal to a predefined threshold $\epsilon \in \mathbb{R}_+$ to fulfill the Explanation requirement. It is important to note that the function Q_E is dependent on the context C, the stakeholder group G, and the targeted *level of explainability readiness* as presented in Section 2.1.1. As an example instantiation for Q_E, a 5-point Likert Scale is suggested. If multiple explanations can be generated and evaluated, they propose calculating the variance σ and restricting it by a certain threshold. The proposed formula notation style will allow us to further specify the Explanation requirements in Chapter 4.

A number of different metrics has been proposed to evaluate and compare different explanations [3, 21, 43, 83, 84, 86, 97, 102, 141, 162, 198][6]. These metrics are shown in the following Table 2.2. The metrics in Table 2.2 are meant as an overview rather than a complete, detailed list. For example, Robustness can be expressed through various sub-metrics [83]. However, as the literature points out (see Definition 2.1.1), Metrics can be deceptive.

Definition 2.1.1 (Quantitative fallacy) *"A [false] criterion of significance which assumes that facts are important in proportion to their susceptibility to quantification." [72].*

One particular troubling aspect of this thesis is the Rashomon Effect[7] (see Definition 2.1.2). A brief introduction is given below. The Rashomon Effect will be further discussed in more detail in Chapter 4.

Definition 2.1.2 (The Rashomon Effect) *"What I call the Rashomon Effect is that there is often a multitude of different descriptions [equations $f(x)$] in a class of functions giving about the same minimum error rate." [23].*

[6] See also `https://github.com/understandable-machine-intelligence-lab/Quantus`.
[7] Also known as *predictive multiplicity* in classification tasks [30, 87, 88].

Metric(s)	Description
Robustness / Stability	Measures the degree to which an explanation is stable when subject to slight perturbations of the input, assuming the output approximately stays the same.
Computational Cost	Measures the cost to generate the explanation.
Faithfulness / Fidelity / Completeness	Measures the degree to which an explanation follows the prediction behavior of the model.
Localization	Tests if the explainable evidence is centered around a region of interest.
Complexity	Measures the complexity of the explanation or model.
Randomization	Measures to what extent the explanations deteriorate as model parameters are increasingly randomized.
Understandability	Measures the degree to which an explanation is understandable.
Consistency / Identity	Measures the degree to which two similar model instances differ in their explanation.
Separability	Measures the degree to which two nonidentical model instances differ in their explanation.
Monotonicity	Measures the monotonically increase in model performance by incrementally adding features according to their importance.
ROAR	Measures the accurarcy drop when retraining the ML model with the most relevant features removed.
Top-k Feature (dis)-agreement	Measures the number of agreed upon most (un)-important top k features.
Correlation coefficient	Measures the correlation between different metrics.
Prediction Gap on Important feature metric (PGI)	Measures the prediction faithfulness by calculating the average error of the prediction with only the k most important features.
Local Lipschitz continuity	Measures the stability of local explanations by comparing local explanations at a data point of interest against each other.
Cohen's Kappa Coefficient (K)	Measures the level of agreement between two annotators on a classification problem [42].
Kendall Rank Correlation Coefficient (τ)	Measures the ranking correlation of two independent rankings [105].
Spearman's Rank Correlation Coefficient (ρ)	Measures the ranking correlation of two independent ranking by how well they follow a monotonic relationship [171].
Weighted Cosine Similarity	Measures the similarity between vectors accounting for randomness of unimportant feature rankings [151].
Structural Similarity Index	Measures the similarity between images [185].
Normalized Mutual Information	Measures the correlation between two images [175].

Table 2.2.: Metrics for Explanations.

First formalized for predictive models in 2001 [23], the Rashomon Effect states that many models may exist for a given data set with equally well-performing but different internal solution strategies [139]. This effect directly affects Explainable Artificial Intelligence (XAI) methods because the explanations itself can differ under it. At the same time, the ML models are seemingly unchanged — at least based on performance metrics like accuracy — giving the user a sense of false security. The Rashomon Effect becomes especially troubling when users are involved because stability regarding explanations is considered mandatory to establish trust [139]. At worst, the Rashomon Effect can lead to explanations that contradict

each other [136]. The set of ML models affected by the Rashomon Effect can be defined as the Rashomon Set (see Definition 2.1.3). The Rashomon Set's size depends on ϵ_R, which controls which ML models are considered equally well-performing.

Definition 2.1.3 (Rashomon Set) *For a given ML model $f_R \in F$, where F denotes the Hypothesis space, $\mathcal{L}$ a loss function, the Rashomon parameter $\epsilon_R > 0$. The Rashomon Set can be defined as: $R_{\mathcal{L},\epsilon_R} = \{f \in F \mid \mathbb{E}[\mathcal{L}(f)] \leq \mathbb{E}[\mathcal{L}(f_R)] + \epsilon_R\}$ [30].*

Calculating the Rashomon Set is an NP-hard problem and computationally infeasible for non-complex ML tasks [41, 56, 108, 121, 189]. Additionally, one can define the following three metrics to measure the severity of the Rashomon Effect [30].

$$\alpha_{\epsilon_R}(f_R) = \frac{1}{n}\sum_{i=1}^{n} \max_{f \in R_{\mathcal{L},\epsilon_R}(f_R)} \mathbb{1}[f(x_i \neq f_R(x_i))]$$

$$\delta_{\epsilon_R}(f_R) = \max_{f \in R_{\mathcal{L},\epsilon_R}(f_R)} \frac{1}{n}\sum_{i=1}^{n} \mathbb{1}[f(x_i) \neq f_R(x_i)]$$

$$\gamma_{\epsilon_R}(f_R) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{|R_{\mathcal{L},\epsilon_R}(f_R)|} \sum_{f \in R_{\mathcal{L},\epsilon_R}(f_R)} \mathbb{1}[f(x_i) \neq f_R(x_i)]$$

α_{ϵ_R} is called the Ambiguity metric and measures the existence of conflicting predictions produced by the functions in a Rashomon Set, compared to the reference function f_R. The predictions are drawn from n observations in the input space. In the same manner, one can define Discrepancy δ_{ϵ_R}, which measures the maximum ratio of conflicting predictions that arise from comparing the reference function to functions in the Rashomon Set, again sampled by n observations. Lastly, one can define Obscurity γ_{ϵ_R}, which measures the average ratios of conflicting prediction between the reference function and functions in the Rashomon Set [30].

2.2. Federated Learning

As stated in Chapter 1, FL is an Artificial Intelligence (AI) paradigm proposed by McMahan et al. in 2016 to train an ML model in a decentralized fashion [132]. Since then, research regarding FL gained much attraction, especially in the medical field, as a means to train ML models by multiple participants without sharing local – and potentially sensitive – data (in contrast to training a centralized ML model that relies on the data availability of all participants for training). So, instead of moving the data to the computation, the computation is moved to the data. A key distinguishing factor from distributed optimization in traditional machine learning is the high degree of system and statistical heterogeneity in FL systems [132].

2.2.1. Process

The general algorithm regarding any FL architecture can be summarized as seen in Algorithm 1.

Algorithm 1 Generalized FL algorithm

1: $\Phi^0 \leftarrow$ initialize global ML model
2: $i \leftarrow 0$
3: **while** Φ^i is not converged **do**
4: $\quad \Gamma_k^i \leftarrow$ send global ML model Φ^i to k local nodes
5: $\quad \Gamma_k^{i+1} \leftarrow$ train local ML model Γ_k^i on local dataset D_k
6: $\quad \Phi^{i+1} \leftarrow$ aggregate ML model updates Γ_k^{i+1} from each client k
7: $\quad i \leftarrow i + 1$
8: **end while**

After the initialization of a global ML model Φ, Φ is sent to every participating client k of the federation. Each client k then trains a local ML model Γ_k on their respective data D_k. Then, each client sends their new ML model updates back, and an aggregation operation is performed to combine the different ML model updates to calculate a new global ML model. This procedure repeats itself till the global ML model Φ converges. Figure 2.2 visualizes the Algorithm.

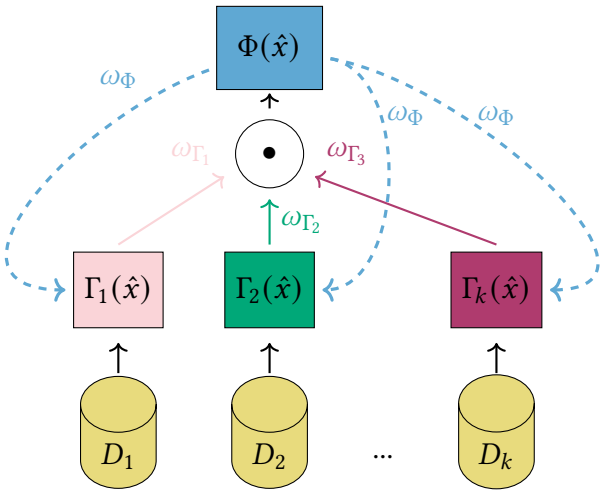


Figure 2.2.: Visualization of the fl Algorithm 1.

2.2.2. Algorithms

Since [132]'s initial proposal with the Federated Averaging (FedAvg) algorithm, the research literature has produced hundreds of different FL algorithms, often adapted for specific tasks or desirable properties. For this thesis, however, we will limit the scope of FL algorithms to a

selection of implementations already provided by the FL framework Flower in version 1.13.0 (see Table 2.3). Additionally, in the Appendix Figure A.3 a sequence diagramm showing the execution of FL strategys in Flower is shown.

Name	Reference
FedSGD	[132]
FedAvg	[132]
FedProx	[123]
FedOpt	[153]
FedAvgM	[89]
Krum	[19]
FedTrimmedAvg	[192]
FedMedian	[17]

Table 2.3.: List of Selected FL Algorithms in Flower Version 1.14.0.

In Table 2.3 Federated Stochastic Gradient Descent (FedSGD), FedAvg, and FedProx are the three most prominent FL algorithms, most often used as baselines algorithms. For this reason, a summary will be given below. For the other algorithms, readers are directed to the referenced papers or the code implementation provided by the Flower FL framework [17].

Algorithm 2 FedSGD [132]

Input: Number of participants K indexed by k, learning rate η, number of federated rounds T, number of local data points n_k by participant k, number of total data points n, local model $\Gamma_k(\omega)$ of each client k taking model weights ω

1: **initialize** ω_0 randomly
2: **for** $t = 0$ to T **do**
3: **for all** participants $k \in \{1, \ldots, K\}$ **do**
4: $g_k \leftarrow \nabla \Gamma_k(\omega_t)$
5: **end for**
6: $\omega_{t+1} \leftarrow \omega_t - \eta \sum_{k=1}^{K} \frac{n_k}{n} g_k$
7: **end for**

The FedSGD algorithm shown in Algorithm 2 is the most basic algorithm for mapping the Stochastic Gradient Descent (SDG) algorithm into a FL setting with FL clients and a central server for coordination. First, a random parameterization for the model weights ω_0 is chosen by the central server. Then, for the number of global FL rounds T, the local gradient is computed for each participant's local model $\Gamma_k(\omega_t)$ with the current model weights ω_t of round t. At the end of each federated round t, the server calculates the new weights ω_{t+1} by stochastic gradient descent with learning rate η over the weighted sum of local gradients

g_k. The linear factors of the weighted sum are the number of local data points n_k seen by participant k divided by the total number of data points n of all participants.

While FedSGD does work, it produces a high communication overhead because each client computes only one step of the gradient descent locally before the central server performs the aggregation. By introducing three new parameters: C, the fraction of clients that perform computation on each federated round t, E, the number of local computation rounds; and B, the local minibatch size used for the client updates, the authors McMahan et al. developed the FedAvg algorithm [132].

Algorithm 3 FedAvg [132]

Input: Number of participants K indexed by k, fraction of number of clients C, learning rate η, number of federated rounds T, number of local data points n_k by participant k, number of local epochs E, mini-batch size B, local data sets D_k for client k

function ServerUpdate
  initialize ω_0 randomly
  for $t = 0$ to T **do**
    $m \leftarrow \max(C \cdot K, 1)$
    $S_t \leftarrow$ {|m| clients randomly chosen}
    for all participants $k \in S_t$ **do**
      $\omega_{t+1}^k \leftarrow$ ClientUpdate(k, w_t)
    end for
    $m_t \leftarrow \sum_{k \in S_t} n_k$
    $\omega_{t+1} \leftarrow \sum_{k \in S_t} \frac{n_k}{m_t} \omega_{t+1}^k$
  end for
end function

function ClientUpdate(k, w_t)
  $B \leftarrow$ {split D_k into batches of size B}
  for local epoch $i = 0$ to E **do**
    for batch $b \in B$ **do**
      $\omega \leftarrow \omega - \eta \nabla \mathcal{L}(\omega, b)$
    end for
  end for
  return ω
end function

The FedAvg algorithm is shown in Algorithm 3. Here, we can see that each client performs multiple local gradient descent steps in the function ClientUpdate(k, w_t) over mini-batches of size B. $\mathcal{L}$ is the loss function for the local model prediction. It should be noted that by setting $E = 1$ and B $= \infty$, FedAvg equals FedSGD. While FedAvg reduces communication overhead, the convergence on non-independent and identically distributed (non-IID) data is much slower than FedSGD [199]. The reason for this is that the locally computed model weights in ClientUpdate(k, w_t) will diverge more on non-IID data the more local epochs

are performed. This leads to a slower convergence of the averaging calculation performed by the central server (see Line 10).

Algorithm 4 FedProx [123]

Input: Number of participants K indexed by k, number of federated rounds T, p_k probability to choose client k, μ scaling factor for proximal term, γ_k^t measurement how much local computation is performed by client k in round t

1: **function** ServerUpdate
2: **initialize** ω_0 randomly
3: **for** $t = 0$ to T **do**
4: $S_t \leftarrow \{|m|$ clients randomly chosen with probability $p_k\}$
5: **for all** participants $k \in S_t$ **do**
6: $\omega_{t+1}^k \leftarrow$ ClientUpdate(k, w_t)
7: **end for**
8: $\omega_{t+1} \leftarrow \frac{1}{K} \sum_{k \in S_t} \omega_{t+1}^k$
9: **end for**
10: **end function**

11: **function** ClientUpdate(k, w_t)
12: $\omega \leftarrow$ Find γ_k^t-inexact solution (see Definition 2.2.1)
13: **return** ω
14: **end function**

As an enhancement of the FedAvg algorithm in a more realistic FL setting with systems and statistical heterogeneity applied, the authors Li et al. introduced the FedProx algorithm [123]. The core idea of the FedProx algorithm is to allow participants to perform variable amounts of work locally across devices, via an added proximal term. This approach allows for more robustness and stability – regarding the convergence – than FedAvg in heterogeneous federated networks.

Definition 2.2.1 (γ_k^t-inexact solution) *Let $h_k(\omega, \omega_t) = \mathcal{L}_k(\omega) + \frac{\mu}{2}\|\omega - \omega_t\|^2$, and $\gamma_k^t \in [0, 1]$ where $\mathcal{L}_k$ is the local loss function of client k. Then ω^* is a γ_k^t-inexact solution for client k at round t for $\min_\omega h_k(\omega, \omega_t)$ if $\left\|\nabla h_k(\omega, w_t)\right\| \leq \gamma_k^t \left\|\nabla h_k(\omega_t, \omega t)\right\|$. Where $\nabla h_k(\omega, \omega_t) = \nabla \mathcal{L}_k(\omega) + \mu(\omega - \omega_t)$ [123].*

FedProx is shown in Algorithm 4. Notice the usage of the γ_k^t-inexact solution in Line 12. By defining the local objective of client k as $h_k(\omega, \omega_t) = \mathcal{L}(\omega) + \frac{\mu}{2}\|\omega - \omega_t\|^2$, the authors added the proximal term mentioned above that controls how much local work is performed by each client. The additional parameter μ acts as a penalization constant which can be tuned to prevent divergence and improve stability [123]. In that sense, the local epoch E of the FedAvg algorithm has been re-parameterized and made variable via γ_k^t (variable epochs) and μ (variable update step size concerning the global model). Note that a smaller γ_k^t corresponds to a higher accuracy regarding the local solution and more local computation.

Choosing the correct parameter for μ is difficult because a large μ forces the updates to be closer to the weights of the global model which can slow down convergence, while a small μ can lead to convergence problems [123]. Therefore, μ is designed to be chosen adaptively based on the model's current performance.

2.2.3. Architectures

As of the state of writing, no agreed-upon overview of different FL architectures exists. The authors of [180] present the most common ones found in the research literature. They categorize the different architectures according to the data distribution, scalability, and coordination mechanisms. One architecture over the other is selected according to requirements or challenges related to data distribution, communication (e.g., volume), and coordination. Each architecture shown here can utilize FL algorithms, as presented in Section 2.2.2.

2.2.3.1. Data Distribution-based

As the name suggests, this type is concerned with the underlying distribution of data used for training. Three types are distinguished. In a Horizontal FL architecture, clients share a similar feature space but differ in the sample space. If the feature space differs but the sample space is the same, it is called a Vertical FL architecture. If both feature space and sample space differ, it is called a Federated Transfer Learning architecture[8].

2.2.3.2. Scale-Driven

Here, architectures are divided according to the number of clients participating in the federation. If only a limited number of clients participate with large data sets, the term Cross-Silo FL is used. On the other hand, if lots of clients with small data sets are participating, the term Cross-Device FL is used.

2.2.3.3. Coordination-based

These architecture types are divided by who coordinates the aggregation and ML model updates. The coordination can be done either centralized (one party is responsible), decentralized (no central authority exists, and participants coordinate themselves), or hierarchical (local aggregators form a hierarchical multi-layered system architecture).

[8] Because transfer learning mechanisms are applied [180].

2.2.4. Evaluation

Federated evaluation is a sub-discipline in FL that intends to assess the model at the client side without sharing the data [73]. This usually involves the computation of model evaluation metrics at the client side and aggregation at the server side. Federated evaluation is noted here because this thesis will use it to evaluate explainability metrics (see 2.1.2) on the client side with the FL framework Flower [17].

Table 2.4 shows a brief overview of metrics to evaluate the ML models [63, 65].

Metric	**Domain**
$R^2 = 1 - \frac{\sum_i (y_i - y_i^*)^2}{\sum_i (y_i - \overline{y})}$	[R]
$RMSE = \sqrt{\frac{\sum_i (y_i - y^*)^2}{N}}$	[R]
$QE = median(\lvert\frac{y_i - y_i^*}{y_i}\rvert)$	[R]
$MAE = \frac{\sum_i \lvert y_i - y_i^*\rvert}{N}$	[R]
$MSE = \frac{\sum_i (y_i - y_i^*)^2}{N}$	[R]
$MAPE = \frac{\sum_i \lvert y_i - y_i^*\rvert}{N}$	[R]
$F1 - Score = 2 * \frac{Precision * Recall}{Precision + Recall}$	[C]
$Precision = \frac{TP}{TP+FP}$	[C]
$Recall = \frac{TP}{TP+FN}$	[C]
$Accuracy = \frac{TP+TN}{TP+FP+TN+FN}$	[C]
$AUROC$ = Area under the ROC Curve	[C]

Table 2.4.: Metrics for Evaluating ML Models. *Legend: [R] Regression, [C] Classification.*

For clarification, y_i denotes the prediction of the ML model of feature i while y_i^* denotes the ground truth of feature i. Furthermore, N denotes the total number of samples in the validation data set. TP, TN, FP, and FN stands for *True positive, True negative, False positive, and False negative*, respectively. *True/False* answers whether a class has been correctly classified, while the later part specifies the class. These concepts are related to Confusion Matrices.

2.2.5. Challenges

FL is still a relatively recent concept. Therefore, some challenges still need to be solved [122]. These are:

- **Expensive Communication:** Model update information needs to be transferred in each round of the FL algorithm. This can be impractical in settings where low-power Internet of Things (IoT) devices are used or network resources are limited.
- **System heterogeneity:** It is difficult to account for the heterogeneity of systems participating in the federation. This aspect is not only limited to the computational power of the participant but also includes other aspects like storage, communication capabilities, electric power usage, availability, or trustworthiness. There are also challenges regarding straggler mitigation and fault tolerance [123].
- **Statistical heterogeneity:** The differences in each client's local data set can vary widely. This problem is also known as non-IID data and is especially difficult in FL because it leads to the weights diverging too much from each other so that the global model can not converge [196].
- **Privacy Concerns:** This challenge relates to the possibility of other participants manipulating the global model (e.g., by poisoning attacks) or extracting sensitive information.

2.3. Explainable Artificial Intelligence (XAI)

Given the success of ever-more-complex ML models — even surpassing human abilities — it is unsurprising that a strong push exists to deploy these ML models in sensitive contexts where trust-related problems occur. To tackle these problems, a new branch of research has emerged that focuses on the interpretability and explainability of AI, namely XAI. While "interpretability" and "explainability" are often used synonymously, there is a solid case to be made to distinguish these two.

- **Interpretability:** This is the concern to which a human can understand the reason for a decision by simple observation [146]. In that sense, "interpretability" is a passive attribute.
- **Explainability:** On the other hand — as stated in Section 2.1 — can be seen as "the currency in which beliefs are exchanged" [146]. This definition matches the "enabling understanding" concept presented in the abovementioned section. In that sense, it is an active element of the system.

While XAI methods can be taxonomized in different ways, this proposal uses the taxonomy from Speith [172]. For this thesis, only post-hoc (after the model is trained) model-agnostic (applicable for every ML model) methods are relevant. Most prominently known are Local Interpretable Model Agnostic Explanation (LIME) and SHapley Additive exPlanations

(SHAP) [146, 154]. These two methods will be briefly described in the following. Additionally, explainability methods are divided into local and global explanations (see Definition 2.3.1 and 2.3.2)[9]. While local explanations generate explanations for individual instances of the ML model, global explanations reason about the ML model holistically. However, global explanations are usually more computationally expensive (see Subsection 2.3.2).

Definition 2.3.1 (Local Explanation) *Let $\vec{x} \in D$ be a concrete instance of a data point for an ML model $f : D \rightarrow Y$ and $f(\vec{x}) = \vec{y}$. Further let $vic_{\vec{x},\epsilon_D}(D) \in D$ be defined as the confined area around $\vec{x}$ limited with parameter ϵ_x and $vic_{\vec{y},\epsilon_Y}(Y)$ the confined area around $\vec{y}$ limited by ϵ_Y accordingly. A local explanation E_{Local} is then defined as a valid Explanation as stated in Definition 1 of Section 2.1 where the explanandum X is additionally locally confined such that only $f(\vec{x} + \vec{\delta_x}) = \vec{y} + \vec{\delta_y}$ with $\vec{\delta_x} \in vic_{\vec{x},\epsilon_D}(D)$ and $\vec{\delta_y} \in vic_{\vec{y},\epsilon_Y}(Y)$ are considered for the explanation.*

Definition 2.3.2 (Global Explanation) *A global explanation can be defined in concordance with local explanations. However, any global explanation must consider all (or at least approximate all) possible $\vec{\delta_x} \in vic_{\vec{x},\epsilon_D}(D)$ and $\vec{\delta_y} \in vic_{\vec{y},\epsilon_Y}(Y)$ where $\epsilon_D, \epsilon_Y \rightarrow \infty$. This way, the local confinement of the explanandum X is lifted, and a holistic approach enforced.*

2.3.1. Local Interpretable Model Agnostic Explanation (LIME)

LIME is an algorithm that tries to fit a local interpretable ML model — a surrogate – around a data point of interest. The fitted ML model shall approximate the predictions of the original ML model in that local area. LIME first creates a neighborhood of synthetic samples around the data point in question via perturbing values in the feature vector of that data point. Then, these synthetic samples are weighted via a weighting kernel, which measures the distance to the original data point. Finally, the synthetic samples are fed into the original ML model to generate output values, thus having everything needed to fit a local surrogate model [146, 154].

$$\xi(x) = \underset{g \in G}{\operatorname{argmin}} \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

Formally speaking (see equation 1), let x be the data point in question, f the original ML model, G a set of potentially interpretable models, $\mathcal{L}$ be the approximation error of the fitted model g, and π_x the weighted kernel centered around x. LIME then tries to fit a model g from G that minimizes the approximation error $\mathcal{L}$ and a complexity penalization term $\Omega(g)$. The local interpretable model g can then be used for local explanations.

[9] Some scholars argue for an additional middle ground called *Cohort Explanations* [134].

2.3.2. SHapley Additive exPlanations (SHAP)

While LIME computes only local explanations, SHAP can compute both local and global explanations. The general concept is derived from game theory, where Shapley values[10] are a way to distribute the total gains of a cooperative game fairly among players according to their contribution. Formally speaking, a Shapley value is the average value of the marginal contribution of a player over all possible coalitions. This theory inspired the authors in [128] to use this concept to produce a new XAI method called SHAP. In their paper, Shapley values are used to explain feature attribution. Feature attribution measures how much a specific feature contributes to the resulting prediction.

$$g(z') = \phi_0 + \sum_{i=1}^{M} \phi_i z'_i \tag{2}$$

Equation (2) shows the gist of this concept. Here $z' \in \{0,1\}^M$ is the so-called simplified input feature vector (whether a feature is present or not), ϕ_i is the Shapley value for the feature i. This sum shall be approximately equal to the predicted value of the original ML model. Therefore, the original ML model prediction can be expressed as the individual contribution of each feature. However, in practice, the calculation of every possible coalition (2^n, where n is the number of features) is computationally costly; therefore, the Shapley Values are usually approximated [176]. For a global explanation, the average value from all Shapley Values of all data instances are being used/approximated. Equation (3) shows the formula for the calculation of the i-th Shapley value ϕ_i which is divided into three parts.

$$\phi_i = \underbrace{\frac{1}{|N|!} \sum_{S \subseteq N \setminus i}}_{\text{average}} \Big(\underbrace{|S|! * (|N| - |S| - 1)!}_{\text{weight}} * \underbrace{[v(S \cup i) - v(S)]}_{\text{marginal contribution}} \Big) \tag{3}$$

The first part of the equation is the sum of all possible coalitions without the i-th feature[11] present divided by the number of all possible coalitions ($|N|!$). The weight term in the summation is the product of the number of permutations without the i-th feature present times the number of permutations of the complement thereof ($N \setminus \{S \cup i\}$). The last part is taking the difference of the value of the coalition with the i-th feature present $v(S \cup i)$ and the coalition without it $v(S)$.

Figure 2.3 shows an example of the calculation of the Shapley values [25, 94, 106]. The feature vector consists of three features {●, ●, ●}. Furthermore, all possible coalitions with

[10] The theory of Shapley values contributed to †Lloyd S. Shapley winning 2012 the Nobel Prize in Economy.
[11] Also called player.

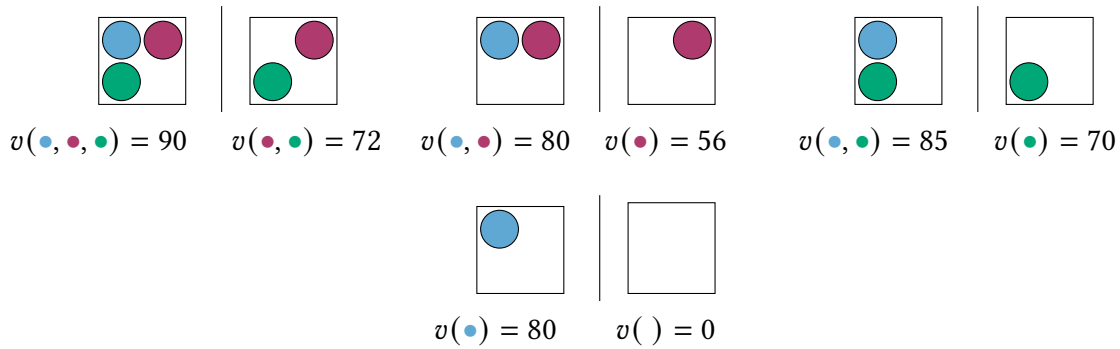


Figure 2.3.: Example Calculation of Shapley Values.

and without the feature vector ● are depicted in the Figure. Therefore, we can calculate the Shapley value $\phi_{●}$ of said feature ● with Equation (4).

$$\phi_{●} = \frac{1}{|\{●, ●, ●\}|!} \sum_{S \subseteq \{●,●,●\} \setminus \{●\}} \Big(|S|! * (|\{●, ●, ●\}| - |S| - 1)! * [v(S \cup \{●\}) - v(S)] \Big) \tag{4}$$

$$\begin{aligned}
\phi_{●} = \frac{1}{6} \Bigg(& \Big(|\{\ \}|! * (|\{●, ●, ●\}| - |\{\ \}| - 1)! * [v(\{●\}) - v(\{\ \})] \Big) \\
& + \Big(|\{●\}|! * (|\{●, ●, ●\}| - |\{●\}| - 1)! * [v(\{●, ●\}) - v(\{●\})] \Big) \\
& + \Big(|\{●\}|! * (|\{●, ●, ●\}| - |\{●\}| - 1)! * [v(\{●, ●\}) - v(\{●\})] \Big) \\
& + \Big(|\{●, ●\}|! * (|\{●, ●, ●\}| - |\{●, ●\}| - 1)! * [v(\{●, ●, ●\}) - v(\{●, ●\})] \Big) \Bigg) \\
\phi_{●} = \frac{1}{6} \Big(& \Big(2 * [80 - 0] \Big) + \Big(1 * [80 - 56] \Big) + \Big(1 * [85 - 70] \Big) + \Big(2 * [90 - 72] \Big) \Big) \\
= 39.1\overline{6}
\end{aligned}$$

The value $39.1\overline{6}$ is the average marginal contribution of the feature ●. For completeness of the example, $\phi_{●} = 20.67$ and $\phi_{●} = 30.17$ respectively. Notice that the sum of the Shapley values equals $v(●, ●, ●) = 90$. However, one aspect that is still not clear is how to omit a feature in the context of ML. For this, the authors [128] used the so-called Shapley Kernel. This approach requires the definition of a background set B with representative data points. Then, the omitted feature is filled in with values of said representative data set while the other features stay fixed. This way, synthetic samples are generated. At last, these synthetic samples are fed into the ML model, and the average over all output values is generated. The Shapley Kernel was the first method that has been proposed to approximate the value function. However, there are many other methods available now for the efficient computation of the value function[12].

The Shapley value suffices the following properties if computed exactly [25, 146, 176]:

[12] https://shap.readthedocs.io/en/latest/api.html

- **Efficiency:** All contributions are fairly redistributed among all features (no more, and no less); see Equation (2).
- **Symmetry:** If two features contributed the same amount to all coalitions, they must receive the same contribution.
- **Linearity:** If the value function v can be presented as the sum of two distinct value functions g, h. Then, the Shapley values of the value function v are equal to the sum of the Shapley values of g and h.
- **Missingness:** If a feature i does not contribute to any possible coalition, its Shapley value must be $\phi_i = 0$.
- **Consistency:** The value of a Shapley value ϕ_i can only increase if the value of i increases, while the other values are fixed. This property implies monotonicity, e.g., ϕ_i increases if i increases.
- **Affine Scale Invariance:** The zero point of a feature and the units thereof do not determine their contribution if the attribution (evaluation of the value function v) is invariant by simultaneous affine transformation. If the ML model output does not change when a feature is measured in meters or inches, then the Shapley value of said feature must also stay the same.

2.3.3. Saliency Map

Saliency maps are one of the simplest forms of calculating the attribution that a pixel has on the prediction of a ML model has. The idea behind saliency maps is: Given an image I with $m * n$ pixels and a particular class C, to calculate through backpropagation the derivative – similar to fitting the model – of a weight vector ω with respect to the given class C and image I [168]. Note that the actual class of the image can differ from class C. The value of the saliency map at position (i, j) is then given as $M_{i,j} = |\omega_{h(i,j)}|$ where $h(i, j)$ is a mapping function that points to the weight value that corresponds to the pixel at position (i, j) of the input image I. In a multi-channel image, e.g., RGB, the channel with the highest weight value gets selected.

2.3.4. Input X Gradient

This is also a straightforward method and is considered an improvement to saliency maps by the authors Shrikumar et al. [166]. As the name suggests, the attribution gets calculated by computing the gradient via backpropagation and then simply multiplying the input image with it. The idea is that the gradient gives a sense of sensitivity because higher gradient values at a particular input position indicates a higher relevance in the output.

2.3.5. Integrated Gradients (IG)

The core idea is the calculation of the integral of gradients along the path of a baseline value x' to the input x [177]. The concrete calculation – an approximation of the underlying integral with m steps – is given in Equation (4).

$$IG_i(x) = (x_i - x_i') \times \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m} \times (x - x'))}{\partial x_i} \times \frac{1}{m} \tag{4}$$

The baseline x' is usually dependent on the given task. For example, the most common baseline value in image classification is the black image because the input images are usually normalized beforehand. Implementation-wise, *IG* is gradient calculation in a for loop for the i-th dimension. Additionally, *IG* is proven to suffice the *implementation invariance* axiom, which states that if two networks produce the same output for all inputs, even with different implementations, then the attribution method – here *IG* – shall produce the same result. Furthermore, it suffices *sensitivity* in the sense that if input and baseline differ in only one feature and produce different predictions, then the attribution of said feature can not be zero [177].

2.3.6. Grad-CAM

Gradient-weighted Class Activation Mapping (Grad-CAM) is another technique to visualize explanations for deep neural networks [164]. It is a generalization of the CAM algorithm proposed in [197] that only applies to Convolutional Neural Network (CNN) architectures that do not contain fully connected layers. Grad-CAM is also discriminative with respect to a target class C, similar to Saliency maps. Since authors' initial publication, many derivative algorithms with additional properties have been proposed and implemented – most notably HiResCAM and ScoreCAM[13].

$$\text{Grad-CAM}^C = ReLU\left(\sum_k \left(\underbrace{\frac{1}{Z}\sum_i\sum_j}_{\text{global average}} * \underbrace{\frac{\partial y^C}{\partial A^k_{i,j}}}_{\text{gradient}}\right) * A^k\right) \tag{5}$$

Equation (5) shows the classical Grad-CAM algorithm for a given class C. Here, A^k are the activation values of a specific convolutional layer k for a given image I. The activation values of the layer k are then weighted with a global average pooling over the gradient values at any given position (i, j) of the activation layer k – therefore, Z equals $i * j$. This

[13] https://github.com/jacobgil/pytorch-grad-cam

process is then repeated with other convolutional layers in the network, and the sum of the values is fed into the *ReLU* function. In practice, Grad-CAM works well. However, there are some reported cases where Grad-CAM does not correctly highlight the relevant parts of a picture [58].

3. Related Work

Research in explainability as a non-functional requirement suffers because scholars often need to recognize it explicitly. Instead, papers often focus solely on algorithmic considerations or very high level contemplation. Both approaches are undoubtedly valid. However, they lack a pragmatic and holistic view of the subject, which will be elaborated on in the following.

The most notable contributions that study explainability as a non-function requirement are presented by the authors Deters, Speith, Köhl, and Chazette. In the papers [34, 35], surveys were executed to determine what users consider advantages or disadvantages regarding explainability as a non-functional requirement. Their results show a "double-edged sword" effect: explainability can increase understanding and usability but also decrease them by providing unnecessary explanations or hindering usage. The authors Deters, Droste, and Schneider [51] present goal-oriented heuristics to assess if a software system fulfills explainability as a non-functional requirement. These heuristics are ten questions that shall be easy to use by software engineers and give them a "rule of thumbishness" to evaluate whether the requirement is fulfilled or not. This line of research study — assessing the explainability of a system — has recently been extended in [53] and [59]. In [53], they present a quality model for explainability based on a systematic literature review comprising ten aspects of explainability, 36 distinct criteria, and 35 associated metrics. Notably, the authors in [33] summarized 57 quality aspects related to explainability, which were also based on a systematic literature review. Both of them suggest the adoption of user-centered practices to develop explainable systems [36, 53]. This approach has been further studied in [59] via a user survey about explainability needs in everyday software. Regardless, while they exhaustively list aspects and criteria, they do not present a thorough and practical analysis thereof, making it hard to assess which criteria and metrics should be chosen and why. In conclusion, the papers above – and comparable work from the authors in that regard – are informing this thesis on a fundamental level. However, the *operationalizing* of explainability as a non-functional requirement remains still unclear. Especially regarding the unique constraints and considerations of the FL context that need to be accounted for.

Recently, researchers have tried to bridge the gap between different communities in the research of explainability (e.g., machine learning, human-computer interaction, etc.). For example, the authors Wang, Huang, and Yao proposed a roadmap with a guideline-like approach [187]. However, while the core questions are clear and refreshingly concrete in design, their application and methodology behind this approach are questionable. The problems arise primarily because of choosing the wrong abstraction level to tackle explainability, which contradicts what we mean in this thesis when discussing explainability. For example,

"How to explain?" is directly mapped via XAI methods to "What to explain?". However, as we shall see in the later part of the thesis, a mapping like this can only be possibly chosen arbitrarily. Furthermore, the question "When to explain?" is reduced to a mere timing in a project rather than a context.

The thesis has already utilized several basic building blocks of explainability, FL, and XAI from various papers in Chapter 2. Instead of iterating over these papers again, readers are directed to the chapter above. Most noteworthy is the Quantus Python Library [83] which provides metrics for the evaluating explanations. The recent empirical evaluation of the Rashomon Effect in the centralized training of ML models [139]. And lastly, the FL framework Flower [17].

One system that claims to be explainable by design is presented in [91]. Here, a provenance-based architecture is envisioned. Domain experts create templates that are automatically filled out by the provenance of data. These templates are then instantiated at run-time to produce explanations. While the architecture and mechanisms are interesting, they do not apply to generating explanations of decisions made by ML models because only the provenance of data can be queried to produce an explanation.

In the realm of FL, several algorithms have been proposed to tackle explainability in light of XAI. This new field of research is, amongst others, motivated by the authors of [14] and called Fed-XAI. However, most current research focuses on the algorithmic part and specific experiments. For example, the authors in [61] compare the explanations of a fuzzy rule-based ML model vs. SHAP-produced ones in a specific data set. Another paper is presented by the same author in [60], where they tackle the problem of needing a representative underlying data set for the SHAP explanation method in the context of FL. They propose to utilize a fuzzy clustering method in the FL architecture to generate artificial samples and produce a representative underlying data set. In the paper from [181], the authors compare the FL explanation provided by SHAP to that of a centralized ML model approach in the 6G network slicing classification field. While all of the papers mentioned in this paragraph have some degree of combination between FL and XAI in place, several aspects are not considered:

- They do not account for different FL architectures, ML scenarios, or the challenges described in 2.2.5.
- Comparisons are difficult because no common baseline is provided.
- The aspects of time, resource usage, trade-offs, and general practicability are omitted.
- The additional value the explanations provide is not further evaluated nor compared to each other.
- It is unclear how the Rashomon Effect affects explanations.

For the above reasons, it is still hard to assess FL under the constraint of fulfilling explainability as a non-functional requirement, which fosters the goals of this thesis and sets it apart from the research literature presented above.

4. Explainability and Federated Learning

In this Chapter, we will apply and experiment with FL in conjunction with XAI methods and metrics. The goal is to grasp how different changeable context parameters affect the explainability.

4.1. Goals and Questions (1)

We follow the Goal Question Metric (GQM) guideline described in [183] to establish a well-defined research approach with the following five evaluation goals for this Chapter:

- **EG1**: Examine the difference between XAI metrics applied to the local ML models and the global ML model.
  - **EG1.Q1** Are there measurable differences regarding XAI metrics?
  - **EG1.Q2**: Are there measurable differences regarding the XAI metrics calculation duration?
  - **EG1.Q3**: Are there measurable differences regarding different FL algorithms and data distributions?
- **EG2**: Examine the stability of different XAI methods.
  - **EG2.Q1**: Which XAI methods are more stable than others?
- **EG3**: Examine the improvement of XAI metrics by applying explanation optimization.
  - **EG3.Q1**: How does the explanation optimization compare to individual XAI methods?
  - **EG3.Q2**: How do different aggregation methods influence the explanation optimization?
  - **EG3.Q3**: Do our results align with those presented in the original paper?
  - **EG3.Q4**: How are cost and performance related to the explanation optimization?
  - **EG3.Q5**: Can the aggregation weights be reused for successive FL rounds?
- **EG4**: Examine the degradation of XAI metrics by applying Differential Privacy (DP).
  - **EG4.Q1**: How much do XAI metrics degrade by applying DP?

  - **EG4.Q2**: How much does DP affect our explanation optimization approach?
- **EG5**: Examine the degradation of XAI metrics by clients that misbehave.
  - **EG5.Q1**: How much do misbehaving FL clients affect XAI metrics?

Each evaluation goal is associated with an experiment in this Chapter (e.g., EG1 and Experiment 1). The metrics for the questions are presented and evaluated in their respective Section. Our GQM goals will be further extended in Chapter 5 when we conduct our user survey and Chapter 6 when we reason about explainability from a human-centric point of view.

4.2. Experiment Preliminaries

Based on two recent comparative analyses for FL frameworks [66, 155], Flower has been selected to implement the experiments [17]. At the time of writing, Flower has $\approx 5.1k$ github stars. Furthermore, Flower provides extensive documentation with examples and has an active development community. The most recent version to date is 1.13.1, released on the 27. November 2024. While Flower is open-source, it is maintained by the Flower Labs GmbH, which resides in Hamburg, Germany. Notably, Flower is ML framework agnostic, platform independent, and can scale to thousands of FL participants.

Internally, Flower uses the Python Library Ray for task scheduling and execution[1]. In a simulation, FL clients are created ephemerally. Ephemerally means that clients only materialize when needed to execute a task and destroyed afterward. This way, resources can be efficiently used. The execution tasks are controlled by Flower, which happens (i) Self-managed (orchestration is done by Flower), (ii) Batchable (execution happens in batches utilizing all available resources), and (iii) Resource-aware (allowing to define which resources are made available to the execution). Figure 4.1 shows the central Flower classes. In the `flwr` module, the `ServerApp` represents the central server in the FL context. `ServerApp` utilizes a concrete `Strategy` which implements several functions that will be called by Flower during simulation. The functions will be presented in the following, a sequence diagramm regarding the call sequence is given in the Appendix A.3.

- `initialize_parameters(...)`: Is called first, and responsible to initialize the global model that is going to be shared with the pariticpating FL clients.
- `configure_fit(...)`: Then creates instructions for the participating FL clients that will then be send to them. This function can be used to adapt client behavior (e.g., change number of local epochs executed by FL client).
- `aggregate_fit(...)`: Is called after the clients execute their `fit` method. This function can be used to aggregate metrics gathered directly after the local training of the FL clients. Because of the flexibility from Flower, it does not presume what metrics a

[1] `https://www.ray.io`

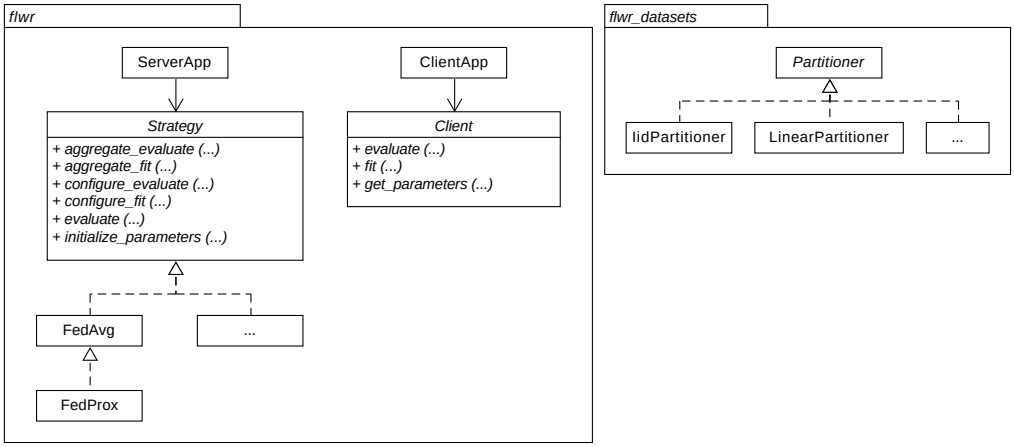


Figure 4.1.: Most Important Flower Classes.

FL client computes (except for the loss). Hence, the developer needs to provide the concrete implementation.

- `evaluate(...)`: Method that evaluates the global model.
- `configure_evaluate(...)`: Will be called before the FL clients start their evaluation. Can be used to adapt the client behavior for their local model evaluation.
- `aggregate_evaluate(...)`: Method that aggregates the results from FL clients after their local evaluation. Again, Flower does not presume any evaluation metrics, hence an implementation must be provided if one desires to aggregate evaluation results.

On the client side, the following functions must be implemented by a `Client`:

- `fit(...)`: Trains the local model and returns the new parameters, as well as performance metrics.
- `get_parameters(...)`: Implements the functionality to retrieve the local model parameters.
- `evaluate(...)`: Evaluates the local model against a validation sets and returns performance metrics.

It should be stressed out because of the ephemerally design of Flower, the `ServerApp` and `ClientApp` create and destroy their `Strategy` or respectively `Client` instances during the simulation. Hence, any form of state management needs to be conducted explicitly. Flower does provide the necessary means via configuration injection mechanisms which is more explained in detail in the official documentation. Lastly, we also have the `flwr_datasets` module that provides the implementation for managing federated data sets. These federated data sets are wrapped by a `Partitioner` which as the name suggests, partitions the dataset among the FL clients. This module provides several concrete classes like `IidPartitioner` which distributes the data evenly among the FL clients, or the `LinearPartitioner` which

distributes the data linearly according to the FL client `partition-id` which is assigned at the beginning of the simulation.

4.2.1. Datasets

We conducted all of our expirements on the following commonly used data set [26, 87, 120, 123, 127]:

- **CIFAR-10:** Contains (32×32×3) images of ten different classes. The data set is comprised of a total of 70,000 images. The task is to classify the images correctly [115].

4.2.2. Implementation

The implementation is provided under the MIT License via the following GitLab Repository: `https://gitlab.kit.edu/Nicolas.Schuler/fl`. Apart from the FL framework Flower[17], we used PyTorch[12], Captum[110], grad-cam[76], SHAP[128], and Quantus[11] for the evaluation of different metrics and attribution methods. For most of our experiments, we utilized the EfficientNetV2-S[179] provided by PyTorch, which was chosen because of the fairly good performance in the image classification task at hand and the rather small memory footprint and training time[179]. We only modified the model's last layer to match with the number of classes in the respective data sets. During the experiments, it became evident that the simulation engine provided by Flower does have problems regarding memory leakages for CPU RAM and GPU VRAM. Therefore, we reimplemented the simulation engine, which now supports two different backends. The first backend uses the ray library[2] and `ray tasks` to execute the FL clients. Based on our testing, this backend is faster and more memory efficient than Flower's simulation engine – which often crashed because of out-of-memory errors[3]. The second backend uses Python's standard `ProcessPool` executor. This implementation does not rely on ray and uses shared memory to exchange parameters between processes. It is not as fast as the ray backend but is more lightweight and generally has a lower continuous CPU RAM usage profile. The `process-pool` backend also supports a locking mechanism for the training process of FL clients so that only a certain number of FL clients can execute the training process in parallel. Locking is beneficial because the training of the ML model is usually the most GPU-intensive operation that will be performed. We also include a CLI interface – called `fl` – for running the simulations and the ability to dynamically change simulation parameters at the beginning of the simulation.

In terms of the performance of the FL simulation, we also noticed that parallelization is strongly limited by the FL loop (see Algorithm 1 in Section 2.2), which requires synchronization between the FL clients and the FL server. Additionally, any server-side-computation, e.g., aggregation or evaluation, will introduce an additional delay before the FL clients can

[2] `https://github.com/ray-project/ray`

[3] Some of our findings are also described in this pull request `https://github.com/adap/flower/pull/3989`.

proceed with computations. This behavior significantly reduced our experiments' throughput and parallelization capability, even with added computational resources. We can elevate the problem by selecting only a fraction of the FL clients for further task proceedings. However, this will inevitably lead to a decrease in model convergence because only a fraction of the information is available.

