business innovation through algorithms

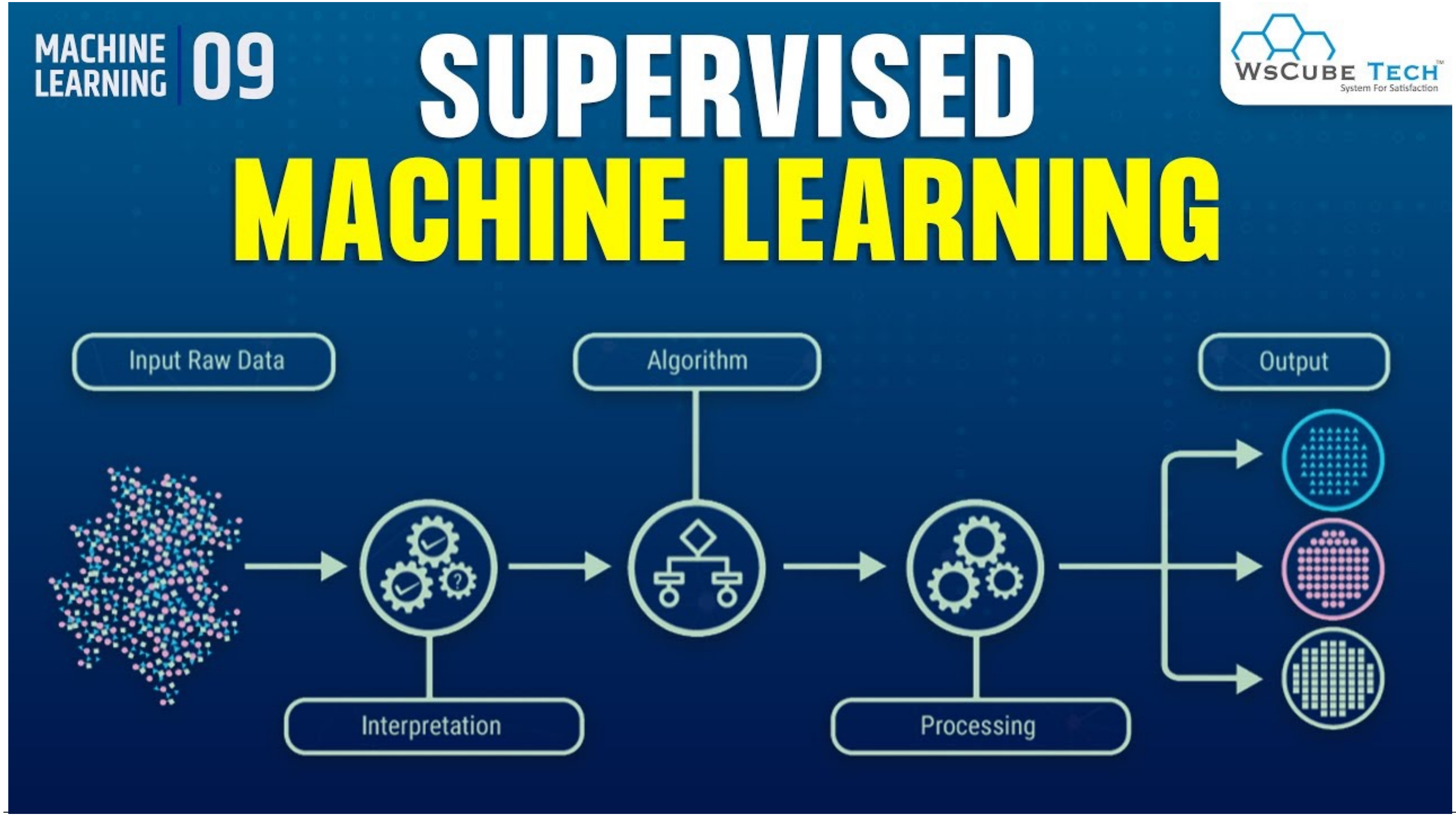
Explainability & Interpretability

Glass Box Model vs Black Box Model

CGnal S.r.l – Corso Venezia 43 - Milano

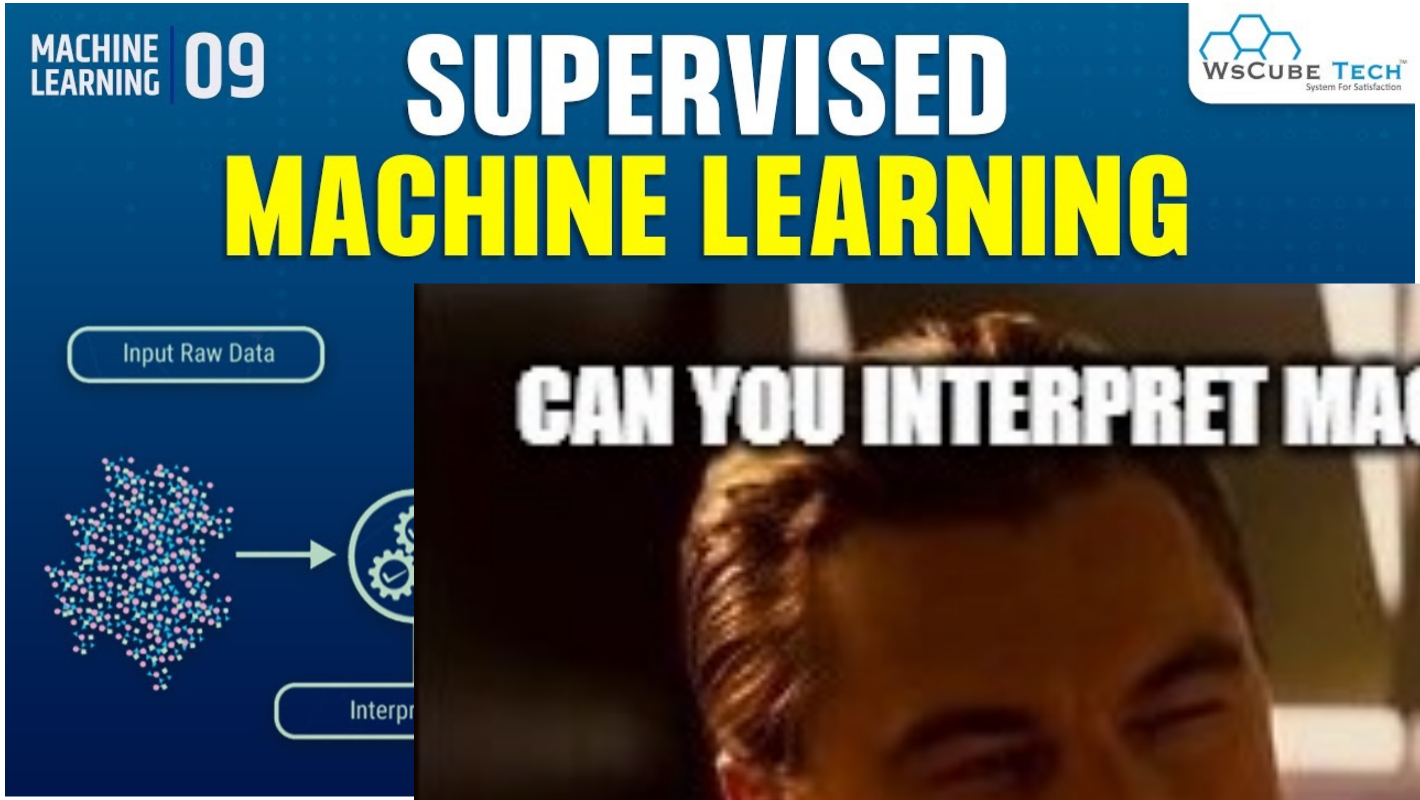
Novembre 2022| Milano

Field



