25.09.2024
Colloqium / Hands - On

TK8117 – Week 05 - Topic 04:

Shapley values
for assessing feature importance

Aafan Ahmad Toor
Andreas Gudahl Tufte
Azimil Gani Alam
Niclas Flehmig

Norwegian University of Science and Technology

Overview

Motivation

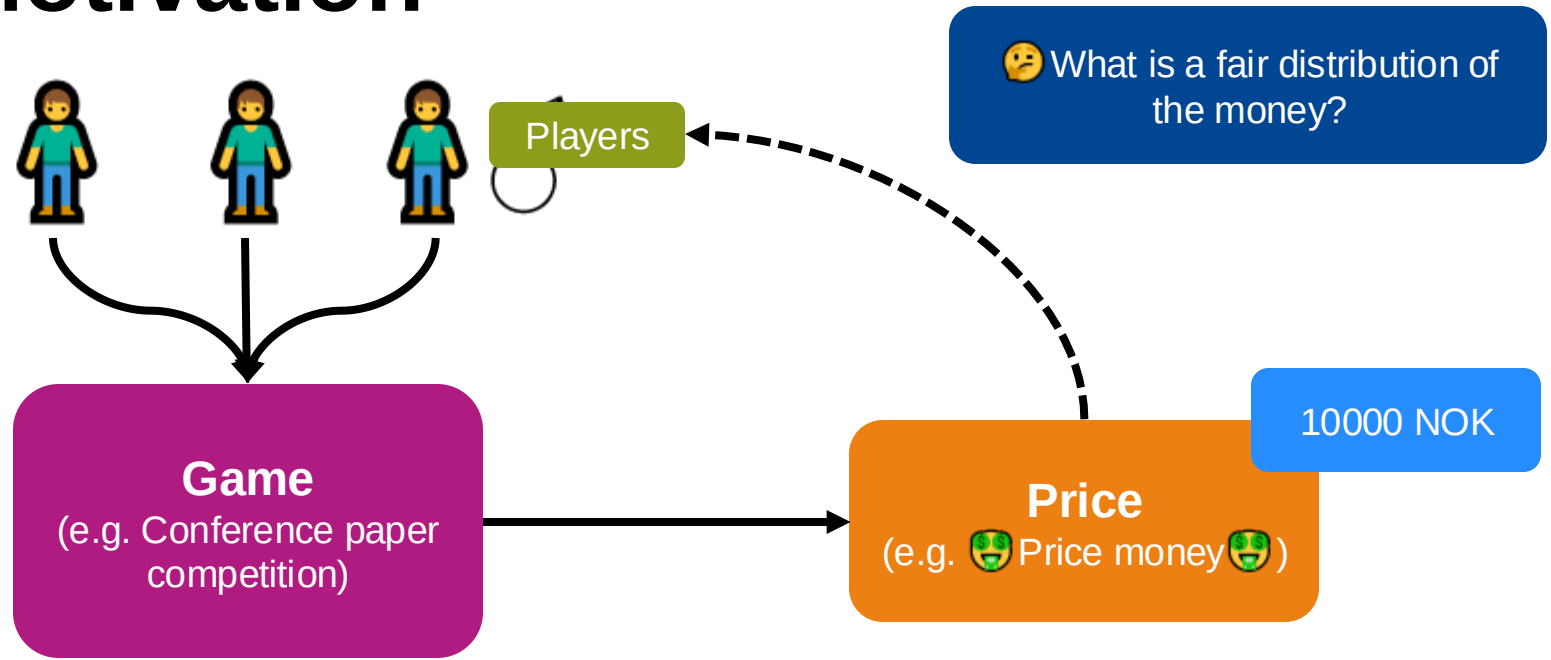
Objective & Calculation of Shapley Values

Pros & Cons

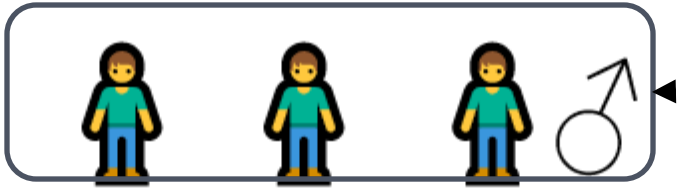
Background Knowledge to Coding Example

Motivation for Shapley values

Motivation


Players
🤔 What is a fair distribution of the money?
Game
(e.g. Conference paper competition)
Price
(e.g. 🤑 Price money 🤑)
10000 NOK