4.3. Experiment 1: Local or Global Model?

In the first series of experiments, we were interested in empirically establishing which ML model should be used for the explanation generation[4]. While in the contemporary research literature, there is a consensus that the global ML model performs better than the clients' ML model, this assumption has not necessarily proven to be the case regarding the performance of XAI methods. Therefore, we first had to verify that this was the case. In addition, during our research efforts on FL, we saw a strong emphasis being placed on the employed FL algorithms to achieve ever-better performance in terms of accuracy. However, it is unclear how much the FL algorithm affects XAI methods or if other factors are more crucial in producing "good" explanations. Lastly, having proven that only the global ML model is relevant in terms of XAI performance will have considerable implications in the practical application of XAI methods in conjunction with FL. Having the ability to offload the explanation-generating process to a usually more powerful FL server instances will reduce computational stress on the FL clients and facilitate the usage of parallelization patterns to XAI applications. Due to the immense computational overhead that the generation process for explanations as well as the evaluation of XAI metrics takes, we have only used one XAI method – Saliency Maps (see Subsection 2.3.3) – that is coincidently also one of the fastest methods available that works on the image classification task at hand.

To test our hypothesis, we tested different FL algorithms (see Table 2.3) with ten FL clients in combinations with different data partitioning schemes provided by the `flower-datasets`[5] library. The following partition schemes were evaluated: (i) independent and identically distributed (IID), (ii) Dirichlet ($\alpha = 0.1$) [90], (iii) linear, (iv) square, and (v) exponential. The partition an individual client gets is computed based on the respective `client-id` assigned at the beginning of the FL simulation. The ML model used was the above-mentioned modified EfficientNetV2-S on the CIFAR-10 data set. The concrete hyperparameters for the FL algorithms and the ML model are listed in the Appendix Table A.2. For a general sense of the model performance regarding explanation stability; we derived a new type of metric which is described in the following.

Definition 4.3.1 (Round-to-Round Metric) *One can compute a round-per-round metric for FL that either calculates a metric for the current FL round or, to the last FL round, effectively comparing the performance of one FL round with the targeted one. For our experiments,*

[4]Because of the volume of the data we gathered, only an excerpt will be shown. The experiment data and its analysis can be found in our GitLab repository.

[5]`https://github.com/adap/flower/tree/main/datasets`

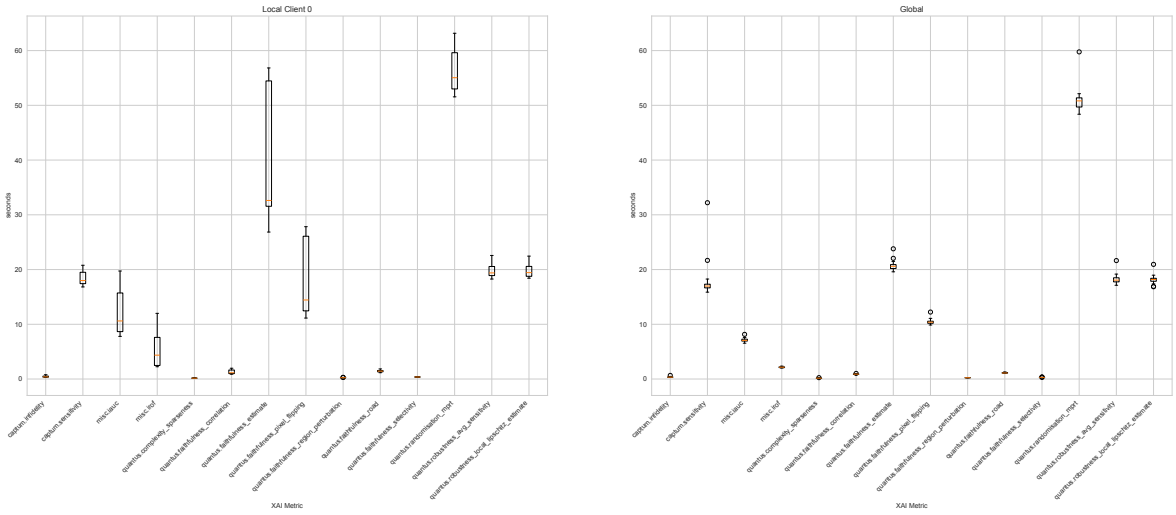
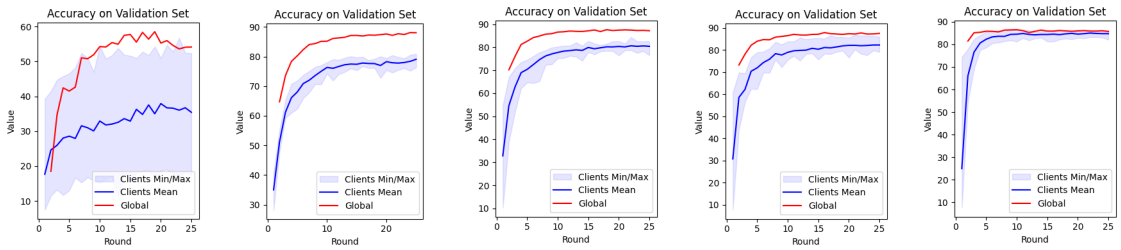


Figure 4.2.: Duration for Metric Calculations. *[Left] Client 0, [Right] Global*

we employed this technique to calculate metrics comparing the intra-performance of local and global model explanations between consecutive rounds and against the last round, and their inter-performance (e.g., comparing current local performance against current global performance). The metrics that we employed are prevalent, and can be easily computed and compared against each other: (i) Mean Squared Error (MSE), (ii) Normalized Mean Squared Error (NMSE), (iii) Structural Similarity Index Measure (SSIM) [185], (iv) Normalized Mutual Information (NMI) [175], (v) Wasserstein Distance, (vi) Peak Signal-to-Noise Ratio (PSNR), (vii) Universal Image Quality Index (UIQ) [186] (viii) Spectral Angle Mapper (SAM) [194], (ix) Signal to Reconstruction Error Ratio (SRE) [117], (x) Pearson's Correlation Coefficient (PEAR) [149], (xi) Spearman's Rank Correlation Coefficient (SPEAR) [171], (xii) Kendall's Rank Correlation Coefficient (TAU) [105], and (xiii) Cosine Similarity.

In addition to the round-to-round metrics, we also computed different XAI metrics which were selected based on their perceived usefulness at the beginning of our experiments. Figure 4.2 shows a boxplot of how long each metric takes to compute. We usually did not deviate from the default values of each metric, but for reproducibility the initialization of the metrics can be found in our above-mentioned GitLab repository.

Notably, some metrics on the client side took longer to compute than on the server side, and the spread is more significant. The reason is that FL clients work in parallel and, therefore, potentially compete against resources in the simulation. In contrast, the FL server execution happens in serial, according to Flowers' standard programming paradigm and API design. Also, FL clients have additional overhead due to parameter sharing between different processes over shared memory. Based on these results, we see that the Model Parameter Randomization (MPRT) metric [2] is the definitively most expensive one because each layer of the ML model will be randomized, and measurements will be taken. Then, six metrics fall in the region of $\approx 10 - 35$ sec, with the faithfulness estimate metric as the most significant [6]. Interestingly, nearly all these metrics are perturbation-based except the Insert Area Under Curve (IAUC) [156] metric, which iteratively adds pixels according



(a) Dirichlet $\alpha = 0.1$ **(b)** IID **(c)** Linear **(d)** Square **(e)** Exponential

Figure 4.3.: Showing the Influence of Data Partitioning on Accuracy (FedAvg).

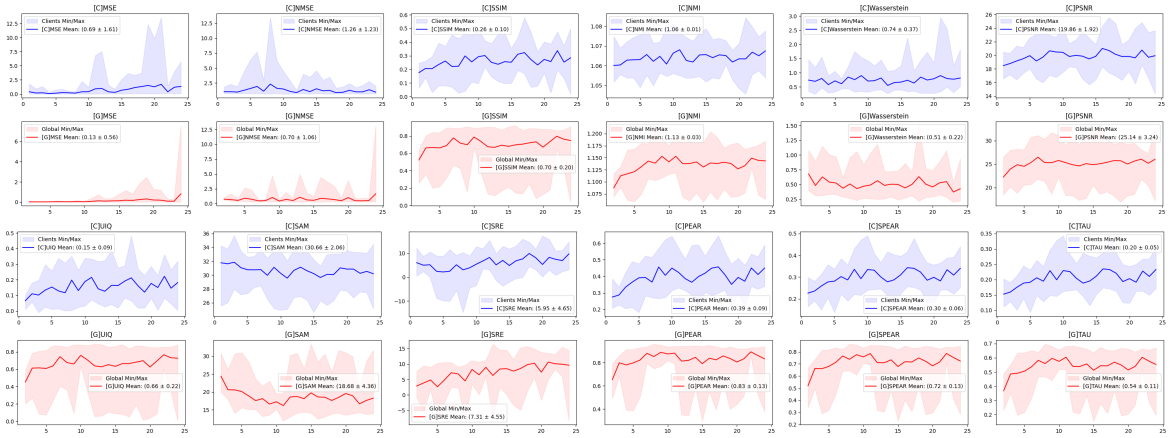


Figure 4.4.: Round-to-Round Metric comparing Client [red] and Global [blue] Attribution of Round T against Round $T-1$ (FedAvg/IID).

to a decreasing importance score of the explanation into a blank image. The rest of the metrics are near or under 5 sec. For reference, the explanation method takes about ≈ 1.3 sec to compute for ten images, the ML model training process takes about ≈ 5.7 sec, and model inference takes $\approx 5.6 - 5.7$ sec for ten thousand images. Therefore, we should expect that the gain of information with the knowledge of the metric should be at least as important as training the ML model. In all of our experiments, it is evident that the global ML model outperforms the respective local ML models in terms of accuracy, with one exception being the Krum algorithm [19]. Here, the local and global models seem very much aligned. However, changing the data partitioning shows clearly that the gap between these two greatly depends on the data distribution. Figure 4.3 is representative of other FL algorithms that we tested, and they all show a very similar pattern (except Krum). The more data one specific client gets, the more the client can learn and share with others, which in turn pulls the weights of the other clients closer to the one of the client with the most data at hand, and this leads to a diminishing gap between client ML model and global ML model.

Before we look at the XAI metrics occurring in Figure 4.2, we look at the round-to-round metrics of the attribution we defined above. From these round-to-round metrics in Figure 4.4, arguably only the correlation coefficients PEAR, SPEAR, and TAU show a recognizable deviation between the global and client model. This is also something that we can see in other experiments that we conducted. The SSIM metric is also recognizable as a useful

Correlation	Dirichlet	IID	Linear	Square	Exponential
SPEAR(M_C, M_G) ↑	0.554752	0.818788	0.925514	0.921067	0.88716

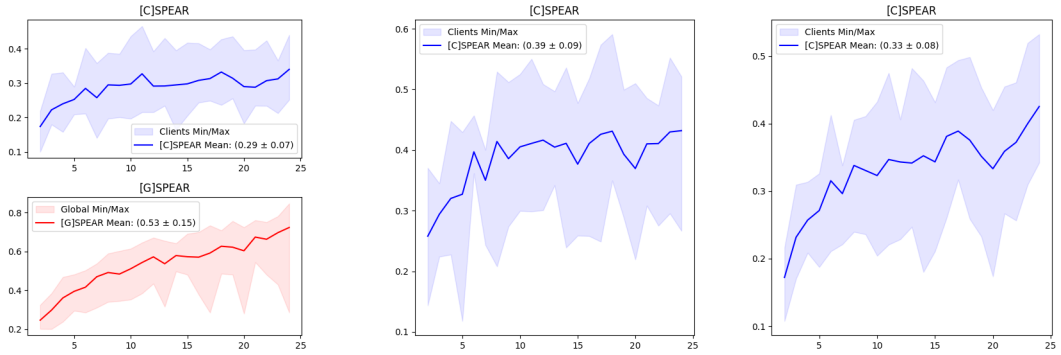
Table 4.1.: Comparing Correlation Between Metrics Taken from Local and Global Model Differing Algorithms.

Correlation	FedAvg	FedProx	FedAvgM	FedMedian	FedTrimmedAvg	Krum	FedOpt
SPEAR(M_C, M_G) ↑	0.585185	0.67037	0.692592	0.9407407	0.7851851	0.892593	0.751852

Table 4.2.: Comparing Correlation Between Metrics Taken from Local and Global Model Differing Data Partitioning.

indicator because it somewhat follows the curve of the correlation coefficients. MSE and NMSE are not useful because they are prone to spiky behavior, and NMI consistently showed a very close value for the client and the global model in our experiments. So, from now on, we will mostly rely on the SPEAR correlation coefficient because it shows monotonic relationships. In addition, we can compare client attribution against global attribution on a round-to-round basis, as shown in Figure 4.5. We notice that only the global model seems to converge to a high SPEAR correlation coefficient, indicating that the global model is the most stable one, at least on the remarks of our round-to-round metric, which adds nearly zero cost to the evaluation. However, this linear increase of the SPEAR correlation coefficient can not be observed in the case of the Dirichlet partitioning, which is the most interesting for FL. Instead, it seemingly platoos very early (Round 5) at a value approximately twice as high as the client model. We also looked at the mean and standard deviations of the attributions and noticed that the global model has lower values than the client models. For the XAI metrics, we prepared Table 4.3, which shows which model performed better according to each metric. The table indicates that the global model performs better in most cases. However, some of the values are very close to each other. We have also computed the correlation matrix of these metrics and compared global and local model against each other (see the last row). Also, we compared the correlation between having executed the same FL algorithm and having the same data partitioning scheme (see Table 4.1 and 4.2). The results indicate that the data partitioning scheme is of more significance since the correlation values are consistently higher in that case, indicating a monotonic relationship. One notable occurrence is that the Iterative Removal of Features (IROF) and IAUC metrics are often reversed. The way the metrics are constructed could indicate that if a model performs better in IROF, it is more focused because the important parts of the image get removed first. A better IAUC metric could indicate that the model relies more on background information because the latent activation on the blank image is higher.

To summarize our findings in this subsection: The global ML model performs better than the local ML model in nearly all cases, and the data partitioning has more influence on the XAI metrics than the executed FL algorithm. Furthermore, for development purposes it makes sense to test the IID, Dirichlet, and Square partitioning scheme because they can show distinct patterns on how the metrics evolve. Therefore, our research indicates that more improvement in terms of the FL algorithm can be gained by focusing on improving



(a) Comparing Against Respective Last Attribution. **(b)** Client Attribution Against Global Attribution. **(c)** Client Attribution Against Last Global Attribution.

Figure 4.5.: Comparison Against Different Attributions.

Metric	FedAvg	FedProx	FedAvgM	FedMedian	FedTrimmedAvg	Krum	FedOpt
Infidelity ↓	0.148/ 0.052	0.168/ 0.064	0.038/ 0.0	0.354/ 0.104	0.152/ 0.063	0.233/ 0.109	0.207/ 0.074
Sparseness ↑	0.39/ 0.4	0.38/ 0.385	0.415/ 0.428	0.388/ 0.392	0.39/ 0.396	0.393 /0.392	0.388/ 0.391
Faithfulness Cor. ↑	-0.001 /-0.007	0.019/ 0.034	0.0 /-0.036	0.018 /0.012	0.021/ 0.027	0.008/ 0.016	0.006/ 0.016
Faithfulness Est. ↑	-0.006/ -0.018	0.018/ 0.034	-0.004/ -0.022	0.023/ 0.025	0.012/ 0.016	0.021/ 0.036	0.015/ 0.017
Pixel Flipping (AOC) ↓	0.19 /0.207	0.182 /0.226	0.116 /0.1	0.222 /0.265	0.241 /0.280	0.248 /0.286	0.249 /0.288
Region Pert. (AOC) ↑	2.737/ 2.818	2.836/ 3.174	0.263 /0.0	3.133/ 3.851	3.857/ 3.891	3.739/ 4.16	3.442/ 4.369
AVG Sensitivity ↓	1.352/ 1.014	1.342/ 1.285	3.280/ 2.176	1.71/ 1.084	1.384/ 0.973	1.299/ 1.106	1.231/ 1.159
MPRT (AOC) ↓	96.351/ 94.155	85.212/ 83.420	101.891/ 99.481	90.642/ 86.372	91.969/ 87.756	92.607/ 86.664	90.826 /91.645
IROF (AUC) ↑	0.483 /0.436	0.543 /0.507	0.308 /0.033	0.540 /0.530	0.562 /0.531	0.571 /0.555	0.561 /0.551
IAUC (AUC) ↑	0.545/ 0.581	0.507/ 0.552	0.718/ 0.971	0.430/ 0.445	0.444/ 0.468	0.419/ 0.445	0.468/ 0.473
SPEAR(M_C, M_G)				≈ 0.82			

Table 4.3.: Some XAI Metrics in the IID Case. *Global [red], Client [blue].*

data partition than focusing on the FL algorithm to be deployed. One interesting aspect could be the introduction of generative AI methods. However, this would be subject for further research and not in the scope of this thesis.

4.4. Experiment 2: Which XAI method is more stable?

While the term stability is ambiguous, many scholars in XAI argue that stability means insensitivity to perturbation, which is precisely how the sensitivity metric is defined. However, we pursue stability in this Section in two ways: (i) stability in light of predictive multiplicity – known as the Rashomon Effect (see Definition 2.1.2) – and (ii) stability in terms of changes to the explanation in consecutive FL rounds. Though sensitivity is an important criterion in many applications – and there are already several studies on it – most end-users would expect a stable explanation to hold to satisfy these two types of stability. However, research literature lacks these two aspects, which is why we conducted these experiments.

4.4.1. Methodology

To measure both aspects, we ran similar experiments as described in Experiment 1 (FedAvg, ten FL-clients) with all available explainers explaining the same ten images for each class instance. This setup was then run 74 times for each of the three data partitioning schemes: IID, Square, and Dirichlet.

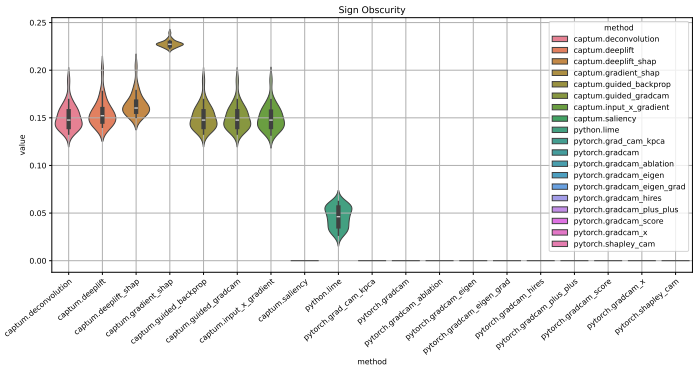
The Rashomon Effect is measured based on the metrics defined in the Foundations Chapter. To quantize the values of the produced explanations (see [30]), we used the *sign* function and also a histogram-based approach with 500 bins in the interval of $[-1, 1]$. We compared it to the reference value/function with the $<$ operator. Then, the mean and the standard deviation of the results are computed.

4.4.2. Results

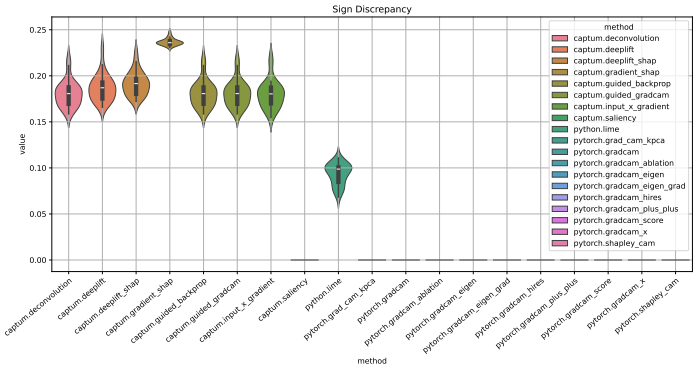
XAI-Method	Sign Obscurity ↓	Sign Discrepancy ↓	Hist. Obscurity ↓	Hist. Discrepancy ↓
Saliency	0.0 ± 0.0	0.0 ± 0.0	0.182354 ± 0.071807	0.005452 ± 0.001293
PyTorch KPCA-CAM	0.0 ± 0.0	0.0 ± 0.0	0.209853 ± 0.003481	0.011519 ± 0.000207
PyTorch GradCAM	0.0 ± 0.0	0.0 ± 0.0	0.236960 ± 0.029444	0.015515 ± 0.001462
PyTorch AblationCAM	0.0 ± 0.0	0.0 ± 0.0	0.216519 ± 0.014479	0.012756 ± 0.000881
PyTorch EigenCAM	0.0 ± 0.0	0.0 ± 0.0	0.210519 ± 0.004349	0.011446 ± 0.000319
PyTorch EigenGradCAM	0.0 ± 0.0	0.0 ± 0.0	0.200418 ± 0.031280	0.014884 ± 0.001591
PyTorch HiResCAM	0.0 ± 0.0	0.0 ± 0.0	0.200404 ± 0.031272	0.014893 ± 0.001588
PyTorch GradCAM++	0.0 ± 0.0	0.0 ± 0.0	0.236130 ± 0.020955	0.015332 ± 0.001123
PyTorch ScoreCAM	0.0 ± 0.0	0.0 ± 0.0	0.250193 ± 0.024496	0.015454 ± 0.001158
PyTorch XGradCAM	0.0 ± 0.0	0.0 ± 0.0	0.223725 ± 0.018838	0.013896 ± 0.001166
PyTorch ShapleyCAM	0.0 ± 0.0	0.0 ± 0.0	0.238422 ± 0.026693	0.015792 ± 0.001403
LIME	0.045991 ± 0.010919	0.093165 ± 0.011554	0.216912 ± 0.020085	0.004401 ± 0.000242
InputXGradient	0.150464 ± 0.012946	0.180130 ± 0.015116	0.269188 ± 0.094524	0.008370 ± 0.002510
Deconvolution	0.150726 ± 0.012748	0.180577 ± 0.014223	0.361272 ± 0.107072	0.010079 ± 0.002137
Guided Backprop	0.150726 ± 0.012748	0.180579 ± 0.014226	0.361245 ± 0.107100	0.010083 ± 0.002135
Captum Guided GradCAM	0.150726 ± 0.012748	0.180579 ± 0.014226	0.016846 ± 0.020292	0.002405 ± 0.002707
Captum Deeplift	0.155078 ± 0.013944	0.186823 ± 0.014578	0.278242 ± 0.088732	0.009014 ± 0.002337
Captum DeepliftSHAP	0.164003 ± 0.012308	0.191474 ± 0.012993	0.287071 ± 0.099401	0.009066 ± 0.002415
Captum GradientSHAP	0.227833 ± 0.003626	0.236654 ± 0.003733	0.236309 ± 0.071842	0.009277 ± 0.002025

Table 4.4.: Measuring the susceptibility of the Rashom Effect for different XAI Methods.

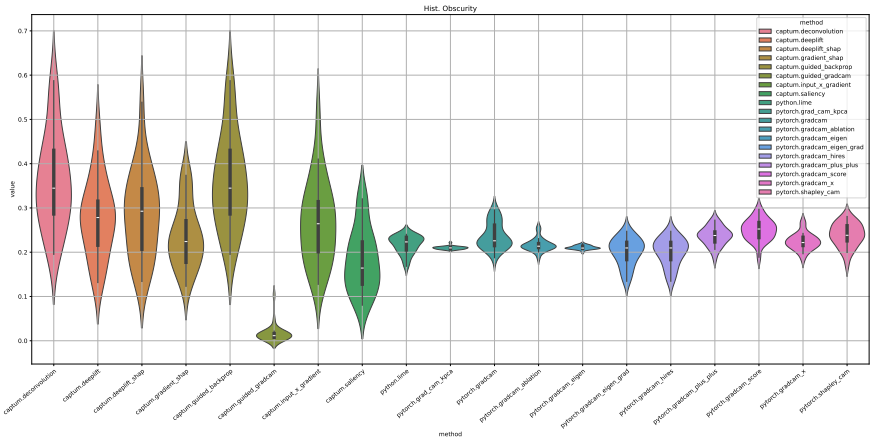
The results are shown in Table 4.4. We can see that different types of XAI methods are differently affected by the Rashomon Effect. First, we notice that for the Sign Obscurity/Discrepancy, some values are zero, which aligns with the fact that these methods only produce a positive feature attribution. Second, we can see that the methods *Captum Guided-GradCAM* and *PyTorch KPCA-CAM* are the most stable concerning to the Rashomon Effect. Interestingly, while the different XAI methods provided by the `grad-cam` library perform well in regard to the histogram obscurity, this seems to be flipped for the discrepancy, which means that while the predictions are somewhat the same data-wise, the maximum ratio of conflicting observations between the predictions as a whole is higher. Figure 4.6 visualizes our results.



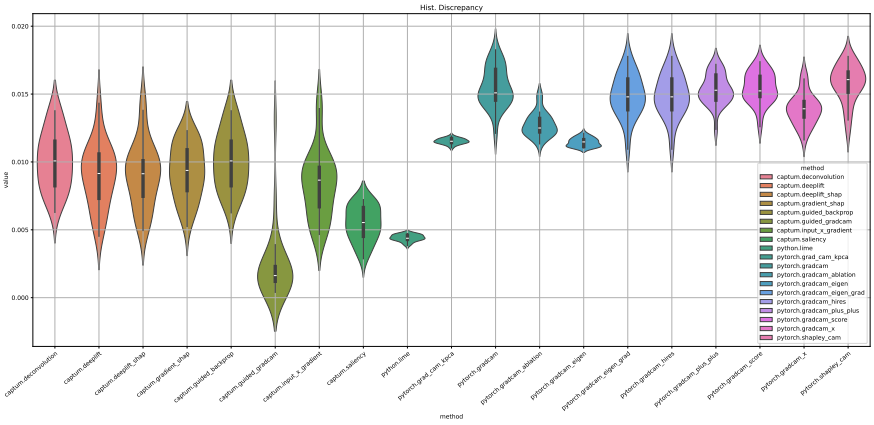
(a) Sign Obscurity.



(b) Sign Discrepancy.

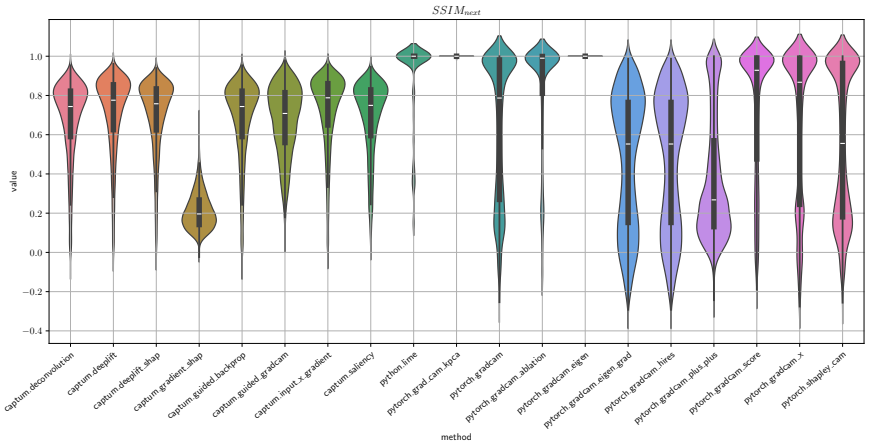


(c) Histogram Obscurity.

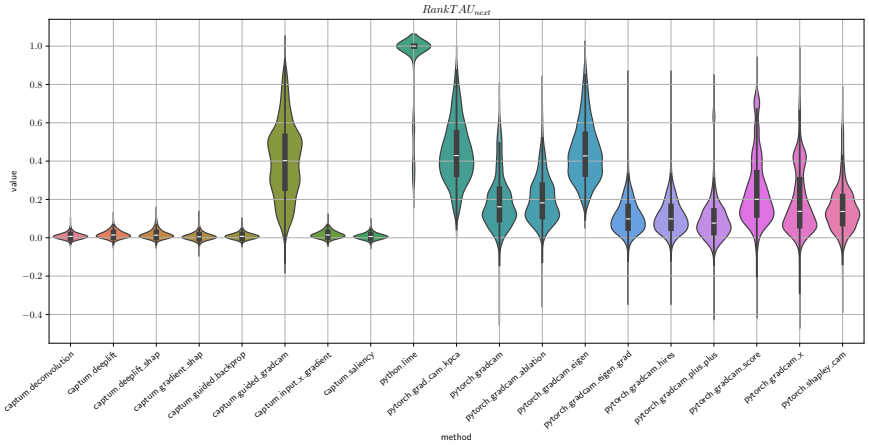


(d) Histogram Discrepancy.

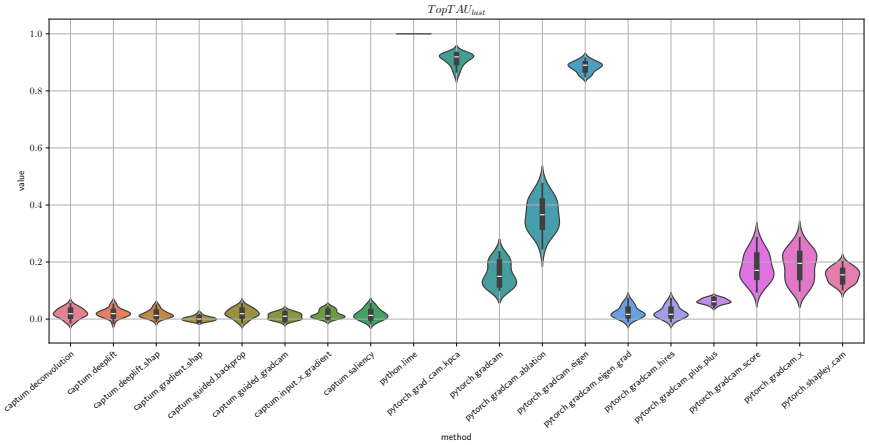
Figure 4.6.: Visualization for Results in Table 4.4.



(a) SSIM comparing consecutive rounds.



(b) RankTAU comparing consecutive rounds.



(c) TopTAU comparing to last round and grouped by rounds.

Figure 4.7.: Example Results for Stability Measurements.

As mentioned above, another important form of stability is the similarity of explanations generated in consecutive FL rounds. Our results are shown in Figure 4.7. We used the SSIM, SPEAR, TAU, RankTAU (indices ranked according to feature importance), and TopTAU (top

eleven indices ranked according to feature importance) values to compare explanations to the next/last round. For the sake of simplicity, we only included the most interesting figures. Additional Figures can be found in the Appendix A.

Given our results, we can infer that especially the XAI methods LIME, PyTorch EigenGrad-CAM, and PyTorch KPCA-CAM seem to be an excellent choice when it comes to stable explanations showing remarkable results in the SSIM, RankTAU, and TopTAU metrics. These results are also aligned with the other data partitioning schemes, Square and Dirichlet (see Appendix A). For the Square paritioning we can see that the standard deviation is smaller, which is reasonable for the FedAvg algorithm. On the other hand, for the Dirichlet partitioning the opposite is the case.

4.4.3. Remarks

Our results try to fill the knowledge gap of not knowing what to expect in terms of stability when using different XAI methods. With the figures and results that we provided, researchers and practitioners are able to estimate and weigh alternatives against each other. For this, we did not want to explicitly state the numbers and compare them against each other; rather, we provide the figures as is with open possibilities for interpretation.

4.5. Experiment 3: Does Optimizing Explanations through Aggregation create better Explanations?

Given that explainability is considered a non-functional requirement, software engineers and requirements engineers want to be able to measure how well an explainability requirement is fulfilled. While one approach would be to gather data directly from the user to measure the degree of fulfillment, in this instance, we could not rely on user feedback, so instead, we opted for using XAI metrics as proxies for evaluating and comparing different explanations against each other (see also Subsection 2.1.2 with the Q_E function). However, during our experiments, it became clear that metrics alone are not actionable, which limits their value immensely. Additionally, taking of said XAI metrics is a tedious task and fallible to be unrepresentative if it is not applied correctly (e.g., most XAI metrics can only be compared relatively against each other and only on the same problem).

To mitigate some of these problems, we adopted the concept of optimized XAI method aggregation as proposed in the paper [50] from Decker et al. and applied and extended it to our FL setting, where we continuously adapt and evaluate the aggregated weights for consumption and optimize for specific metrics.

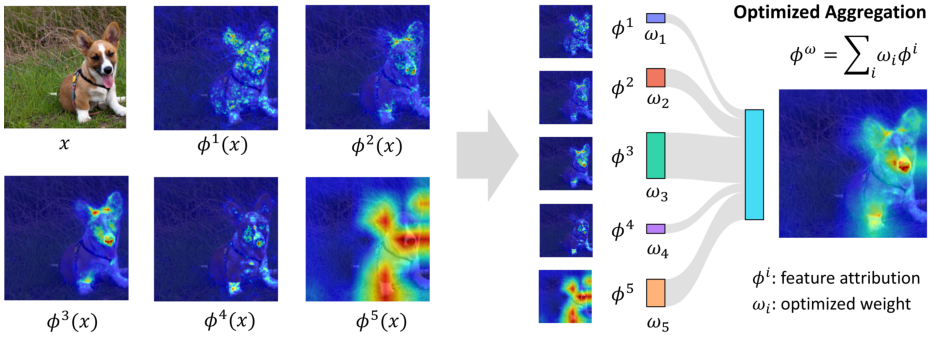


Figure 4.8.: Optimizing Explanations through Aggregation for certain Metrics [50].

4.5.1. Proposed Solution

The concept presented in [50] can be visualized through Figure 4.8, which the authors provide. This Figure, shows that multiple explanations by different explainers are weighted, aggregated and combined into one single explanation. The aggregation weights are calculated by optimizing for one or more metrics. In the original paper, the authors optimized for the metrics of infidelity and sensitivity. However, we extended the optimization also to other XAI metrics and reformulated the problem to a multi-objective problem.

Definition 4.5.1 (Optimizing Explanations) *Let $Q : \mathbb{R}^d \to \mathbb{R}$ be a metric for feature attribution methods $\phi : \mathbb{R}^d \to [0,1]^d$ with $x \in \mathbb{R}^d$. If Q can be transformed to the following form with suitable $\gamma_1 \in \mathbb{R}^{g \times d}$ and $\gamma_2 \in \mathbb{R}^g$:*

$$Q(\phi(x)) = \mathbb{E}_{\gamma_1,\gamma_2}[||\gamma_1\phi(x) - \gamma_2||_2^2]$$

Then, the optimal aggregation of different feature attribution methods $\phi^* = \Phi \times \omega$ with a weight vector $\omega \in \mathbb{R}^k$ for k feature attribution methods and $\Phi = (\phi_1, \ldots, \phi_k) \in \mathbb{R}^{d \times k}$ is given by solving the following convex problem:

$$\min_{\omega} \mathbb{E}[||(\gamma_1\Phi)\omega - \gamma_2||_2^2] \qquad s.t. \quad \omega_i \geq 0, \sum_{i=1}^{k} \omega_i = 1$$

It can be shown "[...] that the quality of the aggregated explanation is at least as good as the equivalently weighted individual attribution qualities [...]." [50].

Given Definition 4.5.1 above, we adopted the proposed optimization for our FL context by first introducing multiple other XAI metrics that can be optimized the same way: (i) Attack Metric (e.g., similar to sensitivity, but with adversarial perturbations), (ii) ROAD, (iii) Selectivity, (iv) Pixel Flipping, and (v) Region Perturbation. Moreover, make metrics available

(e.g., IROF, IAUC, DAUC, Sparseness, MPRT, Faithfulness Correlation, Faithfulness Estimate, Local Lipschitz Estimate) on a "choose-by-best" basis, where the weight vector ω one-hot encodes the selection of the best feature attribution method given a specific metric. Furthermore, we propose that in a classification task, the optimization is done on a per-class-basis because our testing showed that depending on the class instance to be explained, the optimal aggregation weight vector ω can vary significantly. Therefore, respecting this circumstance is crucial for achieving higher performance in the optimization process. The adversarial samples were generated with different methods provided by the Captum[6] [110] Python library (Project Gradient Descent (PGD) [129], Fast Gradient Sign Method (FGSM) [78]), and the Adversarial Robustness Toolbox (ART)[7] [144] (specifically the Auto-Attack [44], Auto Conjugate Gradient [190], Auto Projected Gradient Descent (Auto-PGD) [44], Carlini & Wagner (CW) [28], DeepFool [137], FGSM, Momentum Iterative Method [55], Basic Iterative Method (BIM) [116], NewtonFool [95], Jacobian Saliency Map [147], Shadow Attack [75], Spatial Transformation Attack [67], Square Attack [10], Zeroth Order Optimisation (ZOO) [39], and Elastic Net [38]).

While the authors of the original paper limited the metrics for the aggregation only to sensitivity and infidelity, we can extend to an arbitrary number of metrics, and also impose additional constraints like the costs of a given XAI method, and add metric preferences to the equation, and arrive at the following multi-objective optimization problem.

Definition 4.5.2 (Multi-objective Explanation Optimization) *Let $\omega_1, \ldots, \omega_m \in \mathbb{R}^k$ be the "optimal" individual weights for each metric m, as defined in Definition 4.5.1. Furthermore, let $\Omega = (\omega_1, \ldots, \omega_m) \in \mathbb{R}^{k \times m}$ be the weight-matrix composed of stacking the individual weight vectors for each metric, $c \in \mathbb{R}^k$ be a cost-vector that defines the cost of each of the k feature attribution methods (e.g., normalized median measured times to compute n number of feature attributions), and $p \in \mathbb{R}^m$ be a preference-vector, defining how the individual metrics should be prioritized. Then, the above problem can be defined as the following multi-objective problem with two objectives and the aggregated weight vector $\psi \in \mathbb{R}^k$ as target:*

$$\delta_1 := \min_{\psi} agg\left[(\Omega^\intercal \times diag(c)) \times \psi\right] \quad \delta_2 := \max_{\psi} agg\left[(\Omega \times diag(p))^\intercal \psi\right]$$
$$constr.: \quad \sum_i \psi_i \approx 1 \quad \wedge \quad \forall i \in \{0, \ldots, k\} : 0 \leq \psi_i \leq 1$$

Where δ_1 is to minimize the cost for the feature attribution, and δ_2 is to maximize the metric improvement. The function agg aggregates the value of the resulting vector to a single scalar and is usually just the sum over each element, or to smooth out negative values, the LogSumExp function. The cost objective is here defined as proportional to a chosen explainer's influence e.g., $\psi_x = 0.5$ results in accounting for 0.5 times the cost of the explainer at position x. If this is not the desired behavior than δ_1 should be defined as $\delta_1 := \min_{\psi} agg\left[(\Omega^\intercal \times diag(c)) \times \lceil \psi \rceil\right]$ instead.

[6] https://github.com/pytorch/captum

[7] https://github.com/Trusted-AI/adversarial-robustness-toolbox

As noted before, the multi-objective optimization problem in Definition 4.5.2 is solved for each class instance in the classification task. For solving the computation of the individual weight vectors, we used the `cvxpy`[8] [54] Python library – like the original authors – which uses a solver specifically designed for convex optimization problems. While it is possible to use this library for our multi-objective optimization problem, we would need a mechanism to additionally specify two hyperparameters λ_c, λ_p that weight the objectives against each other so that they can be stated as minimizing one total sum. However, finding these hyperparameters beforehand is challenging and prone to error, so we opted instead for solvers designed for the multi-objective task. Respectively, we used the Python library `pymoo`[9] [20] and `scikit-opt`[10]. These libraries use genetic algorithms like NSGA-II [48] as solvers. Furthermore, our testing showed that using `cvxpy` will most likely result in sparse matrices (meaning not much aggregation, just choosing the best), while the genetic algorithms tend to facilitate aggregation.

4.5.2. Analysis

We conducted multiple experiments to see whether or not our method of explanation optimization is valid. The first series of experiments is dedicated to the method by which the aggregation weights are calculated. Figure 4.9 shows the results. We utilized several different XAI methods (see x-axis) and optimized over all available metrics. The results were then aggregated by averaging or with optimization through `cvxpy` which does not respect the cost of the XAI method, or with `pymoo` which solves the multi-objective optimization problem we described beforehand[11]. In any case, we can see that the aggregation performs – in aggregate – always better than any XAI method alone. Also, we can see that a structured way of computing the aggregates is better than simply averaging. Furthermore, respecting cost as an additional objective does not significantly harm the improvement we get from the aggregation, sometimes even surpassing `cvxpy` in that regard. However, this graph only shows the total improvement, which is calculated by normalizing each metric results via min-max scaling, where the minimum and maximum are chosen from all the results on a given metric. The total improvement is, therefore, an average of all individual improvements for a given metric. Looking at the level of individual metrics (see Figure 4.10), we can see that aggregation sometimes performs worse than a single method. It may not be desirable to optimize simply for all available metrics but target specific metrics that shall be improved.

By representing the problem as a multi-objective optimization one, we can also see in Figure 2.1 that the relation between cost and performance is strongly linear[12]. In this instance, we opted for the pseudo-weight algorithm [49] to select the concrete aggregation weights. In essence, the pseudo weights algorithm respects how much each objective should be weighted against each other. Both were set at 0.5 to balance cost and performance. The

[8] `https://github.com/cvxpy/cvxpy`
[9] `https://github.com/anyoptimization/pymoo`
[10] `https://github.com/guofei9987/scikit-opt`
[11] Results respecting the real cost (see remark in Definition 4.5.2.) can be found in the Attachment A.4.
[12] This changes if we apply the ceiling function as proposed in Definition 4.5.2.

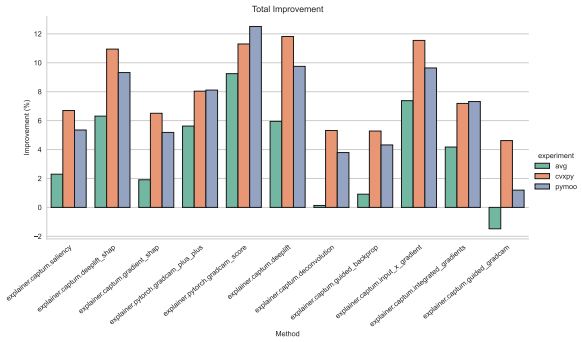


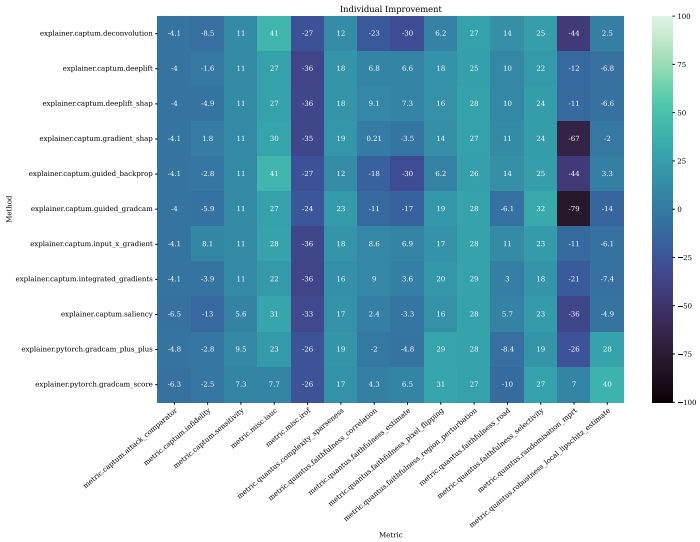
Figure 4.9.: Comparing different Aggregated Weight Computation Methods.

pseudo-weights can be optimized depending on the requirements one has. Furthermore, one can introduce additional constraints to the multi-objective optimization problem e.g., setting a maximum bound for XAI methods to be used or prioritizing specific metrics more than others.

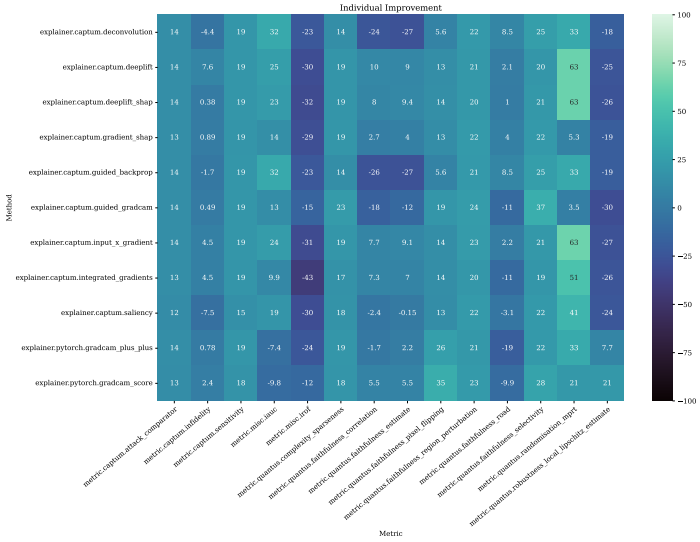
Now, to confirm that our solution performs as well as described in the original paper, we tested it specifically to optimize the metrics infidelity and sensitivity, where sensitivity draws 250 samples from adding random noise and infidelity uses perturbation based on substitution with black squares. Figure 4.12 shows the results. The results mostly align with the original paper, especially the significant boost in performance regarding sensitivity, which resembles the results they obtained. Infidelity is worse. However, in the original paper, they used different XAI methods for sensitivity and infidelity, and their method of perturbation for infidelity differs, which would explain the decrease in the performance for infidelity. Choosing a suitable perturbation method is therefore crucial to realize gains that can be applied to real-world scenarios.

What happens if we optimize for a specific category of metrics, namely perturbation-based methods, was also particularly interesting. Figure 4.13 shows the results. It shows that the overall improvement can benefit by this approach, with an average uplift of around 25%. Interestingly, we can also see that this method improves the robustness against different types of attacks significantly; granted, this does not reflect real-world use cases because the improvement only says that the explanation does change less in light of perturbations generated by attacks but nothing about the prediction.

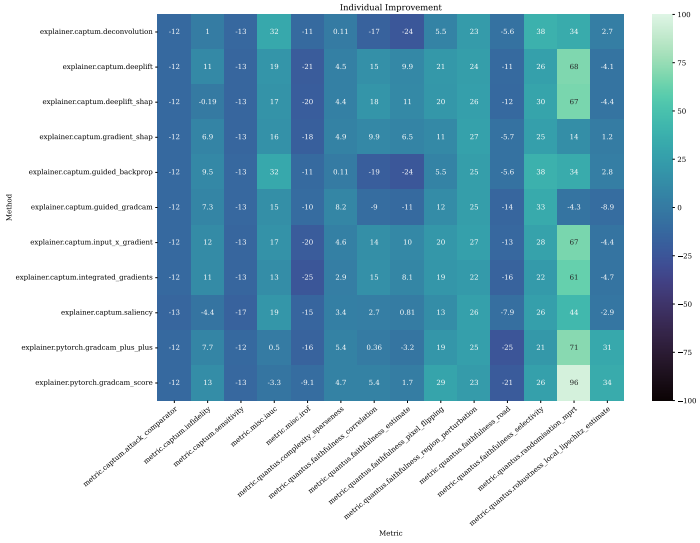
In the next series of experiments, we were interested in whether the aggregation weights could be reused in the next round without a decrease in performance. The results are shown in Figure 4.14. Surprisingly, not only were we able to reuse the weights, but it also had an aggregated improvement compared to the individual XAI method. There can be several reasons for this: (i) First, the measurement was taken for the FL round $T = 24$. Therefore, the ML model had already converged which could be reflected also in the explanations. (ii) Second, we included complexity and randomization metrics that significantly boost the aggregated improvement. (iii) Lastly, it can very well be the case that `pymoo` has not found the best aggregated weight for the given round T but for the next round $T + 1$.



(a) Results Aggregation via Averaging.