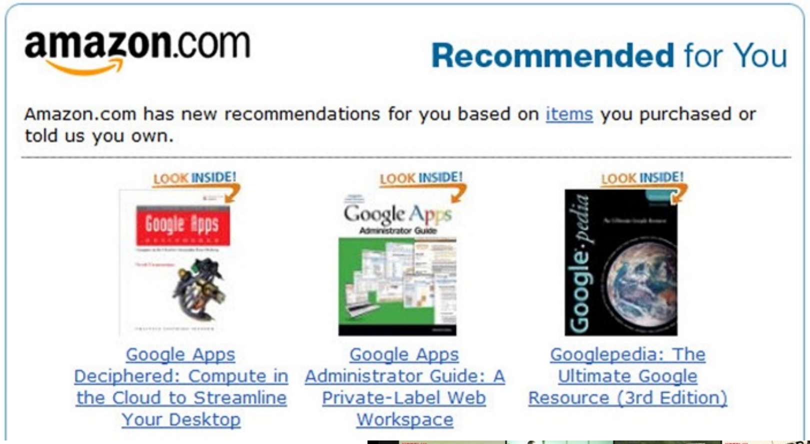
MACHINE LEARNING
09
SUPERVISED
MACHINE LEARNING
WsCube Tech
System For Satisfaction
Input Raw Data
Algorithm
Output
Interpretation
Processing

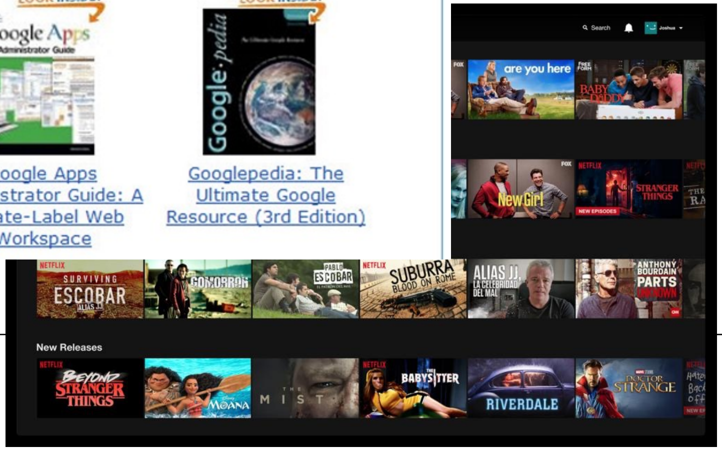
Field


MACHINE LEARNING 09
SUPERVISED
MACHINE LEARNING
WsCube Tech