Motivation



Model all possible subsets (2^n)

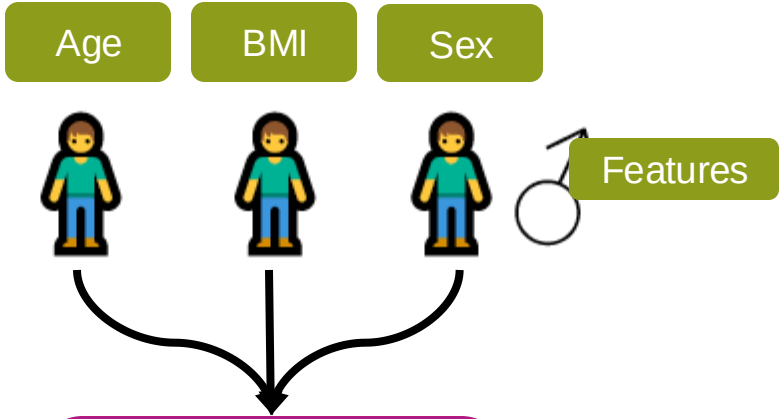
🔍We focus on how the outcome changes

Conference Paper Competition

5000 NOK

🤑Price money🤑

How does it look for ML?



ML model (e.g. RandomForest)

Prediction (e.g. class prediction)

Objective & Calculation

🧐What are Shapley values?

Basic idea from **coalitional game theory**

Group of players (features) compete together in a **game** (ML model)

After the game, we want to **distribute the price** (prediction) in a **fair** manner

Objective!

Shapley value tells us the average **contribution** of a **player** (feature) to a **payout** (prediction)

Calculating those mysterious values

To **calculate Shapley** values we can use **SHAP** (which connects LIME and Shapley values)

👨🍳 **Recipe** 👨🍳

Features are **not independent** so, we create **subsets of features**

Aggregate each feature's contribution over **all possible subsets** to get a **global explanation**

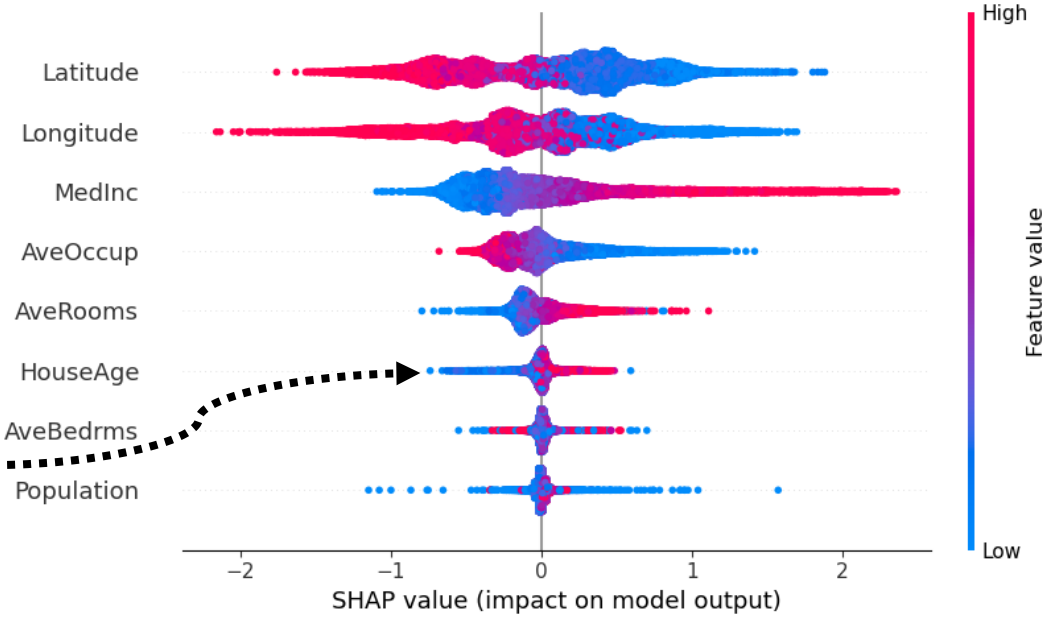
Make the predictions of our model **explainable** and **interpretable**

