GOETHE UNIVERSITÄT FRANKFURT AM MAIN

AI Tools Lab SS2020

Project Description

Todor Ivanov todor@dbis.cs.uni-frankfurt.de

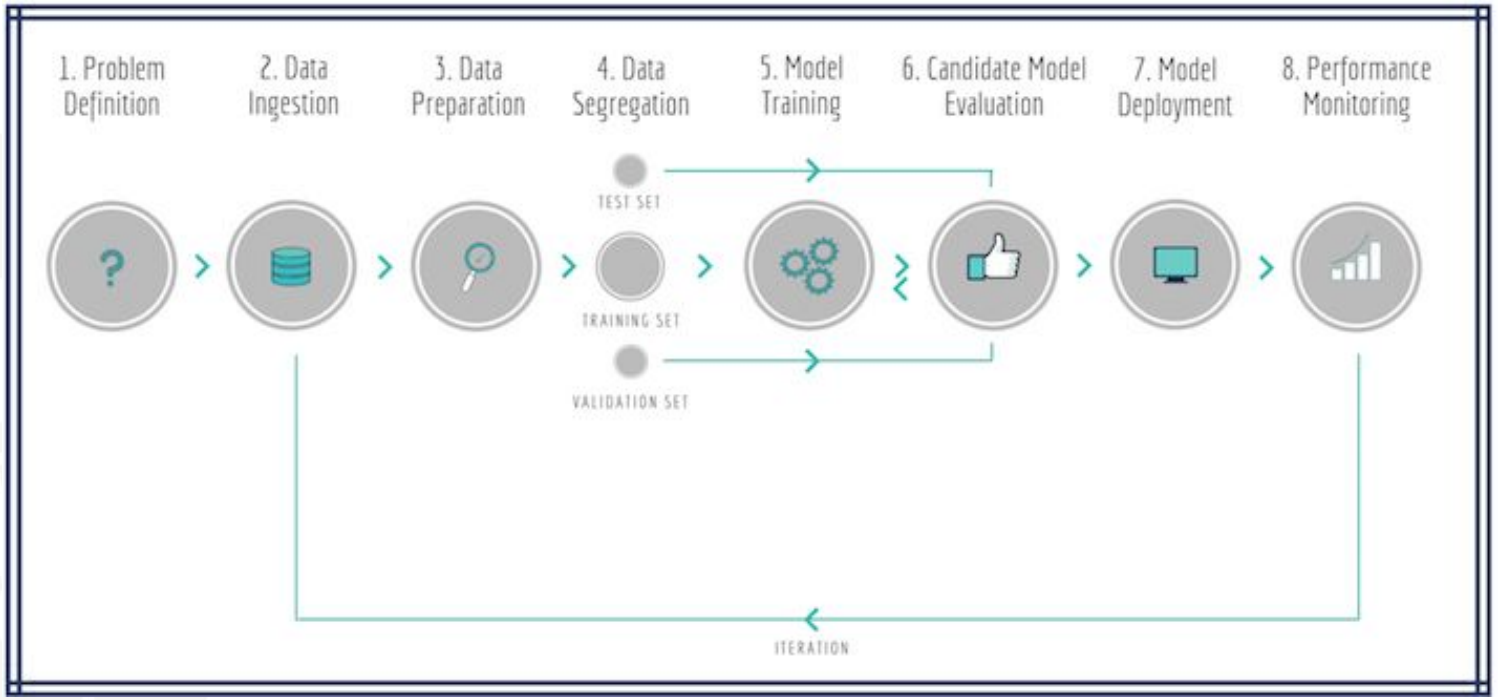
Frankfurt Big Data Lab

GOETHE UNIVERSITÄT

http://www.bigdata.uni-frankfurt.de/

What is Machine Learning?