System For Satisfaction
Input Raw Data

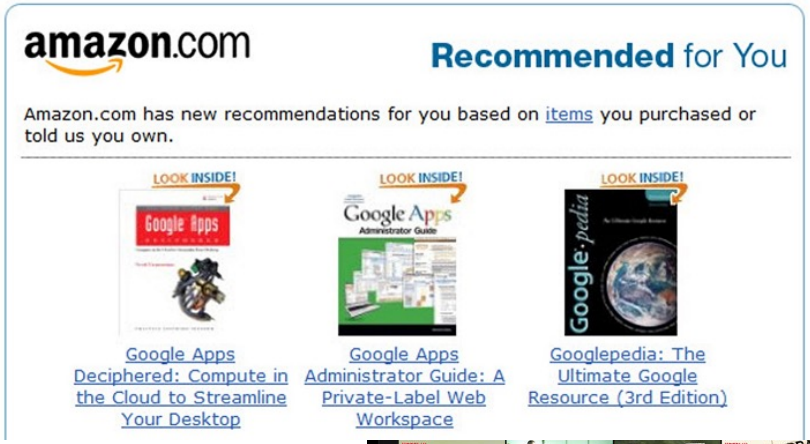

CAN YOU INTERPRET MACHINE LEARNING MODELS
JUST USE MORE MACHINE LEARNING

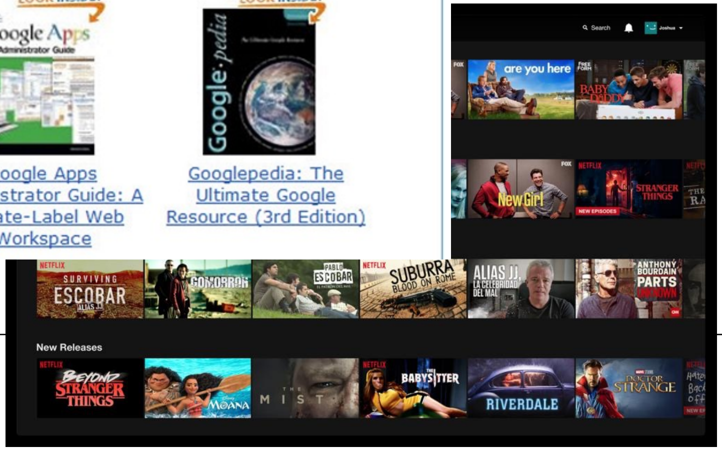
Machine Learning is everywhere





Is Model Understanding Needed Everywhere?