(b) Results Aggregation via `cvxpy`.



(c) Results Aggregation via `pymoo`.

Figure 4.10.: Comparing different Aggregation Methods on a per Metric-basis.

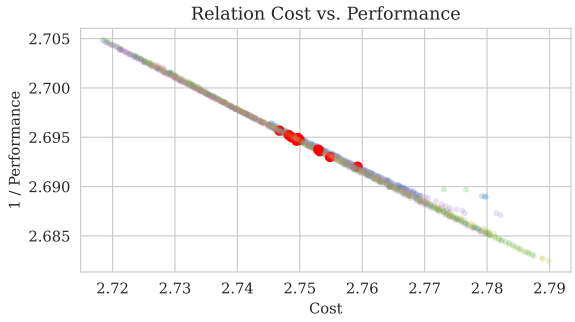


Figure 4.11.: Pareto Front Between the Two Objectives Cost and Performance. *Notice: The Perfomance Axis Is Presented Inverse. [Red] Chosen Trade-off Points for Each Class.*

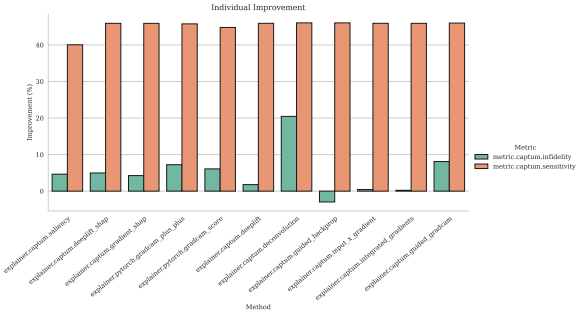
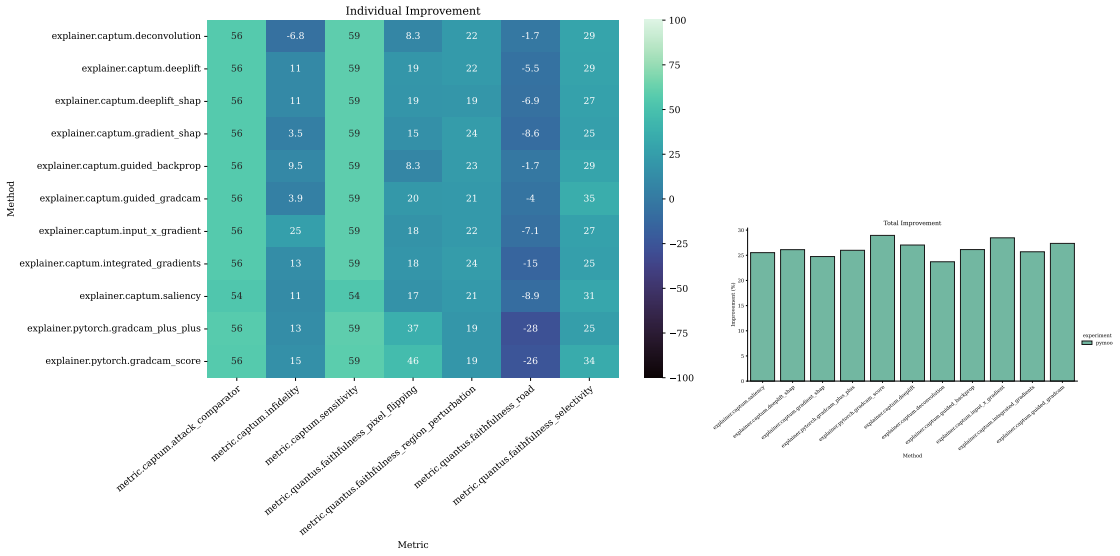


Figure 4.12.: Multi-Objective Optimization on Infidelity and Sensitivity Metrics.



(a) Improvement on individual Metrics. **(b)** Overall Improvement.

Figure 4.13.: Improvement on Perturbation-based XAI Metrics.

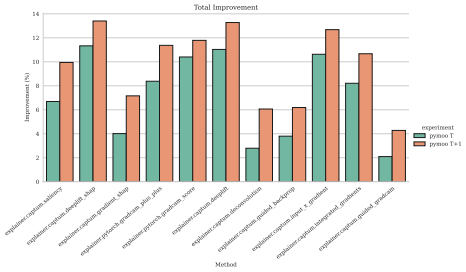
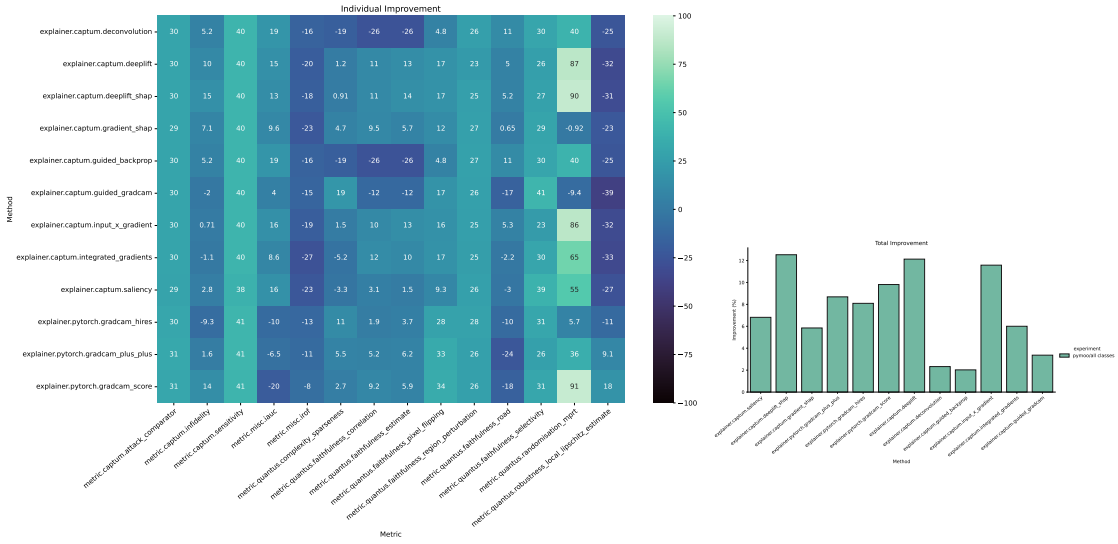


Figure 4.14.: Reusing Aggregated Weights for the next Round.



(a) Improvement on individual Metrics.

(b) Overall Improvement.

Figure 4.15.: Calculating Aggregation Weights over all Classes simultaneously.

Lastly, we wanted to see if optimization should be applied to all classes of the classification problem simultaneously or individually, as we did before. Figure 4.15 shows the results. Compared to Figure 4.10c, we can see that the negative spikes reach further down. However, some metrics, like sensitivity, seem to benefit from this approach.

In conclusion, the presented results (of all conducted experiments in this Section) show that optimizing explanations through solving a multi-objective optimization problem can measurably outperform individual XAI methods while respecting optimization targets.

4.5.3. Application for Requirements Engineering

The idea in the context of this thesis is that requirements engineers can utilize this method to additionally constrain or measurably define target metrics for explainability that should be optimized. The presented framework is very flexible and can be adapted and extended for other measurable metrics. However, before we finish this section, we briefly want to give additional considerations for applying this technique in practice.

Important: Considerations and Pitfalls for Requirements Engineers

- One needs to know how to apply XAI metrics and their limitations. Given the flexibility of some metrics that can be instantiated e.g., with different types of perturbation functions and hyperparameters, one needs to be knowledgeable so that their application has real-world meaning. Figure 4.12 demonstrated this.
- One needs to be knowledgeable of different XAI methods. To optimally balance cost and performance, it also makes sense to limit the number of XAI methods one uses for the aggregation. Some XAI methods hold certain desirable properties that could potentially invalidated by performing the aggregation.
- Importantly, one must be careful about the explanation vs. prediction gap. While explanations should be tightly integrated with the underlying ML model, this may not always be true. For example, the attack metric used throughout this section is essentially a sensitivity metric, but regarding the explanation, not the prediction. So, it only describes how much the explanation changes but not how the prediction changes, which is undoubtedly a crucial part for robustness against attacks.
- While our results indicate that the aggregation weights can be reused, we strongly suggest tying it to some form of adaptive validation mechanism so that if a deterioration in performance occurs, a recomputation of the aggregation weights is started.
- Lastly, we want to encourage imposing additional constraints on the multi-objective optimization problem to further refine the aggregation for desirable properties if possible. This notion is also in line with the idea that every non-functional requirement should be measurable. If this is the case, one can also try to apply this concept to make what is measurable; actionable.

4.6. Experiment 4: How much does Differential Privacy harm the explanations?

DP and other Privacy-Enhancing technologies (PETs) try to minimize the risk of compromising an individual's privacy, where common anonymization techniques would not suffice. In short, DP allows data to be analyzed while ensuring the individual's privacy through certain statistical guarantees [64, 119]. One of the most commonly used notation for DP is presented in Definition 4.6.1. Indeed, one could write a whole book about this very topic. However, in this section, we are only interested in applying the most common concept of DP already present in the FL framework Flower and investigating how it affects the explanation we generate.[13] Specifically, Flower supports the following DP mechanisms:

[13] We also exclude methods like Differential Privacy Stochastic Gradient Descent (DP-SGD) [1] that can be directly applied to the ML model training e.g., for PyTorch with the Python library `opacus` [193].

- DP mechanisms are applied centrally (after ML model aggregation) or locally (before ML model aggregation). Applying DP mechanisms centrally usually results in a higher utility than applying them locally, but it requires additional trust in the FL server instance.
- Fixed (predefined threshold) or adaptive (dynamic threshold) clipping of weight updates and adding Gaussian noise to it. In the adaptive setting, the weight updates are clipped based on the algorithm presented by Andrew et al. [9], while in the fixed setting, flat-clipping is applied, as presented by McMahan et al. [133]. Furthermore, the Gaussian noise can be added on the client side, only on the server side, or both.

Definition 4.6.1 ((ϵ, δ)-differential privacy) *A randomized algorithm $\mathcal{A}$ is (ϵ, δ)-differentially private if for all databases $x, y \in \mathcal{D}^n$ – where databases are assumed to be vectors in $\mathcal{D}^n$ for some domain $\mathcal{D}$ – that differ only in one entry, $\Pr[x]$ the probability function for x, and for all subsets S of outputs from $\mathcal{A}$, the following equation holds:*

$$\Pr[\mathcal{A}(x) \in S] \leq e^{\epsilon} \cdot \Pr[\mathcal{A}(y) \in S] + \delta \quad [101]$$

The effect of the DP mechanisms on the explanations is measured in terms of the explainability metrics, as we did in Experiment 1, stability, as we did in Experiment 2, and in terms of explanation optimization, as we did in Experiment 3. For further simplicity, we only used fixed clipping – and not adaptive clipping – so that we do not need to tune and rely on additional hyperparameters in our experiments. Furthermore, when applying client side DP, we also apply Gaussian noise on the server side, as is the case in Flowers's own reference implementation. Also, we only apply the noise to specific layers – all bias parameters, all fully connected layers, and layer zero and layer seven – of our ML model because otherwise the ML model becomes unusable most of the time. Specifically layers like BatchNorm2D become unusable with clipping and Gaussian noise applied, this is also why the `opacus` Library for DP-SGD converts layers to other nearly equivalent layers e.g., BatchNorm2D to GroupNorm. However, we did not want to change the ML model directly, so this would be part of future work instead.

First and foremost, the results show that the accuracy drops significantly when DP is applied. Notably, we can also see that the XAI metrics and measurements are worst in nearly all cases, meaning that DP harms explainability. This is unsurprising because research literature on it already exists [69, 160]. However, in terms of stability, our results – at least by looking at the figures – show that while the explainability is worse on average, the measurements try to converge very steeply to a similar point to our reference measurement. This behavior leads us to the conclusion that explainability is mostly harmed if not an adequate number of *compensatory rounds*[14] are added. We want to stress out, that the *compensatory* rounds are only meaningful for the explainability and not e.g., the accuracy. Even after 150 FL rounds – six times more FL rounds – both client side ($\overline{accuracy} = 80.94$) and server side ($\overline{accuracy} = 82.08$) DP could not achieve the same accuracy score than without DP[15]. Lastly,

[14] Additional FL rounds that compensate the drawbacks of adding DP.
[15] See Appendix A.19.

Measurement	No DP	Server-side fixed Clipping	Client-side fixed Clipping
Mean Accuracy ↑	87.37	42.38	47.43
Mean $SSIM_{next}$ ↑	0.66 ± 0.23	0.61 ± 0.2	0.56 ± 0.19
Mean $SSIM_{last}$ ↑	0.46 ± 0.23	0.22 ± 0.2	0.18 ± 0.2
Mean Sensitivity ↓	1.678 ± 1.334	18.883 ± 13.559	17.364 ± 33.122
Mean Infidelity ↓	0.049 ± 0.044	338.636 ± 632.826	1561.612 ± 4565.515
Mean Faithfulness Cor. ↑	0.047 ± 0.101	0.015 ± 0.085	0.017 ± 0.1
Mean Pixel Flipping (AOC) ↓	0.254 ± 0.177	0.131 ± 0.203	0.154 ± 0.237
MPRT (AOC) ↓	94.927 ± 5.443	124.512 ± 21.080	121.042 ± 18.716

Table 4.5.: Results for Saliency FedAvg/IID with and without DP.

we could not find a significant difference between server side and client side DP. This could be because of our hyperparameter choice (see Appendix A). However, investigating this further would be subject to future work.

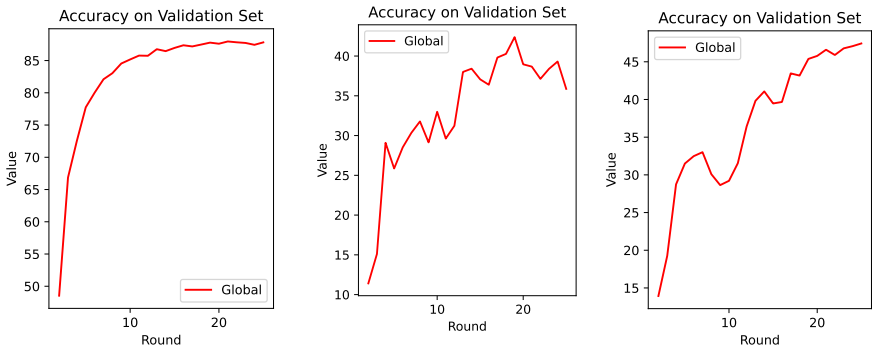
Second, having to redo the methodology presented in Experiment 2, with 15 runs in comparison, we could not see any significant difference between applying DP and not applying DP in terms of the stability related to predictive multiplicity. This is interesting because it suggests that this form of stability can be directly linked to the XAI methods and not to the ML model performance, meaning that our observations in Experiment 2 are most likely generally applicable.

Finally, we wanted to investigate how DP could affect our explanation optimization approach, which was presented in Experiment 3. For this we utilized all of our available XAI methods and all available XAI metrics except metrics that are complexity-based (MPRT, and Sparseness). The results are shown in Figure 4.17. We can clearly see that DP has a very bad influence on our optimization mechanism because both of them are worse than no DP applied. Furthermore, we can see that Client-side DP is more harmful than Server-side DP for XAI metrics. This results is also not that much surprising, given that a lot of XAI metrics seem to perform worse, when the accuracy of the ML model drops – as it is the case when applying DP.

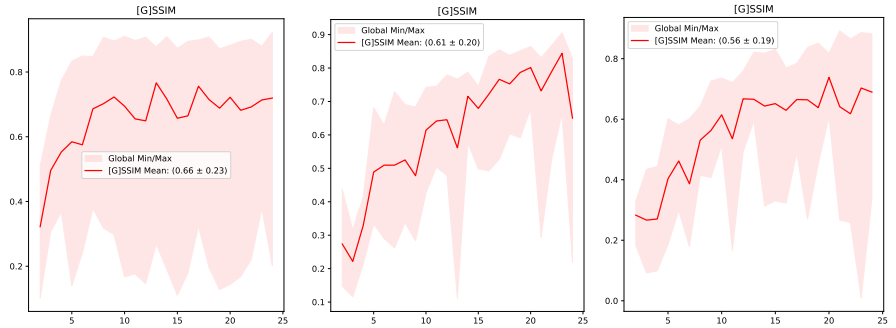
To put it briefly, DP harms explainability, but not as much as the accuracy of the ML model. Surprisingly, the effect on stability in terms of the Rashomon effect is non-existent. Lastly, we investigated whether our explanation optimization approach could again boost explainability performance (measured regarding XAI metrics), which is the case, but not enough to compensate for the harmful effects of applying DP.

4.7. Experiment 5: What if clients misbehave?

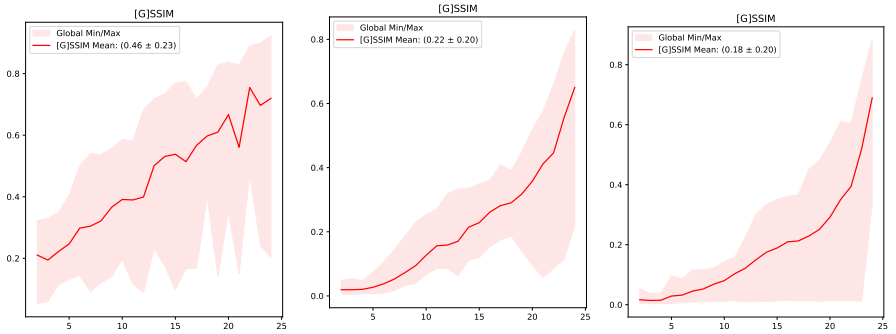
In contemporary FL research literature, the fact that any FL system depends on the FL clients executing the FL algorithm as intended is often assumed. However, it may be difficult



(a) Accuracy without DP **(b)** Accuracy Server-side DP **(c)** Accuracy Client-Side DP



(d) $SSIM_{next}$ without DP **(e)** $SSIM_{next}$ Server-side DP **(f)** $SSIM_{next}$ Client-Side DP



(g) $SSIM_{last}$ without DP **(h)** $SSIM_{last}$ Server-side DP **(i)** $SSIM_{last}$ Client-Side DP

Figure 4.16.: Results for Saliency FedAvg/IID with and without DP.

to rely on this given system property in practice. Even if we do not assume malicious intent, individual FL clients could introduce mistakes or errors in the FL algorithm. Therefore, in this Section, we want to determine the impact of mistakes or errors introduced by individual FL clients on explainability. For this, we tested the following:

- Shifting or Randomizing Labels: An increasing number of clients wrongfully train their local ML model on shifted or randomized labels.
- Reporting the wrong number of processed samples: One client wrongfully counted the number of samples it used to trained its local ML model.

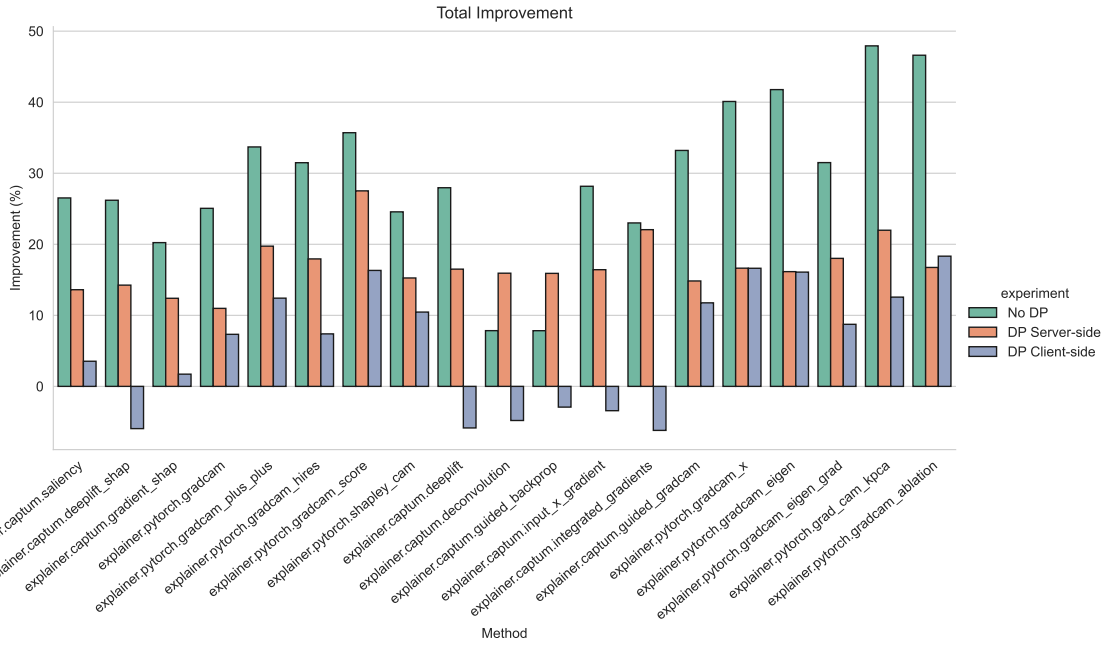


Figure 4.17.: Multi-objective Explanation Optimization with and without DP.

Measurement	Reference	Shift one Client	Shift two Clients	Shift four Clients	Shift eight Clients
Mean Accuracy ↑	83.303	82.632	80.514	57.0325	3.4246
Mean $SSIM_{next}$ ↑	0.71 ± 0.16	0.65 ± 0.22	0.60 ± 0.22	0.52 ± 0.24	0.54 ± 0.20
Mean $SPEAR_{next}$ ↑	0.69 ± 0.13	0.65 ± 0.17	0.62 ± 0.17	0.57 ± 0.19	0.55 ± 0.17
Mean $SSIM_{last}$ ↑	0.46 ± 0.21	0.45 ± 0.21	0.40 ± 0.18	0.42 ± 0.21	0.35 ± 0.16
Mean $SPEAR_{last}$ ↑	0.51 ± 0.16	0.48 ± 0.17	0.56 ± 0.17	0.46 ± 0.18	0.38 ± 0.14
Mean Infidelity ↓	0.046 ± 0.042	0.028 ± 0.034	0.015 ± 0.019	0.014 ± 0.016	0.008 ± 0.007
Mean Sensitivity ↓	1.453 ± 1.166	7.778 ± 8.135	3.947 ± 3.306	3.219 ± 3.201	44.356 ± 36.438
Mean Faith. Cor. ↑	0.038 ± 0.105	0.062 ± 0.110	0.023 ± 0.097	0.032 ± 0.098	0.03 ± 0.089
Mean Pixel Flipping (AOC) ↓	0.288 ± 0.213	0.282 ± 0.207	0.262 ± 0.187	0.205 ± 0.172	0.064 ± 0.072
Mean Region Pert. (AOC) ↑	4.862 ± 0.935	4.648 ± 0.826	4.054 ± 0.987	2.505 ± 0.722	-0.288 ± 0.181
Mean IROF (AOC) ↑	0.604 ± 0.193	0.602 ± 0.196	0.554 ± 0.214	0.518 ± 0.256	0.137 ± 0.189
Mean IAUC (AOC) ↑	0.460 ± 0.215	0.454 ± 0.231	0.441 ± 0.243	0.518 ± 0.238	0.801 ± 0.199

Table 4.6.: Results of Experiment 5 (Shifting Labels): Saliency on FedAvg/IID.

- Randomized Model Weights: An increasing number of clients sends randomized ML model parameters to the FL server instance.

In our first series of experiments (see Table 4.6), we can see that shifting labels will mostly be harmful when a total number of four Clients start shifting labels. However, we can see that some XAI metrics need to be evaluated in the context of multiple metrics to be helpful, given that some metrics are measured to be improved under shifting labels. The

Measurement	Reference	One Client	Two Clients	Four Clients
Mean Accuracy ↑	83.303	81.68	77.359	58.196
Mean $SSIM_{next}$ ↑	0.71 ± 0.16	0.70 ± 0.18	0.52 ± 0.20	0.34 ± 0.18
Mean $SPEAR_{next}$ ↑	0.69 ± 0.13	0.69 ± 0.15	0.59 ± 0.15	0.44 ± 0.13
Mean $SSIM_{last}$ ↑	0.46 ± 0.21	0.53 ± 0.19	0.41 ± 0.17	0.33 ± 0.16
Mean $SPEAR_{last}$ ↑	0.51 ± 0.16	0.52 ± 0.17	0.45 ± 0.14	0.40 ± 0.13
Mean Infidelity ↓	0.046 ± 0.042	0.006 ± 0.006	0.003 ± 0.003	0.001 ± 0.001
Mean Sensitivity ↓	1.453 ± 1.166	2.291 ± 2.135	4.212 ± 2.537	23.647 ± 16.641
Mean Faith. Cor. ↑	0.038 ± 0.105	0.024 ± 0.109	0.025 ± 0.88	0.012 ± 0.104
Mean Pixel Flipping (AOC) ↓	0.288 ± 0.213	0.018 ± 0.120	0.169 ± 0.096	0.124 ± 0.043
Mean Region Pert. (AOC) ↑	4.862 ± 0.935	0.209 ± 0.119	1.649 ± 0.525	0.659 ± 0.273
Mean IROF (AOC) ↑	0.604 ± 0.193	0.555 ± 0.161	0.460 ± 0.171	0.284 ± 0.176
Mean IAUC (AOC) ↑	0.460 ± 0.215	0.467 ± 0.176	0.581 ± 0.179	0.724 ± 0.176

Table 4.7.: Results of Experiment 5 (Randomizing Labels): Saliency on FedAvg/IID.

same applies to randomizing the labels, as shown in Table 4.7, except that the deterioration seems more significant than shifting the labels.

We omitted the results of one client reporting the wrong number of samples because our evaluation only showed an insignificant deviation from the reference.

Finally, we wanted to see how our FL context is affected by introducing clients that send randomized ML model weights to the central FL server instance. Our results – included in the GitLab repository – indicate that only one client performing the randomization is enough to completely break the FL algorithm, resulting in abysmal accuracy and explainability metrics. This result is interesting, given that even in the FedAvg/IID case, one client suffices for this attack.

To summarise our experiment results: It is evident that the demonstrated types of attacks influence explainability and ML model performance, with sending randomized ML model weights being the strongest. Therefore, some form of sanity check should be carried out before the aggregation is done. While sanity checks impose additional overhead on the FL server instance, it may be necessary for specific environments to mitigate the risk of being affected by one of the presented attacks.

5. Evaluation

In this Chapter, we wanted to field-study some of our results presented in Chapter 4. We conducted a survey based on a proxy task that was answered primarily by students and researchers of the computer science department at the Karlsruhe Institute for Technology (KIT) and Politecnico di Milano.

5.1. Goals and Questions (2)

Our GQM goals are further extended with the following research and evaluation goals for this Chapter:

- **RG2**: Examine participants' different opinions regarding explainability.
- **RG3**: Examine participants' acceptance of different explanation approaches.
- **EG6:** Evaluate the effectiveness of the explanation optimization approach for humans.

This time, each associated question is directly evaluated in terms of participants' agreement, satisfaction, or opinions.

5.2. Survey

In the following, we will present our user survey from the preparatory setup to analysis. The reader should have a good understanding of how we designed, conducted, and analyzed the survey. Where needed, we added statistical measurements.

5.2.1. Setup

The survey design is structured into four sections: (i) Participant's experience, (ii) Opinions about explainability, (iii) Evaluation of the explanation optimization. In our survey design, we followed the principles found in contemporary textbook literature [99, 152, 161]. We mainly focused on simplicity so that little to no prior experience was needed to participate in the survey.

κ	Interpretation
< 0	Poor agreement
$0.01 - 0.2$	Slight agreement
$0.2 - 0.4$	Fair agreement
$0.41 - 0.6$	Moderate agreement
> 0.61	Substantial to perfect agreement

Table 5.1.: Fleiss' Kappa Interpretation [118].

5.2.2. Implementation

After the design phase, we implemented the survey via the EFS Survey software[1], which is part of the unipark software suite. The software is proprietary, and a license for it was obtained via the KIT. Custom design changes have been made to make the questionnaire easier to read. For instance, we added a border around the question to visualize their separation better. Furthermore, for some questions that required more space for images, we increased the size of the question boxes for desktop computer screens. For the section regarding the evaluation of the explanation optimization, we rotated (selection of different questions) and shuffled the questions the participants saw to allow for comparison between more XAI methods and different orders. It should be emphasized that the rotation is not a random; instead, the survey software tries to obtain an approximately uniform distribution for all questions. The shuffling, however, is purely random and individual for each user. Lastly, we added an introduction of our research goal and appropriate legal disclaimers to the survey. The survey is attached to this thesis in Appendix A.

5.2.3. Execution

The survey was conducted from 08.04.2025 to 14.04.2025. Subjects of the survey were predominantly students and researchers from the computer science department at the KIT and Politecnico di Milano. Participation was voluentarily. Alongside the survey, participants were offered a small candy treat to engage in the participation. However, participation or proof thereof was no requirement to obtain the candy.

5.3. Results and Discussion

Of the 95 samples, 28 participants completed the survey. To mitigate the effect of a selection bias because of partial results, we only evaluated the 28 completed surveys. Unsurprisingly, most participants identified themselves as male, and most were relatively young (< 34 years). Interestingly, the heterogeneity of the participant group regarding their highest

[1] `https://www.unipark.com`

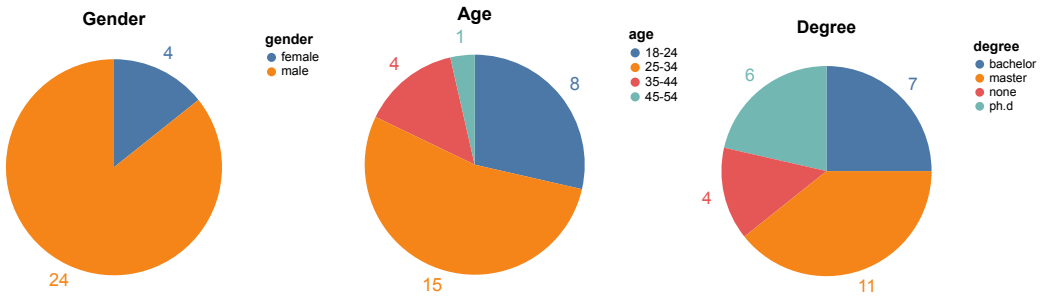


Figure 5.1.: Participant's Heterogenity.

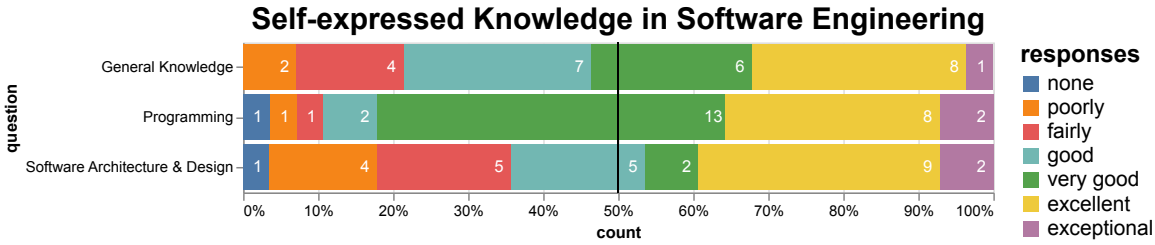


Figure 5.2.: Self-expressed Knowledge in SE.

academic degree is pleasantly somewhat evenly distributed. Nevertheless, most participants were students or researchers who had master's degrees.

Regarding self-expressed qualifications in software engineering (see Figure 5.2), the participants were not shy in proclaiming their knowledge. We are not entirely sure why the figure shows that around 50% of the participants have "very good" or better Software Engineering (SE) knowledge (the one-sided t-test also confirmed this by a standard significance level set to 0.05). This level of qualification is already interesting, raising the question of whether people tend to overestimate their self-reported knowledge. Also interesting is that from the three categories, "Programming" obtained the most people who reported knowledge of "very good" or higher, indicating that programming is easier to learn or people choose to refine the most. With $\kappa \approx 0.402$, the agreement between participants in their qualification was also statistically measurable.

If one participant selected in one of the categories presented in Figure 5.2 a score higher than "good", they were also asked to specify how many years of experience they have in SE. The results can be seen in Figure 5.3. We removed two outliers in the data: one participant answered 25 years and another 99 years. However, the expressed years of experience in SE matches the fact that the self-reported qualifications were very high.

Figure 5.3.: Years of Experiences in SE.

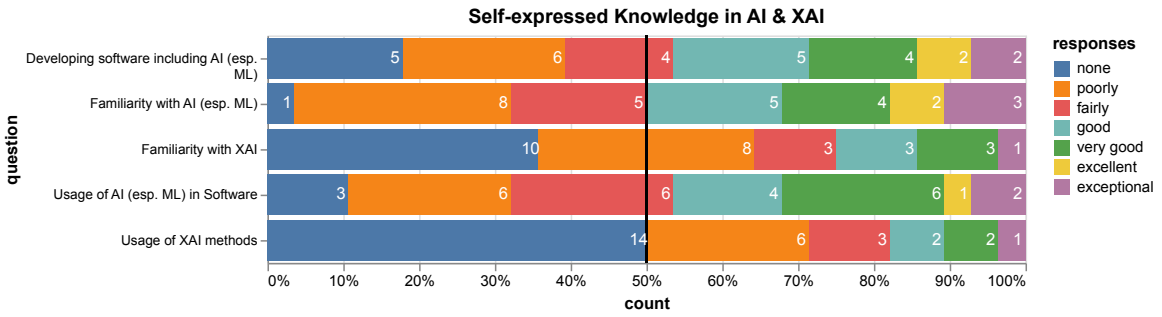


Figure 5.4.: Self-expressed Knowledge in AI and XAI.

Next, in Figure 5.4, participants were asked about their familiarity with AI and XAI. While participants reported a "fairly" or higher score in their knowledge and usage – even in development – of AI, most reported a poor score in their knowledge of XAI. Unsurprisingly, this is also why most participants have not used XAI methods. This indicates that people tend to "just" use or apply AI and raises the question of how the participants know that the AI does what it is supposed to do when they use or apply AI. The participants want more explainability in the later part of the survey results. However, the fact that most of them have not used any XAI method is somewhat astonishing. This can again be statistically measured by first calculating Fleiss' Kappa over all the questions ($\kappa \approx 0.247$), questions regarding just AI ($\kappa \approx 0.686$) and just the questions regarding XAI ($\kappa \approx 0.557$).

In Figure 5.5, we asked participants multiple questions about explainability. Notably, the introduction of ML in software harms the acceptance of an explanation. Also, we can see that while participants answered mostly that they understand how AI works, their understanding of how AI reasons is lower. These two questions are somewhat interesting, given that understanding how something works may arguably also include how it reasons. However, this is not the case here. On the other hand, one could argue that understanding reasoning is something more instance-specific, which is why the understanding is lower. Next, we asked participants whether they have prejudices against AI because of an assumptive lack of explainability, which was mostly agreed upon. This is also interesting, given that a majority of the participants do apply or use AI. The next set of questions were all related to establishing if there is an agreement on the importance of explainability as a non-functional requirement in SE, which is favored in strong agreement between participants. Again, there is a stronger emphasis on applying explainability in conjunction with AI (especially ML).

Figure 5.6 asked which parts regarding an explanation are valuable to a participant (compare to Section 6.3). While participants chose to rate every mentioned aspect to be important ($\kappa \approx 0.409$), the reasoning was chosen more often than others. The presentation of an explanation seemed to be the least important. However, the results are not distinct enough from the other aspects to be definitive. In the free text fields, participants also valued the time needed to understand an explanation (one participant) and the importance of correctness (one participant). One participant also seemed to favor the term observability more than explainability. The idea is that with observability, explainability can be achieved "externally". This notion does relate to the difference between interpretability and explainability, which

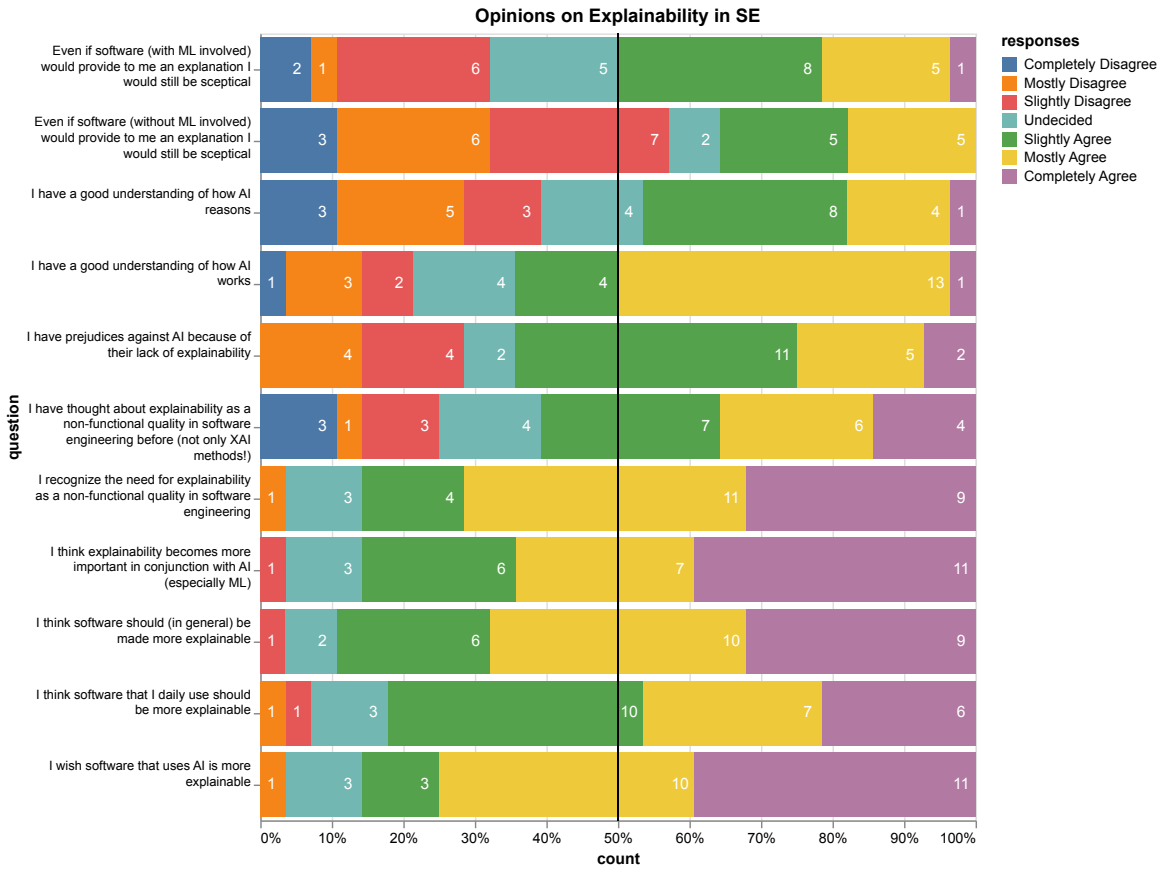


Figure 5.5.: Opinions about Explainability in SE.

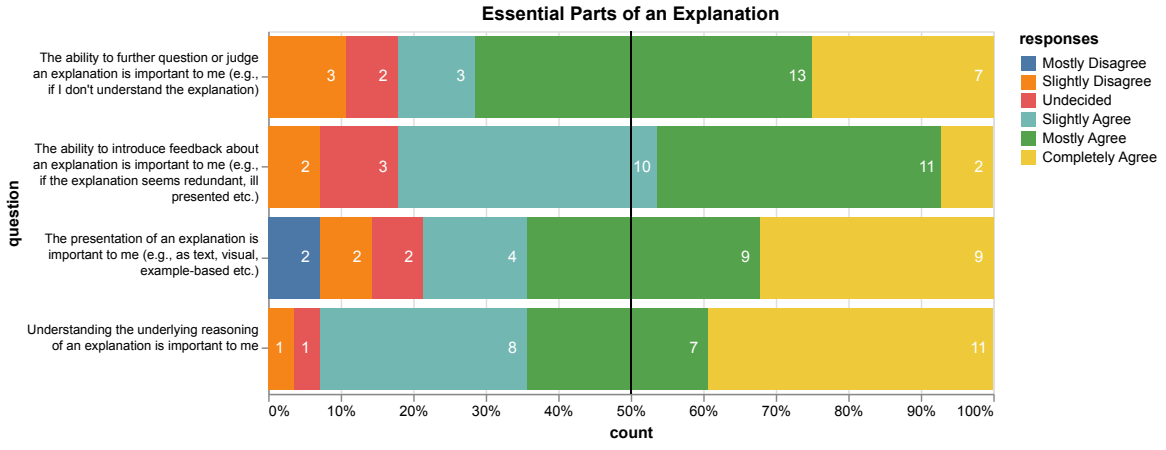


Figure 5.6.: Essential Parts of an Explanation.

we defined in Chapter 2. However, as we stated, interpretability is passive and does not necessarily imply explainability.

In the last question of the first survey question page, we asked participants when they would accept an explanation (see Figure 5.7). The results show that while most people

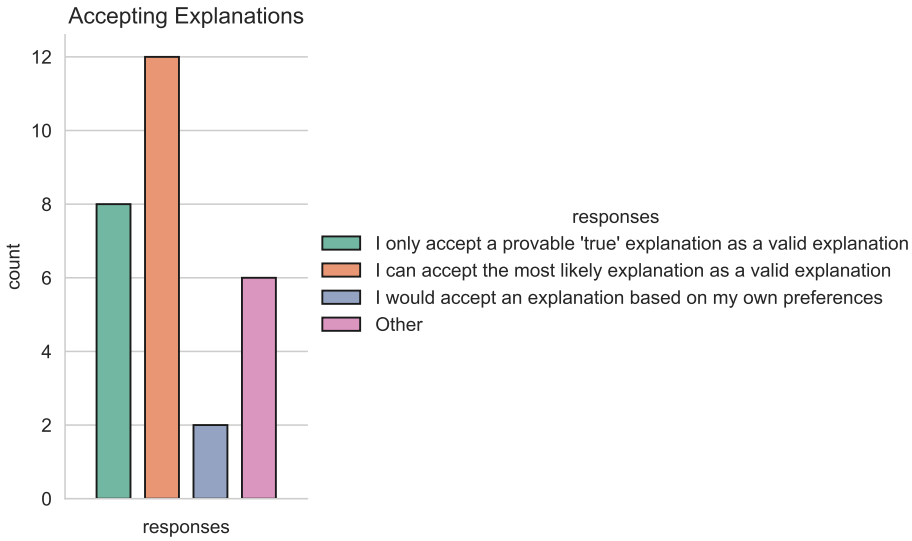


Figure 5.7.: Which Kind of Explanation is Acceptable.

can accept the most likely explanation as valid, this is not enough for the second-highest number of participants, and they demand provable true explanations. In the category of others, people accept an explanation if they also know how likely it is preferable as a score (2 participants). Some would make it context-dependent whether the option exists to have a provable explanation or not, then they would demand a provable explanation (2 participants). Lastly, two other participants declared they wanted a reasonable or objective explanation. Arguably, objectivity is a stronger constraint on an explanation than reasonability because objectivity is only given when a claim is true, even outside the viewpoint of any subject. For example, reports after certain incidents like bridge collapses are usually written to make what happened reasonably explainable. However, if the explanation outcome may be reasonable, it is unclear whether it is also objectively true absent a particular viewpoint of a given subject evaluating the incident. Another example is scientific results under uncertainty, e.g., user studies. What would an objective explanation for a potentially valid conclusion look like in this case? Analyzing the results, one evident thing is that they are distinct opinions, with different participants having different constraints on explanations.

Regarding the image of the cat shown with a heatmap overlayed, participants strongly preferred to see multiple examples before they trusted the explanation. However, they still favored the heatmap as a means of understanding the reasoning of the AI. Regardless, while they somewhat accept the heatmap as an explanation, it is not enough to suffice as a satisfying one.

With the added textual explanation that uses simple reasoning, we could observe a notable shift in the perception of the explanation. Skepticism about the explanation decreased, and satisfaction with the explanation is now over 70%. Notably, only the combination of heatmap and textual reasoning can achieve this level of satisfaction among the participants.

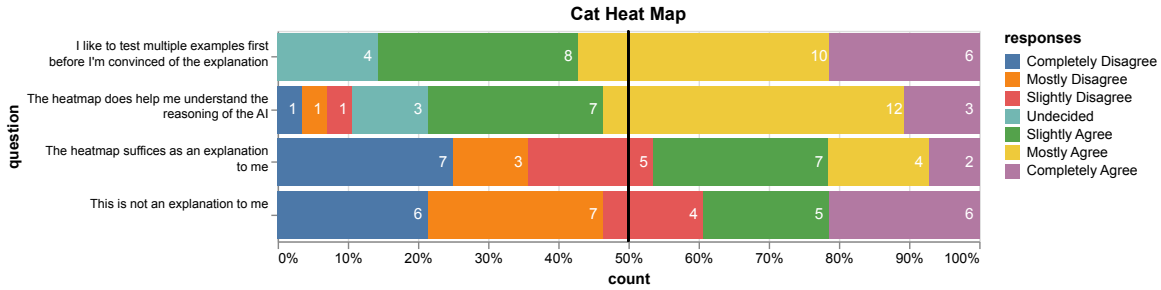


Figure 5.8.: Explanation Acceptance Solely Based on Heatmap.

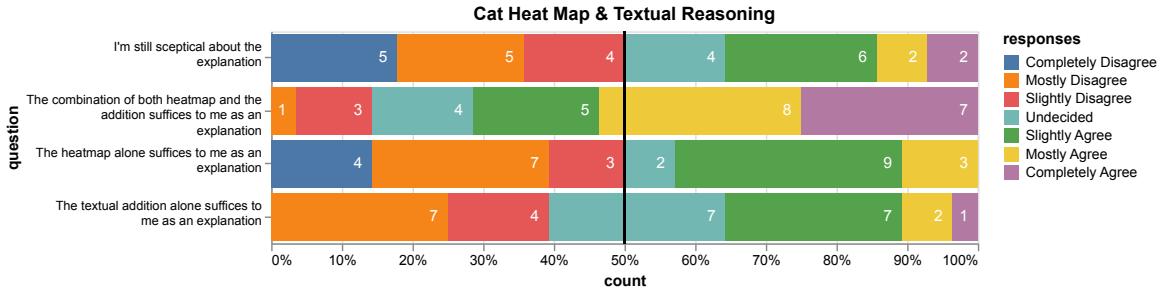


Figure 5.9.: Explanation Acceptance Based on Heatmap and Textual Information.

In this last question, participants were additionally assured that the accuracy of the AI was very high. This question was slightly changed at approximately one-fourth of the survey because participants wanted to answer something "other" than the pre-defined responses. This option was then added. Interestingly, explanation satisfaction was not measurably as high as the question before. Indicating that the satisfaction is still too ambigous to say something clearly about it. Others include total satisfaction (2 participant), relatively more satisfaction but with further own examples (3 participants), own agreement with the AI (2 participants), undecided depends on consequences (1 participant).

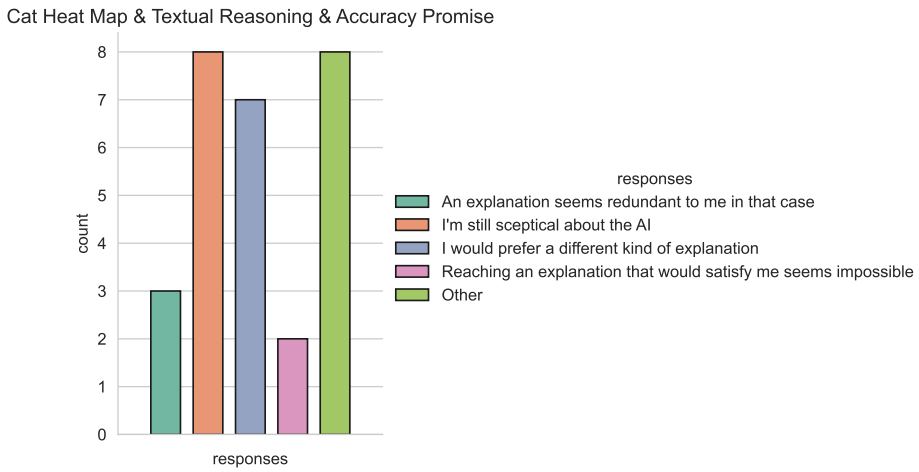


Figure 5.10.: Explanation Acceptance Solely Based on Heatmap, Textual Information and Accuracy.