Examples

Examples of SHAP

Summary plot

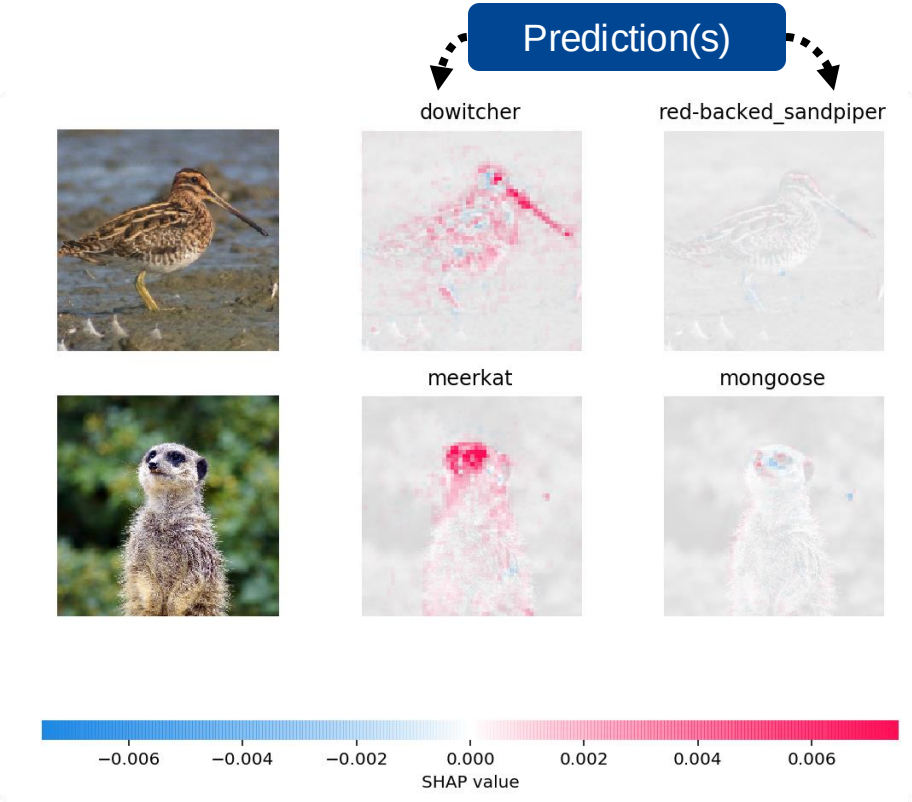
Each point is a Shapley value for a feature and an instance



Relationship between value of a feature and impact on the prediction

NTNU

Examples of SHAP



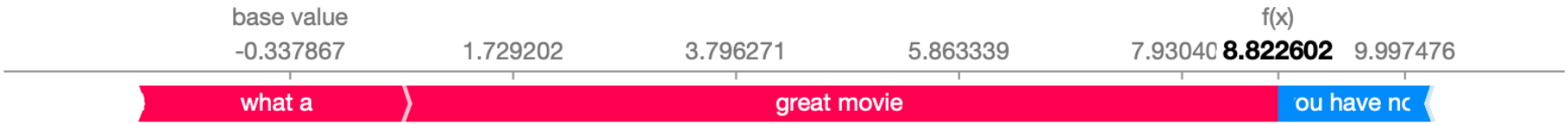
🥵 Heatmap of SHAP values

🧐 We can identify which features are important for the model to make its decision

Examples of SHAP

Average prediction of the ML model

Prediction for this input (e.g. the next token/word)