When Model Understanding is needed?

Model Understanding is not needed when/because:

- Little to no consequences for incorrect predictions
- Problem is well studied and models are extensively validated in real world application
- …

Model Understanding is needed when/because:

- **The choice** can have **impacts** on the human lives, health or finances
- Problems are **not well studied** and it not possible to extensively validate in real world application
- **Accuracy (measures) of the model** is no longer enough: for example when the train and test data are not representative of new data encounter
- The model must to be fair (nondiscriminance)
- …

When Model Understanding is needed?

A Typical Machine Learning Example

- I have data, and I want to solve a problem. So, just deploy a model!
- But in real life, things are much more complicated.
- You have various parties: ML Scientists, Product Managers, End Users.

ML Scientist:

Which model features should I use? Does my model perform well?

Product Managers:

Can I trust/deploy this model? Is it fair for all parties?

End User:

Why did it give me this prediction?

If the users do not trust a model or a prediction, they will not use it

black box *vs* glass box

A Typical Machine Learning Example

- I have data, and I want to solve a problem. So, just deploy a model!
- But in real life, things are much more complicated.
- You have various parties: ML Scientists, Product Managers, End Users.

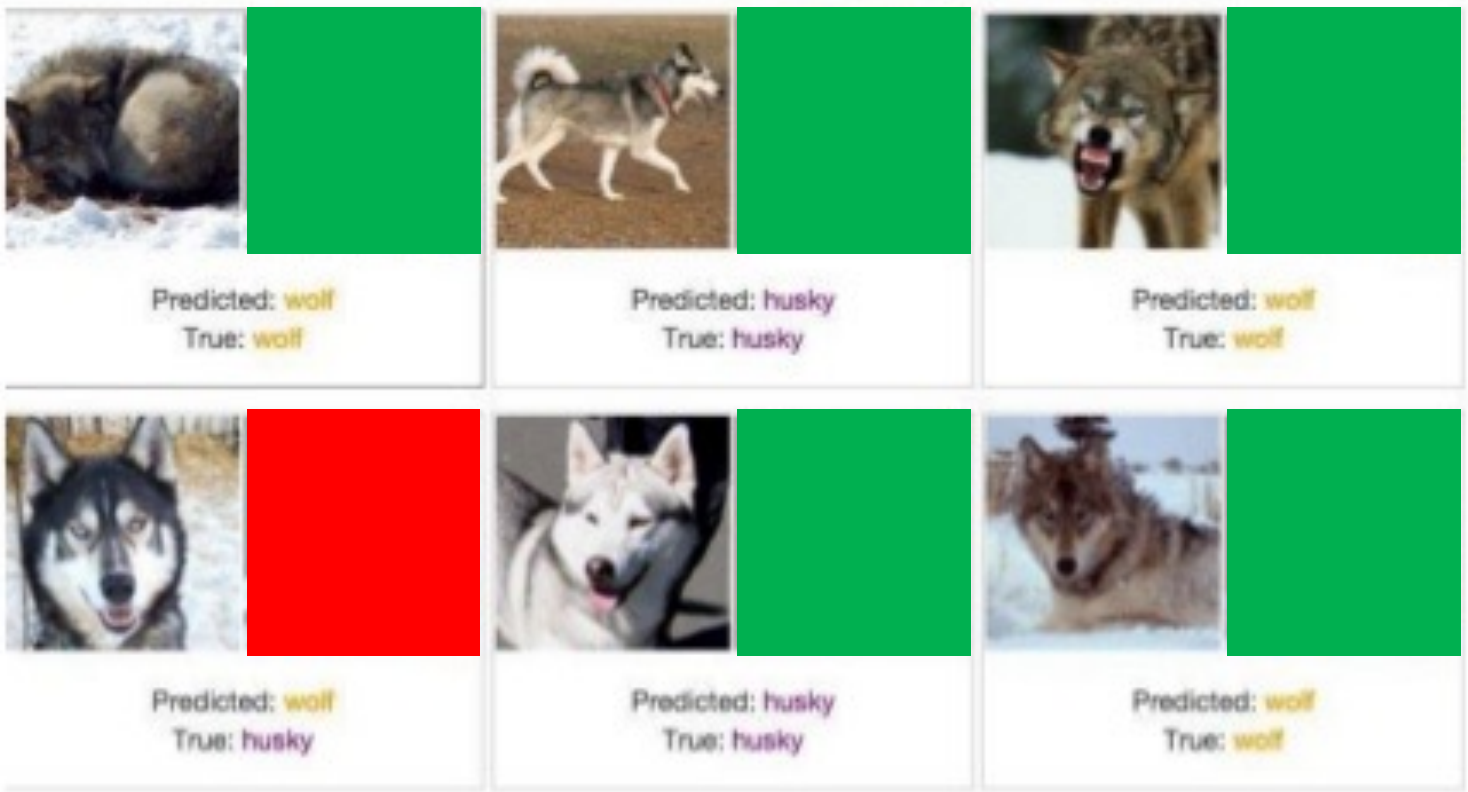
ML Scientist:
Which model features should I use? Does my model perform well?