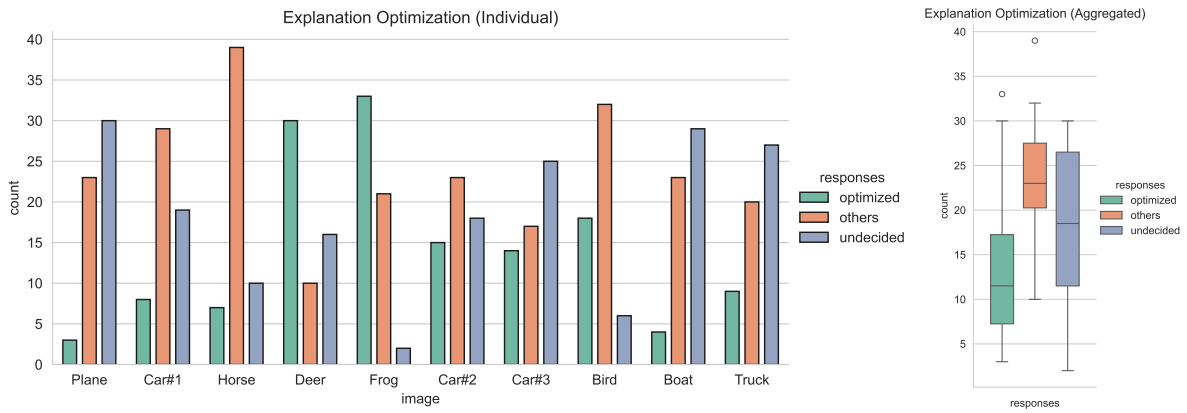


Figure 5.11.: Comparing Optimized Explanations against other XAI methods.

At last, we wanted to know whether our approach of optimizing explanation via aggregation will result in perceived better explanations (see Figure 5.11). As mentioned above, participants were shown ten different images from the CIFAR-10 data set. We created different heat maps with different XAI methods (the same XAI methods and metrics as in the other experiments) from these images. Participants saw two question pages (out of four) for each image, with four possible answers each, the optimized version, two other XAI methods applied, and a "cannot decide" button. First of all, we noticed that certainly numerous participants used the option rather not to make a decision. Because of the simplicity of the survey, this behavior could be explained by the fact that no additional context was given to the participants to make a proper decision, or the XAI methods were perceived as too similar. Apart from the "dog" and "frog" images, most participants did not choose the optimized explanation. Our results are also confirmed by performing a t-test regarding the null hypothesis that *optimized* boxplot's mean is statistically significantly different than the *others* (p-value ≈ 0.0328) and *undecided* (p-value ≈ 0.0375). We also conducted a binomial test on the total number of picking the optimized version ($k = 141$) compared to the total number of trials ($n = 28 * 10 * 2 = 560$) and an expected probability of twice than an even distribution ($p = \frac{2}{3}$) resulting in a clear result that the probability should be less than the expected one (p-value ≈ 0) [195]. Calculating the confidence interval of the result (with $ci = 0.95$) and the alternative hypothesis that the probability is less than expected gives a range from $[0, 0.2838]$. So, the probability of choosing the *optimized* version should lay somewhere between $[0.2838, \frac{1}{3}]$. These are important results that indicate that even after optimizing an explanation according to different XAI metrics, this does not necessarily map to user satisfaction with the explanation. So, while XAI metrics seem to be numerically interesting, their value in an application for the receiving end of a user is not measurable. One could even argue that our results indicate a negative correlation because participants saw the optimized explanation two times more than any other XAI method. While conducting the survey, two participants commented in person about the heat maps. They stated that they most likely chose the one heatmap that precisely encapsulated the object in question and potentially highlighted certain important features. This opens up an interesting question: Each applied XAI method validly explains the AI in question. However, statements like these indicate that participants have problems separating their mental state from how an

explanation should look versus what is a potentially valid explanation, as presented with the different images. Sadly, we were not able to obtain more statements from participants. If these indications apply to other participants, then it stands to reason that the development of AI should integrate explainability as a requirement very early on and develop methods that allow for some form of integration of the mental models of users. Otherwise, users – as these two participants – will end up dissatisfied with the explanation of an AI, regardless of whether it is a valid explanation or the AI is good at what it is supposed to do. This insight also aligns with the theory of abduction we will investigate in the following Chapter 6.

For all of the figures above, we also investigated whether there were noticeable differences in the answers of participants according to their highest academic degree, but this was not the case (apart from the self-reported qualifications).

5.4. Threats to Validity

We now want to discuss the threats to the validity of our survey. This section is divided into four subsections, each referring to a distinct consideration.

5.4.1. Construct Validity

In this subsection, we want to discuss whether our survey measures the target to be measured. For most of the survey, we wanted to discover specific indications matching this thesis's content. Three of the four survey sections were more or less dedicated to discovery, while the last section focused on evaluating the explanation optimization approach. Therefore, the first three sections are fine regarding construct validity. However, readers are advised that the goal was only to show an indication. This indication would be subject to further research on one's desires. For example, our explanations were limited to the domain of image classification, and our methods of explanation were also limited. Other researchers have conducted user studies with a more diverse set of explanations [107]. For the last section of our survey, when comparing the optimized explanations against other XAI methods, we should have introduced different kinds of explanations other than heatmaps for comparison. At the end of the survey, one participant commented that the heat maps were not intended for user consumption; therefore, this would explain the high number of participants answering "undecided". Hence, some participants stated they wanted more context in this scenario to be able to decide between different instances. To summarize, it stands to reason that our survey only applies to this very scenario of heat map comparison and not in general.

5.4.2. Internal Validity

The survey participants were primarily people who have or want to achieve an academic degree in computer science. Therefore, the validity of our survey is also limited to this

specific target group. While we wanted to have a more diverse set of people, this was not feasible in the short time window we had planned for the survey. To increase the internal validity, we shuffled and rotated the question as mentioned above. However, the questions themselves were selected based on fulfilling the interest of this thesis. Usually, every question had the opportunity to be answered negatively or positively. Sadly, the dropout rate for this survey was very high. To mitigate the severity of statistical side-effects in including partial results, these must have been omitted from the analysis. Furthermore, because it was an anonymous online survey, we could not ask for the reason for not completing the survey, which would have been interesting to know. We also had planned to integrate the randomization of the question order for other questions. However, we ultimately went against it because we were unsure if introducing randomization would also add noise to our analysis that may not be fully explainable by the change of order. Moreover, we would have to acquire more participants to measure statistical relevance. However, this would have allowed us to measure a priming effect on participants in some relevant instances, such as the question related to the concept of abduction with logical reasoning.

5.4.3. External Validity

The external validity reasons about to what extent a cause-and-effect relationship can not be explained by the study itself but by other external factors. In this instance, this could be the daytime and mood of the participants. Because we did not have any controls in place for external factors, we do not have any means to assess the severity of this aspect. We tried to make our survey the most accessible and readable by introducing custom changes to the layout so that participants were not forced to use a specific device to participate in the survey. Furthermore, there may be external factors in place as to why the dropout rate of the participants was so high, such as time pressure to arrive at certain events. However, we could potentially reach more people because we did not require participants to be there in person to participate in the survey. Furthermore, this could have reduced the Hawthorne effect of participants behaving differently because they know they are being studied.

5.4.4. Repeatability

One crucial factor for the repeatability of our survey is the selection and generation of the images for the survey. While the images we provided are all generated based on the XAI methods and metrics we used in previous experiments, we can not guarantee different results when recomputing the images. There are just too many randomized factors that inhibit this. For example, choosing the weights by which the aggregation happens is controlled by an evolutionary algorithm. Which strongly makes use of randomization to calculate a solution. We did investigate multiple runs of the same computation to inspect how much this could impact the results visually, and we did not see any hard evidence to assume otherwise, but this is still something to be aware of. This also relates to our results about the predictive multiplicity in Section 4.4 Experiment 2, by which we investigated the severity of the predictive multiplicity with different XAI methods. We could measurably show

differences between consecutive runs only by simply rerunning the same experiment with different XAI methods.

5.5. Lessons Learned (1)

In this Section, we briefly iterate over what we have learned from the previous Chapter 4 and this Chapter. However, instead of re-iterating each result, we want to take a bird-view of the results presented.

We have ultimately learned from each of the experiments and user survey that while we could numerically improve explanations and investigate different aspects of explainability in the context of FL, the most important factor in explainability remains the human evaluation part. Based on the limited context of our research, explainability needs to be approached from a human-centric approach to be meaningful for end users. Still, while we could show no correlation between optimizing explanations with XAI metrics and user acceptance, this does not mean that our numerical results should be discarded as not meaningful. For some applications, it is important to improve specific metrics or stability as we have done, e.g., robustness against attacks.

The next Chapter will, therefore, be focused more on the human-centric side of explainability by taking a multidisciplinary approach to explainability derived from a literature search. In said Chapter, we will see how nuanced the concept of explainability can become and why abduction plays a crucial role in accepting explanations.

6. Conceptualizing Explainability

This Chapter aims to introduce, exemplify, show, and reason about nuances when it comes to explainability that are often not accounted for. While we rather gently introduced explainability in the Foundation Chapter 2 – as it is often done in contemporary research literature – this Chapter will start on a very high level for explainability and focus on essential and basic elements thereof. The goal of this Chapter is to create a shared understanding of what is desirable from explainability and why it should be pursued in the presented way[1].

6.1. Goals and Questions (3)

We define the following research goal and associated questions for this Chapter:

- **RG1**: Examine the possible viewpoints regarding explainability from other disciplines.
  - **RG1.Q1**: What are different viewpoints on explainability in the research literature?
  - **RG1.Q2**: What are the most common elements found in these viewpoints?
  - **RG1.Q3**: How can a categorization of explainability be approached?

Granted, our GQM plan cannot define metrics for this Chapter. Hence, we try only to argumentatively examine different viewpoints on explainability and not evaluate their applicability. However, as stated before, we will synthesize our findings at the end of this Chapter.

6.2. Entangling Explainability

In order to assess the explainability of a system, we have already presented several definitions and metrics in Subsection 2.1.1 of the Foundation Chapter. However, to approach a revision of the *levels of explainability readiness* for a broader spectrum of applications, we first need a means to talk about explanations in a more pronounced way, enabling us to reason about explanations at an abstract – but still tangible – level. While residing in a computer scientist's perspective seems tempting at first, it will most certainly limit any conclusion to

[1] This Chapter is solely argumentative; no evaluation of the proposed characterization is done, unlike the previous Chapters.

be drawn by technological means. Therefore, we will propose a different, multidisciplinary approach. This section will present various articles [46, 85, 96, 103, 104, 125, 126, 174] and their work closely related to describing and modeling explainability. After introducing the core ideas of each source, we will try to synthesize our findings. This way, readers are invited to follow the thinking process that leads to our conclusion and, later on, to our model for assessing the explainability of a system.

6.2.1. Explanations as Proofs

It is reasonable to assume that while we already defined the term *explanation* in Chapter 2, other fields of research use different definitions. Indeed, "[t]heories of explanation date back at least as far as the times of Plato and Aristotle [...]" [174]. Aristotle, also commonly known as the father of logic, inspired the predecessor of all other research fields: philosophy. In philosophy – or more precisely, the philosophy of logic – explanations can be seen as proofs [174]. These proofs can be categorized into *deduction, induction, and abduction* [85, 125].

Definition 6.2.1 (Induction) *"An inductive generalization is an inference that goes from the characteristics of some observed sample of individuals to a conclusion about the distribution of those characteristics in some larger population." In essence, induction is a form of generalization.*

All observed A's are B's.

Therefore all A's are B's. [96]

Often, in inductive reasoning, it is necessary to distinguish between the "event of observing some fact and the fact observed", because the conclusion in the induction can only explain the observation but not the underlying facts that are indeed being observed. This becomes evident in a simple example of drawing randomly balls from a hat, where all balls are red. If one draws a ball from the hat, we anticipate that the ball must be red (because all observed balls are red). However, the induction can not explain why the particular instance of a ball is red in the first place; it can only suggest that there may exist a cause in relation to all observed balls being red (all A's were B's). The generalization can not explain the instance itself [96]. Inductive processes also play a crucial role in how we currently conduct XAI. For example, if one trained an ML model to classify cats and dogs with high accuracy and then wanted to explain the classifier with current state-of-the-art XAI methods, one would probably observe that all images that contain whiskers and pointy ears are correctly classified as cats. The XAI method would highlight these features – as shown in Figure 6.1. Given these observations, we can now assume, or rather create the explanatory hypotheses, that the classifier will likely classify every image with whiskers and pointy ears correctly as cats. However, this conclusion is misleading and leads to a wrong understanding and usage of XAI methods. In this example, we implicitly based our explanatory hypothesis on inductive reasoning without proper recognition thereof. Notice the distinct difference between explaining an observed event and explaining the underlying fact, as we mentioned

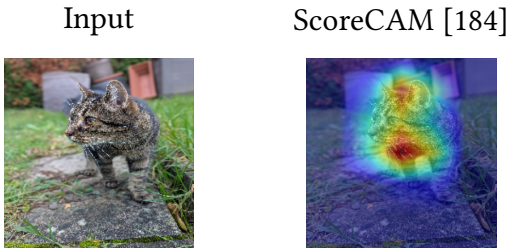


Figure 6.1.: XAI Method applied to ResNet-50.

above. Rather than explaining why a particular instance is a cat, we only (can) explain the observed event that all images containing whisker, and pointy ears are cats, but not why this is the case in the first place. This example makes it even more necessary to recognize XAI methods as explanatory hypotheses-forming techniques rather than – strictly speaking – explanation techniques because one would assume that an explanation provides also the necessary reasoning of the underlying fact. By this reasoning, induction does not suffice to cover the explanatory hypotheses space adequately.

Definition 6.2.2 (Deduction)

$$\frac{A \longrightarrow B \text{ and } A \text{ is true.}}{\text{Therefore } B.}$$

Deduction follows an inverse approach to induction. A statement is deductive valid, *if and only if (iff)* the conclusion follows from the premises [96]. We can observe deductive reasoning, for instance, in rule-based systems. Applying XAI is straight forward in this case because the rules explain the prediction of the model. If one wants to execute a model prediction on a specific instance, then all that needs to be done is to evaluate the rules leading to a conclusion. However, therein lies another fallacy that is often omitted because, in deductive reasoning, one must distinguish between validity and soundness. Referencing our above cats and dogs example, we can easily create a valid deductive explanation of why the classifier predicts cats correctly. Given the two premises, which are that all cats have whiskers and all dogs have no whiskers, one can deductively reason that the classifier is valid if it recognizes an image with whiskers and predicts a cat (see Figure 6.2). However, validity alone does not suffice for soundness because, for soundness, the premises must be true in all cases. If, however, our classifier sees an seal, and recognizes the whiskers,

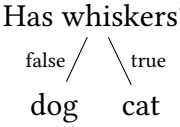


Figure 6.2.: Simple Rule-based Classifier.

then it would be a valid deductive argument to assume, that the seal is a cat, but obviously this is not a sound explanation[2]. Again, the term explanation is overloaded and unspecified compared to the correct depiction of explanation forming through generating explanatory hypotheses. By reasoning about the example mentioned above, we still need to catch – something – to cover the explanatory hypothesis space adequately. In the next paragraph, we will introduce the concept of abduction, which is essential for understanding what we mean with explanatory hypothesis space in the first place.

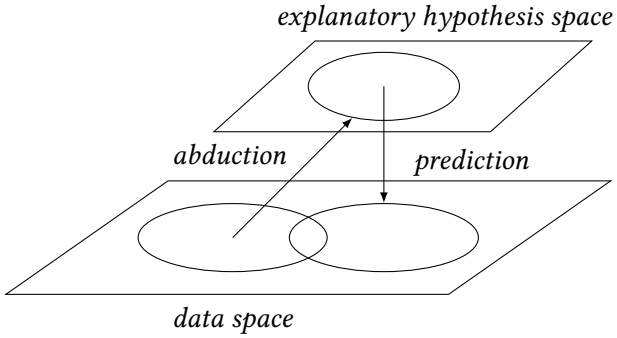


Figure 6.3.: Visualization of Abduction in Contrast to Prediction [96].

While induction and deduction are undisputed parts of any valid proof-based system, abduction stands out as a somewhat controversial inclusion [85, 96, 130].

Definition 6.2.3 (Abduction in Philosophy) *"Abduction, or inference to the best explanation, is a form of inference that goes from data describing something to a hypothesis that best explains or accounts for the data. Thus abduction is a kind of theory-forming or interpretive inference."*

D is a collection of data comprising facts, observations, assumptions.
H explains D sufficiently.
No other hypthoses explains D as well as H.

H is probably true. [96]

The concept of abduction – stipulated by Charles Sanders Peirce (1893) – has been known in AI research since at least 1987, when the authors Charniak et al. [31] characterized abduction as "modus ponens turned backward, inferring the cause of something, generation of explanations for what we see around us, and inference to the best explanation" [96]. It should be noted that abduction is not limited to the generation process of the explanatory hypothesis but also includes the evaluation of which explanation is probably considered the best explanatory hypothesis [96]. In contrast to deduction, abduction is a matter of judgment of likelihood and acceptance of the explanatory hypothesis. The considerations

[2]There exists also selective cat breeds without whiskers.

of accepting an explanatory hypothesis are primarily based on plausibility, explanatory power, and pragmatic considerations:

1. Does H decisively surpass all alternatives?
2. Does H decisively support itself (not considering weighing alternatives)?
3. Is the data reliable?
4. Is there confidence that all plausible explanations have been considered?
5. What are the costs for inferring and evaluating alternatives?
6. Does a decision need to be taken promptly?

As the authors Josephson further state: "[A]bductive inference depends on an evaluation that ranges over all possible hypotheses, or at least a set of them large enough to guarantee that it includes the true one." While it is generally anticipated that the *best* explanation is the true one, in light of not having direct access to a judgment of truth, one needs to fall back on "a summary judgments of accessible explanatory virtues. [...] Abductions are fallible, and doubt cannot be completely eliminated. Nevertheless, by the aid of abductive inferences, knowledge is possible even in the face of uncertainty" [96]. The authors Josephson, even recognizes the process of abduction as several optimization problems where the target is:

- "[M]aximizing explanatory coverage consistent with maintaining confidence above some preset threshold."
- "[M]aximiz[ing] explanatory coverage while minimizing specific kinds of error costs."
- "[M]aximizing explanatory coverage in a given amount of processing time."

In the article [85] from the authors Hoffman, Miller, and Clancey, they argue that XAI should develop "systems capable of engaging in meaningful interactions with people to support **their** abductive reasoning" (emphasis added). They proceeded to say, "[XAI] has the purpose of helping people develop good mental models of how the AI system works and when, why, and how it fails. [...] The user learns and benefits from the AI, but additionally, the XAI improves based on the actions and feedback of the user, such as improving its ability to adjust [...] or eliminate certain hypotheses". The hypotheses meant here are the explanatory hypotheses of the user interacting with the XAI system. This notion of defining XAI integrates several desirable key aspects. First, XAI involves an active dialog "in which an explainer and a learner collaborate, explore, and co-adapt." Second, it allows the definition of *true* self-explainable systems because users are enabled in their "abducti[on] to understand what, how, and why the AI does what it does." Lastly, "abduction depends on propositions from the reasoner's knowledge – propositions that come from beyond the given rule and the given observation" – which is inherently different from inductive and deductive reasoning. By respecting this last aspect, XAI systems are argued to be aware of

context and the implicit knowledge of the reasoner [85]. As seen in the paragraph above, this is often crucial to build sound explanatory hypotheses. Appendix Table A.1 presents an overview of proposed explainability requirements to support abductive reasoning in AI. Figure A.2 in the Appendix describes the abduction process graphically.

Considering our cats and dogs image classifier example, the abductive process starts with an observation that the deployed XAI method marks whiskers and pointy ears in a given image and classifies them consistently as cats if these features are present. Now, we can start forming explanatory hypotheses about the classifier. The apparent hypothesis from a user is that the classifier detects the whiskers and pointy ears and classifies them as cats. However, this explanation is just a hypothesis that needs to be evaluated first based on the abovementioned considerations. If we consider the explanation sound at this early stage without further evaluation we would just fall back to inductive reasoning. Ultimately, we want AI to help us with our abductive reasoning, particularly in generating and evaluating appropriate explanatory hypotheses. However, in light of not having access to such an AI the user is responsible for the abductive reasoning process. To evaluate the explanatory hypothesis, one could test the classifier with images of animals with whiskers or pointy ears, cats without whiskers or pointy ears, or dogs with whiskers or pointy ears and see how the classifier and the deployed XAI method react to it. While testing these examples sounds daunting, it is the most reasonable approach to ensure that the explanatory hypothesis is the most likely one. More importantly, it enables the user to understand better how the image classifier works without exposure to technical aspects of the ML model. Now, it is likely that a user is not satisfied with the explanatory hypothesis after conducting an evaluation as outlined above. Either it is because of the evaluation results, meaning that the ML model does not perform as expected, or the explanatory hypothesis is still considered unsatisfactory. In both cases, the user gained significant knowledge about the system in question and is enabled to assess it better. Furthermore, it is still possible to evaluate other explanatory hypotheses. Enabling abductive reasoning does not mean enforcing the acceptance of explanatory hypotheses or accepting the predictions of an ML model in general. It is just a way of *theory-forming and inference to the best explanation* [96].

Given this introduction to abduction, we will now reconcile results from the subdiscipline of computational logic in computer science [112]. Here, abduction has been known in the form of *abductive logical programming* since at least the early 1980s [113]. The most prominent figure in this field is Prof. em. Robert Anthony Kowalski, a major contributor to the programming language Prolog and winner of the IJCAI Award for research excellence lifetime achievements in the year 2011 [92].[3] Certainly, his paper, *Algorithm = Logic + Control* is one of his most recognized ones [111].

Definition 6.2.4 (Abduction in Computational Logic) *An abductive program is a triple $\langle P, I, A \rangle$, where P is a logic program, I is a set of integrity constraints, and A is a set of abducible predicates (also known as abducible hypotheses). Given the triple $\langle P, I, A \rangle$ and a goal clause G*

[3] Some of his research is currently under revitalization under the term Neuro-Symbolic AI [165].

(also known as observations), an abductive solution (or explanation) of G is a subset Δ of A, such that:

$$P \cup \Delta \models G$$
$$P \cup \Delta \cup I \quad \textit{is consistent.} \qquad [74, 93, 98, 113, 114]$$

Note: *Sometimes, it is also stated that $P \cup \Delta$ shall be a minimal model, which is automatically the case when using Horn Clauses [113].*

Abduction, as presented in Definition 6.2.4, is commonly referred to as non-monotonic reasoning because abducible hypotheses may become invalid if new evidence or rules are added [68]. The integrity constraints I, are often in the form of denials so that specific abductive hypotheses are rejected, because, e.g., they would result in logical inconsistencies.

Now, reconsider our cats and dogs example from above; it can be represented in the following logical program with P being the program, $\Delta = \{cat, dog, seal\}$ as our abducibles and only one integrity Constraint I, as shown in Line 15.

Algorithm 5 Cats and Dogs Classifier based on abductive logic

Input: Occurence probabilities of $\langle p_{cat}, p_{dog}, p_{seal} \rangle$.

```
1:  % Probabilities assignment.
2:  p_cat :: cat.
3:  p_dog :: dog.
4:  p_seal :: seal.

5:  % Logical Program.
6:  is(X) :- X.
7:  whiskers(X) :- X = cat, is(X).
8:  whiskers(X) :- X = seal, is(X).
9:  ears(X) :- X = cat, is(X).
10: ears(X) :- X = seal, is(X).
11: ears(X) :- X = dog, is(X).
12: animal(X, Y) :- X, Y.
13: animal(X) :- X.

14: % Integrity Constraints.
15: whiskers(X) :- X = dog, not(is(dog)).
```

Algorithm 5 has been written in Problog, which utilizes a subset of the Prolog commands [47, 71][4]. The added benefit of using Problog is that it allows us to add a probabilistic model on top of our logical model, which facilitates the meaning of *inference of the best explanation* because the best explanation can then be expressed as also the most likely one in terms of probabilities. For example, suppose the program is now queried with Query 6. In that case,

[4]Concretely it uses the Yet Another Prolog (YAP) compiler `https://www.dcc.fc.up.pt/~vsc/yap`.

Query 6 Abducting Animals with Ears

```
?- query(animal(ears(X))).
```

$animal(ears(cat)) : p_{cat}$
$animal(ears(dog)) : p_{dog}$
$animal(ears(seal)) : p_{seal}$

it will list all possibilities of what `X` can be substituted for – a process commonly referred to as unification – and additionally respect and show the associated probabilities that `X` is the case. In this example, it would substitute `X` for cat, dog, and seal and their associated probabilities. The process by which Problog tries to solve subgoals that are defined in goal clauses is also referred to as backtracking. To compute explanatory hypotheses like that is certainly a noteworthy achievement and makes the usage of abduction also practically applicable. Also, because we use a logic program to form these hypotheses, our explanations are fully explainable by design.

6.2.2. Explanations as Causal Reasoning

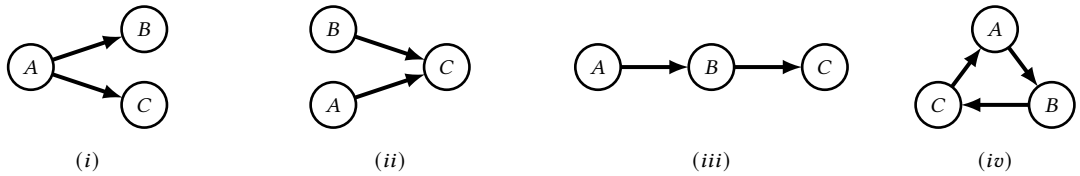


Figure 6.4.: Overview of Common Causal Relations [174].

Causal reasoning is another way of referring to explanations [104, 174]. Causal relations can often take the following forms (see also Figure 6.4):

(i) **Common Cause:** A single cause can be identified that provides all the information necessary to explain an explanandum X. Notice that this cause can result in multiple branches of causes and effects afterward. However, these can be backtracked to this single root cause.

(ii) **Common Effect:** Several causes can result in a singular effect; knowing the causes suffices as an explanation.

(iii) **Linear Chains:** A linear chain of causes and effects. Backtracking the chain suffices as an explanation.

(iv) **Causal Homeostasis:** These explanations reason about the reinforcement or circle-like stability of a systems property and follow the form $A \rightarrow B \rightarrow C \rightarrow A$. For example, a good mood leads to work being done, which leads to self-esteem, which leads to a good mood. Notice that every point in this chain can suffice as a starting point for an explanation.

For humans, these cause-and-effect relations often suffice as an explanation. While it is related to the proof-based reasoning we introduced at the beginning of the section, it is still different, because it does not have this strict connection to the field of logical calculus in philosophy. Another instance of causal explanations are counterfactuals – commonly known as "*what if?*" questions. Producing counterfactuals "involves imagining alternatives to one or more features of a perceived event" [131]. To produce a counterfactual explanation, one needs to be able to reason about cause and effect, which shows a form of high intelligence we desire in intelligent computer systems [131, 173]. In XAI, counterfactuals are also perceived as intuitive and helpful in generating explanations that satisfy users [138]. Indeed, in the previous section, we deliberately used counterfactuals to evaluate the explanatory hypothesis for our image classifier example by imagining and then evaluating variations of specific features in our explanation.

6.2.3. Explanations in Social Science

Lastly, explanations can be seen as related to social systems [46, 103]. In [103], the authors applied Luhmann's social system theory (see Definition 6.2.5). The three aspects of information, utterance, and understanding are further highlighted by the authors Keenan and Sokol and directly mapped to XAI. The viewpoint from social science is different from others because it pronounces the importance of modeling relationships and the effects thereof between agents, as well as distinguishing understanding from communication.

Definition 6.2.5 (Luhmann's System Theory) *"Systems theory is [...] concerned with the conditions and operations of meaningful communication. According to the theory, society is a complex of self-referential autonomous systems of communication. Communication is a self-organising process of differentiation that, independently of any central control, evolves and differentiates codes and structural processes, and does so using only its own processes. Systems make meaning possible by reducing the complexity of the world in order to communicate about it, which in turn makes society more complex as it communicates about itself and its environment. [...] [N]o system communicates directly with any other system and information cannot "transfer" from one system to another. Rather, each system observes other systems as elements in its environment, and responds to its observations of other systems only on its own terms. [...] In each communication, understanding is the key moment; it occurs through the drawing of a distinction between selected information and utterance. Understanding – the making of meaning – is observer dependent and arises in the decoding of what is communicated from how it is communicated" [103].*

The authors Dazeley et al. also regard explanations as a social process. In their paper [46], they present a framework called *Broad-XAI*. This framework divides explanation into a social and cognitive process. First, they present levels of explanation as a bottom-up model inspired by animal cognitive ethology's levels of intentionality. This model comprises:

- **Zero-order (reactive) explanations:** "an explanation of an agent's reaction to immediately perceived inputs." These explanations are considered the foundation for all other explanations.
- **First-order (disposition) explanations:** "an explanation of an agent's underlying internal disposition towards the environment and other actors that motivated a particular decision." These are questions regarding the agent's "current internal disposition and how it influenced its reaction."
- **Second-order (social) explanations:** "an explanation of a decision based on an awareness or belief of its own or other actors' mental states." Questions regarding the anticipation of other agents' behavior are included here.
- **Nth-order (cultural) explanations:** "an explanation of a decision made by the agent based on what is determined is expected of it culturally, separate from its primary objective, by other actors." "This level of reasoning is equated to third-order intentionality [171] because person A, not only has a model of what person B will do, but recognises that person B will expect person A to do likewise — and vice versa. This represents an ever increasing recursive level of mentalisation [128,129] indicating an understanding of a set of cultural rules about behaviour."
- **Meta (reflective) explanations:** "an explanation detailing the process and factors that were used to generate, infer or select an explanation." Encompasses the idea of reflection on the explanatory process. This process happens orthogonal to the levels above.

In addition to what they call *social process*, they also suggest a *cognitive process* that utilizes (i) perception, (ii) a Merkwelt Model (agents' internal model), (iii) an actor's Model (model of other agents), and (iv) a behavior model (mapping that combines all aspects mentioned into agent behavior) [46]. The concept of a Merkwelt Model is derived from *mental models* in cognitive sciences. Mental models are internal, highly individual representations of the workings of a system. Explanations are derived from instantiation and interpretation of the mental model, which often involves simplification [104, 174]. Again, this model is targeted more towards agent-based systems. Indeed, the authors applied this concept to reinforcement learning-based agents in a recently published article [45]. Our simple image classification example does arguably not incorporate any social or cultural component that we can leverage; therefore, we propose another example based on connected autonomous vehicles. In this example, autonomous vehicles need to be able to react to observations they made of the environment, for example, seeing a stop sign. Furthermore, they often need to anticipate their behavior in advance, for instance slowing, down when seeing a stop sign or changing lanes. However, they also need to anticipate the behavior of others, such as pedestrians crossing the road. While we could attribute these effects to causal reasoning, it is somehow still distinctively different than that because it does not capture the cognitive elements associated with a *social process* (e.g., thinking about the internal disposition of others). However, it is also different than a solely *cognitive process* because it does not holistically capture the complexity of what scholars mean when they use the term *society* to describe such a system.

6.2.4. Abduction as the Unifying Element

In light of the research presented above, a general categorization of explainability levels is still not directly feasible, at least if we strive for completeness and a multi-faceted view of explainability. While each of the papers above presented a unique approach to explainability, they inherently still share common elements that can and – we argue – should not be separated.

Instead, we propose a different approach with abduction as the essence of explainability. We adopt the notion of an *explanatory hypothesis space* that can be utilized to refine explanations. This way, we can define explainability as a problem of choosing the right explanatory coverage, as presented above. Based on our research, we propose dividing the explanatory hypothesis space into *social, cognitive, causal, and abductive hypothesis space* $\mathcal{H}$. Figure 6.5 visualizes this approach. The distinction is adopted as presented in the section above. While we show a distinct separation, this may not always be true. The task of explainability practitioners is now to find the explanatory hypothesis space that most suits their individual needs. We could enforce a separation by further specifying boundaries – e.g., through associated characteristics – but we argue that the notion of "correctness" remains ambiguous. The proposed model also aligns with most current available XAI methods because we do not have – and neither produce – ground truth explanations. Instead, XAI methods generate explanation hypotheses based on mathematical properties that are then assumed proxies of a ground truth explanation. Introducing the notion of explanatory hypothesis does clarify this difference decisively. We now want to build upon our introduced notion of $\mathcal{H}$ and explore characteristics of it. In the paper: *"Explanation Is Effective Because It Is Selective"*;

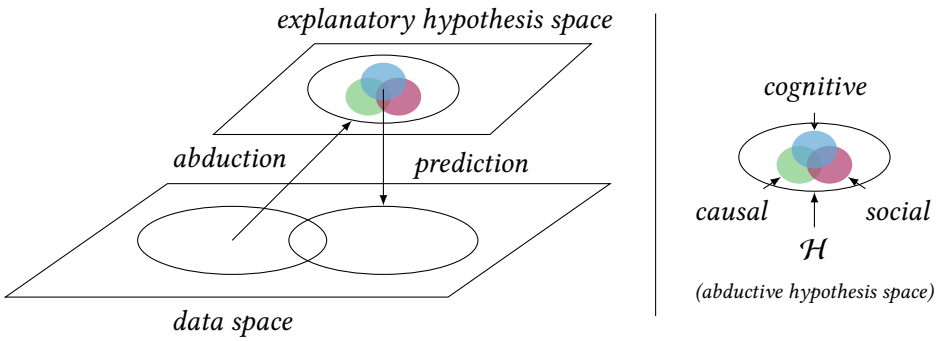


Figure 6.5.: Combining Different Views with Abduction.

the authors reason about two observed phenomena in human explanations [126]. First, we observe that the process of "seeking, generating and evaluating explanations [actively] support learning and generalization[…]." By seeking explanations from others and world phenomena, as well as explaining them to others and ourselves, human learning can be facilitated. They argue that this process works most for humans because of the selectivity by which we seek and evaluate explanations. This leads to the second observation: Humans tend to seek explanations only for a subset of encountered phenomena. Not only that, even if we seek explanations, we are also selective in evaluating what we consider a satisfactory explanation or not. We can map the concept of phenomena selectivity and explanation

satisfactory to our introduced notion. Phenomena selectivity can be interpreted as the fact that only some points of our data space can start the process of abduction, which results in the seeking of explanations. We notice that this introduces another necessary adjustment to our unified model. Our data space now needs to be able to capture the phenomena selectivity. One approach would be associating the abduction process with probabilities dependent on the subject that seeks an explanation. Another approach would be to redefine the data space as inherently subjective, so instead of considering it as ground truth and universally agreed upon, the data space itself depends on the subject in question. So that different subjects perceive the data space differently. On a side note, the last notion ties back to the Rashomon Effect we described in the Foundation chapter. The name "Rashomon Effect" was chosen by Breiman because of the equally named Japanese movie 羅生門, Rashōmon [23]. In this movie, each character witnesses the same crime but testifies differently. It becomes evident that human perception can never reflect the exact reality. This finding is fundamental in today's philosophical, epistemological theory – also known as the "problem of basic statements" [152]. In that sense, we do not introduce a notion of ambiguity in our model by introducing subjectivity, but instead, we recognize the fact that reality itself is perceived subjectively. We still need to address the selectivity of what is deemed a satisfactory explanation. In the paper above from Lombrozo and Liquin, they argue that selectivity comes partially from the intrinsic recognition that selectivity promotes effective learning. In essence, selectivity serves a purpose or goal even if the individual does not recognize this directly. The paper itself limits the purpose to promoting effective learning. However, we argue that this can be further extended to any goal an individual seeks, either knowingly or unknowingly. While the authors Josephson primarily focus on logical and pragmatic conclusions to accept an explanatory hypothesis, we argue that this notion lacks *human characteristics* [96]. In our view, the acceptance of an explanatory hypothesis comes from a sufficient overlap of an individual's value system with the perceived intrinsic and extrinsic values that an explanatory hypothesis provides. If the perceived sum of values exceeds a certain threshold, an individual considers the explanatory hypothesis satisfactory. This also means that the explanatory hypothesis space $\mathcal{H}$ must include all necessary information to deem an explanation satisfactory. The segmentation of the hypothesis space $\mathcal{H}$ into the fundamental components of *social, cognitive, and causal* that we collected by literature research provides the most common denominator we found.

While keeping our current abstraction level[5], we can now define the properties of robustness, faithfulness, and complexity on our unified model [40]. Robustness in XAI – or often also called sensitivity – is usually defined as a measurement of how much explanation changes when the input is perturbated (often considering only small changes) [40]. Looking at Figure 6.5, we can see that our explanations suffice robustness if the abductive hypothesis space $\mathcal{H}$ stays the same after perturbation in the data space – or at least close to the same when we allow for minor changes. However, this property is insufficient for faithfulness because different abduction processes can still come to the same abductive hypothesis space $\mathcal{H}$ regardless of whether the data space stays the same. For this reason, faithfulness must suffice that the predictions we draw from our abductive hypothesis space $\mathcal{H}$ points back to the data space that started the abductive process. Lastly, complexity measures

[5]For more formal and concrete definitions see [7, 8, 18, 40, 97, 141].

how complex explanations are for human comprehension. For this, we assume that the representation of an explanation is encoded in our explanatory hypothesis space $\mathcal{H}$. This ties back to the selectivity of explanations we reasoned above, with the result that the explanatory hypothesis space must encode all information necessary to deem an explanation satisfactory.

Important: Why do we need abstractions?

As the authors Josephson point out: "When reasoning about complex systems, both human experts and expert systems are necessarily compelled to use high-level, qualitative symbols, whether or not complete numerical models are available."

This can be exemplified in a simple image recognition task of a cleaning robot.

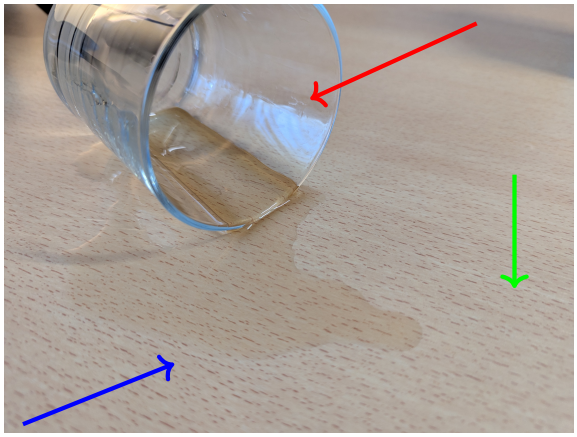


Figure 6.6.: Image of a Spilled Cup of Liquid.

Even if the operation and perception of the robot can be described numerically, to recognize that the cup has been spilled, it is far easier to reason about it on a high-level symbolic abstraction than on a numerical model. Given the image, simply by inferring very basic primitives like "table", "water", and "cup vertical", we can abduct that the most likely explanation of this scene is that the "cup has been spilled". This high-level symbolic reasoning is challenging for pure neural network-based AI because it had to be trained on it beforehand – with a potentially large set of training data, which might not be available – and furthermore, it could still produce unpredictable, undesired results – defying even the most basic reasoning. With abduction and our unifying model, we have at least a model that acknowledges that an explanation might be wrong. Moreover, this type of reasoning about the underlying problem structure would not be possible without the abstraction used throughout this whole chapter for reasoning about explainability. While human experts sometimes need to switch to the numerical representation, "the numerical analysis [remains] under the overall control of a qualitative, symbolic reasoning system [by the human expert]." [96]

To foster the validity of our model, we will now apply it to an example in the autonomous vehicle industry.

Example:
A large number of scholars consider autonomous driving vehicles beneficial to society because they promise to reduce accidents and improve economic aspects like energy consumption. This becomes especially important in light of yearly ≈1.19 million deaths as a result of car crashes and the fact that road injuries are the leading cause of death for five to 29-year-olds [145].

For our example, we consider an autonomous vehicle, which satisfies level four in the J3016 standard proposed by the Society of Automobile Engineers [158]. These vehicles are capable of autonomous driving under certain conditions, which include lane changing – which is amongst the riskiest maneuvers –, pedestrian detection, and advanced path planning [148]. Now, to apply our unified model, we need an observation that sparks the interest of the vehicle driver to start the abductive reasoning. In this example, we may consider the autonomous changing of the lane as the starting point of abductive reasoning. Noticing the lane change, the driver wants to know why the car just changed the lane; if the driver is used to similar behavior in the past, it is very likely that an explanatory hypothesis has already been formed, and only the confirmation thereof is necessary – granted if the driver even deems this necessary. On the other hand, if the driver is not aware of an explanatory hypothesis, further assistance from the vehicle is required. This assistance should provide the driver with at least one explanatory hypothesis as to why the vehicle executed the lane change. There could very well be multiple reasons, but ultimately, the one that is the most satisfactory will also be the one that persuades the driver to accept it the most. For the driver to accept an explanation, the explanation should encompass *social, cognitive, and causal* components that align with the driver's value system. Consider the following explanatory hypothesis provided by the vehicle: "Lane change was sought out to save fuel." This explanation can be inherently problematic given a particular driver's value system; while it may be socially adequate to save as much fuel as possible to reduce the environmental footprint, certainly not all drivers would be satisfied with such an explanation. If this were not an abductive process, we would now stop further reasoning because the system gave its explanation. However, through the abduction process, we can utilize another process: the co-adaption between driver and vehicle. This co-adaption process aims to allow the driver to explore the explanations provided by the vehicle and ultimately will enable the vehicle to understand the driver's disposition and adapt itself to it. For example, the driver might further ask the vehicle how much fuel it saves, and the driver deems the value of the answer as negligible and can give the vehicle feedback on what value he would consider significant enough to justify lane changes in future events. Counterfactuals as part of the causal reasoning can also play a role here; given the answer of the vehicle, a counterfactual question would be: "What if the fuel saving is half of that? Would this still justify the lane change?" Notice also the implicit *causal reasoning* of lane changing as a cause for the effect of fuel being saved, which we have not explicitly acknowledged yet. We could also attribute this reasoning as inductive – therefore *cognitive* – assuming that changing the lane to the slower one does, in general, save fuel.

An interesting use case for FL is given in the medical industry and will be presented in the following. The to-be-presented use case actively uses FL and XAI. However, we will also include additional concepts like differential privacy if needed. This way, we can reason about explainability with a concrete example using FL and XAI.

Use Case:
The most prominent use case for FL is in the medical field[†]. FL promises to preserve data privacy by mitigating the need to consolidate the data before training an ML model happens [62, 157].

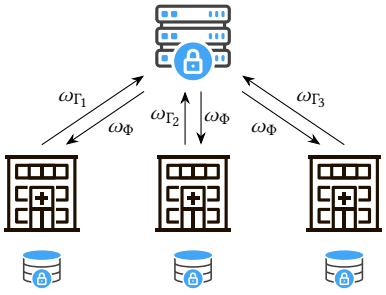


Figure 6.7.: Hospital Use Case Overview with FL.

Imagine the following scenario: Several hospitals are interested in applying ML techniques to the screening process of medical image data, e.g., detecting abnormalities in tissue samples. This *pre-screening* process happens orthogonal to the already established process of doctors looking at the images themselves. While this approach may not completely erase the need for a doctor to look at the medical images, it promises to reduce the time a doctor will take to look at them, and it will give doctors additional assurance if something abnormal is indicated to proceed with escalating the instance for further diagnostics.

However, some key questions need to be addressed.

1. A simple indication (benign or malignent) may not be as helpful to the doctor because he can not infer, why the ML model decided this way. Not understanding the reasoning behind the model's decision would significantly hamper trust in the *pre-screening* process and also limit further diagnostics.

2. From a doctor's perspective, the well-being of the patient is of the utmost importance. There are serious concerns that the ML model could harm the patient by facilitating misdiagnosis. For example, *pre-screening* could lead to a quality decrease in the screening process, and wrongful escalation of instances. Harm that stems from breaches of confidentiality regarding the patient data and, lastly, biased predictions in the ML model.

3. Given each hospital's data set size, it is not directly feasible to train an ML model with it, that would achieve a high enough accuracy. Because of (2), consolidating the data is not possible. Another approach would be to share the ML model, train it on the local data of one hospital, and then hand it to the next one. However,

this would lead to a strong recency bias in the ML model, and it would be prone to model inversion attacks.

To tackle these problems, the concept of FL with XAI seems fitting (see Figure 4.6). With FL, we are able to train an ML model without sharing the patient data to achieve a high model accuracy (this could also be enhanced by using differential privacy mechanisms to decrease the risk of model inversion attacks). Furthermore, by applying XAI methods, we can make the predictions of the ML model better understandable for the doctors. However, before we focusing too much on the technologies, let us try to decipher the term *explainability* in this context based on the theoretical foundations we laid beforehand (see Table 4.1).

Element	**Description**
System S	Group of hospitals connected via a network. A central instance managed by a trusted third party, called *fl server*, is responsible for managing the federation. Each hospital represents one *fl client* participating in the federation with its own respective set of medical patient data.
Target Group G	Medical personnel, e.g., medical doctors (Dr. med.), x-ray technicians.
Context C	*Pre-screening* process of medical image data taken from patients of the respective hospital.
Explanandum X	Diagnosis of the ML model regarding medical image data taken from patients of a given hospital.
Aspect Y	Suspicious parts of the patient's medical images that lead to the diagnosis of the ML model.
Data Space	An abstraction of the real world in terms of data. In this case, the most important aspects are the medical images of a patient, the explanation provided for the prediction, and the system itself the user interacts with.
Prediction	Predicted diagnosis of the ML model in the *pre-screening* process.
Explanatory Hypothesis Space	An abstraction of the reasoning process of a user in the target group G.

Table 6.1.: Explainability Elements That Need to Be Identified.

With the most necessary elements defined, we can now go into more detail about approaching the problem. Granted, some elements are still very much abstract – namely, the *data space* and the *explanatory hypothesis space* – but they are abstract by design,

because otherwise we would not be able to express them adequately. We will not go into implementation details regarding the proposed use case; this was shown in Chapter 4, where evaluated closely related experiments applicable to this use case. Instead, we will now focus on expectation management.

Expectations and Goals

As presented in the sections before, the term explanation in the context of XAI is overloaded and inherits a strong connotation of giving *ground-truth explanations* for an explanandum. Therefore, the expectations one has regarding the explainability of a system in question are usually ill-defined to begin with. So, let's concretely devise what is to expected in this use case scenario of an explanation. Because we will use an ML algorithm for the multiclass classification problem, we are bound to use XAI methods that can work with these algorithms. Having a good overview of XAI methods will lead us to the conclusion that the explanation will most certainly be a heat map of the original image that will show the most relevant areas of the image that lead the ML model to its prediction. While the concrete XAI method will dictate how the importance of the prediction is measured, it suffices to say that simply some pixels of the image have been selected as more important than others in the prediction of the ML algorithm. It is important to understand that this is not what we mean as an explanation in the context of our unified model – it is simply just data in the data space. The concrete explanation will be devised by the abduction process of the user in the target group G, forming explanatory hypotheses. We can not expect the XAI method to "inject" understanding into a user's consciousness.

As software engineers, we are also interested in specifying how to verify and validate explainability, e.g., what will be written in our specification document. While we can not anwser this question in the context of this thesis, we will weigh alternatives of how to approach FL and XAI against each other and provide recommendations for implementation.