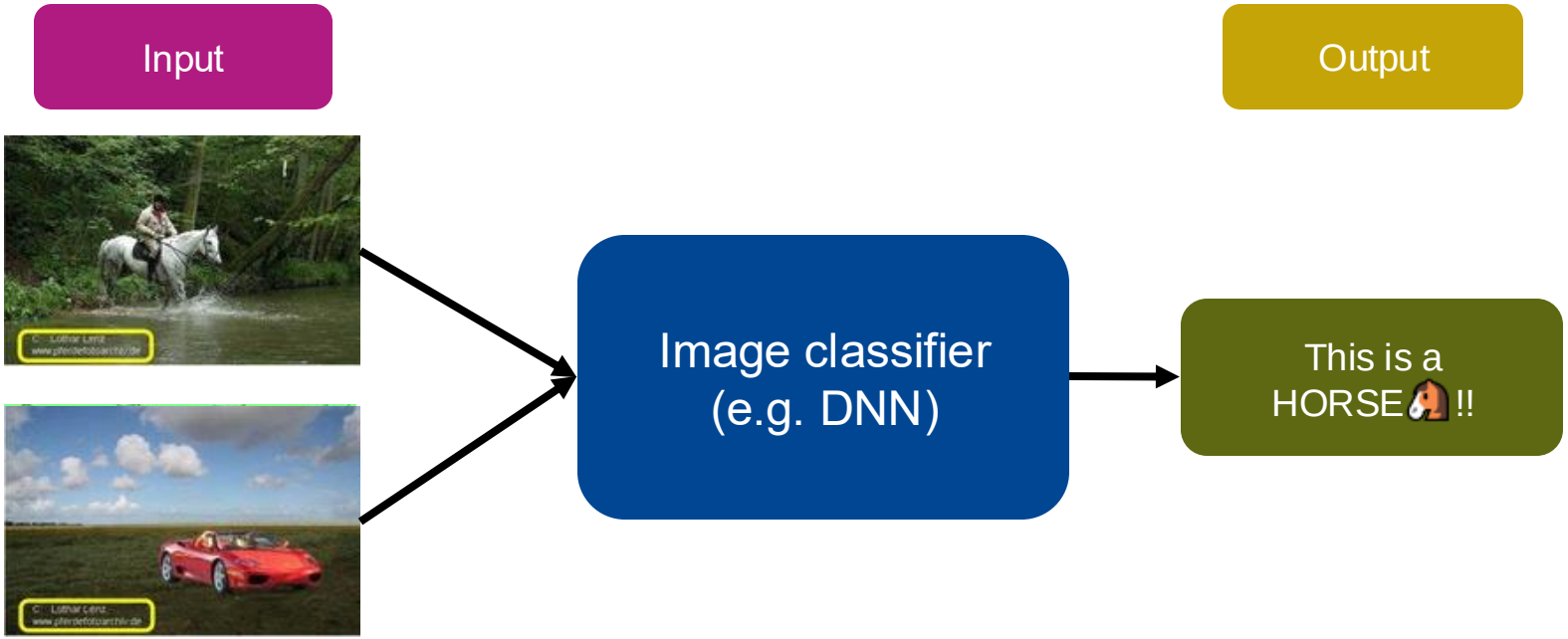


what a great movie ! . . . if you have no taste .

Input / instance

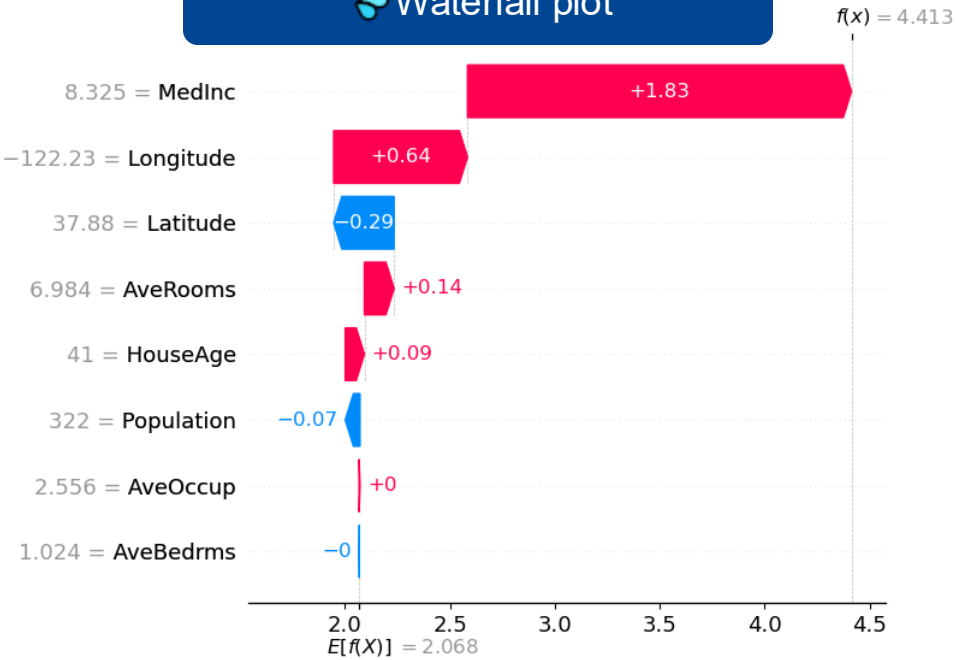
💡This could be ChatGPT!💡

Why do I want explainability??