- **Machine learning is a set of methods that computers use to make and improve predictions or behaviors based on data.**
- **Machine Learning (ML) Pipelines**



source:https://towardsdatascience.com/architecting-a-machine-learning-pipeline-a847f094d1c7

Machine Learning Terminology: https://christophm.github.io/interpretable-ml-book/terminology.html

Interpretability - the “Why?” question

- **Interpretability is the degree to which a human can understand the cause of a decision.**
  - Miller, Tim. “Explanation in artificial intelligence: Insights from the social sciences.” arXiv Preprint arXiv:1706.07269. (2017) - https://arxiv.org/pdf/1706.07269.pdf

- **Interpretability is the degree to which a human can consistently predict the model’s result.**
  - Kim, Been, Rajiv Khanna, and Oluwasanmi O. Koyejo. “Examples are not enough, learn to criticize! Criticism for interpretability.” Advances in Neural Information Processing Systems (2016). - https://people.csail.mit.edu/beenkim/papers/KIM2016NIPS_MMD.pdf

- **The higher the interpretability of a machine learning model, the easier it is for someone to comprehend why certain decisions or predictions have been made.**

- **A model is better interpretable than another model if its decisions are easier for a human to comprehend than decisions from the other model.**

  https://christophm.github.io/interpretable-ml-book/interpretability.html

Interpretability Questions

- How does the algorithm create the model?

  → ***algorithm transparency***

- How does the trained model make predictions?

  → ***global, holistic model interpretability***

- How do parts of the model affect predictions?

  → ***global model interpretability on a modular level***

- Why did the model make a certain prediction for an instance?

  → ***local interpretability for a single prediction***

- Why did the model make specific predictions for a group of instances?

  → ***local interpretability for a group of predictions***

Lipton, Zachary C. “The mythos of model interpretability.” - https://arxiv.org/pdf/1606.03490.pdf

Interpretable Models

- **The easiest way to achieve interpretability is to use only a subset of algorithms that create interpretable models.**

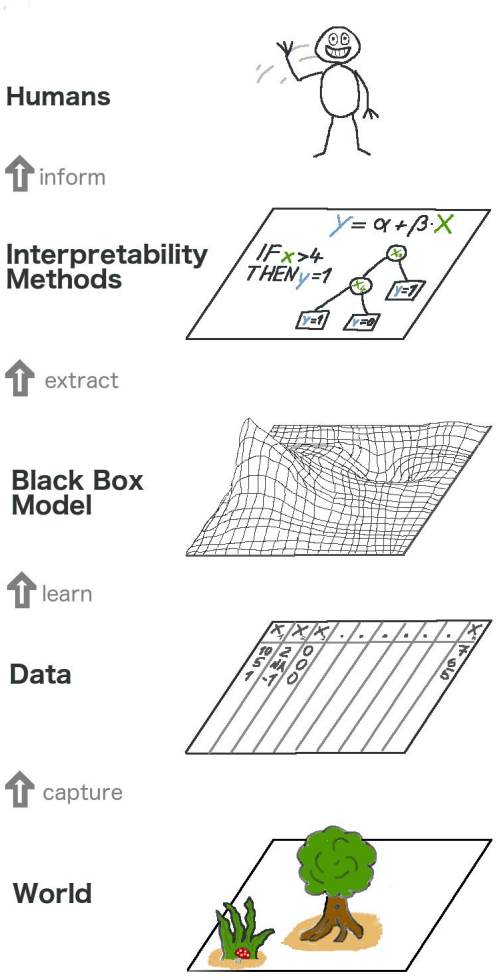
- Examples for such models are:
  - linear regression
  - logistic regression
  - decision tree
  - decision rules

More: https://christophm.github.io/interpretable-ml-book/simple.html

Model-Agnostic Methods

Separating the explanations from the machine learning model (= model-agnostic interpretation methods) has multiple advantages:

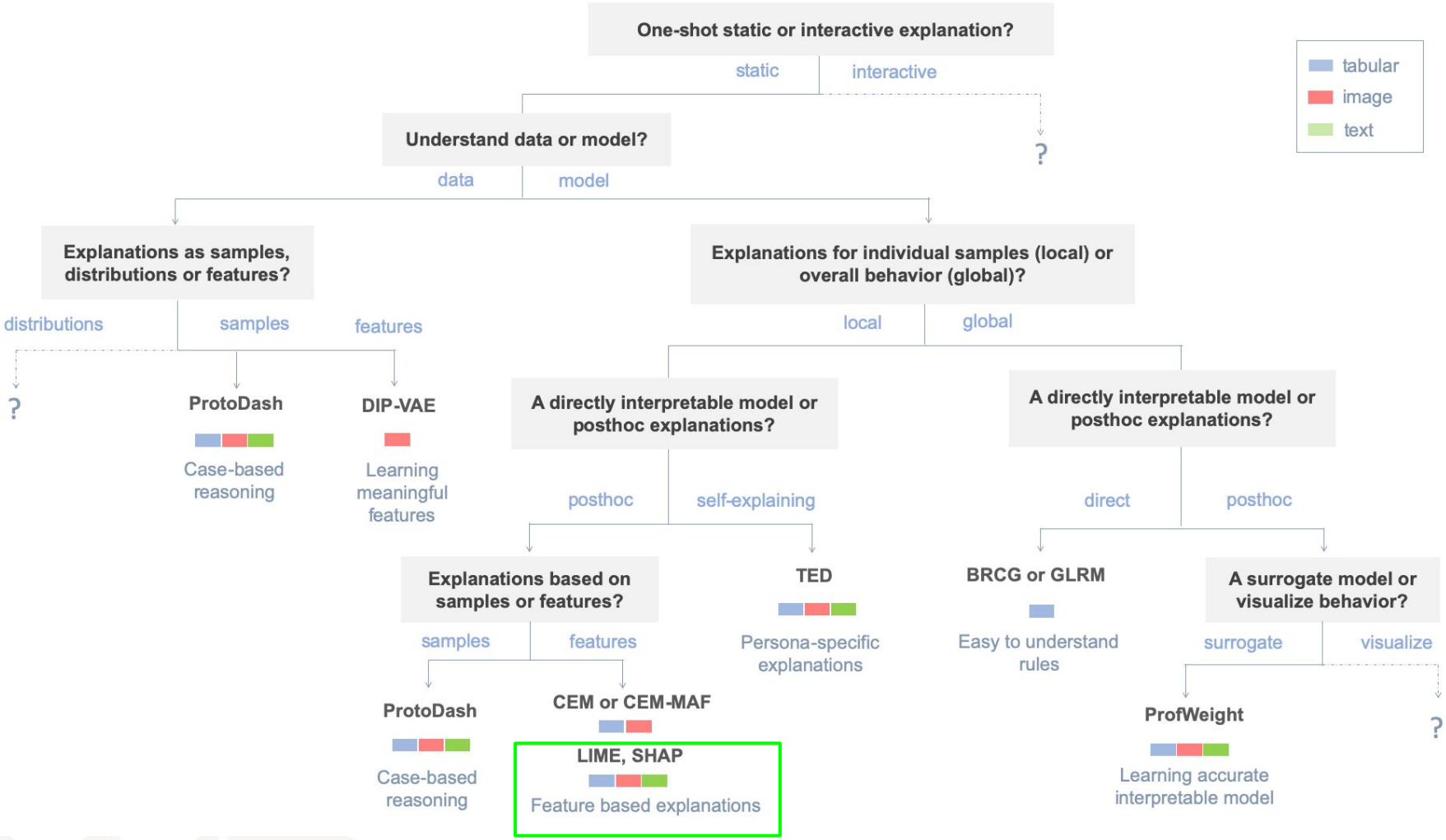
- **Model flexibility:** The interpretation method can work with any machine learning model, such as random forests and deep neural networks.
- **Explanation flexibility:** You are not limited to a certain form of explanation.
- **Representation flexibility:** The explanation system should be able to use a different feature representation as the model being explained.



Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Model-agnostic interpretability of machine learning." ICML Workshop on Human Interpretability in Machine Learning. - https://arxiv.org/pdf/1606.05386.pdf