[†]Interestingly, the concept of abduction – which was extensively discussed by the authors Josephson [96] – also used the medical context (creation of diagnosis hypotheses) for reasoning about explainability, as well as the authors in [150].

6.3. Approaching a Categorization for Explainability

Based on our findings and remarks in the last section, we can reason about assessing the explainability of systems. Figure 6.8 shows our proposed approach, which we will present in the following.

Figure 6.8 comprises four categories we extracted from our analysis and the examples presented in the last section. These categories are ordered as indicated in the figure, and each category has its own fulfillment score associated, meaning they are independently

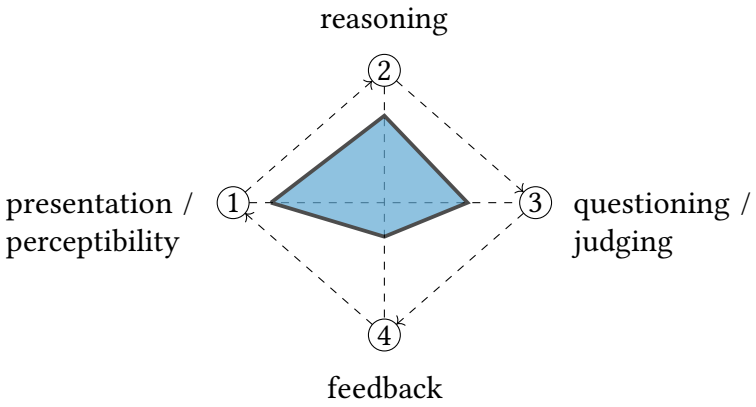


Figure 6.8.: Proposed Explainability Assessment Categorization.

calculated from each other. Therefore, instead of proposing a – strictly speaking – level-based approach, we propose a fluid, continuous categorization.

The first category is presentation / perceptibility. This category indicates how well the system in question can make observed phenomena that occur during operation perceivable by the user. A good perceptibility score is important to highlight events that can trigger the user's abduction process. Only what is made perceptible by the system can also be the target of an explanation seeking by the user. Additionally, we include the event's presentation in this category because how something is presented can influence the user's perception of it. There is also a point to distinguish between making relevant and non-relevant phenomena available to the user. For example, a system could present everything to the user. However, such a system should score poorly in this category.

Next, we have the category reasoning. A system needs the means to generate or list explanatory hypotheses to the user. The score in this category indicates how diverse (quantity) and how good the explanations are (quality). For example, an explanation that plainly states observations does most likely not help the user in understanding the phenomena to be explained. On the other hand, even if an explanation is – by quality aspects – good, a user may still prefer a different type of explanation, for example, a counterfactual one to help him in their abduction process.

In the second last category questioning / judging, we score the capability of the system to allow the user to question and judge the explanatory hypotheses the system may provide. This category captures how well the system allows the user to explore – through their abduction – the explanatory hypotheses space. For example, a system that can be questioned by the user in the form of counterfactuals would certainly score very high in that regard. Meanwhile, a system that does not allow for this degree of freedom would score much less because it does not allow for a meaningful exploration of the explanatory hypotheses space by the user and, therefore, limits the explainability of the system. While the first two categories are commonly seen in current research literature about explainability, this category, and the next are still unexplored. One reason is that it is tough to develop systems that would score high in this regard.

Lastly, we have the category feedback. This category scores the system's capability to co-adapt based on the feedback that the user provides while interacting with the system. For example, a system that would score high in this category can presumably infer adaptions by suggestions incorporated in the line of questions by the user. However, also more direct means are imaginable. A system that adapts to context and the user can arguably also provide better explanations.

6.4. Remarks

In this Section, we first want to reconcile what is still open for discussion given the proposed classification approach described in Section 6.3. Then, we will briefly discuss the concept of abduction regarding the design, evaluation, and methods. At last, we want to summarize what we have learned from this Chapter and how it informs this thesis.

6.4.1. What is Missing?

If one wants to use the proposed categorization, it becomes evident that to assess a score for each category, one needs guidelines or reference values. Otherwise the score itself becomes meaningless. While it is possible to qualitatively describe systems in regard to the four categories, because of the lack of references and guidelines, one can not assign an order like $A \leq B$ to compare the explainability of system A and B against each other – even when only comparing inside a particular category. For guidelines and reference values, one would need to capture much empirical data about the explainability of systems related to each category we described and then compile this data into a scoring system.

However, even if one would do this, it stands to reason that comparisons based on these scoring systems are valid. Imagine the example of a system A being the autonomous vehicle AI we described in the last section and a system B a Large Language Model (LLM) like GitHub Copilot. Even if we have reference values and guidelines to make a comparison, these two systems are inherently different. So, comparing different types of systems against each other is most likely not desirable in the first place. Instead, adopting guidelines and reference values on a per-system-like basis is most likely more beneficial. The overarching problem is now that by doing this one encodes domain / application-specific knowledge into the scoring system, which forces the practitioner of the scoring system inadvertently into a lower abstraction level. Hence, the ability to compare the explainability of systems at the highest possible meta-level is lost.

Another problem with a scoring-based system is that it will most likely fall back to the usage of heuristics, because either the system's complexity is too high or the complexity of the question one has regarding the explainability is too difficult to answer [51]. For example, a relatively recent research field is the development of benchmarks that compare the reasoning capability of different LLMs [70]. In this instance, the question of how good

LLMs can reason is too difficult to assess otherwise, so heuristics that can at least test for some instances are being used.

We must still tie our findings to the practical aspects of explainability we mentioned in the Foundations Chapter 2. Which we will present in the following.

In the aforementioned Chapter, we subdivided explainability into roughly three main concerns: (i) Design, (ii) Evaluation, and (iii) Methods (XAI). We will follow this approach so readers can cross-reference the appropriate literature if further guidance is needed.

Design

We began this thesis by definitions of explainability as a non-functional requirement in current research literature. These definitions are still valid and align mostly with our proposed unified model. However, we somewhat disregarded the word "systems" in favor of a generalized form of an abstract *data space* that is – strictly speaking – not confined to system boundaries. Certainly, one can define systems and system boundaries rather generously and vaguely to capture different interaction levels of systems. However, we have purposely chosen not to do so. This is because systems usually need an a priori definition itself before one can work with it – they do usually occur naturally, but their definition is in this case, still obligatory. In the *data space*, however, these boundaries are naturally by design and a strict a priori definition is not mandatory. A noticeable difference between our proposed model and the presented definitions is the definition of the means M, which provides the explanation. With abduction as the core of the explainability process, we diverged from the notion of "providing" an explanation to the notion of "arriving" at an explanation. Our notion is not focused on mere matters of receiving an explanation to understand an explanandum X, but on arriving at an explanatory hypothesis that is satisfying to a specific individual. Moreover, in light of that, we further derived from state-of-the-art research literature that the core elements of an explanation are *cognitive, causal, and social* and that humans evaluate explanations by these means. In that sense, our notion is more concrete. We then presented a list of goals that an explainable system shall satisfy. These goals are orthogonal to our model because we rely on an individuals' discretion to pursue them. Then, we introduced the *levels of explainability*, which we wanted to revisit to encompass systems that are not agent-based. For this, we introduced the categorization in the last section. While we could not end up with a level-based distinction, we strived for a categorization that can – in theory – be utilized for creating a scoring-based system to assess the explainability of a system. Lastly, we have shown the assumed relation between model accuracy and model explainability, which is commonly found in research literature. Truthfully, empirical validation is necessary, but this is not the subject of this thesis.

Evaluation and Methods

In the evaluation section, we focused on available metrics for (quantitatively) assessing and comparing the goodness of explanations. We noted that with computed metrics, one

relies on assumed proxies of *goodness* to assess the explainability of a system. Also, we presented the quality function $Q_E(M)$ from the authors Bersani et al. [16], which we tried to re-parameterize with already available XAI metrics to assess and compare different explanations in furtherance of guiding a sensible FL architecture that incorporates XAI methods. While this approach did not work out, we tried in this Chapter to broaden our understanding of what explainability means with the concept of abduction.

Lastly, we mentioned the Rashomon Effect, which we already tied to our unified model by recognizing it not only as a mathematical property but also as a source of unavoidable subjectivity. We can even visualize this effect in Figure 6.5 as multiple spaces in the explanatory hypothesis space that point – via *prediction* – to the same or close to the same data space. Hence, the also commonly referred name of *predictive multiplicity*. Unsurprisingly, we could also measure this effect in our experiments and noted differences in how participants of our user study reacted to it.

6.4.2. Lessons Learned (2)

To put it briefly and conclude this Chapter, we have seen that explainability is a multifaceted and multidisciplinary problem. We have seen and acknowledged each facet and tried to analyze them using the concept of abduction. While we only argumentatively reasoned about the importance of each facet, it becomes evident that for explainability to be comprehensible and implementable for practitioners, more fundamental research in this area is necessary. In essence, we showed the contrast of two different approaches in this thesis; in the first part, we wanted to approach this problem technically, and in this part, we wanted to approach it from a top-down perspective. Both approaches were reasonably valid and produced scientifically valid results, which further research can build on.

7. Conclusion

This chapter will first summarize our findings in 7.1 and provide an outlook for future research directions in 7.2.

7.1. Summary

In this master's thesis, we integrated and analyzed explainability as a non-functional requirement in the context of FL. We addressed this problem by applying explainability in the FL context and conducted several experiments. The first experiment (see Subsection 4.3) validated that the global ML model of the FL context shall be used for generating explanations and that the FL algorithm is not that important in contrast to the data distribution. The second experiment (see Subsection 4.4) evaluated the effect of predictive multiplicity or the Rashomon Effect. We experimentally validated that different XAI methods are more susceptible to this effect than others and should preferably be used if predictive multiplicity is a valid concern. In experiment 3 (see Subsection 4.5), we explored the possibilities of aggregating different XAI methods to create and optimize explanations. By using this approach, we could measurably improve XAI metrics and have the opportunity to balance the cost and performance of explainability methods by solving the associated multi-objective optimization problem. Lastly, we explored how DP and misbehaving FL clients affect explanations by measurably reducing XAI and performance metrics (see Subsections 4.6 and 4.7).

Our empirical acquired data was then (in parts) complemented by a user survey about explainability in Chapter 5. Our results indicate that while the demand for explainability is very high, there exists an inconsistency in how explanations are deemed satisfactory. Also, we could show that while our optimized explanations result in numerically better explanations, this does not map to a measurable increase in user acceptance.

Lastly, we took a multidisciplinary approach to defining explainability and its most common elements (see Section 6.2). For this, we used the theory of abduction, which explained the core principles of how explanations are formed and determined to be satisfactory by a user. This was done in response to the previous Chapters to better understand the "human problems" associated with explainability.

7.2. Future Work

During our experiments with the FL framework Flower, it became more and more evident that it did not fulfill all our requirements, especially since the current implementation of the simulation engine was a limiting factor. Our implementation of the simulation engine can still be further refined to allow for a better parallelization and utilization of available resources. Currently, it is limited to a single computer and a single Graphics Processing Unit (GPU). The simulation engine can be extended for cluster usage and a more optimized, adaptive management of the computing resources to allow for more real-world simulations with many independent devices while still having the benefit of a simulation's controlled environment. Exciting features are:

- Auto-scaling mechanisms and checkpointing.
- Simulation of failures and other more elaborated scenarios (e.g., simulation of networks with package loss).
- Out-of-band communication allowing for interventions.
- Allow for further variability of the FL loop.
- Easier integration of monitoring solutions.

We were particularly surprised that the Flower community does not emphasize the use of the simulation engine further, which is, in our eyes, one of the most interesting applications of FL in research.

If one desires to conduct further research on the optimization of explanations, we suggest further evaluating the influence and computation capabilities with evolutionary algorithms. In this thesis, we only scratched the surface and used the most basic way of using them. Further optimizations can most certainly achieved by further customization of these algorithms. Generally speaking, we were also pleased with the concept of *making something measurable, actionable*. Particularly, we would want to investigate how this concept can be further applied to other problems in the domain of non-functional requirements.

Next is the aspect of FL in conjunction with security-related questions. So, as stated before, FL is only helpful when there is a specific need to conduct the training in a federated fashion. However, this still does not guarantee that data will not leak. One method we introduced for mitigation was DP. As we documented, DP strongly cuts into the performance of the ML model and, in turn, into the usefulness of XAI methods. It would be exciting if a method could be developed that simultaneously satisfies both privacy, and explainability requirements.

Another interesting research direction is the potential application of LLMs for retrieving or generating explanations [140]. What is interesting for us is the ability to retrieve explanation scenarios from software artifacts so that engineers can further improve the overall explainability of the software in question. Another interesting aspect is the possibility of

generating explanatory hypotheseses, as presented in Chapter 6. Google already conducts research in this area, as seen in their technical paper about a co-scientist agent helping generate scientific hypotheses [79]. While we do not have the resources for research of this kind, we could come up with possible use cases and evaluation mechanisms for a co-scientist used for explainability purposes. LLMs are particularly interesting in terms of explainability because, at the time of writing, there are not many methods for explainability that can be applied to it.

Lastly, we want to mention the human-machine collaboration that needs to be respected in terms of explainability. While some research is already done in this area, we found that there are still possible research gaps. For example, to the author's best knowledge, there is currently no viable engineering approach for explainability in software engineering that includes both the human-machine relation and, on the other hand, the practical implementation of explainability mechanisms. So we strongly suggest further research in this area to make explainability more approachable.

For this master's thesis, Grammarly has been used to improve grammar and sentence structure.

Bibliography

[1] Martín Abadi et al. "Deep Learning with Differential Privacy". In: *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. Oct. 24, 2016, pp. 308–318. DOI: 10.1145/2976749.2978318. arXiv: 1607.00133 [stat]. URL: http://arxiv.org/abs/1607.00133 (visited on 02/26/2025).

[2] Julius Adebayo et al. "Sanity Checks for Saliency Maps". In: *Advances in Neural Information Processing Systems*. Ed. by S. Bengio et al. Vol. 31. Curran Associates, Inc., 2018. URL: https://proceedings.neurips.cc/paper_files/paper/2018/file/294a8ed24b1ad22ec2e7efea049b8737-Paper.pdf.

[3] Chirag Agarwal et al. *Rethinking Stability for Attribution-based Explanations*. Mar. 2022. arXiv: 2203.06877 [cs]. (Visited on 11/07/2024).

[4] Ahmad Ajalloeian et al. "On Smoothed Explanations: Quality and Robustness". In: *Proceedings of the 31st ACM International Conference on Information & Knowledge Management*. Atlanta GA USA: ACM, Oct. 2022, pp. 15–25. ISBN: 978-1-4503-9236-5. DOI: 10.1145/3511808.3557409. (Visited on 11/07/2024).

[5] Sajid Ali et al. "Explainable Artificial Intelligence (XAI): What We Know and What Is Left to Attain Trustworthy Artificial Intelligence". In: *Information Fusion* 99 (Nov. 2023), p. 101805. ISSN: 15662535. DOI: 10.1016/j.inffus.2023.101805. (Visited on 10/13/2024).

[6] David Alvarez-Melis and Tommi S. Jaakkola. *Towards Robust Interpretability with Self-Explaining Neural Networks*. Dec. 3, 2018. DOI: 10.48550/arXiv.1806.07538. arXiv: 1806.07538 [cs]. URL: http://arxiv.org/abs/1806.07538 (visited on 12/11/2024). Pre-published.

[7] Leila Amgoud and Jonathan Ben-Naim. "Axiomatic Foundations of Explainability". In: *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence*. Vienna, Austria: International Joint Conferences on Artificial Intelligence Organization, July 2022, pp. 636–642. ISBN: 978-1-956792-00-3. DOI: 10.24963/ijcai.2022/90. (Visited on 10/31/2024).

[8] Leila Amgoud, Martin C. Cooper, and Salim Debbaoui. *Axiomatic Characterisations of Sample-based Explainers*. Aug. 2024. arXiv: 2408.04903 [cs]. (Visited on 11/11/2024).

[9] Galen Andrew et al. *Differentially Private Learning with Adaptive Clipping*. May 9, 2022. DOI: 10.48550/arXiv.1905.03871. arXiv: 1905.03871 [cs]. URL: http://arxiv.org/abs/1905.03871 (visited on 02/26/2025). Pre-published.

[10] Maksym Andriushchenko et al. *Square Attack: a query-efficient black-box adversarial attack via random search*. 2020. arXiv: 1912.00049 [cs.LG]. URL: https://arxiv.org/abs/1912.00049.

[11] Hedström Anna et al. "Quantus: An Explainable AI Toolkit for Responsible Evaluation of Neural Network Explanations". In: (2022). DOI: 10.48550/ARXIV.2202.06861.

[12] Jason Ansel et al. "PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation". In: *29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS '24)*. ACM, Apr. 2024. DOI: 10.1145/3620665.3640366. URL: https://pytorch.org/assets/pytorch2-2.pdf.

[13] Puja Banerjee and Rajesh P. Barnwal. "Methods and Metrics for Explaining Artificial Intelligence Models: A Review". In: *Explainable AI: Foundations, Methodologies and Applications*. Ed. by Mayuri Mehta, Vasile Palade, and Indranath Chatterjee. Cham: Springer International Publishing, 2023, pp. 61–88. ISBN: 978-3-031-12807-3. DOI: 10.1007/978-3-031-12807-3_4. URL: https://doi.org/10.1007/978-3-031-12807-3_4.

[14] José Luis Corcuera Bárcena et al. "Fed-XAI: Federated Learning of Explainable Artificial Intelligence Models". In: *Proceedings of XAI. it 2022 Italian Workshop on Explainable Artificial Intelligence 2022* (2022), pp. 1–14. URL: https://ceur-ws.org/Vol-3277/paper8.pdf (visited on 10/05/2024).

[15] Alejandro Barredo Arrieta et al. "Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI". In: *Information Fusion* 58 (June 2020), pp. 82–115. ISSN: 15662535. DOI: 10.1016/j.inffus.2019.12.012. (Visited on 10/13/2024).

[16] Marcello M. Bersani et al. "A Conceptual Framework for Explainability Requirements in Software-Intensive Systems". In: *2023 IEEE 31st International Requirements Engineering Conference Workshops (REW)*. Hannover, Germany: IEEE, Sept. 2023, pp. 309–315. ISBN: 9798350326918. DOI: 10.1109/REW57809.2023.00059. (Visited on 10/05/2024).

[17] Daniel J. Beutel et al. *Flower: A Friendly Federated Learning Research Framework*. Mar. 2022. arXiv: 2007.14390 [cs, stat]. (Visited on 10/05/2024).

[18] Umang Bhatt, Adrian Weller, and José M. F. Moura. *Evaluating and Aggregating Feature-based Model Explanations*. May 2020. arXiv: 2005.00631 [cs]. (Visited on 11/07/2024).

[19] Peva Blanchard et al. *Byzantine-Tolerant Machine Learning*. Mar. 2017. arXiv: 1703.02757 [cs]. (Visited on 11/01/2024).

[20] J. Blank and K. Deb. "pymoo: Multi-Objective Optimization in Python". In: *IEEE Access* 8 (2020), pp. 89497–89509.

[21] Francesco Bodria et al. "Benchmarking and Survey of Explanation Methods for Black Box Models". In: *Data Mining and Knowledge Discovery* 37.5 (Sept. 2023), pp. 1719–1778. ISSN: 1384-5810, 1573-756X. DOI: 10.1007/s10618-023-00933-9. (Visited on 10/13/2024).

[22] Clara Bove et al. *Why Do Explanations Fail? A Typology and Discussion on Failures in XAI*. May 2024. arXiv: 2405.13474 [cs]. (Visited on 11/07/2024).

[23] Leo Breiman. “Statistical Modeling: The Two Cultures”. In: *Statistical science* 16.3 (2001), pp. 199–231.

[24] David A Broniatowski. *Psychological Foundations of Explainability and Interpretability in Artificial Intelligence*. Tech. rep. NIST IR 8367. Gaithersburg, MD: National Institute of Standards and Technology (U.S.), Apr. 2021, NIST IR 8367. DOI: 10.6028/NIST.IR.8367. (Visited on 10/05/2024).

[25] Olivier Caelen. *What is the Shapley value ?* Dec. 2022. URL: https://medium.com/the-modern-scientist/what-is-the-shapley-value-8ca624274d5a (visited on 11/04/2024).

[26] Sebastian Caldas et al. *LEAF: A Benchmark for Federated Settings*. Dec. 2019. arXiv: 1812.01097 [cs]. (Visited on 11/19/2024).

[27] Matteo Camilli, Raffaela Mirandola, and Patrizia Scandurra. “XSA: Explainable self-adaptation”. In: *Proceedings of the 37th IEEE/ACM International Conference on Automated Software Engineering*. 2022, pp. 1–5.

[28] Nicholas Carlini and David Wagner. *Towards Evaluating the Robustness of Neural Networks*. 2017. arXiv: 1608.04644 [cs.CR]. URL: https://arxiv.org/abs/1608.04644.

[29] Diogo V. Carvalho, Eduardo M. Pereira, and Jaime S. Cardoso. “Machine Learning Interpretability: A Survey on Methods and Metrics”. In: *Electronics* 8.8 (July 2019), p. 832. ISSN: 2079-9292. DOI: 10.3390/electronics8080832. (Visited on 10/13/2024).

[30] Mustafa Cavus and Przemysław Biecek. *An Experimental Study on the Rashomon Effect of Balancing Methods in Imbalanced Classification*. July 2024. arXiv: 2405.01557 [cs]. (Visited on 11/10/2024).

[31] Eugene Charniak et al. *Artificial Intelligence Programming*. 2. ed. Erlbaum, 1987. ISBN: 0-89859-609-2.

[32] Larissa Chazette, Wasja Brunotte, and Timo Speith. “Explainable software systems: from requirements analysis to system evaluation”. In: *Requirements Engineering* 27.4 (2022), pp. 457–487.

[33] Larissa Chazette, Wasja Brunotte, and Timo Speith. “Exploring Explainability: A Definition, a Model, and a Knowledge Catalogue”. In: *2021 IEEE 29th International Requirements Engineering Conference (RE)*. Notre Dame, IN, USA: IEEE, Sept. 2021, pp. 197–208. ISBN: 978-1-66542-856-9. DOI: 10.1109/RE51729.2021.00025. (Visited on 10/05/2024).

[34] Larissa Chazette, Oliver Karras, and Kurt Schneider. “Do End-Users Want Explanations? Analyzing the Role of Explainability as an Emerging Aspect of Non-Functional Requirements”. In: *2019 IEEE 27th International Requirements Engineering Conference (RE)*. Jeju Island, Korea (South): IEEE, Sept. 2019, pp. 223–233. ISBN: 978-1-72813-912-8. DOI: 10.1109/RE.2019.00032. (Visited on 10/05/2024).

[35] Larissa Chazette and Kurt Schneider. "Explainability as a Non-Functional Requirement: Challenges and Recommendations". In: *Requirements Engineering* 25.4 (Dec. 2020), pp. 493–514. ISSN: 0947-3602, 1432-010X. DOI: 10.1007/s00766-020-00333-1. (Visited on 10/05/2024).

[36] Larissa Chazette et al. "How Can We Develop Explainable Systems? Insights from a Literature Review and an Interview Study". In: *Proceedings of the International Conference on Software and System Processes and International Conference on Global Software Engineering*. Pittsburgh PA USA: ACM, May 2022, pp. 1–12. ISBN: 978-1-4503-9674-5. DOI: 10.1145/3529320.3529321. (Visited on 10/05/2024).

[37] Larissa Chazette et al. "Requirements on Explanations: A Quality Framework for Explainability". In: *2022 IEEE 30th International Requirements Engineering Conference (RE)*. Melbourne, Australia: IEEE, Aug. 2022, pp. 140–152. ISBN: 978-1-66547-000-1. DOI: 10.1109/RE54965.2022.00019. (Visited on 10/05/2024).

[38] Pin-Yu Chen et al. *EAD: Elastic-Net Attacks to Deep Neural Networks via Adversarial Examples*. 2018. arXiv: 1709.04114 [stat.ML]. URL: https://arxiv.org/abs/1709.04114.

[39] Pin-Yu Chen et al. "ZOO: Zeroth Order Optimization Based Black-box Attacks to Deep Neural Networks without Training Substitute Models". In: *Proceedings of the 10th ACM Workshop on Artificial Intelligence and Security*. CCS '17. ACM, Nov. 2017. DOI: 10.1145/3128572.3140448. URL: http://dx.doi.org/10.1145/3128572.3140448.

[40] Zixi Chen et al. *What Makes a Good Explanation?: A Harmonized View of Properties of Explanations*. July 2024. arXiv: 2211.05667 [cs]. (Visited on 11/07/2024).

[41] Martino Ciaperoni, Han Xiao, and Aristides Gionis. *Efficient Exploration of the Rashomon Set of Rule Set Models*. June 2024. arXiv: 2406.03059 [cs]. (Visited on 11/10/2024).

[42] Jacob Cohen. "A Coefficient of Agreement for Nominal Scales". In: *Educational and Psychological Measurement* 20.1 (Apr. 1960), pp. 37–46. ISSN: 0013-1644, 1552-3888. DOI: 10.1177/001316446002000104. (Visited on 11/24/2024).

[43] Loredana Coroama and Adrian Groza. "Evaluation Metrics in Explainable Artificial Intelligence (XAI)". In: *Advanced Research in Technologies, Information, Innovation and Sustainability*. Ed. by Teresa Guarda, Filipe Portela, and Maria Fernanda Augusto. Cham: Springer Nature Switzerland, 2022, pp. 401–413. ISBN: 978-3-031-20319-0.

[44] Francesco Croce and Matthias Hein. *Reliable evaluation of adversarial robustness with an ensemble of diverse parameter-free attacks*. 2020. arXiv: 2003.01690 [cs.LG]. URL: https://arxiv.org/abs/2003.01690.

[45] Mattia Daole et al. "OpenFL-XAI: Federated Learning of Explainable Artificial Intelligence Models in Python". In: *SoftwareX* 23 (July 2023), p. 101505. ISSN: 23527110. DOI: 10.1016/j.softx.2023.101505. (Visited on 10/05/2024).

[46] Richard Dazeley et al. “Levels of Explainable Artificial Intelligence for Human-Aligned Conversational Explanations”. In: *Artificial Intelligence* 299 (Oct. 2021), p. 103525. ISSN: 00043702. DOI: 10.1016/j.artint.2021.103525. (Visited on 11/11/2024).

[47] Luc De Raedt, Angelika Kimmig, and Hannu Toivonen. “ProbLog: A probabilistic Prolog and its application in link discovery”. In: *IJCAI 2007, Proceedings of the 20th international joint conference on artificial intelligence*. IJCAI-INT JOINT CONF ARTIF INTELL. 2007, pp. 2462–2467.

[48] K. Deb et al. “A fast and elitist multiobjective genetic algorithm: NSGA-II”. In: *IEEE Transactions on Evolutionary Computation* 6.2 (2002), pp. 182–197. DOI: 10.1109/4235.996017.

[49] Kalyanmoy Deb. *Multi-objective optimization using evolutionary algorithms*. Wiley-Interscience series in systems and optimization. Includes bibliographical references (p. [471] - 490) and index. Chichester ; Wiley, 2001. URL: http://www.loc.gov/catdir/toc/onix06/2001022514.html.

[50] Thomas Decker et al. *Provably Better Explanations with Optimized Aggregation of Feature Attributions*. June 2024. arXiv: 2406.05090 [cs]. (Visited on 11/11/2024).

[51] Hannah Deters, Jakob Droste, and Kurt Schneider. “A Means to What End? Evaluating the Explainability of Software Systems Using Goal-Oriented Heuristics”. In: *Proceedings of the 27th International Conference on Evaluation and Assessment in Software Engineering*. Oulu Finland: ACM, June 2023, pp. 329–338. ISBN: 9798400700446. DOI: 10.1145/3593434.3593444. (Visited on 10/05/2024).

[52] Hannah Deters et al. “Exploring the Means to Measure Explainability: Metrics, Heuristics and Questionnaires”. In: *Information and Software Technology* 181 (May 2025), p. 107682. ISSN: 09505849. DOI: 10.1016/j.infsof.2025.107682. URL: https://linkinghub.elsevier.com/retrieve/pii/S0950584925000217 (visited on 02/18/2025).

[53] Hannah Deters et al. “How Explainable Is Your System? Towards a Quality Model for Explainability”. In: *Requirements Engineering: Foundation for Software Quality*. Ed. by Daniel Mendez and Ana Moreira. Vol. 14588. Cham: Springer Nature Switzerland, 2024, pp. 3–19. DOI: 10.1007/978-3-031-57327-9_1. (Visited on 10/05/2024).

[54] Steven Diamond and Stephen Boyd. “CVXPY: A Python-embedded modeling language for convex optimization”. In: *Journal of Machine Learning Research* 17.83 (2016), pp. 1–5.

[55] Yinpeng Dong et al. *Boosting Adversarial Attacks with Momentum*. 2018. arXiv: 1710.06081 [cs.LG]. URL: https://arxiv.org/abs/1710.06081.

[56] Jon Donnelly et al. *The Rashomon Importance Distribution: Getting RID of Unstable, Single Model-based Variable Importance*. Apr. 2024. arXiv: 2309.13775 [cs]. (Visited on 11/10/2024).

[57] Finale Doshi-Velez and Been Kim. *Towards A Rigorous Science of Interpretable Machine Learning*. Mar. 2017. arXiv: 1702.08608 [stat]. (Visited on 10/30/2024).

[58] Rachel Lea Draelos and Lawrence Carin. *Use HiResCAM Instead of Grad-CAM for Faithful Explanations of Convolutional Neural Networks*. Nov. 2021. DOI: 10.48550/arXiv.2011.08891. arXiv: 2011.08891 [eess]. (Visited on 01/14/2025).

[59] Jakob Droste et al. "Framing What Can Be Explained - an Operational Taxonomy for Explainability Needs". In: *Requirements Engineering* (Mar. 25, 2025). ISSN: 0947-3602, 1432-010X. DOI: 10.1007/s00766-025-00440-x. URL: https://link.springer.com/10.1007/s00766-025-00440-x (visited on 04/07/2025).

[60] Pietro Ducange et al. "Consistent Post-Hoc Explainability in Federated Learning through Federated Fuzzy Clustering". In: *2024 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. Yokohama, Japan: IEEE, June 2024, pp. 1–10. ISBN: 9798350319545. DOI: 10.1109/FUZZ-IEEE60900.2024.10611761. (Visited on 10/05/2024).

[61] Pietro Ducange et al. "Federated Learning of XAI Models in Healthcare: A Case Study on Parkinson's Disease". In: *Cognitive Computation* (Aug. 2024). ISSN: 1866-9956, 1866-9964. DOI: 10.1007/s12559-024-10332-x. (Visited on 10/05/2024).

[62] Pietro Ducange et al. "Federated Learning of XAI Models in Healthcare: A Case Study on Parkinson's Disease". In: *Cognitive Computation* 16.6 (Nov. 2024), pp. 3051–3076. ISSN: 1866-9956, 1866-9964. DOI: 10.1007/s12559-024-10332-x. URL: https://link.springer.com/10.1007/s12559-024-10332-x (visited on 01/16/2025).

[63] Meedeniya Dulani. *Deep Learning : A Beginners' Guide*. Vol. First edition. Chapman and Hall/CRC, 2024. ISBN: 9781032473246.

[64] Cynthia Dwork and Aaron Roth. "The Algorithmic Foundations of Differential Privacy". In: *Foundations and Trends® in Theoretical Computer Science* 9.3–4 (2013), pp. 211–407. DOI: 10.1561/0400000042. URL: http://www.nowpublishers.com/articles/foundations-and-trends-in-theoretical-computer-science/TCS-042 (visited on 02/26/2025).

[65] Mohammed Amine El Mrabet, Khalid El Makkaoui, and Ahmed Faize. "Supervised Machine Learning: A Survey". In: *2021 4th International Conference on Advanced Communication Technologies and Networking (CommNet)*. Rabat, Morocco: IEEE, Dec. 2021, pp. 1–10. ISBN: 978-1-66540-306-1. DOI: 10.1109/CommNet52204.2021.9641998. (Visited on 10/24/2024).

[66] Ismail M. Elshair et al. "Evaluating Federated Learning Simulators: A Comparative Analysis of Horizontal and Vertical Approaches". In: *Sensors* 24.16 (Aug. 2024), p. 5149. ISSN: 1424-8220. DOI: 10.3390/s24165149. (Visited on 11/07/2024).

[67] Logan Engstrom et al. *Exploring the Landscape of Spatial Robustness*. 2019. arXiv: 1712.02779 [cs.LG]. URL: https://arxiv.org/abs/1712.02779.

[68] Kave Eshghi and Robert A Kowalski. "Abduction Compared with Negation by Failure." In: *ICLP*. Vol. 89. Citeseer. 1989, pp. 234–255.

[69] Fatima Ezzeddine et al. "Differential Privacy for Anomaly Detection: Analyzing the Trade-off between Privacy and Explainability". In: *Explainable Artificial Intelligence*. Ed. by Luca Longo, Sebastian Lapuschkin, and Christin Seifert. Cham: Springer Nature Switzerland, 2024, pp. 294–318. ISBN: 978-3-031-63800-8.

[70] Bahare Fatemi et al. *Test of Time: A Benchmark for Evaluating LLMs on Temporal Reasoning*. June 2024. DOI: 10.48550/arXiv.2406.09170. arXiv: 2406.09170 [cs]. (Visited on 12/15/2024).

[71] Daan Fierens et al. "Inference and Learning in Probabilistic Logic Programs Using Weighted Boolean Formulas". In: *Theory and Practice of Logic Programming* 15.3 (May 2015), pp. 358–401. ISSN: 1471-0684, 1475-3081. DOI: 10.1017/S1471068414000076. URL: https://www.cambridge.org/core/product/identifier/S1471068414000076/type/journal_article (visited on 02/01/2025).

[72] David Hackett Fischer. *Historians' fallacies; toward a logic of historical thought*. 1970. URL: https://archive.org/details/historiansfallac00fisc/page/90 (visited on 11/04/2024).

[73] Patrick Foley et al. "OpenFL: The Open Federated Learning Library". In: *Physics in Medicine & Biology* 67.21 (Nov. 2022), p. 214001. ISSN: 0031-9155, 1361-6560. DOI: 10.1088/1361-6560/ac97d9. (Visited on 10/05/2024).

[74] Dov M. Gabbay and John Woods. *A Practical Logic of Cognitive Systems :* A practical logic of cognitive systems ; Includes bibliographical references (p. 443-472) and index. Amsterdam ; Elsevier, 2005. URL: https://www.sciencedirect.com/science/bookseries/18745075/2.

[75] Amin Ghiasi, Ali Shafahi, and Tom Goldstein. *Breaking certified defenses: Semantic adversarial examples with spoofed robustness certificates*. 2020. arXiv: 2003.08937 [cs.LG]. URL: https://arxiv.org/abs/2003.08937.

[76] Jacob Gildenblat and contributors. *PyTorch library for CAM methods*. https://github.com/jacobgil/pytorch-grad-cam. 2021.

[77] M. Glinz. "On Non-Functional Requirements". In: *15th IEEE International Requirements Engineering Conference (RE 2007)*. Delhi: IEEE, Oct. 2007, pp. 21–26. ISBN: 978-0-7695-2935-6. DOI: 10.1109/RE.2007.45. (Visited on 10/05/2024).

[78] Ian J. Goodfellow, Jonathon Shlens, and Christian Szegedy. *Explaining and Harnessing Adversarial Examples*. 2015. arXiv: 1412.6572 [stat.ML]. URL: https://arxiv.org/abs/1412.6572.

[79] Juraj Gottweis et al. *Towards an AI co-scientist*. 2025. arXiv: 2502.18864 [cs.AI]. URL: https://arxiv.org/abs/2502.18864.

[80] David Gunning and David W. Aha. "DARPA's Explainable Artificial Intelligence Program". In: *AI Magazine* 40.2 (June 2019), pp. 44–58. ISSN: 0738-4602, 2371-9621. DOI: 10.1609/aimag.v40i2.2850. (Visited on 11/04/2024).

[81] Umm-e- Habiba et al. *How Mature Is Requirements Engineering for AI-based Systems ? A Systematic Mapping Study on Practices, Challenges, and Future Research Directions*. Sept. 2024. arXiv: 2409.07192 [cs]. (Visited on 10/05/2024).

[82] Leif Hancox-Li. "Robustness in Machine Learning Explanations: Does It Matter?" In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. Barcelona Spain: ACM, Jan. 2020, pp. 640–647. ISBN: 978-1-4503-6936-7. DOI: 10.1145/3351095.3372836. (Visited on 10/11/2024).

[83] Anna Hedström et al. *Quantus: An Explainable AI Toolkit for Responsible Evaluation of Neural Network Explanations and Beyond*. Apr. 2023. arXiv: 2202.06861 [cs]. (Visited on 10/07/2024).

[84] Davin Hill et al. *Axiomatic Explainer Globalness via Optimal Transport*. Nov. 2024. arXiv: 2411.01126 [cs]. (Visited on 11/11/2024).

[85] Robert R. Hoffman, Timothy Miller, and William J. Clancey. "Psychology and AI at a Crossroads: How Might Complex Systems Explain Themselves?" In: *The American Journal of Psychology* 135.4 (Dec. 2022), pp. 365–378. ISSN: 0002-9556, 1939-8298. DOI: 10.5406/19398298.135.4.01. (Visited on 11/07/2024).

[86] Sara Hooker et al. *A Benchmark for Interpretability Methods in Deep Neural Networks*. Nov. 2019. arXiv: 1806.10758 [cs]. (Visited on 11/07/2024).

[87] Hsiang Hsu and Flavio du Pin Calmon. *Rashomon Capacity: A Metric for Predictive Multiplicity in Classification*. Oct. 2022. arXiv: 2206.01295 [cs]. (Visited on 11/10/2024).

[88] Hsiang Hsu et al. *Dropout-Based Rashomon Set Exploration for Efficient Predictive Multiplicity Estimation*. Feb. 2024. arXiv: 2402.00728 [cs]. (Visited on 11/10/2024).

[89] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. *Measuring the Effects of Non-Identical Data Distribution for Federated Visual Classification*. Sept. 2019. arXiv: 1909.06335 [cs]. (Visited on 11/01/2024).

[90] Tzu-Ming Harry Hsu, Hang Qi, and Matthew Brown. *Measuring the Effects of Non-Identical Data Distribution for Federated Visual Classification*. Sept. 13, 2019. DOI: 10.48550/arXiv.1909.06335. arXiv: 1909.06335 [cs]. URL: http://arxiv.org/abs/1909.06335 (visited on 01/20/2025). Pre-published.

[91] Trung Dong Huynh et al. "A Methodology and Software Architecture to Support Explainability-by-Design". In: *arXiv preprint arXiv:2206.06251* (2022).

[92] International Joint Conferences on Artificial Intelligence Organization IJCAI. 2011. URL: https://www.ijcai.org/awards/.

[93] Katsumi Inoue. "Automated Abduction". In: *Computational Logic: Logic Programming and Beyond*. Ed. by Antonis C. Kakas and Fariba Sadri. Red. by G. Goos, J. Hartmanis, and J. Van Leeuwen. Vol. 2408. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, pp. 311–341. ISBN: 978-3-540-45632-2. DOI: 10.1007/3-540-45632-5_13. URL: http://link.springer.com/10.1007/3-540-45632-5_13 (visited on 01/31/2025).

[94] James. *Shapley Value Calculator*. URL: https://nonzerosum.games/shapleyvalue.html (visited on 11/04/2024).

[95] Uyeong Jang, Xi Wu, and Somesh Jha. "Objective Metrics and Gradient Descent Algorithms for Adversarial Examples in Machine Learning". In: *Proceedings of the 33rd Annual Computer Security Applications Conference*. ACSAC '17. Orlando, FL, USA: Association for Computing Machinery, 2017, pp. 262–277. ISBN: 9781450353458. DOI: 10.1145/3134600.3134635. URL: https://doi.org/10.1145/3134600.3134635.

[96] John R. Josephson and Susan G. Josephson. *Abductive Inference: Computation, Philosophy, Technology*. Cambridge: Cambridge University Press, Oct. 2009. ISBN: 978-0-511-53012-8.

[97] Md Abdul Kadir, Amir Mosavi, and Daniel Sonntag. *Assessing XAI: Unveiling Evaluation Metrics for Local Explanation, Taxonomies, Key Concepts, and Practical Applications*. May 2023. DOI: 10.31224/2989. (Visited on 11/07/2024).

[98] A. C. Kakas, R. A. Kowalski, and F. Toni. "Abductive Logic Programming". In: *Journal of Logic and Computation* 2.6 (1992), pp. 719–770. ISSN: 0955-792X, 1465-363X. DOI: 10.1093/logcom/2.6.719. URL: https://academic.oup.com/logcom/article-lookup/doi/10.1093/logcom/2.6.719 (visited on 01/31/2025).

[99] Konrad Wolfgang Kallus. *Erstellung von Fragebogen*. 2., aktualisierte und überarbeitete Auflage. UTB ; Literaturverzeichnis: Seite 145 - 150. Wien : facultas, 2016. URL: https://elibrary.utb.de/doi/book/10.36198/9783838544656.

[100] Margot E Kaminski. "The right to explanation, explained". In: *Research Handbook on Information Law and Governance*. Edward Elgar Publishing, 2021, pp. 278–299.

[101] Shiva P. Kasiviswanathan and Adam Smith. "On the Semantics of Differential Privacy: A Bayesian Formulation". In: *Journal of Privacy and Confidentiality* 6.1 (June 1, 2014). ISSN: 2575-8527. DOI: 10.29012/jpc.v6i1.634. URL: https://journalprivacyconfidentiality.org/index.php/jpc/article/view/634 (visited on 02/26/2025).

[102] Rémi Kazmierczak et al. *Benchmarking XAI Explanations with Human-Aligned Evaluations*. Nov. 2024. arXiv: 2411.02470 [cs]. (Visited on 11/11/2024).

[103] Bernard Keenan and Kacper Sokol. *Mind the Gap! Bridging Explainable Artificial Intelligence and Human Understanding with Luhmann's Functional Theory of Communication*. July 2024. arXiv: 2302.03460 [cs]. (Visited on 11/07/2024).

[104] Frank C. Keil. "Explanation and Understanding". In: *Annual Review of Psychology* 57.1 (Jan. 2006), pp. 227–254. ISSN: 0066-4308, 1545-2085. DOI: 10.1146/annurev.psych.57.102904.190100. (Visited on 11/07/2024).

[105] M. G. KENDALL. "A NEW MEASURE OF RANK CORRELATION". In: *Biometrika* 30.1-2 (June 1938), pp. 81–93. ISSN: 0006-3444. DOI: 10.1093/biomet/30.1-2.81. eprint: https://academic.oup.com/biomet/article-pdf/30/1-2/81/423380/30-1-2-81.pdf. URL: https://doi.org/10.1093/biomet/30.1-2.81.

[106] KIE. *Shapley Additive Explanations (SHAP)*. June 2021. URL: https://www.youtube.com/watch?v=VB9uV-x0gtg (visited on 11/04/2024).

[107] Sunnie S. Y. Kim et al. ""Help Me Help the AI": Understanding How Explainability Can Support Human-AI Interaction". In: *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*. Hamburg Germany: ACM, Apr. 2023, pp. 1–17. ISBN: 978-1-4503-9421-5. DOI: 10.1145/3544548.3581001. URL: https://dl.acm.org/doi/10.1145/3544548.3581001 (visited on 03/12/2025).

[108] Katarzyna Kobylińska et al. *Exploration of the Rashomon Set Assists Trustworthy Explanations for Medical Data*. Sept. 2023. arXiv: 2308.11446 [cs]. (Visited on 11/10/2024).

[109] Maximilian A. Kohl et al. "Explainability as a Non-Functional Requirement". In: *2019 IEEE 27th International Requirements Engineering Conference (RE)*. Jeju Island, Korea (South): IEEE, Sept. 2019, pp. 363–368. ISBN: 978-1-72813-912-8. DOI: 10.1109/RE.2019.00046. (Visited on 10/05/2024).

[110] Narine Kokhlikyan et al. *Captum: A unified and generic model interpretability library for PyTorch*. 2020. arXiv: 2009.07896 [cs.LG].

[111] Robert Kowalski. "Algorithm = Logic + Control". In: *Communications of the ACM* 22.7 (1979), pp. 424–436.

[112] Robert Kowalski. *Computational Logig and Human Thinking: How to Be Artificially Intelligent*. 1st ed. Cambridge University Press, July 21, 2011. ISBN: 978-0-521-19482-2. DOI: 10.1017/CBO9780511984747. URL: https://www.cambridge.org/core/product/identifier/9780511984747/type/book (visited on 01/31/2025).

[113] Robert Kowalski. *Logic for problem solving*. Vol. 75. Department of Computational Logic, Edinburgh University, 1974.

[114] Robert Kowalski. "Logic Programming". In: *Handbook of the History of Logic*. Vol. 9. Elsevier, 2014, pp. 523–569. ISBN: 978-0-444-51624-4. DOI: 10.1016/B978-0-444-51624-4.50012-5. URL: https://linkinghub.elsevier.com/retrieve/pii/B9780444516244500125 (visited on 02/04/2025).

[115] Alex Krizhevsky, Geoffrey Hinton, et al. "Learning multiple layers of features from tiny images". In: *University of Toronto* (2009).

[116] Alexey Kurakin, Ian Goodfellow, and Samy Bengio. *Adversarial examples in the physical world*. 2017. arXiv: 1607.02533 [cs.CV]. URL: https://arxiv.org/abs/1607.02533.

[117] Charis Lanaras et al. "Super-Resolution of Sentinel-2 Images: Learning a Globally Applicable Deep Neural Network". In: *ISPRS Journal of Photogrammetry and Remote Sensing* 146 (Dec. 2018), pp. 305–319. ISSN: 09242716. DOI: 10.1016/j.isprsjprs.2018.09.018. URL: https://linkinghub.elsevier.com/retrieve/pii/S0924271618302636 (visited on 01/23/2025).

[118] J. Richard Landis and Gary G. Koch. "The Measurement of Observer Agreement for Categorical Data". In: *Biometrics* 33.1 (Mar. 1977), p. 159. ISSN: 0006341X. DOI: 10.2307/2529310. JSTOR: 2529310. URL: https://www.jstor.org/stable/2529310?origin=crossref (visited on 04/16/2025).

[119] Moritz Leitner. "Federated Learning for Private Synthetic Data Generation". MA thesis. Karlsruher Institut für Technologie (KIT) / Karlsruher Institut für Technologie (KIT), 2023. 109 pp. DOI: 10.5445/IR/1000172145.

[120] Quentin Lhoest et al. “Datasets: A Community Library for Natural Language Processing”. In: *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Online and Punta Cana, Dominican Republic: Association for Computational Linguistics, Nov. 2021, pp. 175–184. arXiv: 2109.02846 [cs.CL]. URL: https://aclanthology.org/2021.emnlp-demo.21.

[121] Sichao Li and Amanda Barnard. *Variance Tolerance Factors For Interpreting ALL Neural Networks*. May 2023. arXiv: 2209.13858 [cs]. (Visited on 11/10/2024).

[122] Tian Li et al. “Federated Learning: Challenges, Methods, and Future Directions”. In: *IEEE Signal Processing Magazine* 37.3 (May 2020), pp. 50–60. ISSN: 1053-5888, 1558-0792. DOI: 10.1109/MSP.2020.2975749. (Visited on 10/05/2024).

[123] Tian Li et al. *Federated Optimization in Heterogeneous Networks*. Apr. 2020. arXiv: 1812.06127 [cs]. (Visited on 11/01/2024).