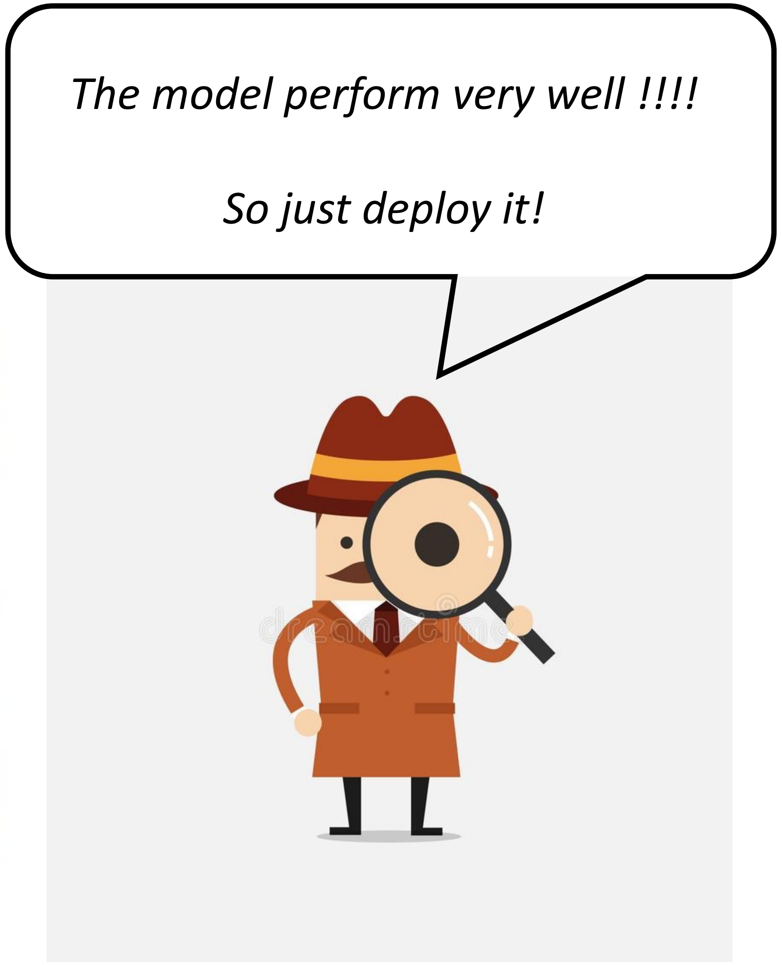
Product Managers:
Can I trust/deploy this model? Is it fair for all parties?