Be careful…

Waterfall plot



NOT the difference in prediction when we would remove feature!

Return simple value per feature, **no prediction model**

Cannot be used to make **statements** about **changes** in **prediction** for **changes** in **input**

Pros & Cons

Pros & Cons

Solid theoritical foundation (i.e. Game theory)

Fairly distributed among feature values

Global model interpretations

Can be applied on different models (e.g. Tree-based, linear, DNN)

Can be misinterpreted

Computationally expensive (e.g. complex models)

No exact solution for non-linear models

Application

Need for **explainable AI** (e.g. EU regulations, supervisor asks me what is going on)

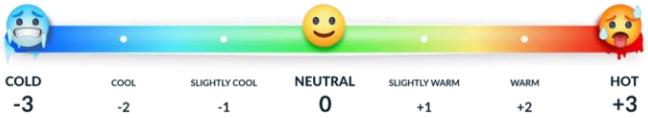
Enhance the **trust** in my model

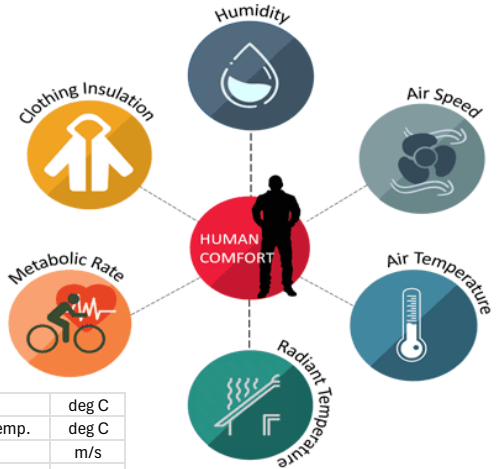
Levarage the **interpretability** of my model and potentially **improve** it

Increase **safety** of my model

👦💻Coding example

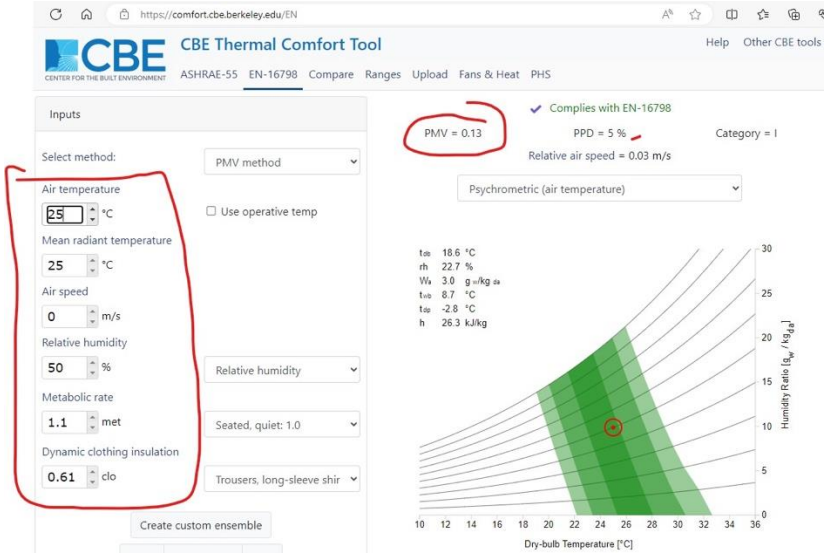
Background Knowledge

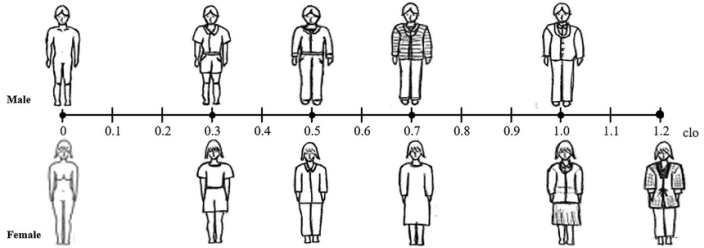




Air Temperature	deg C
Mean Radiant Temp.	deg C
air velocity	m/s
Relative Humidity	%
Metabolic Rate	-
Cloth Thick Rate	-
Thermal Sensation	-