[124] Q. Vera Liao et al. “Connecting Algorithmic Research and Usage Contexts: A Perspective of Contextualized Evaluation for Explainable AI ”. In: *Proceedings of the AAAI Conference on Human Computation and Crowdsourcing* 10.1 (Oct. 2022), pp. 147–159. ISSN: 2769-1349, 2769-1330. DOI: 10.1609/hcomp.v10i1.21995. (Visited on 11/12/2024).

[125] Peter Lipton. “Inference to the best explanation”. In: *A Companion to the Philosophy of Science* (2017), pp. 184–193.

[126] Tania Lombrozo and Emily G. Liquin. “Explanation Is Effective Because It Is Selective”. In: *Current Directions in Psychological Science* 32.3 (June 2023), pp. 212–219. ISSN: 0963-7214, 1467-8721. DOI: 10.1177/09637214231156106. (Visited on 11/07/2024).

[127] Luis M. Lopez-Ramos et al. *Interplay between Federated Learning and Explainable Artificial Intelligence: A Scoping Review*. Nov. 2024. arXiv: 2411.05874 [cs]. (Visited on 11/21/2024).

[128] Scott M Lundberg and Su-In Lee. “A Unified Approach to Interpreting Model Predictions”. In: *Advances in Neural Information Processing Systems 30*. Ed. by I. Guyon et al. Curran Associates, Inc., 2017, pp. 4765–4774. URL: http://papers.nips.cc/paper/7062-a-unified-approach-to-interpreting-model-predictions.pdf.

[129] Aleksander Madry et al. *Towards Deep Learning Models Resistant to Adversarial Attacks*. 2019. arXiv: 1706.06083 [stat.ML]. URL: https://arxiv.org/abs/1706.06083.

[130] Lorenzo Magnani. *Abduction, Reason and Science*. Boston, MA: Springer US, 2001. ISBN: 978-1-4419-8562-0. DOI: 10.1007/978-1-4419-8562-0. URL: http://link.springer.com/10.1007/978-1-4419-8562-0 (visited on 02/06/2025).

[131] David R Mandel and Darrin R Lehman. “Counterfactual thinking and ascriptions of cause and preventability.” In: *Journal of personality and social psychology* 71.3 (1996), p. 450.

[132] Brendan McMahan et al. “Communication-efficient learning of deep networks from decentralized data”. In: *Artificial intelligence and statistics*. PMLR. 2017, pp. 1273–1282.

[133] H. Brendan McMahan et al. *Learning Differentially Private Recurrent Language Models*. Feb. 24, 2018. DOI: 10.48550/arXiv.1710.06963. arXiv: 1710.06963 [cs]. URL: http://arxiv.org/abs/1710.06963 (visited on 03/02/2025). Pre-published.

[134] Fanyu Meng et al. *CohEx: A Generalized Framework for Cohort Explanation*. Oct. 2024. arXiv: 2410.13190 [cs]. (Visited on 11/10/2024).

[135] Dang Minh et al. "Explainable Artificial Intelligence: A Comprehensive Review". In: *Artificial Intelligence Review* 55.5 (June 2022), pp. 3503–3568. ISSN: 0269-2821, 1573-7462. DOI: 10.1007/s10462-021-10088-y. (Visited on 10/13/2024).

[136] Christoph Molnar et al. "General Pitfalls of Model-Agnostic Interpretation Methods for Machine Learning Models". In: *xxAI - Beyond Explainable AI: International Workshop, Held in Conjunction with ICML 2020, July 18, 2020, Vienna, Austria, Revised and Extended Papers*. Ed. by Andreas Holzinger et al. Cham: Springer International Publishing, 2022, pp. 39–68. ISBN: 978-3-031-04083-2. DOI: 10.1007/978-3-031-04083-2_4. URL: https://doi.org/10.1007/978-3-031-04083-2_4.

[137] Seyed-Mohsen Moosavi-Dezfooli, Alhussein Fawzi, and Pascal Frossard. *DeepFool: a simple and accurate method to fool deep neural networks*. 2016. arXiv: 1511.04599 [cs.LG]. URL: https://arxiv.org/abs/1511.04599.

[138] Ramaravind K Mothilal, Amit Sharma, and Chenhao Tan. "Explaining machine learning classifiers through diverse counterfactual explanations". In: *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*. 2020, pp. 607–617.

[139] Sebastian Müller et al. *An Empirical Evaluation of the Rashomon Effect in Explainable Machine Learning*. June 2023. arXiv: 2306.15786 [cs]. (Visited on 10/11/2024).

[140] Lauritz Munch and Jens Christian Bjerring. "Can Large Language Models Help Solve the Cost Problem for the Right to Explanation?" In: *Journal of Medical Ethics* (Sept. 2024), jme-2023–109737. ISSN: 0306-6800, 1473-4257. DOI: 10.1136/jme-2023-109737. (Visited on 12/09/2024).

[141] Meike Nauta et al. "From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review on Evaluating Explainable AI". In: *ACM Computing Surveys* 55.13s (Dec. 2023), pp. 1–42. ISSN: 0360-0300, 1557-7341. DOI: 10.1145/3583558. (Visited on 11/10/2024).

[142] Sidra Naveed, Gunnar Stevens, and Dean Robin-Kern. "An Overview of the Empirical Evaluation of Explainable AI (XAI): A Comprehensive Guideline for User-Centered Evaluation in XAI". In: *Applied Sciences* 14.23 (Dec. 2024), p. 11288. ISSN: 2076-3417. DOI: 10.3390/app142311288. (Visited on 12/09/2024).

[143] An-phi Nguyen and María Rodríguez Martínez. *On Quantitative Aspects of Model Interpretability*. July 2020. arXiv: 2007.07584 [cs]. (Visited on 11/07/2024).

[144] Maria-Irina Nicolae et al. "Adversarial Robustness Toolbox v1.2.0". In: *CoRR* 1807.01069 (2018). URL: https://arxiv.org/pdf/1807.01069.

[145] World Health Organization. *Road traffic injuries*. Dec. 2023. URL: https://www.who.int/news-room/fact-sheets/detail/road-traffic-injuries (visited on 12/02/2024).

[146] Evandro S. Ortigossa, Thales Gonçalves, and Luis Gustavo Nonato. “EXplainable Artificial Intelligence (XAI)—From Theory to Methods and Applications”. In: *IEEE Access* 12 (2024), pp. 80799–80846. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2024.3409843. (Visited on 10/05/2024).

[147] Nicolas Papernot et al. *The Limitations of Deep Learning in Adversarial Settings*. 2015. arXiv: 1511.07528 [cs.CR]. URL: https://arxiv.org/abs/1511.07528.

[148] Darsh Parekh et al. “A Review on Autonomous Vehicles: Progress, Methods and Challenges”. In: *Electronics* 11.14 (July 2022), p. 2162. ISSN: 2079-9292. DOI: 10.3390/electronics11142162. (Visited on 12/02/2024).

[149] Karl Pearson. “Note on Regression and Inheritance in the Case of Two Parents”. In: *Proceedings of the Royal Society of London Series I* 58 (Jan. 1895), pp. 240–242.

[150] Yun Peng and James A. Reggia. *Abductive Inference Models for Diagnostic Problem-Solving*. New York, NY: Springer New York, 1990. ISBN: 978-1-4419-8682-5. DOI: 10.1007/978-1-4419-8682-5. URL: http://link.springer.com/10.1007/978-1-4419-8682-5 (visited on 02/06/2025).

[151] Clement Poiret et al. *Can We Agree? On the Rashōmon Effect and the Reliability of Post-Hoc Explainable AI*. Aug. 2023. arXiv: 2308.07247 [cs]. (Visited on 11/10/2024).

[152] Jürgen Raithel. *Quantitative Forschung :* 2., durchgesehene Auflage. Lehrbuch. Literaturverzeichnis: Seiten 211-213. Wiesbaden : VS Verlag für Sozialwissenschaften, 2008. URL: http://d-nb.info/989642267/04.

[153] Sashank Reddi et al. *Adaptive Federated Optimization*. Sept. 2021. arXiv: 2003.00295 [cs]. (Visited on 11/01/2024).

[154] 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*. San Francisco California USA: ACM, Aug. 2016, pp. 1135–1144. ISBN: 978-1-4503-4232-2. DOI: 10.1145/2939672.2939778. (Visited on 10/08/2024).

[155] Pascal Riedel et al. “Comparative Analysis of Open-Source Federated Learning Frameworks - a Literature-Based Survey and Review”. In: *International Journal of Machine Learning and Cybernetics* 15.11 (Nov. 2024), pp. 5257–5278. ISSN: 1868-8071, 1868-808X. DOI: 10.1007/s13042-024-02234-z. (Visited on 10/18/2024).

[156] Laura Rieger and Lars Kai Hansen. “IROF: a low resource evaluation metric for explanation methods”. In: *CoRR* abs/2003.08747 (2020). arXiv: 2003.08747. URL: https://arxiv.org/abs/2003.08747.

[157] Nicola Rieke et al. “The Future of Digital Health with Federated Learning”. In: *npj Digital Medicine* 3.1 (Sept. 14, 2020), p. 119. ISSN: 2398-6352. DOI: 10.1038/s41746-020-00323-1. URL: https://www.nature.com/articles/s41746-020-00323-1 (visited on 01/13/2025).

[158] On-Road Automated Driving (ORAD) committee. *Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles*. DOI: 10.4271/J3016_202104. (Visited on 12/02/2024).

[159] Avi Rosenfeld and Ariella Richardson. "Explainability in Human–Agent Systems". In: *Autonomous Agents and Multi-Agent Systems* 33.6 (Nov. 2019), pp. 673–705. ISSN: 1387-2532, 1573-7454. DOI: 10.1007/s10458-019-09408-y. (Visited on 10/05/2024).

[160] Saifullah Saifullah et al. "The Privacy-Explainability Trade-off: Unraveling the Impacts of Differential Privacy and Federated Learning on Attribution Methods". In: *Frontiers in Artificial Intelligence* 7 (July 3, 2024), p. 1236947. ISSN: 2624-8212. DOI: 10.3389/frai.2024.1236947. URL: https://www.frontiersin.org/articles/10.3389/frai.2024.1236947/full (visited on 10/05/2024).

[161] Rainer Schnell and Paul B. Hill. *Methoden der empirischen Sozialforschung*. 11., überarbeitete Auflage. De Gruyter Studium. Literaturverzeichnis: Seite 463-513. Berlin ; De Gruyter Oldenbourg, 2018. URL: http://www.blickinsbuch.de/item/650655bd5c7d06de54d85e500b16a14d.

[162] Gesina Schwalbe and Bettina Finzel. "A Comprehensive Taxonomy for Explainable Artificial Intelligence: A Systematic Survey of Surveys on Methods and Concepts". In: *Data Mining and Knowledge Discovery* 38.5 (Sept. 2024), pp. 3043–3101. ISSN: 1384-5810, 1573-756X. DOI: 10.1007/s10618-022-00867-8. (Visited on 10/13/2024).

[163] Maike Schwammberger, Raffaela Mirandola, and Nils Wenninghoff. "Explainability Engineering Challenges: Connecting Explainability Levels to Run-Time Explainability". In: *Explainable Artificial Intelligence*. Ed. by Luca Longo, Sebastian Lapuschkin, and Christin Seifert. Vol. 2156. Cham: Springer Nature Switzerland, 2024, pp. 205–218. DOI: 10.1007/978-3-031-63803-9_11. (Visited on 10/05/2024).

[164] Ramprasaath R. Selvaraju et al. "Grad-CAM: Visual Explanations from Deep Networks via Gradient-based Localization". In: *International Journal of Computer Vision* 128.2 (Feb. 2020), pp. 336–359. ISSN: 0920-5691, 1573-1405. DOI: 10.1007/s11263-019-01228-7. arXiv: 1610.02391 [cs]. (Visited on 11/09/2024).

[165] Amit Sheth, Kaushik Roy, and Manas Gaur. *Neurosymbolic AI – Why, What, and How*. May 1, 2023. DOI: 10.48550/arXiv.2305.00813. arXiv: 2305.00813 [cs]. URL: http://arxiv.org/abs/2305.00813 (visited on 02/01/2025). Pre-published.

[166] Avanti Shrikumar et al. *Not Just a Black Box: Learning Important Features Through Propagating Activation Differences*. Apr. 11, 2017. DOI: 10.48550/arXiv.1605.01713. arXiv: 1605.01713 [cs]. URL: http://arxiv.org/abs/1605.01713 (visited on 01/14/2025). Pre-published.

[167] Herbert A Simon. "What Is an Explanation of Behavior?" In: *PSYCHOLOGICAL SCIENCE* 3.3 (1992).

[168] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. *Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps*. Apr. 19, 2014. arXiv: 1312.6034 [cs]. URL: http://arxiv.org/abs/1312.6034 (visited on 11/09/2024). Pre-published.

[169] Ayush Somani, Alexander Horsch, and Dilip K. Prasad. "Introduction to Interpretability". In: *Interpretability in Deep Learning*. Cham: Springer International Publishing, 2023, pp. 1–67. ISBN: 978-3-031-20639-9. DOI: 10.1007/978-3-031-20639-9_1. URL: https://doi.org/10.1007/978-3-031-20639-9_1.

[170] Francesco Sovrano et al. “A Survey on Methods and Metrics for the Assessment of Explainability Under the Proposed AI Act”. In: *Frontiers in Artificial Intelligence and Applications*. Ed. by Erich Schweighofer. IOS Press, Dec. 2021. DOI: 10.3233/FAIA210342. (Visited on 10/31/2024).

[171] C. Spearman. “The Proof and Measurement of Association between Two Things”. In: *The American Journal of Psychology* 15.1 (1904), pp. 72–101. ISSN: 00029556. URL: http://www.jstor.org/stable/1412159 (visited on 11/24/2024).

[172] Timo Speith. “A Review of Taxonomies of Explainable Artificial Intelligence (XAI) Methods”. In: *2022 ACM Conference on Fairness, Accountability, and Transparency*. Seoul Republic of Korea: ACM, June 2022, pp. 2239–2250. ISBN: 978-1-4503-9352-2. DOI: 10.1145/3531146.3534639. (Visited on 10/05/2024).

[173] Barbara A Spellman and David R Mandel. “Causal Reasoning, Psychology Of”. In: *Encyclopedia of Cognitive Science*. Ed. by Lynn Nadel. 1st ed. Wiley, Jan. 2006. DOI: 10.1002/0470018860.s00491. (Visited on 11/25/2024).

[174] Ramya Srinivasan and Ajay Chander. “Explanation Perspectives from the Cognitive Sciences—A Survey ”. In: *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*. Yokohama, Japan: International Joint Conferences on Artificial Intelligence Organization, July 2020, pp. 4812–4818. ISBN: 978-0-9992411-6-5. DOI: 10.24963/ijcai.2020/670. (Visited on 10/05/2024).

[175] C. Studholme, D.L.G. Hill, and D.J. Hawkes. “An Overlap Invariant Entropy Measure of 3D Medical Image Alignment”. In: *Pattern Recognition* 32.1 (Jan. 1999), pp. 71–86. ISSN: 00313203. DOI: 10.1016/S0031-3203(98)00091-0. (Visited on 11/29/2024).

[176] Mukund Sundararajan and Amir Najmi. “The Many Shapley Values for Model Explanation”. In: (2020), pp. 9269–9278.

[177] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. *Axiomatic Attribution for Deep Networks*. June 2017. arXiv: 1703.01365 [cs]. (Visited on 11/07/2024).

[178] Hiwot Belay Tadesse et al. *Directly Optimizing Explanations for Desired Properties*. Oct. 2024. arXiv: 2410.23880 [cs]. (Visited on 11/11/2024).

[179] Mingxing Tan and Quoc V. Le. *EfficientNetV2: Smaller Models and Faster Training*. June 2021. DOI: 10.48550/arXiv.2104.00298. arXiv: 2104.00298 [cs]. (Visited on 12/15/2024).

[180] Asadullah Tariq et al. “Trustworthy Federated Learning: A Comprehensive Review, Architecture, Key Challenges, and Future Research Prospects”. In: *IEEE Open Journal of the Communications Society* (2024).

[181] Rafael Teixeira et al. *Balancing Privacy and Explainability in Federated Learning*. Dec. 2023. DOI: 10.21203/rs.3.rs-3714454/v1. (Visited on 10/05/2024).

[182] Niko Tsakalakis et al. “A Typology of Explanations for Explainability-by-Design”. In: *ACM Journal on Responsible Computing* (2024).

[183] Rini van Solingen (Revision) et al. "Goal Question Metric (GQM) Approach". In: *Encyclopedia of Software Engineering*. John Wiley & Sons, Ltd, 2002. ISBN: 978-0-471-02895-6. DOI: 10.1002/0471028959.sof142. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/0471028959.sof142. URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/0471028959.sof142.

[184] Haofan Wang et al. *Score-CAM: Score-Weighted Visual Explanations for Convolutional Neural Networks*. Apr. 2020. DOI: 10.48550/arXiv.1910.01279. arXiv: 1910.01279 [cs]. (Visited on 11/26/2024).

[185] Z. Wang et al. "Image Quality Assessment: From Error Visibility to Structural Similarity". In: *IEEE Transactions on Image Processing* 13.4 (Apr. 2004), pp. 600–612. ISSN: 1057-7149. DOI: 10.1109/TIP.2003.819861. (Visited on 11/29/2024).

[186] Zhou Wang and A.C. Bovik. "A universal image quality index". In: *IEEE Signal Processing Letters* 9.3 (2002), pp. 81–84. DOI: 10.1109/97.995823.

[187] Ziming Wang, Changwu Huang, and Xin Yao. "A Roadmap of Explainable Artificial Intelligence: Explain to Whom, When, What and How?" In: *ACM Transactions on Autonomous and Adaptive Systems* 19.4 (Dec. 31, 2024), pp. 1–40. ISSN: 1556-4665, 1556-4703. DOI: 10.1145/3702004. URL: https://dl.acm.org/doi/10.1145/3702004 (visited on 03/19/2025).

[188] Jonas Wanner et al. "Stop Ordering Machine Learning Algorithms by Their Explainability! An Empirical Investigation of the Tradeoff Between Performance and Explainability". In: *Responsible AI and Analytics for an Ethical and Inclusive Digitized Society*. Ed. by Denis Dennehy et al. Cham: Springer International Publishing, 2021, pp. 245–258. ISBN: 978-3-030-85447-8.

[189] Rui Xin et al. *Exploring the Whole Rashomon Set of Sparse Decision Trees*. Oct. 2022. arXiv: 2209.08040 [cs]. (Visited on 11/10/2024).

[190] Keiichiro Yamamura et al. *Diversified Adversarial Attacks based on Conjugate Gradient Method*. 2022. arXiv: 2206.09628 [cs.LG]. URL: https://arxiv.org/abs/2206.09628.

[191] Chih-Kuan Yeh et al. *On the (In)Fidelity and Sensitivity for Explanations*. Nov. 2019. arXiv: 1901.09392 [cs]. (Visited on 11/07/2024).

[192] Dong Yin et al. *Byzantine-Robust Distributed Learning: Towards Optimal Statistical Rates*. Feb. 2021. arXiv: 1803.01498 [cs]. (Visited on 11/01/2024).

[193] Ashkan Yousefpour et al. *Opacus: User-Friendly Differential Privacy Library in PyTorch*. Aug. 22, 2022. DOI: 10.48550/arXiv.2109.12298. arXiv: 2109.12298 [cs]. URL: http://arxiv.org/abs/2109.12298 (visited on 02/26/2025). Pre-published.

[194] Roberta H Yuhas, Alexander FH Goetz, and Joe W Boardman. "Discrimination among semi-arid landscape endmembers using the spectral angle mapper (SAM) algorithm". In: (1992).

[195] Jerrold Zar. *Biostatistical Analysis*. Pearson Deutschland, 2013, p. 760. ISBN: 9781292024042. URL: https://elibrary.pearson.de/book/99.150005/9781292037110.

[196] Yue Zhao et al. “Federated Learning with Non-IID Data”. In: (2018). DOI: 10.48550/arXiv.1806.00582. arXiv: 1806.00582 [cs, stat]. (Visited on 10/05/2024).

[197] Bolei Zhou et al. *Learning Deep Features for Discriminative Localization*. Dec. 2015. DOI: 10.48550/arXiv.1512.04150. arXiv: 1512.04150 [cs]. (Visited on 01/14/2025).

[198] Jianlong Zhou et al. “Evaluating the Quality of Machine Learning Explanations: A Survey on Methods and Metrics”. In: *Electronics* 10.5 (Mar. 2021), p. 593. ISSN: 2079-9292. DOI: 10.3390/electronics10050593. (Visited on 10/13/2024).

[199] Hangyu Zhu et al. “Federated Learning on Non-IID Data: A Survey”. In: *Neurocomputing* 465 (Nov. 2021), pp. 371–390. ISSN: 09252312. DOI: 10.1016/j.neucom.2021.07.098. (Visited on 11/01/2024).

Acronyms

AI Artificial Intelligence. viii, 1, 8, 15, 28–30, 37, 43, 55, 76, 78–80, 109

ART Adversarial Robustness Toolbox. 1, 61

Auto-PGD Auto Projected Gradient Descent. 1, 61

BIM Basic Iterative Method. 1, 61

CNN Convolutional Neural Network. 1, 20

CW Carlini & Wagner. 1, 61

DAUC Delete Area Under Curve. 1

DNN Deep Neural Network. 1, 5

DP Differential Privacy. vii, ix, 1, 2, 50, 51, 67–69, 71, 83, 84

DP-SGD Differential Privacy Stochastic Gradient Descent. 1, 68

FedAvg Federated Averaging. 1, 9–12

FedSGD Federated Stochastic Gradient Descent. 1, 10, 11

FGSM Fast Gradient Sign Method. 1, 61

FL Federated Learning. vii, ix, 1, 2, 5, 8–10, 13–15, 23–25, 38, 39, 41, 44, 47–56, 60, 61, 65, 68–72, 83, 84

GPU Graphics Processing Unit. 1, 84

GQM Goal-Question-Metric. 1

GQM Goal Question Metric. 1, 25, 50, 51, 73

Grad-CAM Gradient-weighted Class Activation Mapping. 1, 20

IAUC Insert Area Under Curve. 1, 53, 54

iff if and only if. 1, 27

IID independent and identically distributed. ix, 1, 51, 53–56, 69, 71

IoT Internet of Things. 1, 15

IROF Iterative Removal of Features. 1, 54

KIT Karlsruhe Institute for Technology. 1, 73, 74

LIME Local Interpretable Model Agnostic Explanation. 1, 15–17, 59

LLM Large Language Model. 1, 43, 84, 85

ML Machine Learning. ix, 1, 2, 5, 7–9, 13–19, 24, 26, 30, 39–41, 47, 49–51, 53, 54, 65, 67–69, 71, 72, 76, 83, 84

MPRT Model Parameter Randomization. 1, 53, 69

MSE Mean Squared Error. 1, 52, 53

NMI Normalized Mutual Information. 1, 52, 53

NMSE Normalized Mean Squared Error. 1, 52, 53

non-IID non-independent and identically distributed. 1, 11, 12, 15

PEAR Pearson's Correlation Coefficient. 1, 52, 53

PET Privacy-Enhancing technology. 1, 67

PGD Project Gradient Descent. 1, 61

PGI Prediction Gap on Important feature metric. 1, 7

PSNR Peak Signal-to-Noise Ratio. 1, 52

RIS Relative Input Stability. 1

ROS Relative Output Stability. 1

SAM Spectral Angle Mapper. 1, 52

SDG Stochastic Gradient Descent. 1, 10

SE Software Engineering. 1, 75, 76

SHAP SHapley Additive exPlanations. 1, 15, 17, 24

SPEAR Spearman's Rank Correlation Coefficient. 1, 52–54, 56, 72, 115, 117, 118, 121, 122

SRE Signal to Reconstruction Error Ratio. 1, 52

SSIM Structural Similarity Index Measure. 1, 52, 53, 56, 58, 59, 69, 114, 115, 117, 121

TAU Kendall's Rank Correlation Coefficient. 1, 52, 53, 56, 115, 116, 118, 122

UIQ Universal Image Quality Index. 1, 52

XAI Explainable Artificial Intelligence. vii, ix, 1, 2, 7, 15, 17, 24–27, 29, 30, 33, 35, 36, 38–41, 43, 44, 47, 50–56, 59–63, 65, 67–69, 71, 74, 76, 79–84

YAP Yet Another Prolog. 1, 31

ZOO Zeroth Order Optimisation. 1, 61

A. Appendix

Explainability

TABLE 3. Supporting Abduction in AI

Process	Explainability requirements
1. Observation of an event or phenomenon.	Design clear interfaces to make it easy to identify what has happened.
	Design tools that make it easy to highlight events that could be considered anomalous or unusual for a particular domain.
2. Generation of one or more possible explanations for some observed event or phenomenon.	Design interfaces and affordances that either list potential (archived) hypotheses or help people compose a list of new potential hypotheses. Hypotheses in this case would be the "causes" or "reasons" that influence the AI's outputs.
3. Judging the plausibility of the candidate explanations.	Design interactions and affordances that make it easy for people to learn how the causes (inputs) affect system outputs. This would include support for contrastive analysis: How would a decision of the AI change if some variable had been different? Why would an AI decision *not* change if some variable were different?
4. Resolving the explanation.	Make it easy for people to explore how the AI operates when it is pushed to the boundaries of its competence envelope, to surmise the *when*, *how*, and *why*.
5. Extending the explanation.	Make it easy for people to revisit and revise any of their determinations if they receive new information or there are new surprises, or if they have new insights.
	Make it easy for people to explore follow-up questions that the explanatory or sensemaking process may have raised for them.
	Make it easy for people to access and share information about the AI.

Figure A.1.: Explainability Requirements to Support Abductive AI [85].

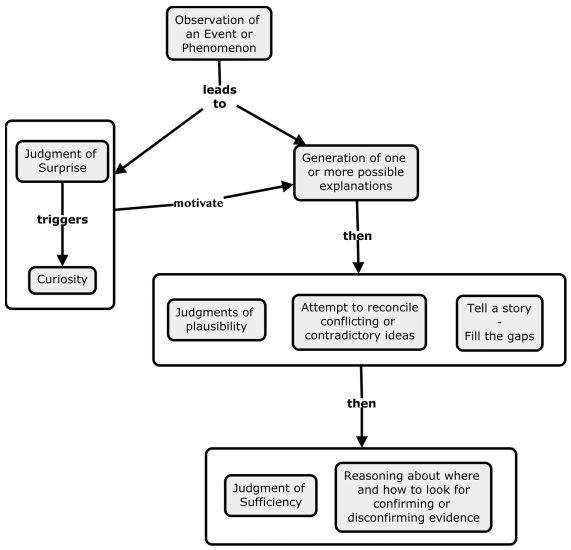


Figure A.2.: Process Model for Abduction [85].

Property	Reference
Abornmality	[5]
Actionability	[124]
Agreement	[22]
Biased inferences	[22]
Causality	[5, 15]
Certainty	[5, 15, 124, 141]
Coherence	[7, 8, 104, 124, 141]
Compactness	[40, 124, 141]
Compatibility	[18]
Completeness	[7, 8, 124, 141]
Complexity	[18, 40, 50, 141, 143]
Comprehensibility	[5, 15, 124]
Consistency	[5, 141]
Continuity	[141]
Contrastivity	[141]
Controllability	[141]
Convex Combination	[18]
Conviction	[18]
Coreness	[7]
Correctness	[141]
Counter-Monotonicity	[8]
Counterintuitive	[22]
Exhaustivity	[7]
Fairness	[5, 15]
Faithfulness	[18, 40, 50, 84, 97, 124, 178]
Feasibility	[7, 8]
Fidelity	[4, 8, 40, 50, 143, 191]
Non-Circularity	[104]
Globalness	[84]
Homogeneity	[40]

Property	Reference
Identity	[18]
Implementation Invariance	[177]
Interactivity	[15, 124]
Interpretability	[5, 15]
Irreducibility	[7, 8]
Linearity	[177]
Mismatch	[22]
Monotonicity	[8, 143]
Non-Contradictory	[22]
Non-Misleading	[22, 82]
Novelty	[124]
Personalization	[124]
Privacy	[5, 15]
Relevance	[7, 104]
Representativeness	[5, 7]
Robustness	[4, 5, 40, 50, 82, 104, 178]
Selectivity	[126]
Sensitivity	[5, 18, 40, 50, 97, 143, 177, 191]
Self-evidencing	[104]
Separability	[18]
Simplicity	[104, 143]
Smoothness	[178]
Sociological	[5]
Sparseness	[4]
Specifity	[5]
Stability	[3, 5, 22, 124]
Success	[7, 8]
Symmetry	[177]
Translucence	[5, 124]
Validity	[8]

Table A.1.: Explainability Properties from different Papers.

Federated Learning

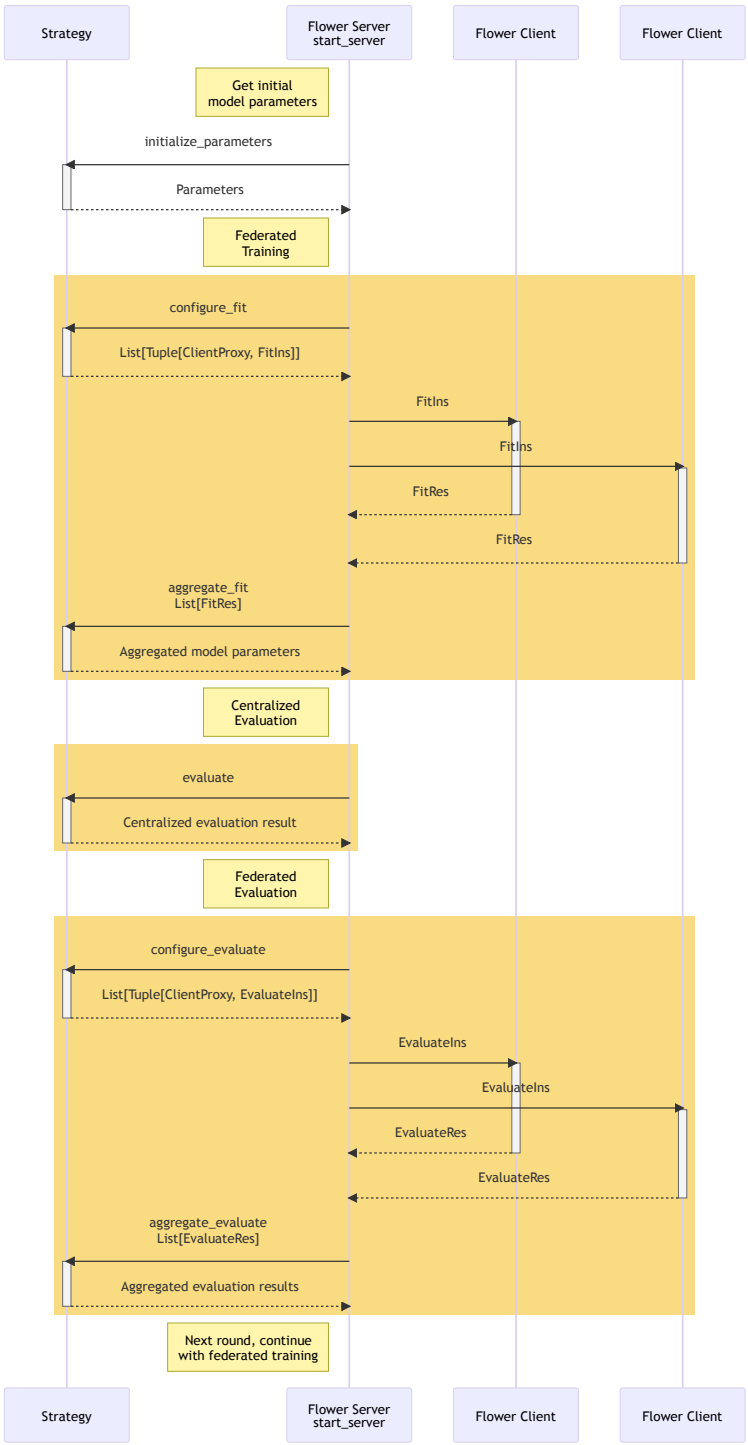


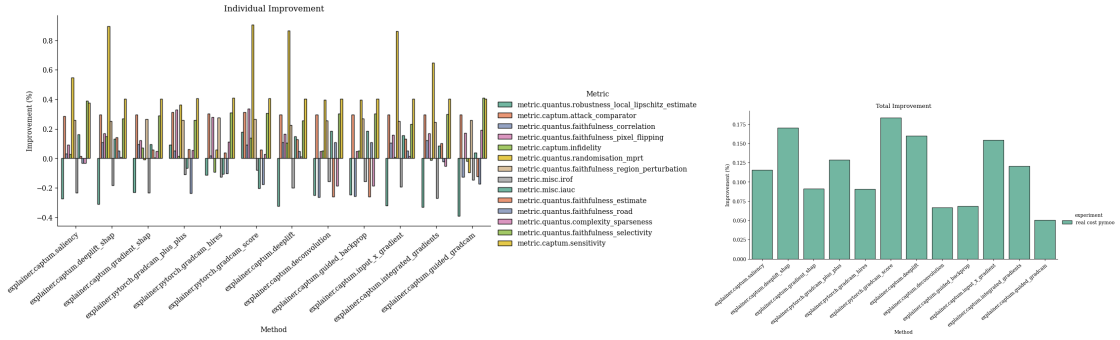
Figure A.3.: Flower Strategy Sequence Diagram [17].

Experiment Settings

Key	Value
num-server-rounds	25
fraction-fit	1.0
local-epochs	1
fraction-evaluate	1.0
batch-size	256
alpha	0.1
proximal_mu	1.0
beta_1	0.1
beta_2	0.1
eta	0.1
tau	1E-9
eta_l	0.1
server_momentum	0.0
server_learning_rate	1E-3
beta	0.2
num_malicious_clients	1
num_clients_to_keep	5
learning_rate	1E-3
Optimizer & Loss	CrossEntropyLoss & Adam
clipping_norm	0.5
dp-sensitivity	1.2
dp-epsilon	5.0
dp-delta	0.0001
dp-noise-multiplier	2

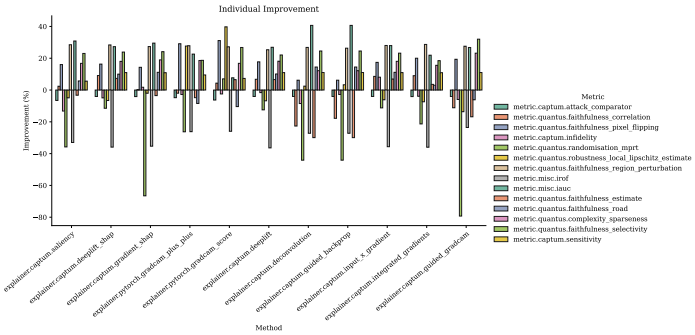
Table A.2.: Hyperparameters for Series of Experiments 1.

Experiment Data

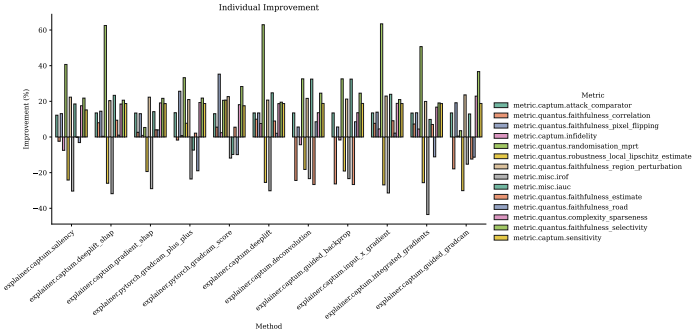


(a) Individual Improvement. **(b)** Overall Improvement.

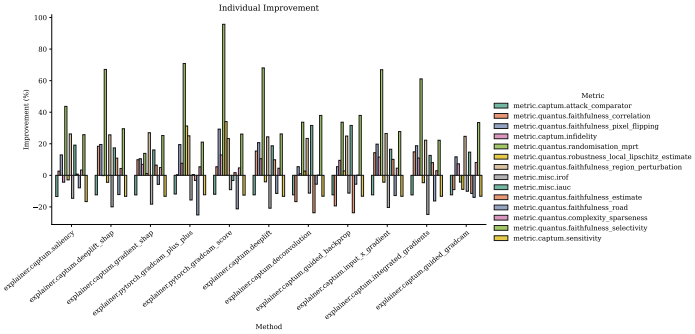
Figure A.4.: Results for Respecting the Real Cost e.g., Using $\lceil \psi \rceil$.



(a) Individual improvement via Averaging.



(b) Individual improvement via cvxpy.



(c) Individual improvement via pymoo.

Figure A.5.: Improvement via Optimization.

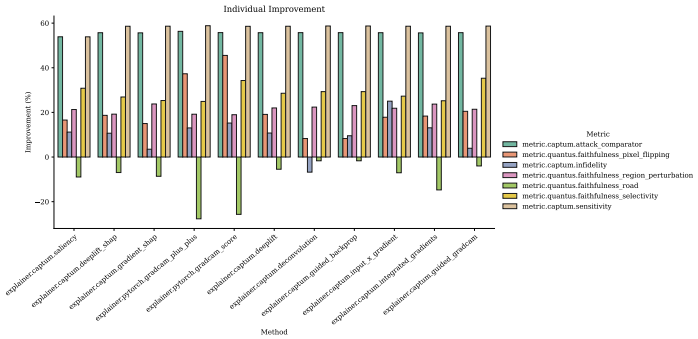


Figure A.6.: Individual improvement on Perturbation-based XAI Methods.

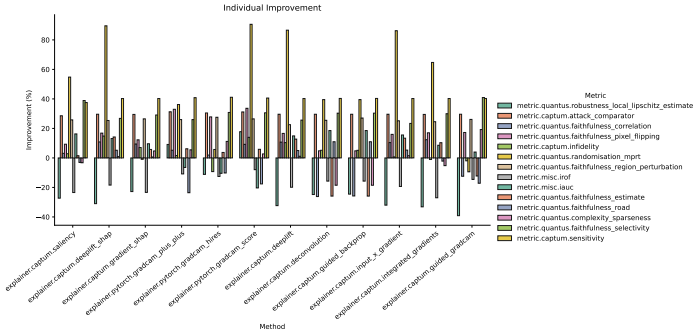
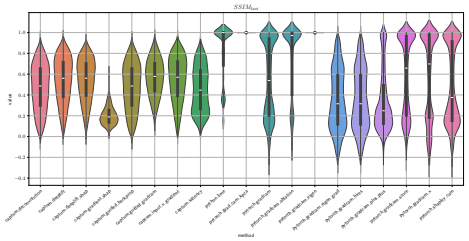
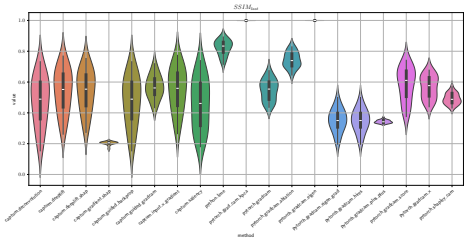


Figure A.7.: Individual improvement over all Classes simultaniously.

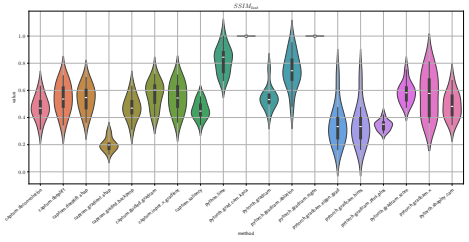
Stability Measurements



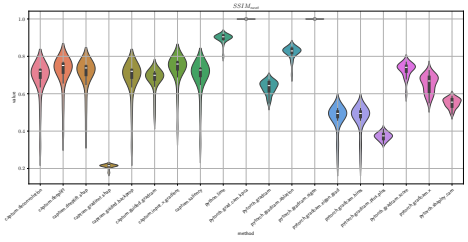
(a) SSIM comparing to last round.



(b) SSIM comparing to last round and grouped by round.

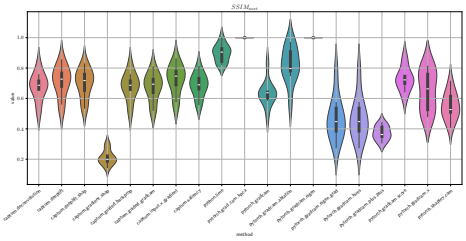


(c) SSIM comparing to last round and grouped by class.

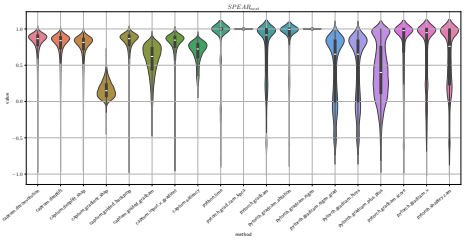


(d) SSIM comparing to next round and grouped by round.

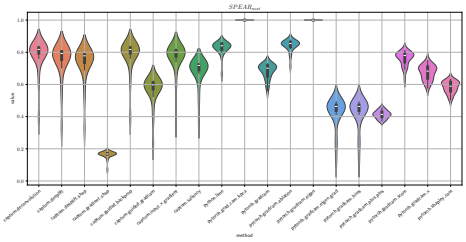
Figure A.8.: Measuring Stability FedAvg/IID (1).



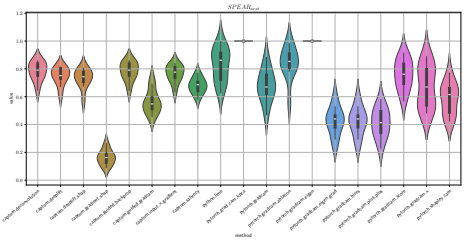
(a) SSIM comparing to next round and grouped by class.



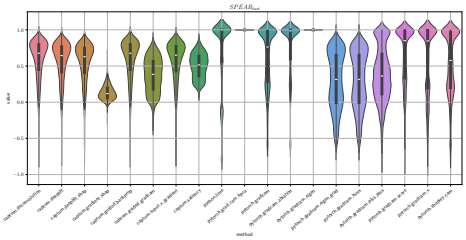
(b) SPEAR comparing to next round.



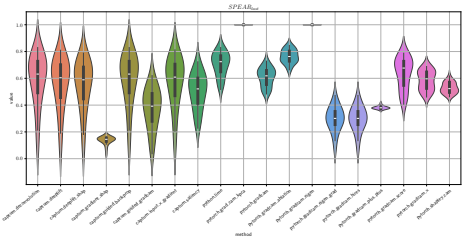
(c) SPEAR comparing to next round and grouped by round.



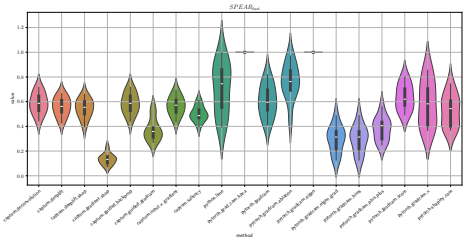
(d) SPEAR comparing to next round and grouped by class.



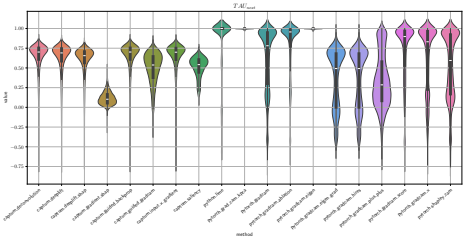
(e) SPEAR comparing to last round.



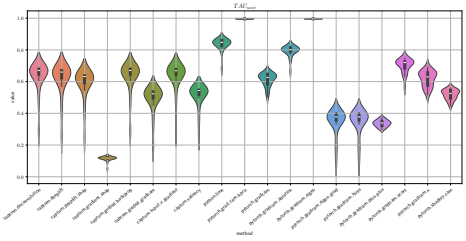
(f) SPEAR comparing to last round and grouped by round.



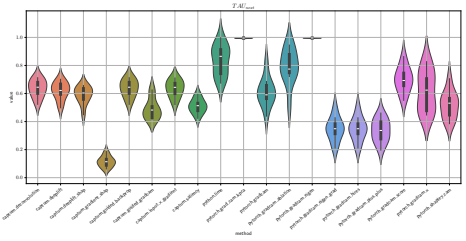
(g) SPEAR comparing to last round and grouped by class.



(h) TAU comparing to next round.

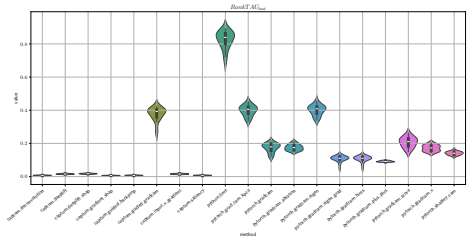


(i) TAU comparing to next round and grouped by round.

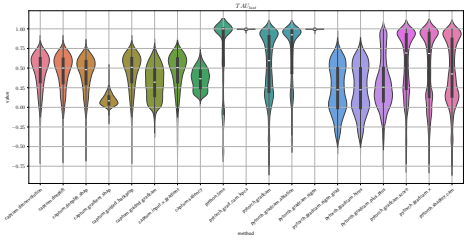


(j) TAU comparing to next round and grouped by class.

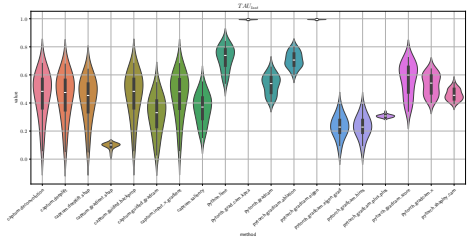
Figure A.9.: Measuring Stability FedAvg/IID (2)



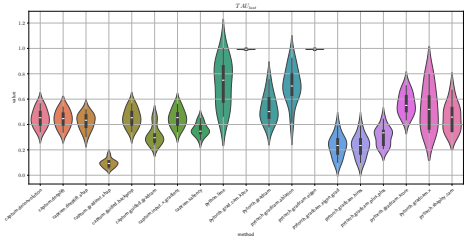
(a) RankTAU comparing to last round and grouped by round.



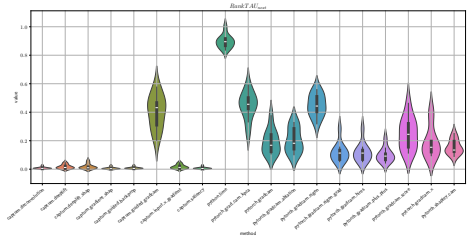
(b) TAU comparing to last round.



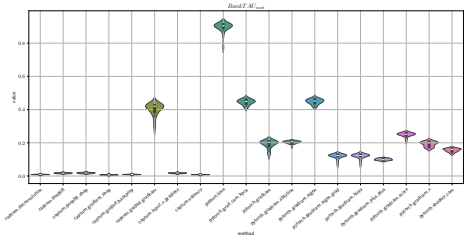
(c) TAU comparing to last round and grouped by round.



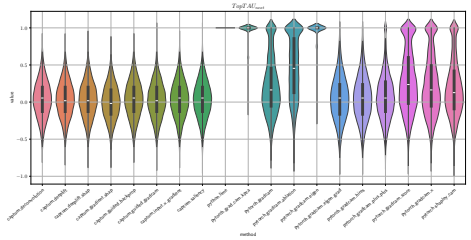
(d) TAU comparing to last round and grouped by class.



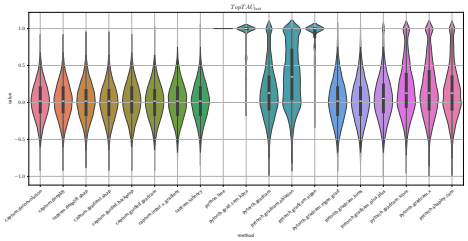
(e) RankTAU comparing to next round and grouped by class.



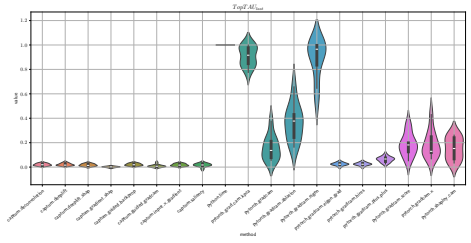
(f) RankTAU comparing to next round and grouped by round.