End User:
Why did it give me this prediction?
If the users do not trust a model or a prediction, they will not use it

For each actor it is necessary and usefull the model ***intepretability***

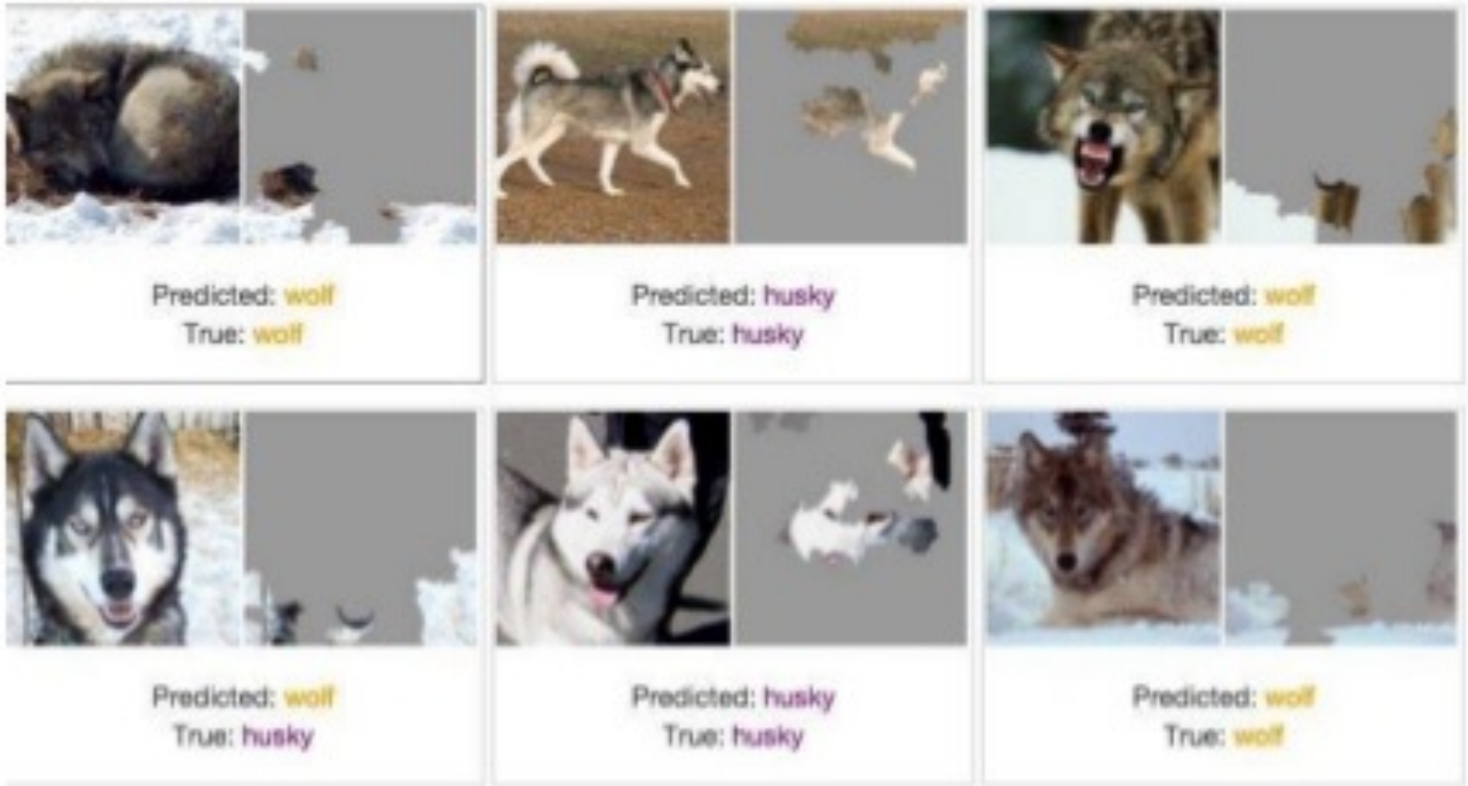