https://christophm.github.io/interpretable-ml-book/agnostic.html

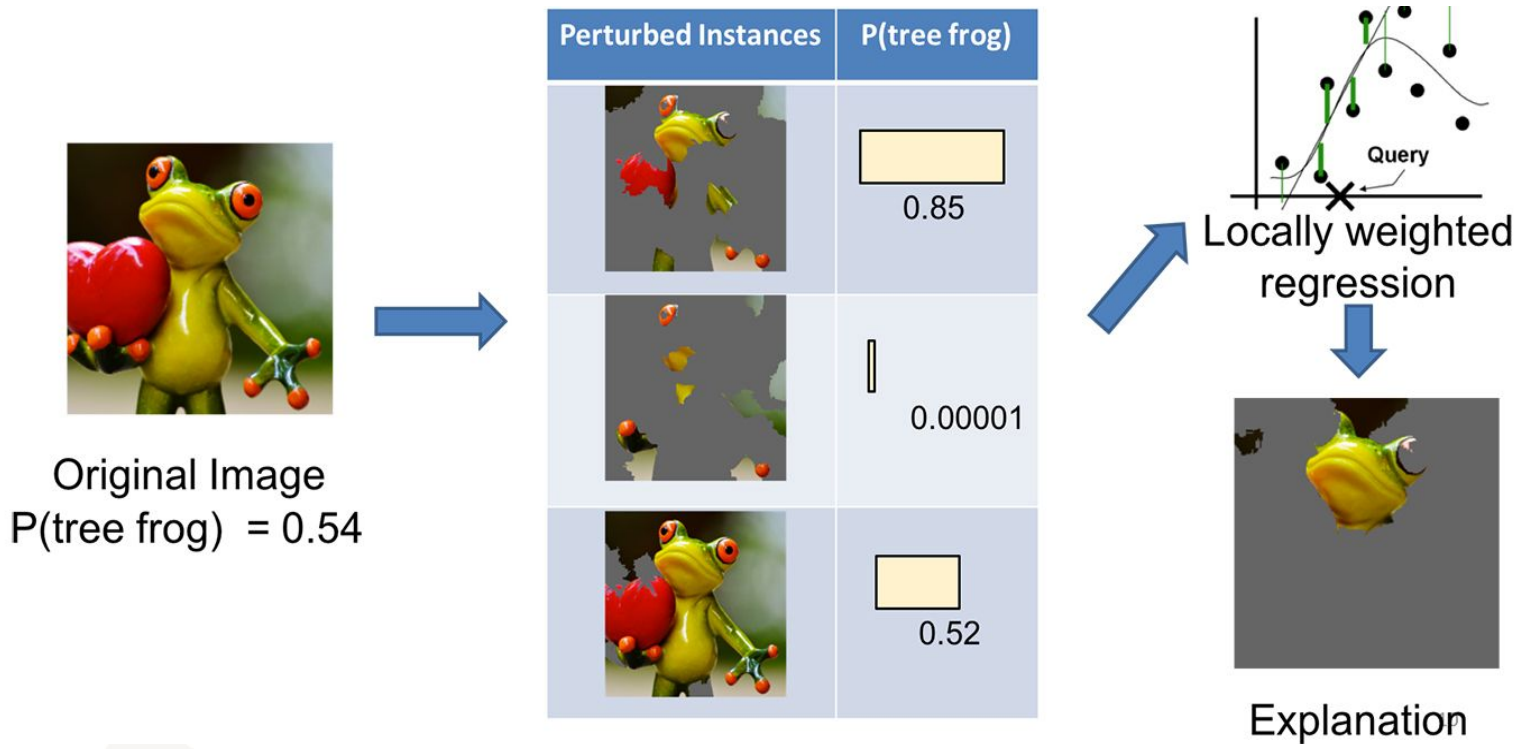
Explainability Algorithms in IBM AI Explainability 360



source: https://github.com/IBM/AIX360/blob/master/aix360/algorithms/README.md

Local Interpretable Model-Agnostic Explanations(LIME)

LIME is a novel explanation technique that explains the predictions of *any classifier in an interpretable* and faithful manner, by learning an *interpretable model locally* around the prediction. (https://arxiv.org/abs/1602.04938)



https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/

Local Interpretable Model-Agnostic Explanations(LIME)

Paper: https://arxiv.org/abs/1602.04938

Presentation: "Why Should I Trust you?" Explaining the Predictions of Any Classifier

Source code: https://github.com/marcotcr/lime

Blog post: https://www.oreilly.com/content/introduction-to-local-interpretable-model-agnostic-explanations-lime/

Videos

- Interpretable Machine Learning with LIME - How LIME works?
- Tutorial on model explanation with LIME

SHAP (SHapley Additive exPlanations)

SHAP (SHapley Additive exPlanations) assigns each feature an importance value for a particular prediction. Its novel components include:

(1) the identification of a new class of additive feature importance measures, and

(2) theoretical results showing there is a unique solution in this class with a set of desirable properties.

The new class **unifies six existing methods**, notable because several recent methods in the class lack the proposed desirable properties.