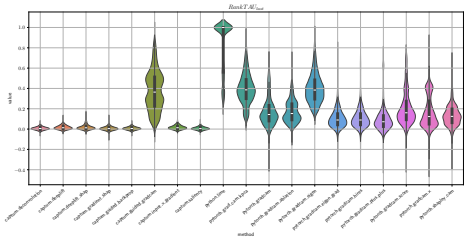
(g) TopTAU comparing to next round.



(h) TopTAU comparing to last round.

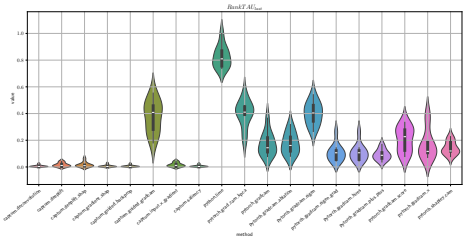


(i) TopTAU comparing to last round and grouped by class.

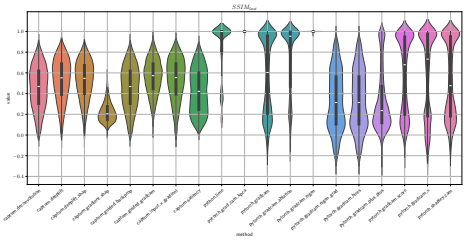


(j) RankTAU comparing to last round.

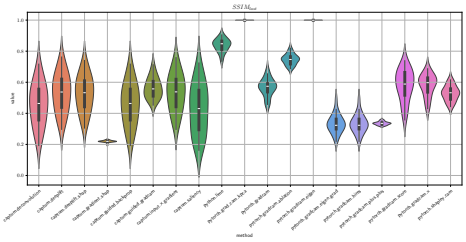
Figure A.10.: Measuring Stability FedAvg/IID (3).



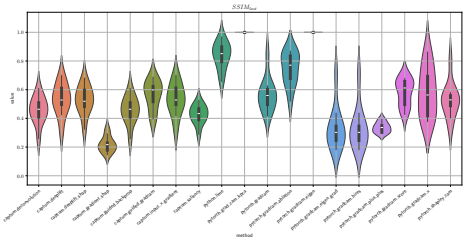
(a) RankTAU comparing to last round and grouped by class.



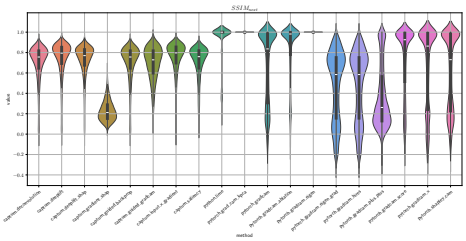
(b) SSIM comparing to last round.



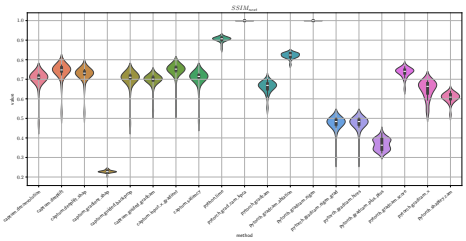
(c) SSIM comparing to last round and grouped by round.



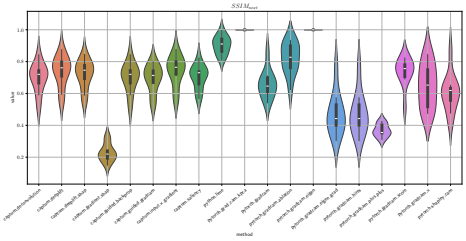
(d) SSIM comparing to last round and grouped by class.



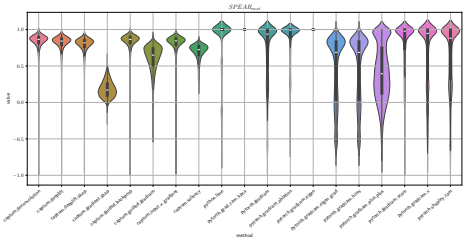
(e) SSIM comparing to next round.



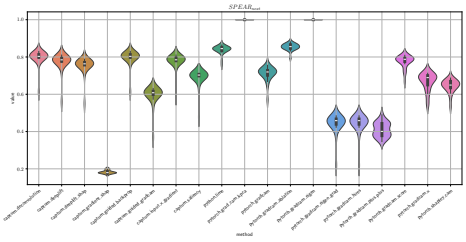
(f) SSIM comparing to next round and grouped by round.



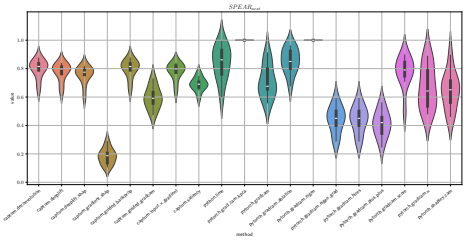
(g) SSIM comparing to next round and grouped by class.



(h) SPEAR comparing to next round.

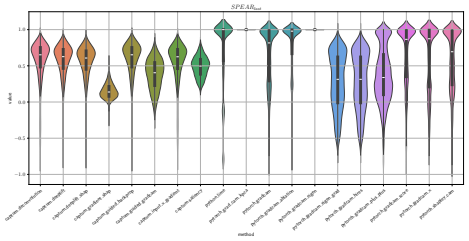


(i) SPEAR comparing to next round and grouped by round.

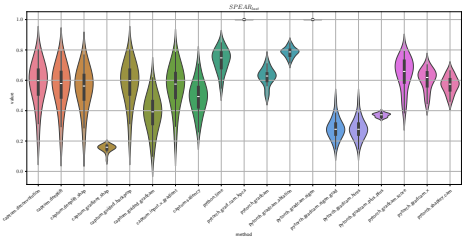


(j) SPEAR comparing to next round and grouped by class.

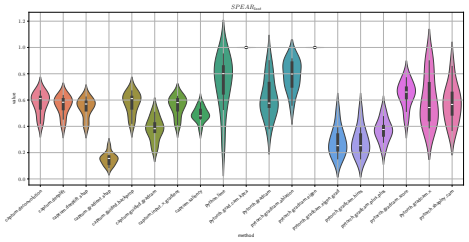
Figure A.11.: Measuring Stability FedAvg/Square (1).



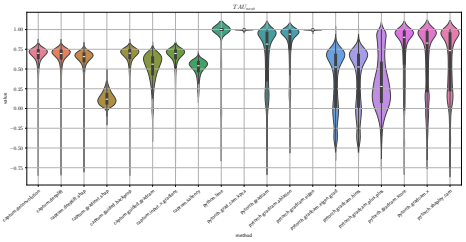
(a) SPEAR comparing to last round.



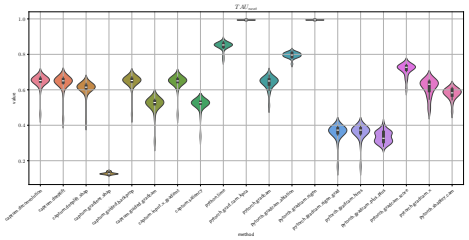
(b) SPEAR comparing to last round and grouped by round.



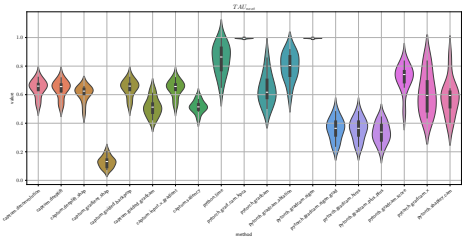
(c) SPEAR comparing to last round and grouped by class.



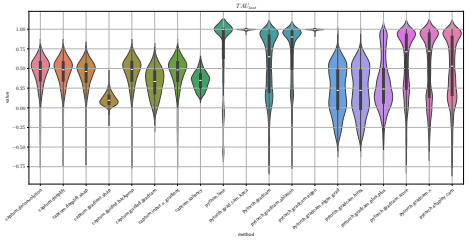
(d) TAU comparing to next round.



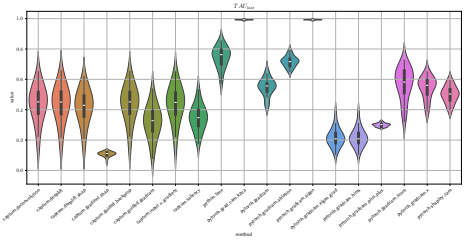
(e) TAU comparing to next round and grouped by round.



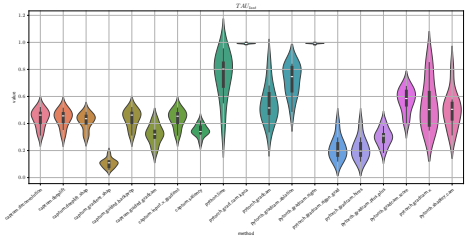
(f) TAU comparing to next round and grouped by class.



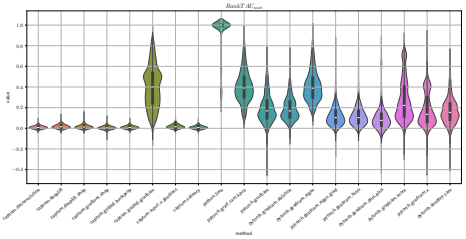
(g) TAU comparing to last round.



(h) TAU comparing to last round and grouped by round.

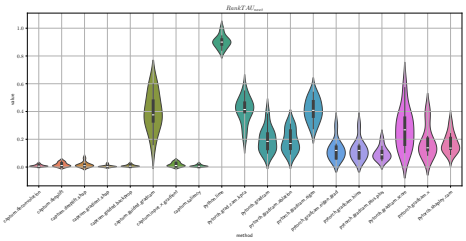


(i) TAU comparing to last round and grouped by class.

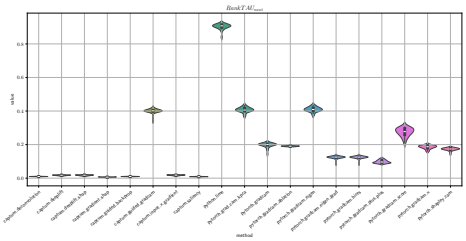


(j) RankTAU comparing to next round.

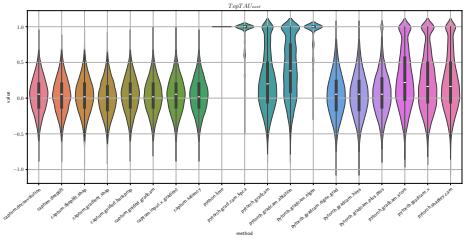
Figure A.12.: Measuring Stability FedAvg/Square (2).



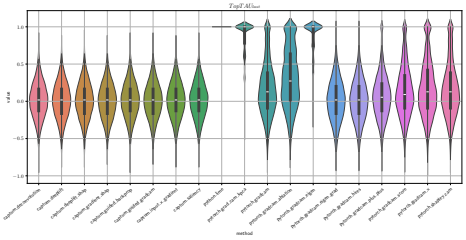
(a) RankTAU comparing to next round and grouped by class.



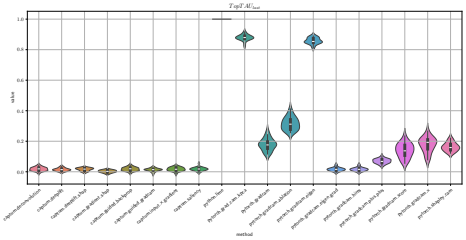
(b) RankTAU comparing to next round and grouped by round.



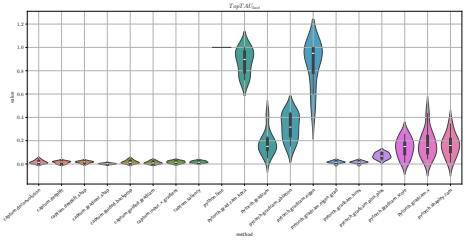
(c) TopTAU comparing to next round.



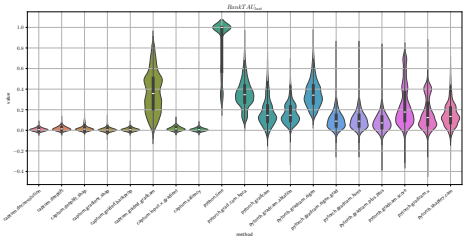
(d) TopTAU comparing to last round.



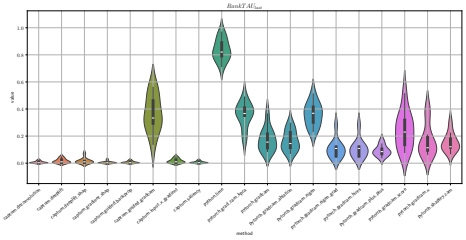
(e) TopTAU comparing to last round.



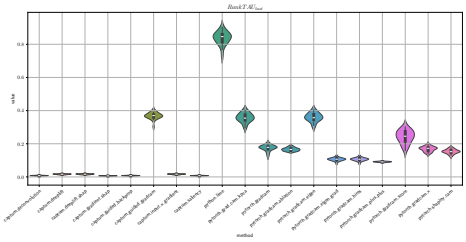
(f) TopTAU comparing to last round and grouped by class.



(g) RankTAU comparing to last round.

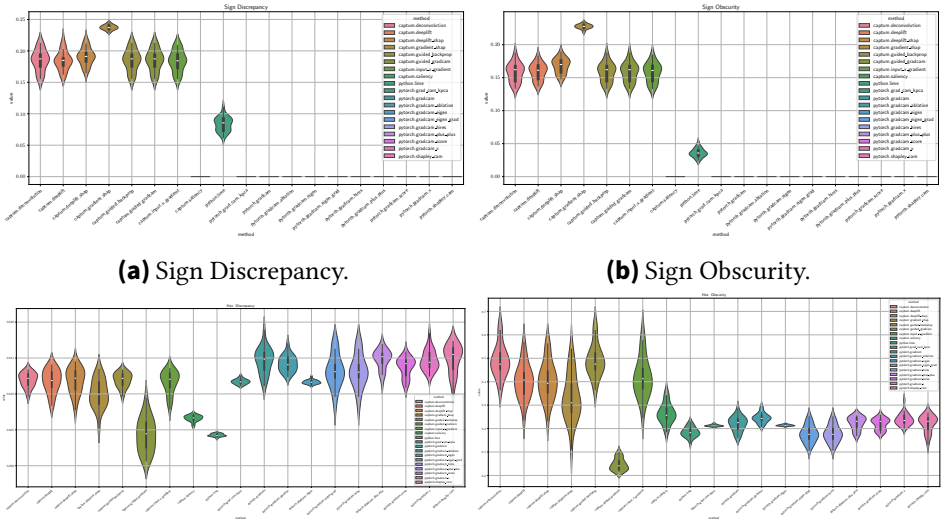


(h) RankTAU comparing to last round and grouped by class.



(i) RankTAU comparing to last round and grouped by round.

Figure A.13.: Measuring Stability FedAvg/Square (3).



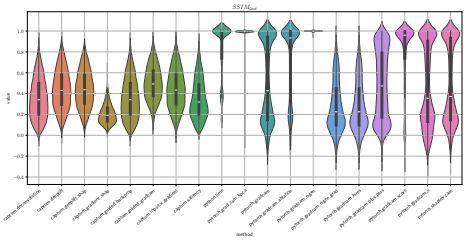
(a) Sign Discrepancy.

(b) Sign Obscurity.

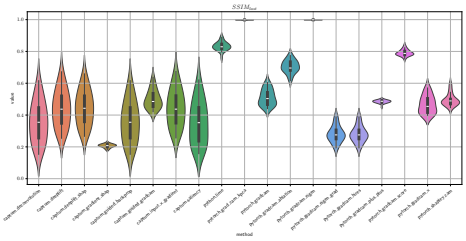
(c) Histogram Discrepancy.

(d) Histogram Obscurity.

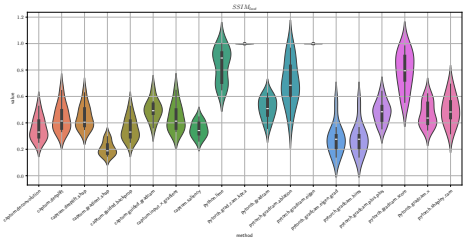
Figure A.14.: Measuring Rashomon Effect FedAvg/Square.



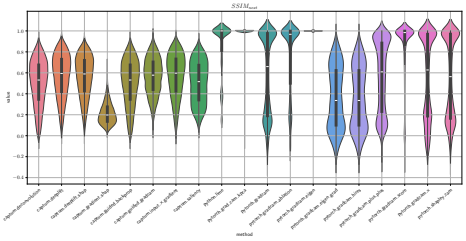
(a) SSIM comparing to last round.



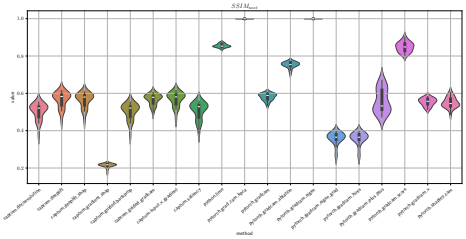
(b) SSIM comparing to last round and grouped by round.



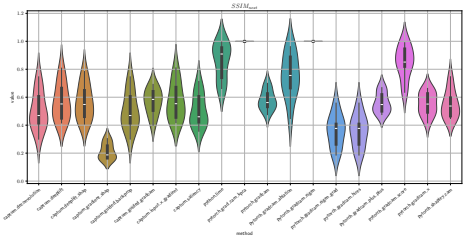
(c) SSIM comparing to last round and grouped by class.



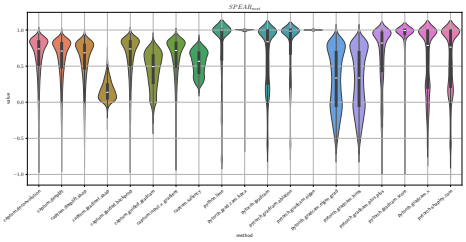
(d) SSIM comparing to next round.



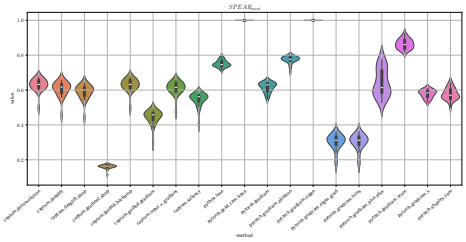
(e) SSIM comparing to next round and grouped by round.



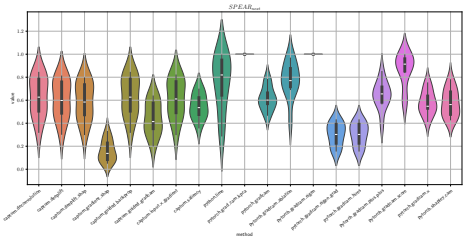
(f) SSIM comparing to next round and grouped by class.



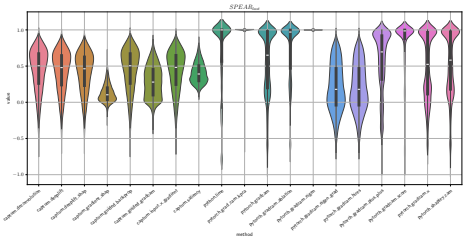
(g) SPEAR comparing to next round.



(h) SPEAR comparing to next round and grouped by round.

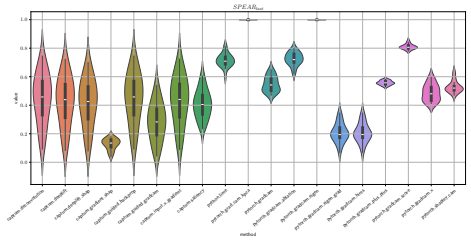


(i) SPEAR comparing to next round and grouped by class.

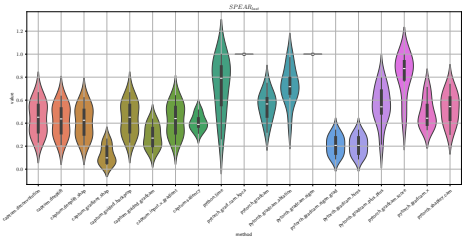


(j) SPEAR comparing to last round.

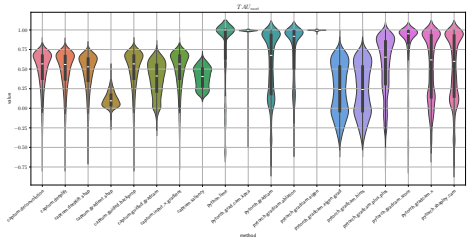
Figure A.15.: Measuring Stability FedAvg/Dirichlet (1).



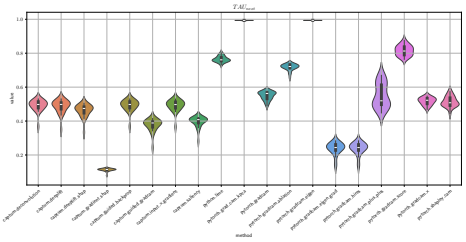
(a) SPEAR comparing to last round and grouped by round.



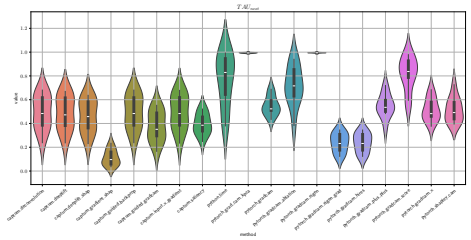
(b) SPEAR comparing to last round and grouped by class.



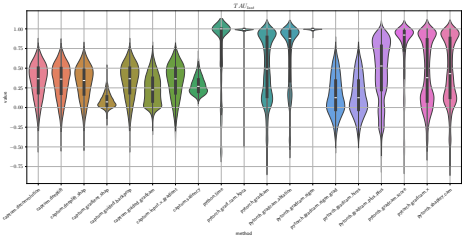
(c) TAU comparing to next round.



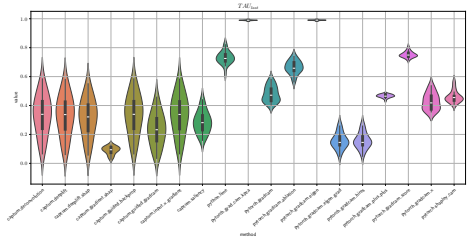
(d) TAU comparing to next round and grouped by round.



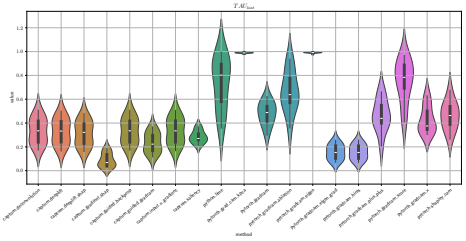
(e) TAU comparing to next round and grouped by class.



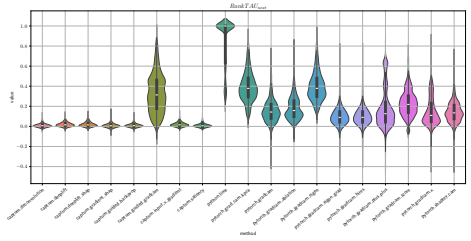
(f) TAU comparing to last round.



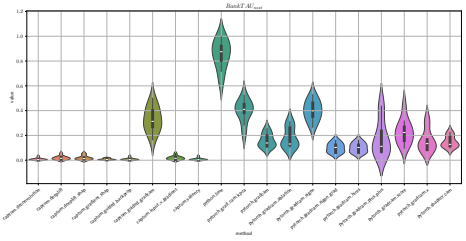
(g) TAU comparing to last round and grouped by round.



(h) TAU comparing to last round and grouped by class.

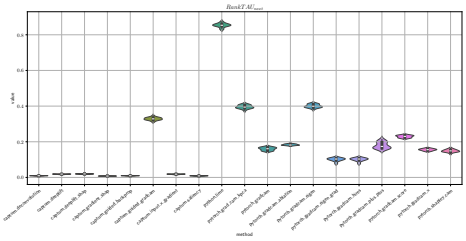


(i) RankTAU comparing to next round.

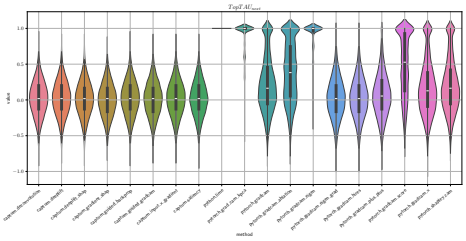


(j) RankTAU comparing to next round and grouped by class.

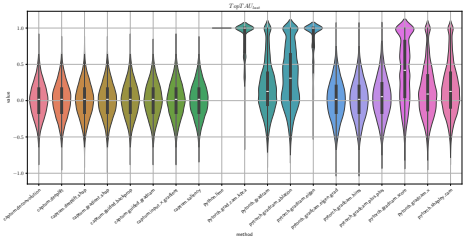
Figure A.16.: Measuring Stability FedAvg/Dirichlet (2).



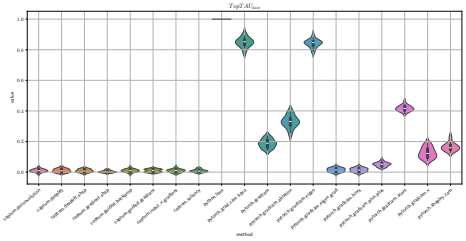
(a) RankTAU comparing to next round and grouped by round.



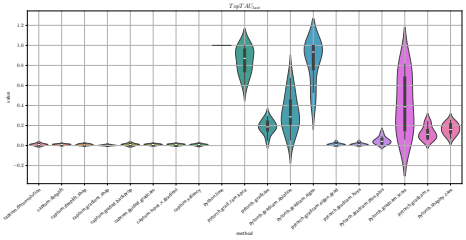
(b) TopTAU comparing to next round.



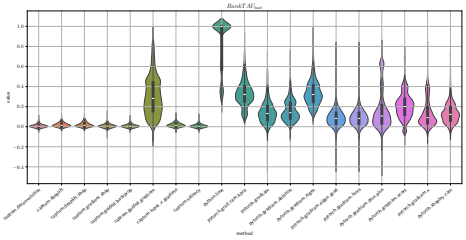
(c) TopTAU comparing to last round.



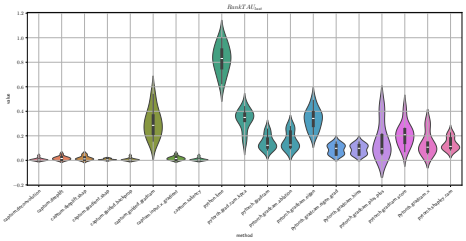
(d) TopTAU comparing to last round.



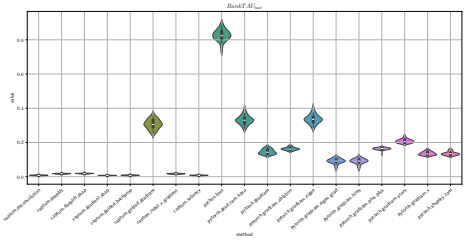
(e) TopTAU comparing to last round and grouped by class.



(f) RankTAU comparing to last round.

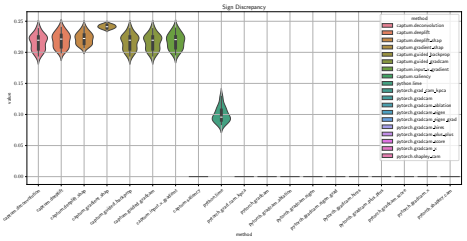


(g) RankTAU comparing to last round and grouped by class.

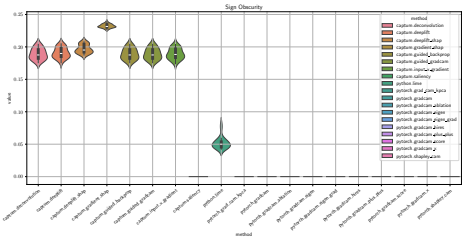


(h) RankTAU comparing to last round and grouped by round.

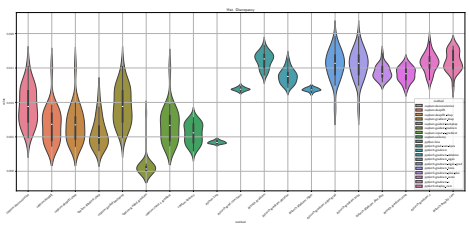
Figure A.17.: Measuring Stability FedAvg/Dirichlet (3).



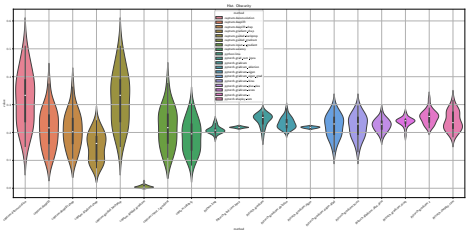
(a) Sign Discrepancy.



(b) Sign Obscurity.

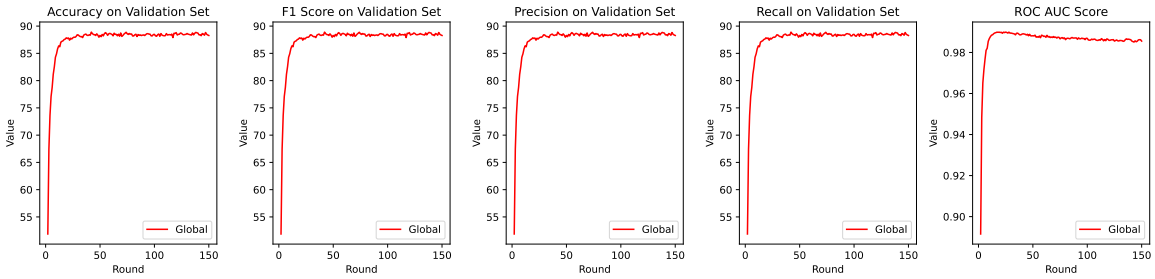


(c) Histogram Discrepancy.

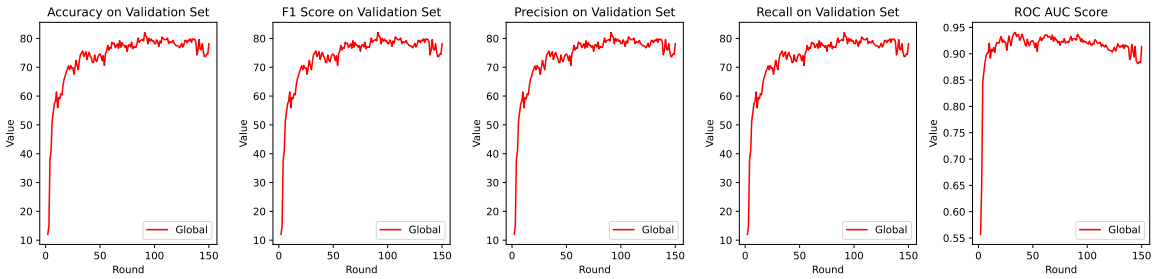


(d) Histogram Obscurity.

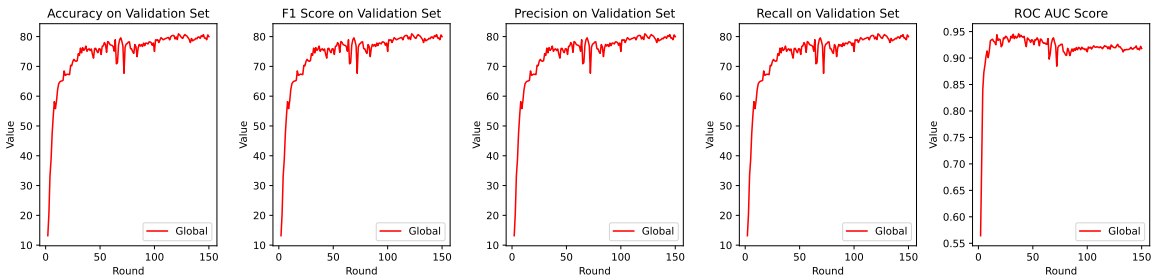
Figure A.18.: Measuring Rashomon Effect FedAvg/Dirichlet.



(a) No DP.



(b) With Server-side fixed clipping.



(c) With Client-side fixed clipping.

Figure A.19.: Results for Experiment 4: 150 FL Rounds.

Survey

Fragebogen

1 About You

Which gender are you?

I don't want to answer
Female
Male
Diverse

How old are you?

What is the highest academic degree that you have?

I don't want to answer
None
Bachelor
Master
Ph.D.

How knowledgeable are you in software engineering?

	None	Poorly	Fairly	Good	Very Good	Excellent	Exceptional
General Knowledge	○	○	○	○	○	○	○
Programming	○	○	○	○	○	○	○
Software Architecture and Design	○	○	○	○	○	○	○

How knowledgeable are you in AI and Explainable AI (XAI)?

	None	Poorly	Fairly	Good	Very Good	Excellent	Exceptional
Familiarity with AI (especially Machine Learning (ML))	○	○	○	○	○	○	○
Usage of AI (especially ML) in software	○	○	○	○	○	○	○
Developing software that includes AI (especially ML)	○	○	○	○	○	○	○
Familiarity with Explainable AI (XAI)	○	○	○	○	○	○	○
Usage of XAI methods	○	○	○	○	○	○	○

How many years of software engineering experience do you have?

2 Explainability explanation

(Please read before continuing)

What is explainability as a non-functional quality?

There is no definitive, agreed-upon definition of explainability as a non-functional quality. However, the most intuitive way to describe explainability is as:

> Enabling understanding of a particular aspect of a system that needs to be explained.

Examples:

- A car navigation system changes the selected route. The aspect to be explained is why the route has been changed. An explanation could be that a better route has been found (in terms of fuel consumption, etc.).
- An image classification tool classifies an image in a certain way. The aspect to be explained is why it is classified this way. One explanation could be that certain parts of the images contributed to higher activation scores, so the proposed classification was selected.

Attention: If you don't know the answer to any of the questions in the context of software engineering, think about an explanation someone gives you orally.
There is no right or wrong answer.

3 Explainability Basics

Give your opinions on explainability in software engineering

	Completely Disagree	Mostly Disagree	Slightly Disagree	Undecided	Slightly Agree	Mostly Agree	Completely Agree
I have thought about explainability as a non-functional quality in software engineering before (not only XAI methods!)	○	○	○	○	○	○	○
I think software should (in general) be made more explainable	○	○	○	○	○	○	○
I think software that I daily use should be more explainable	○	○	○	○	○	○	○
I wish software that uses AI is more explainable	○	○	○	○	○	○	○
I have prejudices against AI because of their lack of explainability	○	○	○	○	○	○	○

	Completely Disagree	Mostly Disagree	Slightly Disagree	Undecided	Slightly Agree	Mostly Agree	Completely Agree
Even if software (without ML involved) would provide to me an explanation I would still be sceptical	○	○	○	○	○	○	○
Even if software (with ML involved) would provide to me an explanation I would still be sceptical	○	○	○	○	○	○	○
I recognize the need for explainability as a non-functional quality in software engineering	○	○	○	○	○	○	○
I think explainability becomes more important in conjunction with AI (especially ML)	○	○	○	○	○	○	○
I have a good understanding of how AI works	○	○	○	○	○	○	○

	Completely Disagree	Mostly Disagree	Slightly Disagree	Undecided	Slightly Agree	Mostly Agree	Completely Agree
I have a good understanding of how AI reasons	○	○	○	○	○	○	○

Regarding an explanation, the following is essential to me:

Please try to be as conscise as possible in your agreement.

	Completely Disagree	Mostly Disagree	Slightly Disagree	Undecided	Slightly Agree	Mostly Agree	Completely Agree
The presentation of an explanation is important to me (e.g., as text, visual, example-based etc.)	○	○	○	○	○	○	○
Understanding the underlying reasoning of an explanation is important to me	○	○	○	○	○	○	○
The ability to further question or judge an explanation is important to me (e.g., if I don't understand the explanation)	○	○	○	○	○	○	○
The ability to introduce feedback about an explanation is important to me (e.g., if the explanation seems redundant, ill presented etc.)	○	○	○	○	○	○	○

Is something else more important to you regarding an explanation? (optional)

Please write it down.

You are running two identical programs with the same inputs and receive an explanation from each program.

Consider the scenario described and give your agreement of the statements below.

	Completely Disagree	Mostly Disagree	Slightly Disagree	Undecided	Slightly Agree	Mostly Agree	Completely Agree
I'm aware that the explanations could be different from each other	○	○	○	○	○	○	○
I accept explanations that are different from each other	○	○	○	○	○	○	○
I accept slightly different explanations (e.g., difference is measured based on some metric)	○	○	○	○	○	○	○
I do not accept different explanations	○	○	○	○	○	○	○
I only accept different explanations if I knew that AI (especially ML) is involved	○	○	○	○	○	○	○
I only accept slightly different explanations (e.g., difference is measured based on some metric) if I knew that AI (especially ML) is involved	○	○	○	○	○	○	○

Which statement do you agree with the most?

Please give your opinions below.

- ○ I only accept a provable "true" explanation as a valid explanation
- ○ I can accept the most likely explanation as a valid explanation
- ○ I would accept an explanation based on my own preferences
- ○ Other (Please write down)

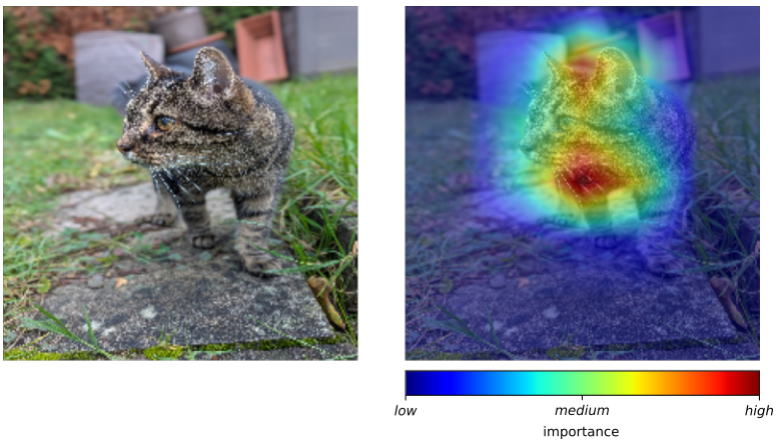
4 Explainability Optimization

Task Introduction

A common way of visualizing what a machine learning model "sees" is via a heatmap.

For example, we have the following input image (left) and a heatmap (right).

Please answer the following questions regarding these two images.



Do you agree with these statements?

	Completely Disagree	Mostly Disagree	Slightly Disagree	Undecided	Slightly Agree	Mostly Agree	Completely Agree
The heatmap does help me understand the reasoning of the AI	○	○	○	○	○	○	○
The heatmap suffices as an explanation to me	○	○	○	○	○	○	○
I like to test multiple examples first before I'm convinced of the explanation	○	○	○	○	○	○	○
This is not an explanation to me	○	○	○	○	○	○	○

Now, additionally the following sentence is provided with the explanation: *"I detected whiskers and pointy ears in the image; therefore, I classified the image as a cat."*

	Completely Disagree	Mostly Disagree	Slightly Disagree	Undecided	Slightly Agree	Mostly Agree	Completely Agree
The heatmap alone suffices to me as an explanation	○	○	○	○	○	○	○
The combination of both heatmap and the addition suffices to me as an explanation	○	○	○	○	○	○	○
The textual addition alone suffices to me as an explanation	○	○	○	○	○	○	○
I'm still sceptical about the explanation	○	○	○	○	○	○	○

Additionally, you are assured that the AI's accuracy is very high, e.g., ~99.8%. Which statement do you agree with the most?

- ○ An explanation seems redundant to me in that case
- ○ I'm still sceptical about the AI
- ○ I would prefer a different kind of explanation
- ○ Reaching an explanation that would satisfy me seems impossible
- ○ None of these (please write into Question 15)

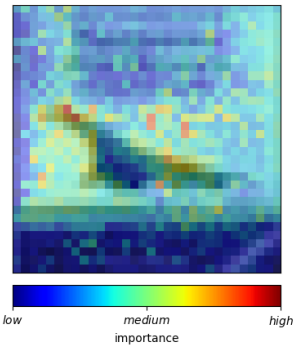
Do you have any other notes or comments you want to make before continuing (optional)?

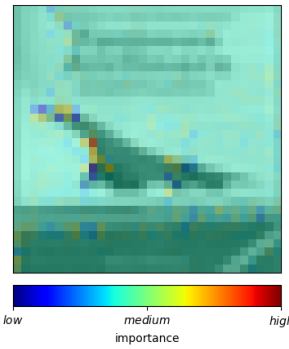
5.1.1.1 User1

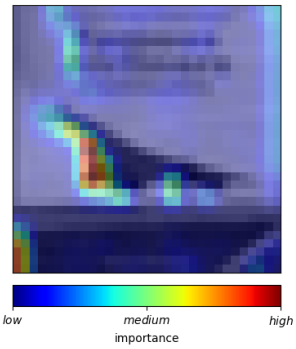
The following picture was classified as "airplane":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.