Activity	Metabolic Rate (Met)
Resting	
Sleeping	0.8
Seating, quiet	1.0
Standing, relaxed	1.2
Sport and Activities	
Archery	4.3
Badminton	5.5
Basketball	8.0
Bicycling	7.5
Boxing	12.8
Calisthenics	3.5
Dancing	7.8
Fencing	6.0
Fishing	3.5

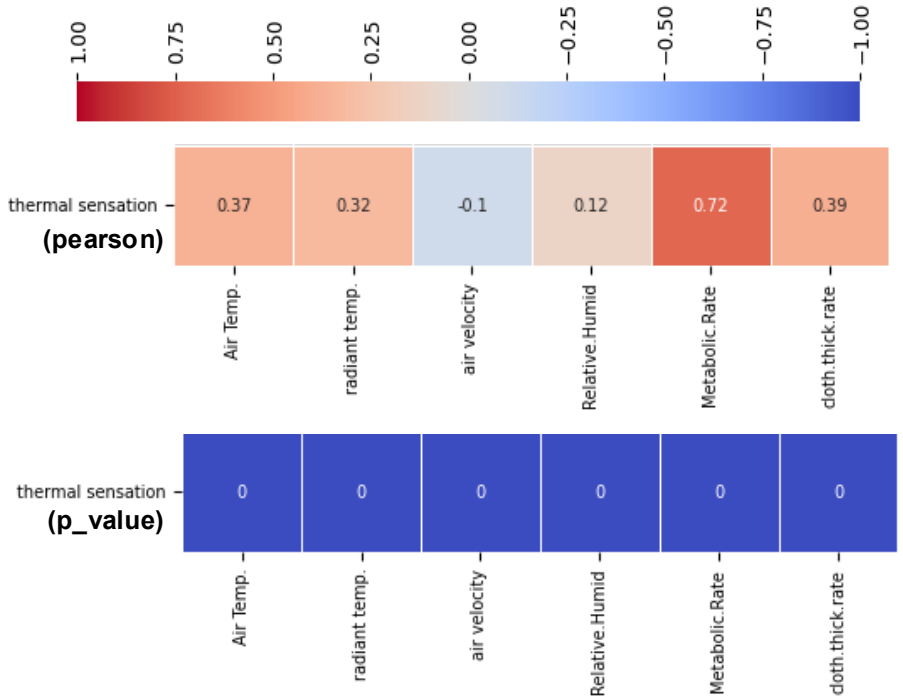




Dataset

- Thermal Comfort Data
- Predict Thermal Sensation

	Air Temp.	radiant temp.	air velocity	Relative.Humid	Metabolic.Rate	cloth.thick.rate	thermal sensation
count	630697.000000	630697.000000	630697.000000	630697.000000	630697.000000	630697.000000	630697.000000
mean	21.506338	21.700489	0.099925	38.757755	1.429113	0.741664	-0.257738
std	2.290107	2.606948	0.070668	18.493891	0.341060	0.185883	0.818509
min	18.000000	18.000000	0.000000	15.000000	1.000000	0.500000	-3.000000
25%	20.000000	19.000000	0.050000	25.000000	1.100000	0.570000	-0.780000
50%	22.000000	22.000000	0.100000	40.000000	1.400000	0.670000	-0.150000
75%	24.000000	24.000000	0.150000	50.000000	1.800000	0.960000	0.360000
max	25.000000	26.000000	0.200000	70.000000	2.000000	1.000000	1.610000



https://raw.githubusercontent.com/rayunacute/MVDA/refs/heads/main/PMV%20PPD/PMV%20example%20data.csv

References

- https://github.com/shap/shap?tab=readme-ov-file
- https://shap.readthedocs.io/en/latest/example_notebooks/overviews/An%20introduction%20to%20explainable%20AI%20with%20Shapley%20values.html
- https://link.springer.com/chapter/10.1007/978-3-031-24628-9_41
- https://christophm.github.io/interpretable-ml-book/shap.html
- https://www.youtube.com/watch?v=9haIOplEIGM
- https://proceedings.neurips.cc/paper_files/paper/2017/hash/8a20a8621978632d76c43dfd28b67767-Abstract.html