- paper: https://arxiv.org/pdf/1705.07874.pdf
- source code: https://github.com/slundberg/shap
- video: https://www.youtube.com/watch?v=wjd1G5bu_TY

Hands-on Tutorial

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

- Open the Black Box: an Introduction to Model Interpretability with LIME and SHAP - Kevin Lemagnen

- slides: https://speakerdeck.com/klemag/pydata-nyc-2018-open-the-black-box-an-introduction-to-model-interpretability-with-lime-and-shap

- source code: https://github.com/klemag/pydata_nyc2018-intro-to-model-interpretability/blob/master/Introduction%20to%20Model%20Interpretability.ipynb

Project Description

1. Install and get to know AIX 360 IBM and InterpretML Microsoft.
2. Choose one of the interpretability methods/algorithms LIME or SHAP.
3. Familiarize yourself with the chosen method by running an existing example (demo) with provided data set and model directly in AIX 360 and InterpretML.
4. Select a new use case with data set and model that is not already provided in both tools.
5. Integrate it in both AIX 360 and InterpretML and perform explainability evaluation using LIME or SHAP (chosen in step 2).
6. Integrate your code into notebook and upload it into github.
7. Evaluate and compare both tools (AIX 360 and InterpretML) based on your use case.
8. Prepare your final 15 slides documenting your findings and use case implementation.

Bonus Task

1. Integrate your use case in What If Google.
2. Compare the experience, functionality and features from What if with AIX 360 and InterpretML.
3. Document your findings and implementation.

Course Organization

- Work in teams of 2 students.
- Send me email (aitoolslabss2020@gmail.com) with team member names and selected method/algorithm (LIME or SHAP). Write the module name (DB-MPR, M-DS-PR-K, M-DS-PR-A, M-DS-PR-B, DB-PR, M-SIW-PRA, M-SIW-PRB, DB-PR) under which you want to register the Hands-on lab/Praktikum.
- From next week 10 - 15 min. general meeting from 10 am on Tuesday.
- Team meeting (20 min.) every Tuesday in the same time slot.

More Resources

- What If Google paper: https://arxiv.org/pdf/1907.04135.pdf
- What-If Tool & SHAP: https://pair-code.github.io/what-if-tool/wit_fat_2020.pdf
- AIX 360 IBM paper: https://arxiv.org/pdf/1909.03012.pdf
- InterpretML Microsoft paper: https://arxiv.org/pdf/1909.09223.pdf

Python Environments (Cloud-based)

- Google Colaboratory Tools - https://colab.research.google.com/notebooks/intro.ipynb
- IBM Notebooks - https://dataplatform.cloud.ibm.com/docs/content/wsj/analyze-data/notebooks-parent.html

What-If Tool

GOETHE UNIVERSITÄT FRANKFURT AM MAIN

What-if Tool is an interactive visual interface designed to probe your models better.
https://pair-code.github.io/what-if-tool/

- Compare multiple models within the same workflow
- Visualize inference results
- Visualize feature attributions
- Arrange data points by similarity
- Edit a datapoint and see how your model performs
- Compare counterfactuals to data points
- Use feature values as lenses into model performance
- Experiment using confusion matrices and ROC curves
- Test algorithmic fairness constraints

AI Fairness & Explainability 360

AIF 360 is an open source toolkit that can help you ***examine, report, and mitigate discrimination and bias in machine learning models*** throughout the AI application lifecycle. Containing over ***70 fairness metrics*** and ***10 state-of-the-art bias mitigation algorithms*** developed by the research community, it is designed to translate algorithmic research from the lab into the actual practice of domains as wide-ranging as finance, human capital management, healthcare, and education.
https://aif360.mybluemix.net/

AIX 360 is an open source toolkit that can help you ***comprehend how machine learning models predict labels by various means*** throughout the AI application lifecycle. Containing ***eight state-of-the-art algorithms for interpretable machine learning*** as well as ***metrics for explainability***, it is designed to translate algorithmic research from the lab into the actual practice of domains as wide-ranging as finance, human capital management, healthcare, and education.
http://aix360.mybluemix.net/

InterpretML - https://github.com/interpretml/interpret

- **InterpretML** is an open-source python package for training interpretable machine learning models and explaining black-box systems.

- Interpretability is essential for:
  - Model debugging - Why did my model make this mistake?
  - Detecting bias - Does my model discriminate?
  - Human-AI cooperation - How can I understand and trust the model's decisions?
  - Regulatory compliance - Does my model satisfy legal requirements?
  - High-risk applications - Healthcare, finance, judicial, ...