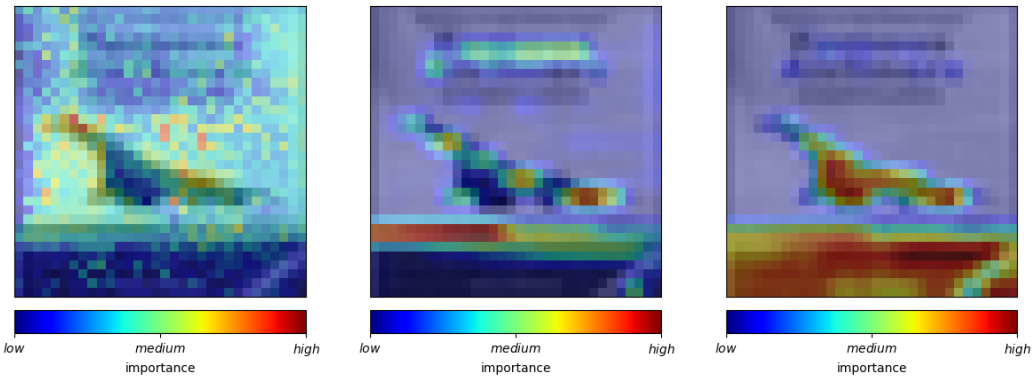
I can't decide

5.1.2.1 User2

The following picture was classified as "airplane":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



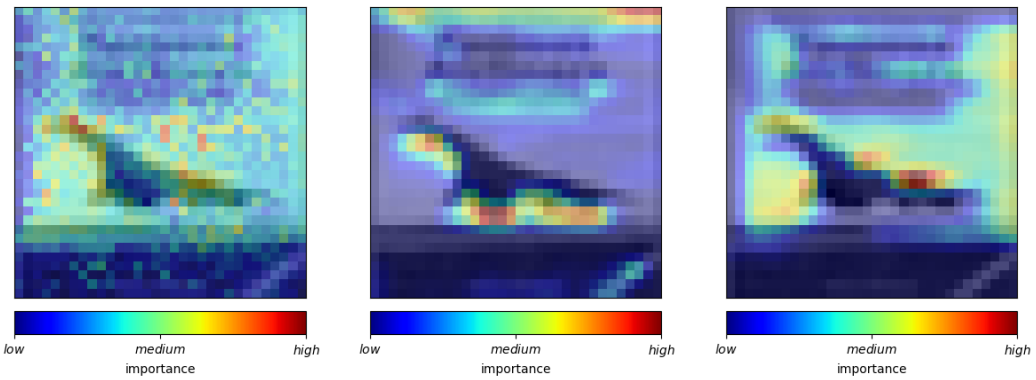
I can't decide

5.1.3.1 User3

The following picture was classified as "airplane":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



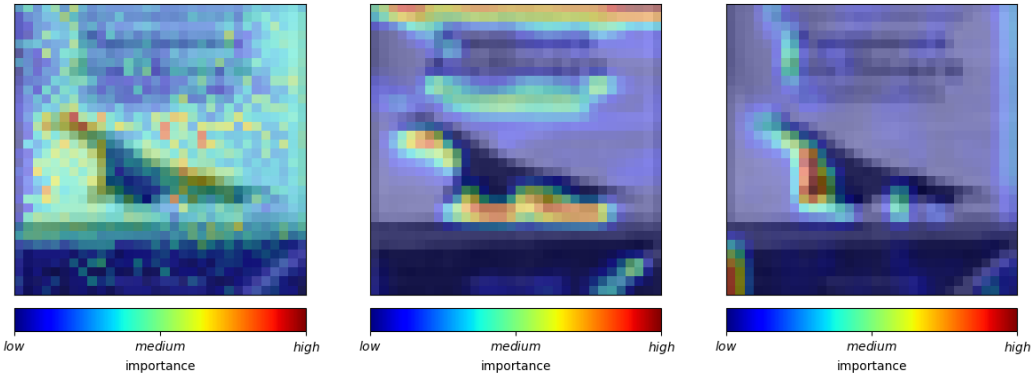
I can't decide

5.1.4.1 User4

The following picture was classified as "airplane":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



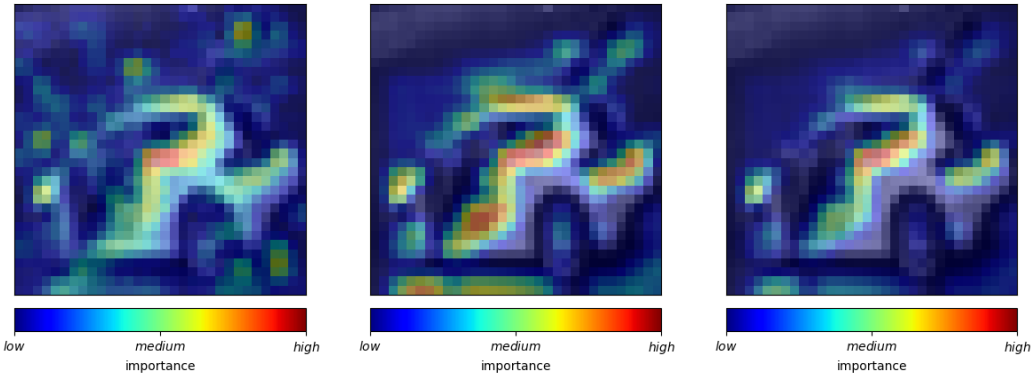
I can't decide

5.2.1.1 User1

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



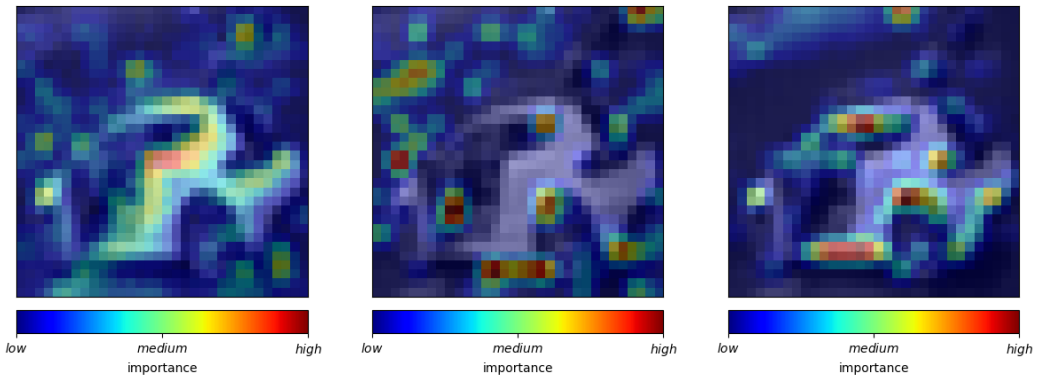
I can't decide

5.2.2.1 User2

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



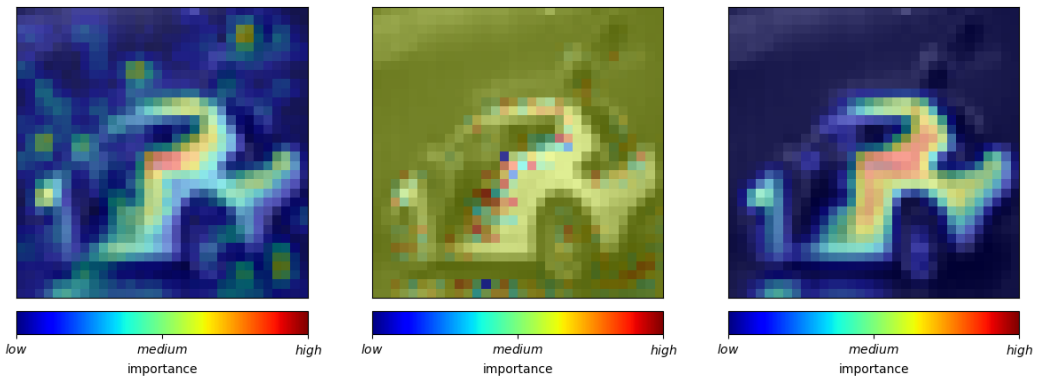
I can't decide

5.2.3.1 User3

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



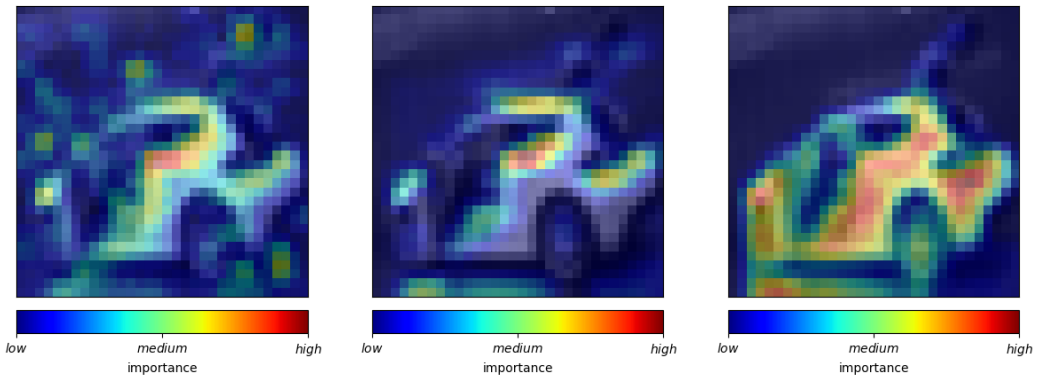
I can't decide

5.2.4.1 User4

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



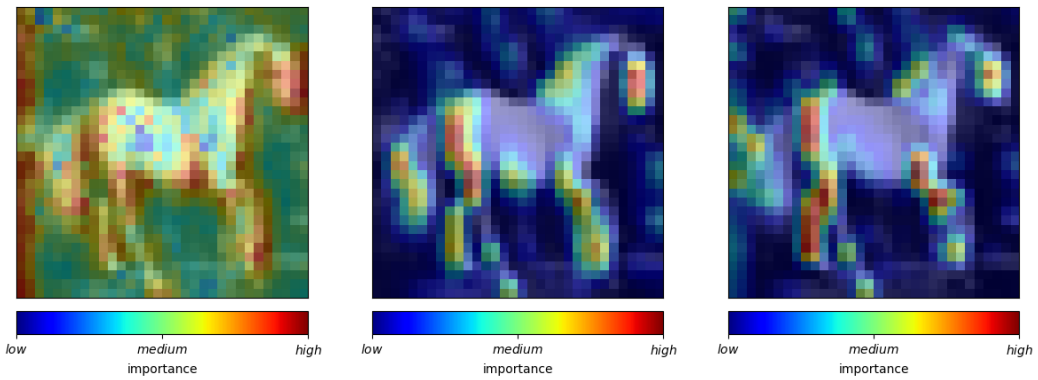
I can't decide

5.3.1.1 User1

The following picture was classified as "horse":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



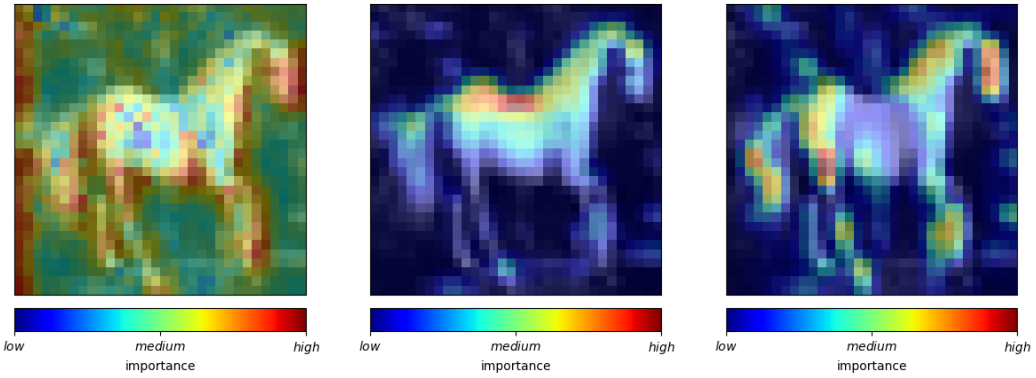
I can't decide

5.3.2.1 User2

The following picture was classified as "horse":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



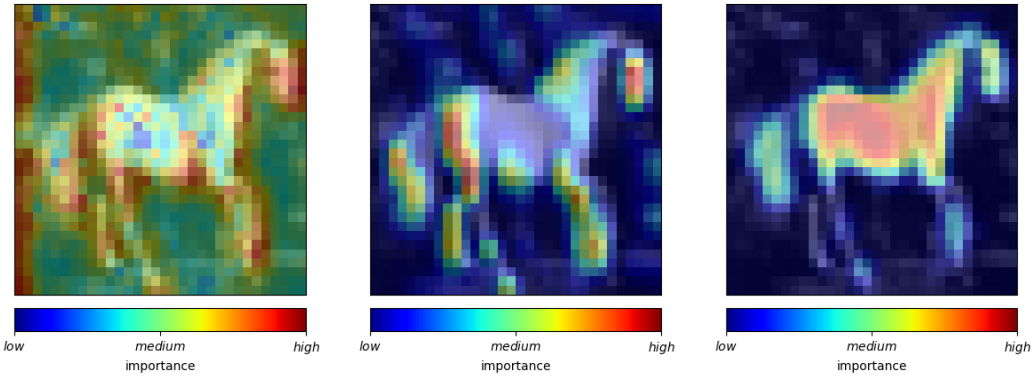
I can't decide

5.3.3.1 User3

The following picture was classified as "horse":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



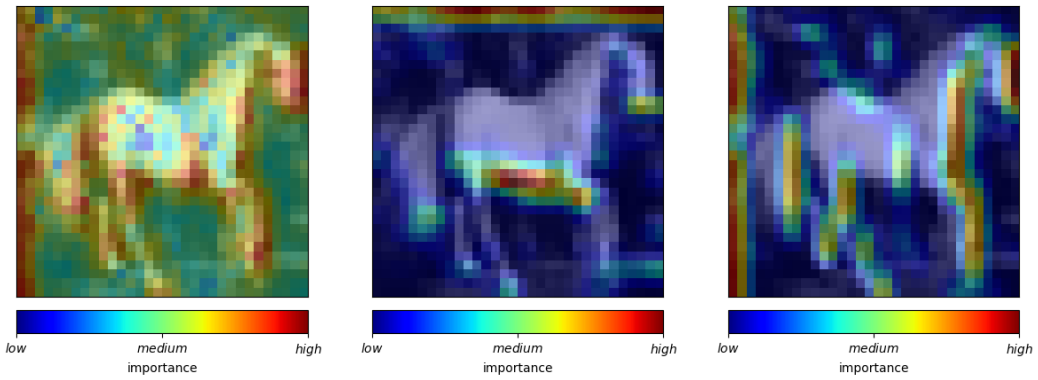
I can't decide

5.3.4.1 User4

The following picture was classified as "horse":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



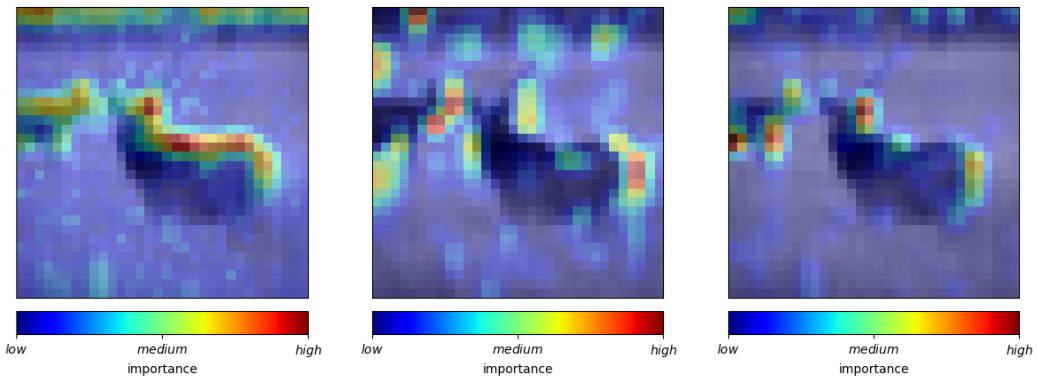
I can't decide

5.4.1.1 User1

The following picture was classified as "deer":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



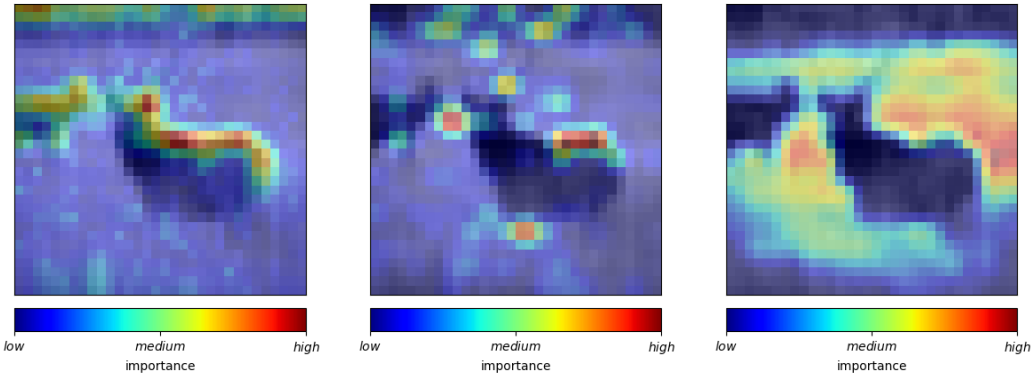
I can't decide

5.4.2.1 User2

The following picture was classified as "deer":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



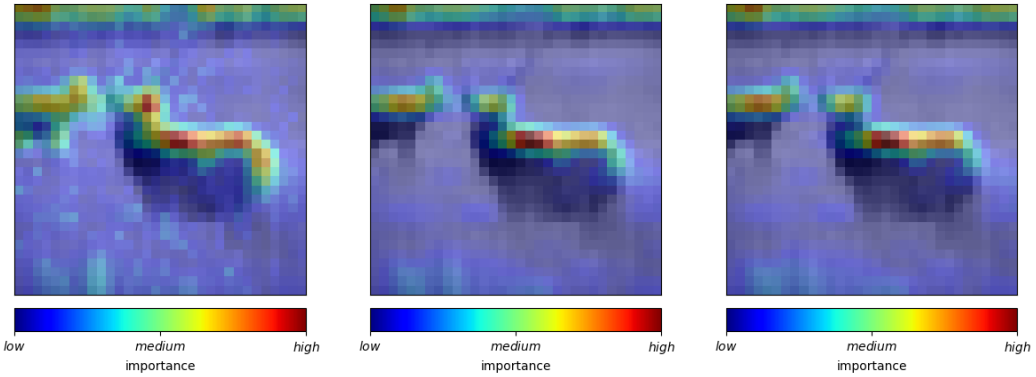
I can't decide

5.4.3.1 User3

The following picture was classified as "deer":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



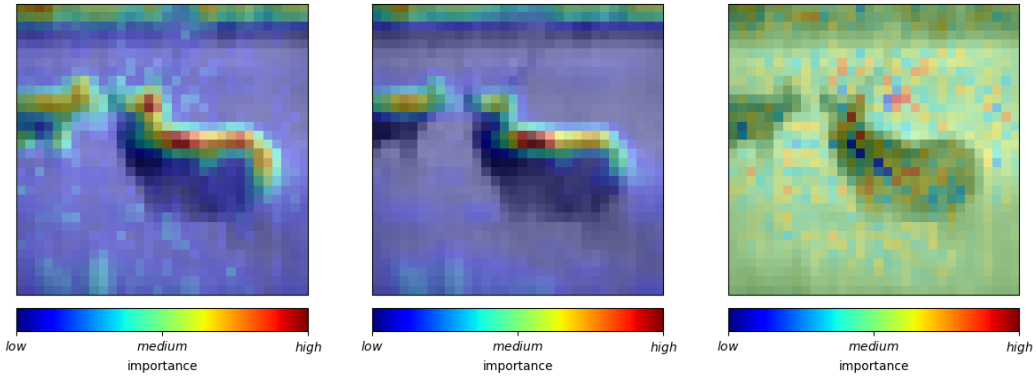
I can't decide

5.4.4.1 User4

The following picture was classified as "deer":

Which explanation of the classification would you find more convincing?

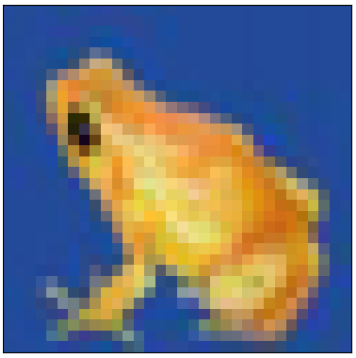
Please select the answer you agree with the most by clicking on it.



I can't decide

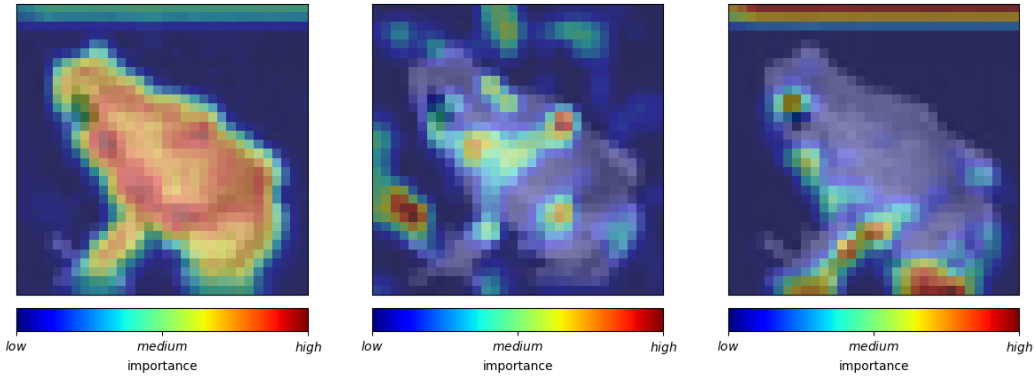
5.5.1.1 User1

The following picture was classified as "frog":



Which explanation of the classification would you find more convincing?

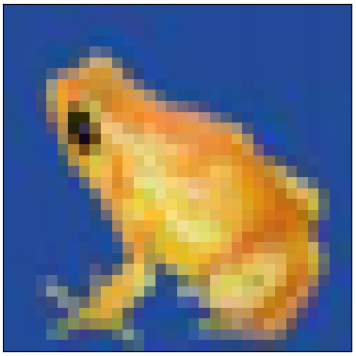
Please select the answer you agree with the most by clicking on it.



I can't decide

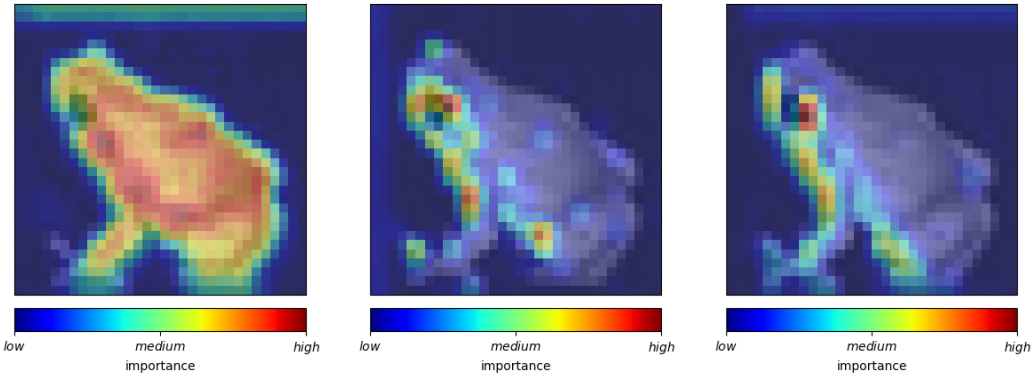
5.5.2.1 User2

The following picture was classified as "frog":



Which explanation of the classification would you find more convincing?

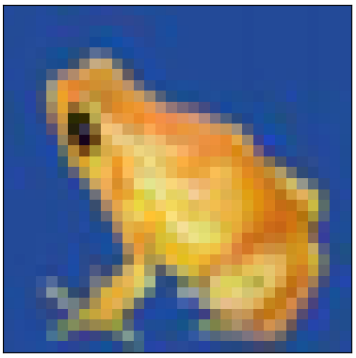
Please select the answer you agree with the most by clicking on it.



I can't decide

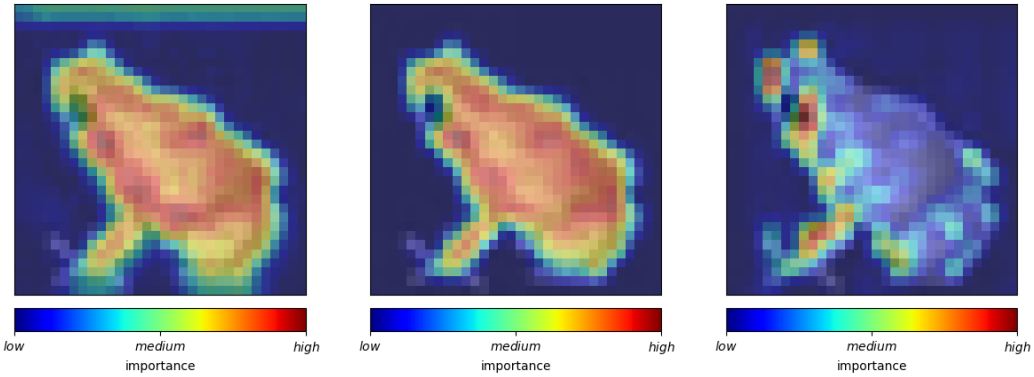
5.5.3.1 User3

The following picture was classified as "frog":



Which explanation of the classification would you find more convincing?

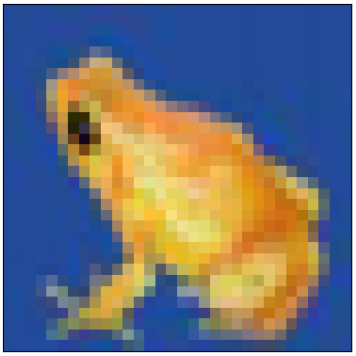
Please select the answer you agree with the most by clicking on it.



I can't decide

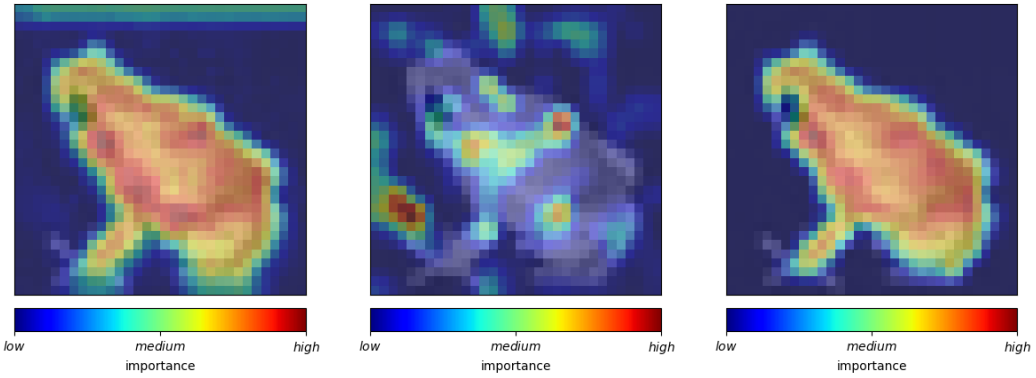
5.5.4.1 User4

The following picture was classified as "frog":



Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it. The classification result is written above each image.



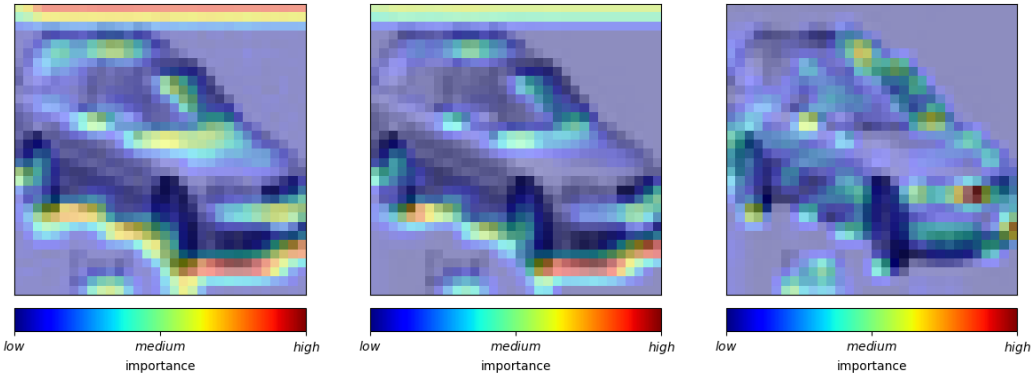
I can't decide

5.6.1.1 User1

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



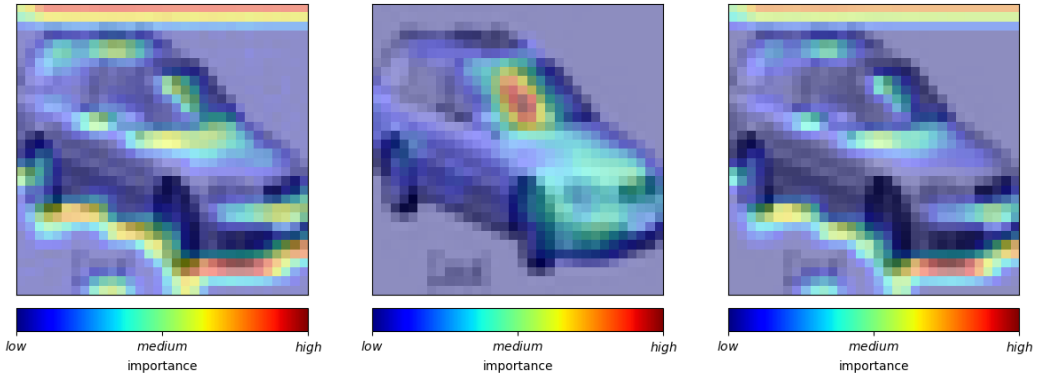
I can't decide

5.6.2.1 User2

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



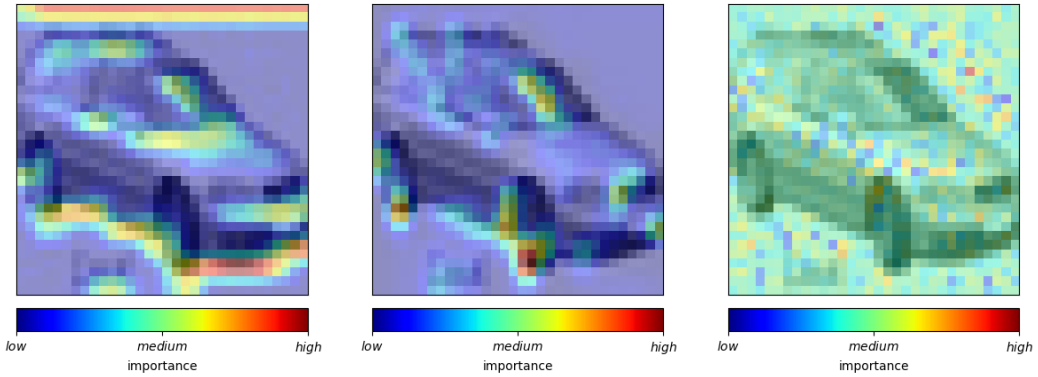
I can't decide

5.6.3.1 User3

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



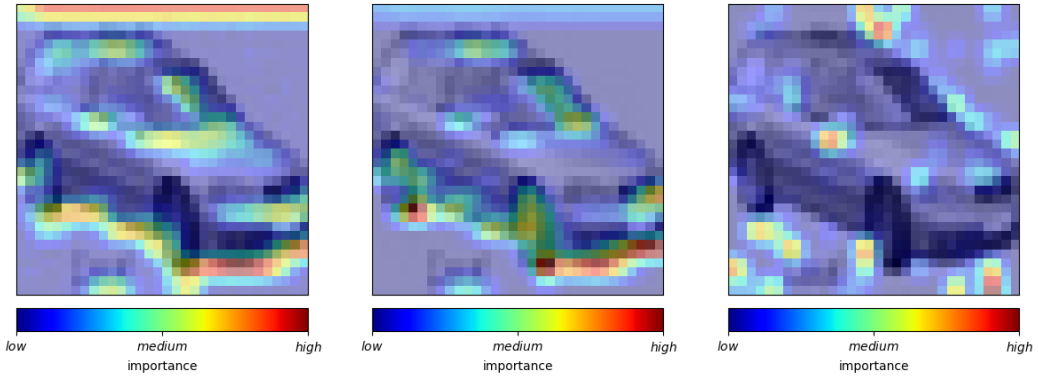
I can't decide

5.6.4.1 User4

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



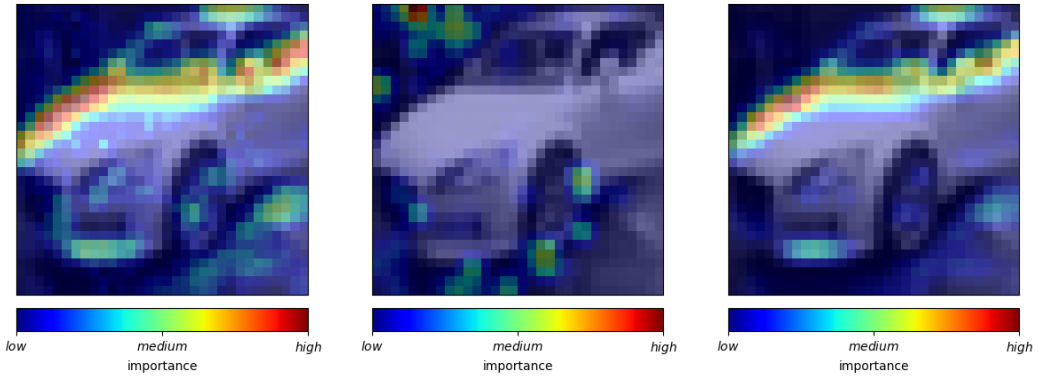
I can't decide

5.7.1.1 User1

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



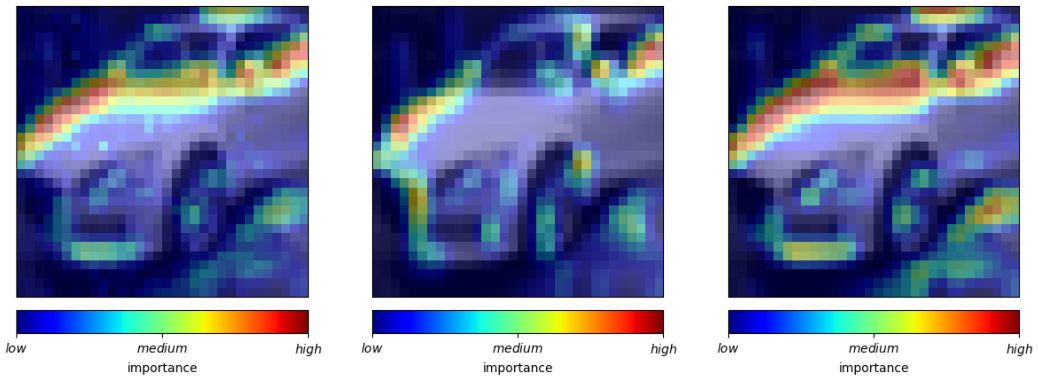
I can't decide

5.7.2.1 User2

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



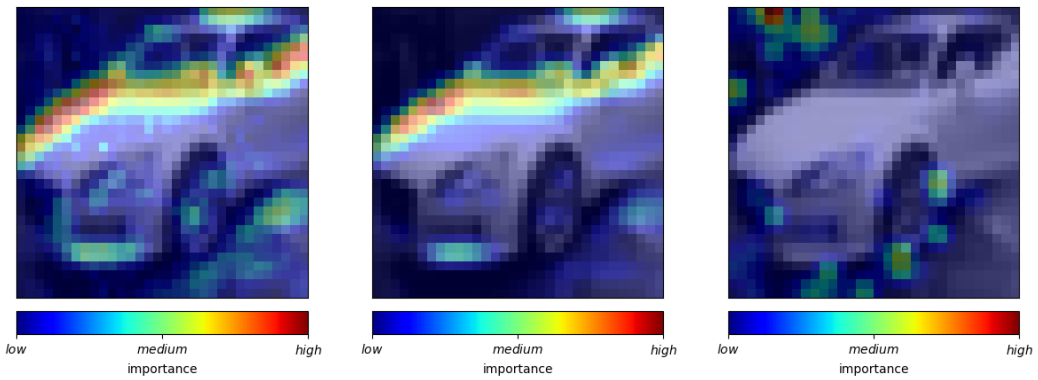
I can't decide

5.7.3.1 User3

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



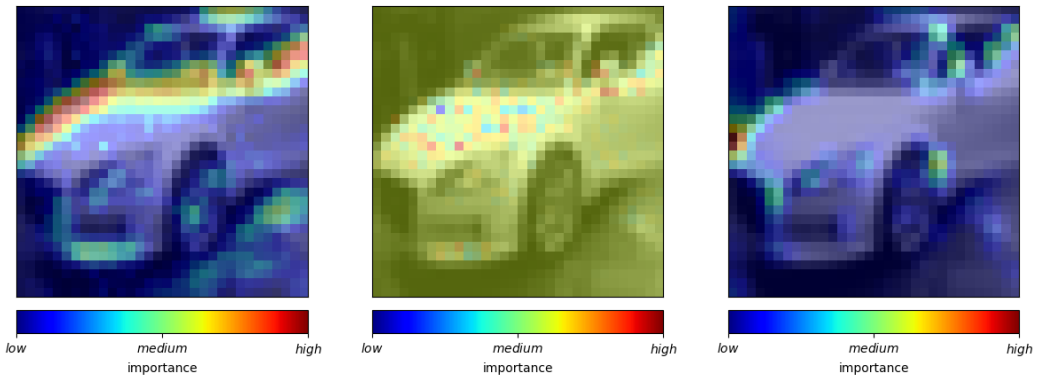
I can't decide

5.7.4.1 User4

The following picture was classified as "car":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



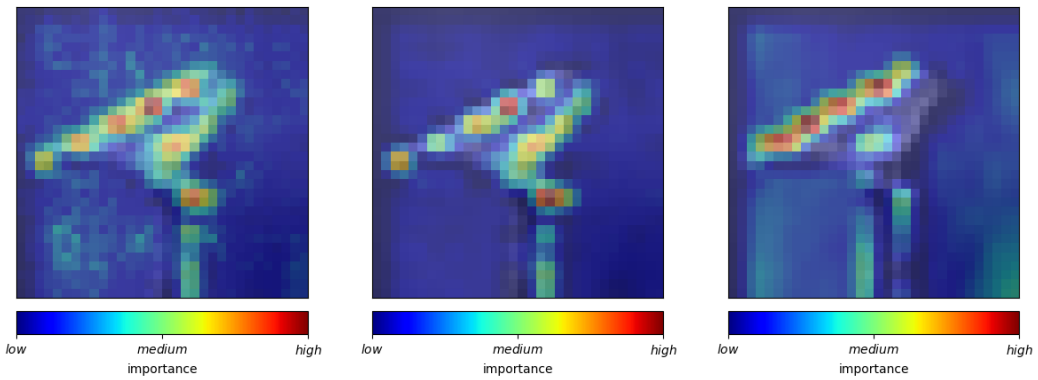
I can't decide

5.8.1.1 User1

The following picture was classified as "bird":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



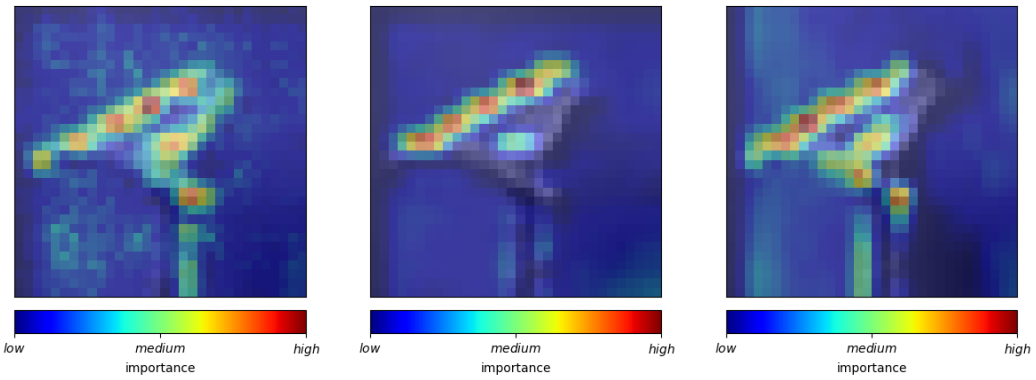
I can't decide

5.8.2.1 User2

The following picture was classified as "bird":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



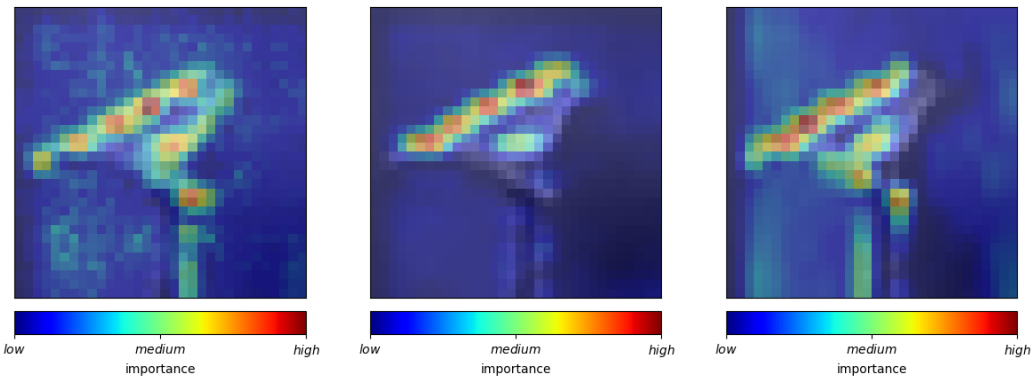
I can't decide

5.8.3.1 User3

The following picture was classified as "bird":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



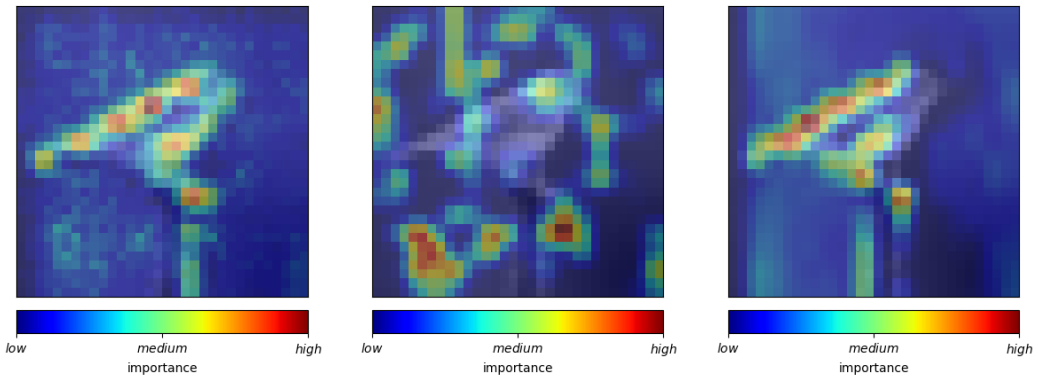
I can't decide

5.8.4.1 User4

The following picture was classified as "bird":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



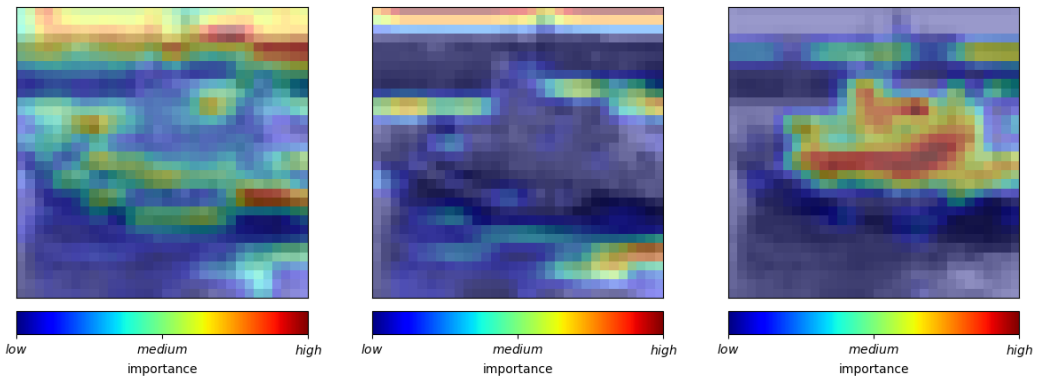
I can't decide

5.9.1.1 User1

The following picture was classified as "boat":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



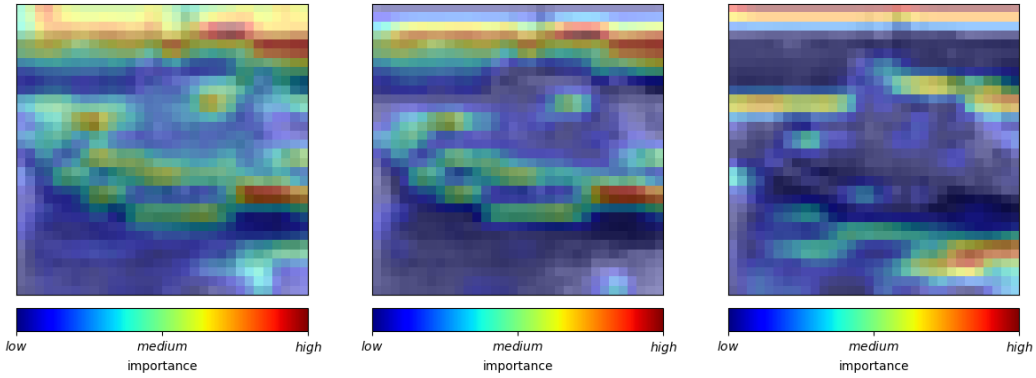
I can't decide

5.9.2.1 User2

The following picture was classified as "boat":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



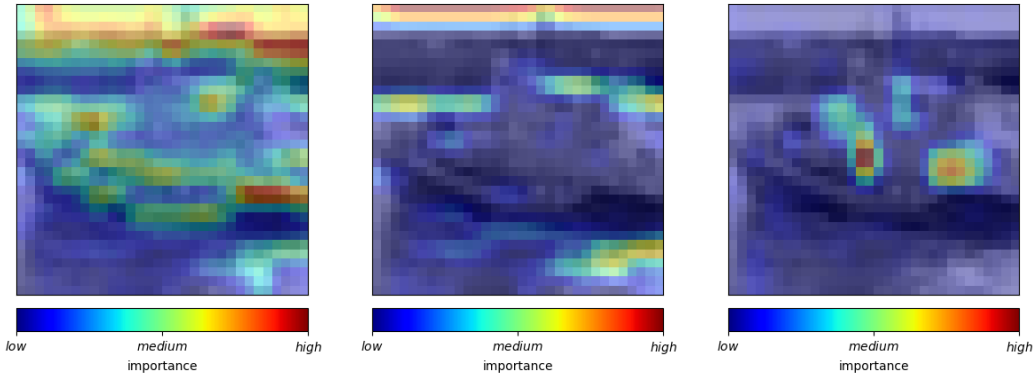
I can't decide

5.9.3.1 User3

The following picture was classified as "boat":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



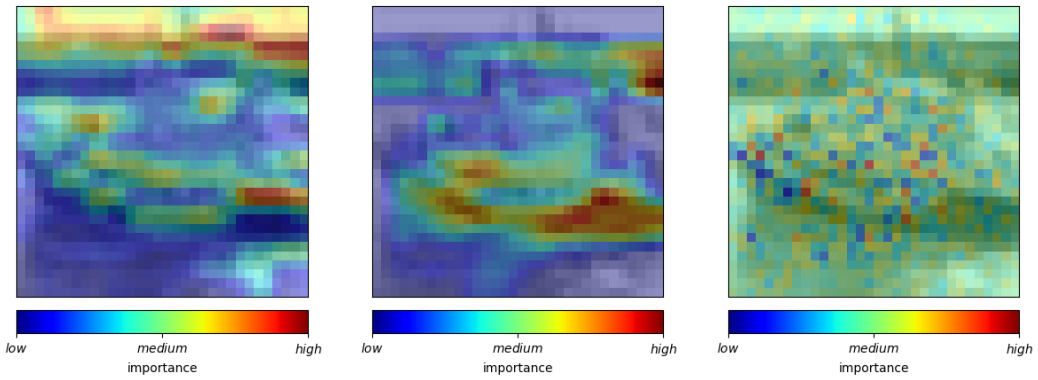
I can't decide

5.9.4.1 User4

The following picture was classified as "boat":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



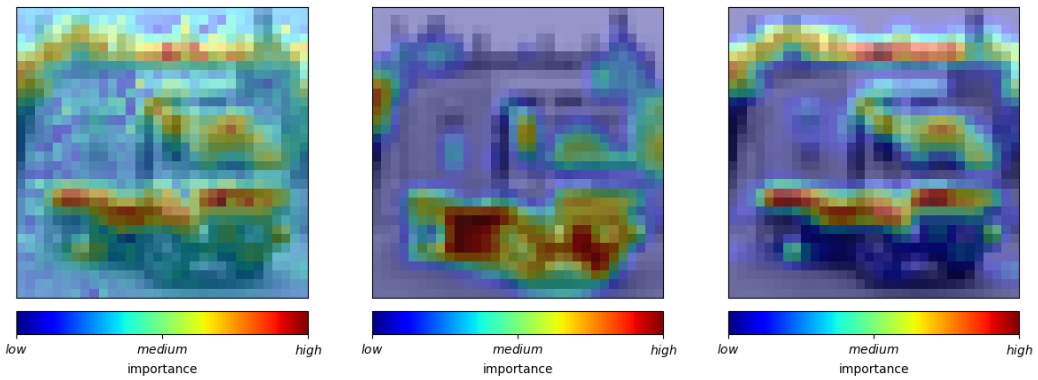
I can't decide

5.10.1.1 User1

The following picture was classified as "truck":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



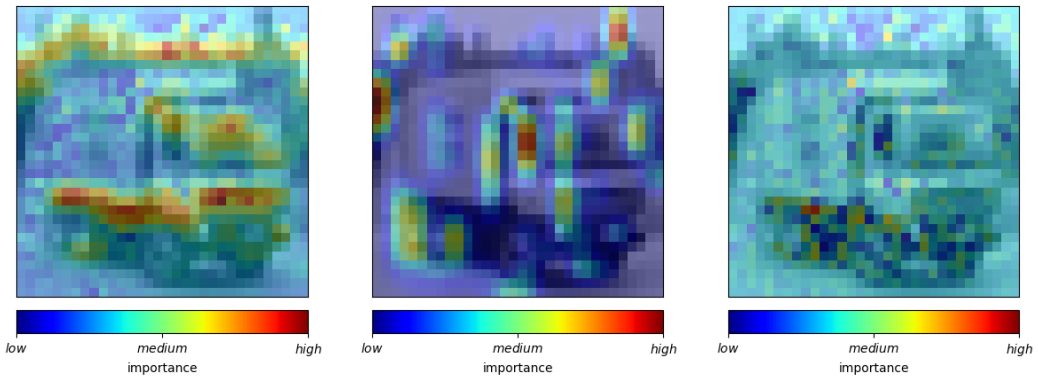
I can't decide

5.10.2.1 User2

The following picture was classified as "truck":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



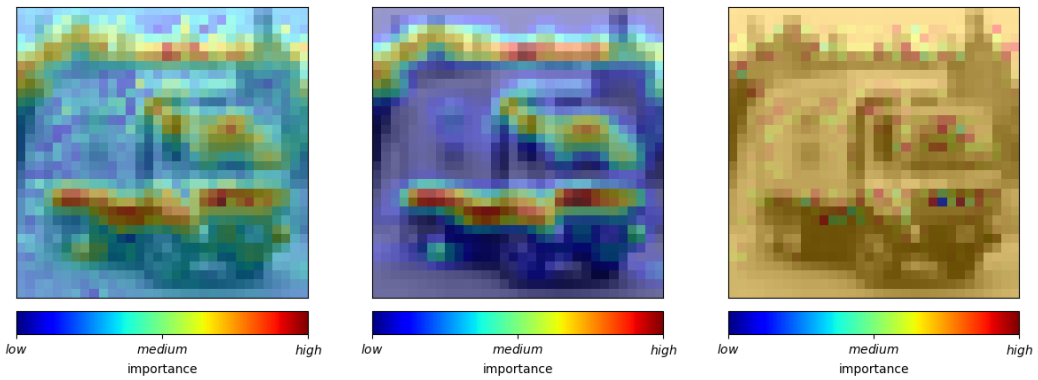
I can't decide

5.10.3.1 User3

The following picture was classified as "truck":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



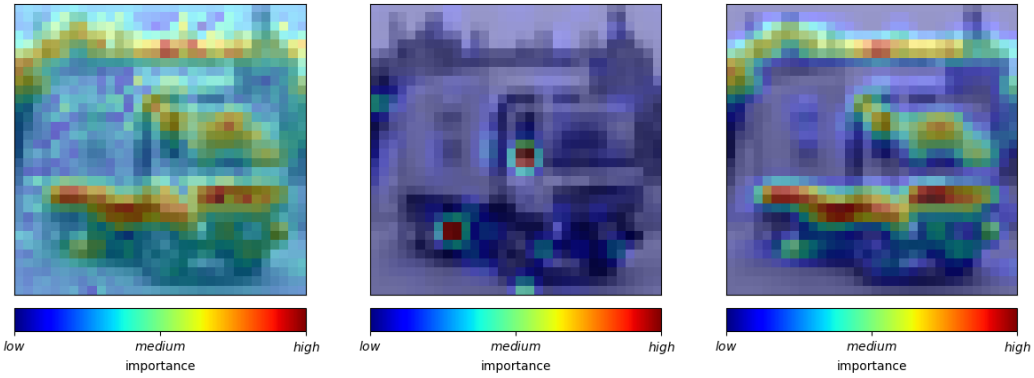
I can't decide

5.10.4.1 User4

The following picture was classified as "truck":

Which explanation of the classification would you find more convincing?

Please select the answer you agree with the most by clicking on it.



I can't decide

6 Commitment

Do you have any other notes or comments you want to make before continuing (optional)?

Please click CONTINUE on the screen to submit your answers. Once submitted, they can not be changed.

Do you want to submit your answers?

7 Endseite

Thank you very much for your participation!

You can close this window now.

If you have any questions regarding the survey, please contact us via nicolas.schuler@student.kit.edu.

Have a nice day!

CLOSE WINDOW