Understanding the model



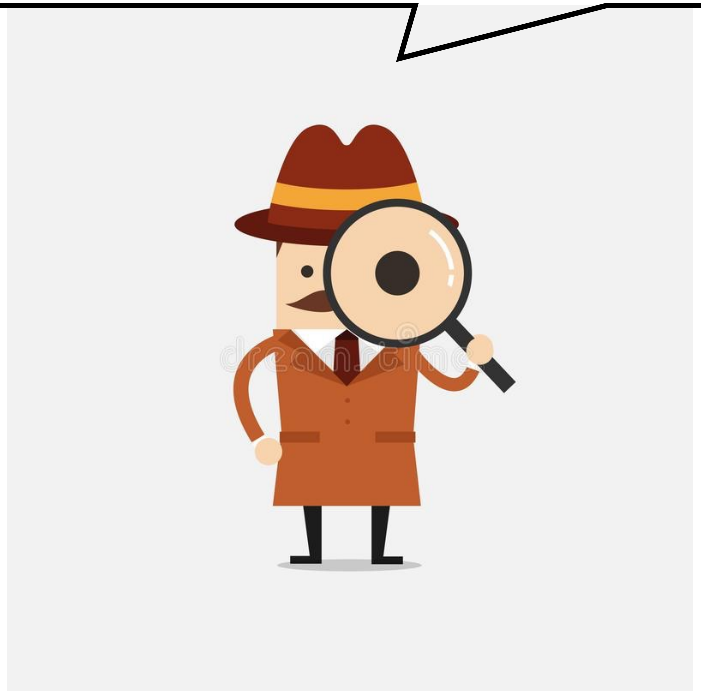


Understanding the model

… YES, IF YOU WANT TO BUILD A GREAT SNOW DETECTOR!



No the model is biased by the snow

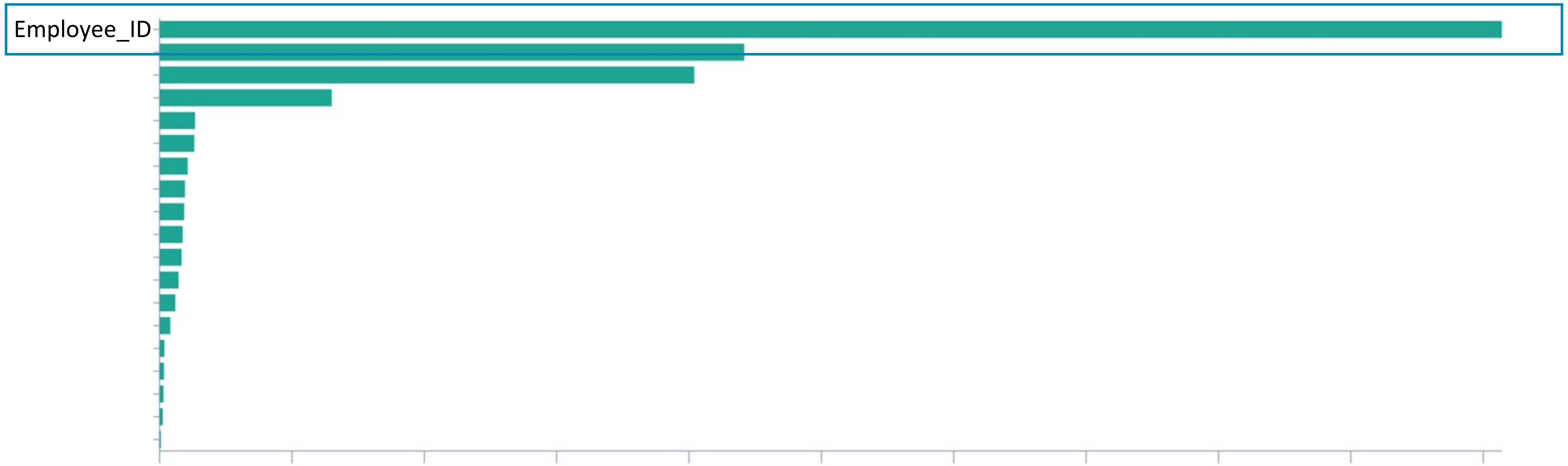


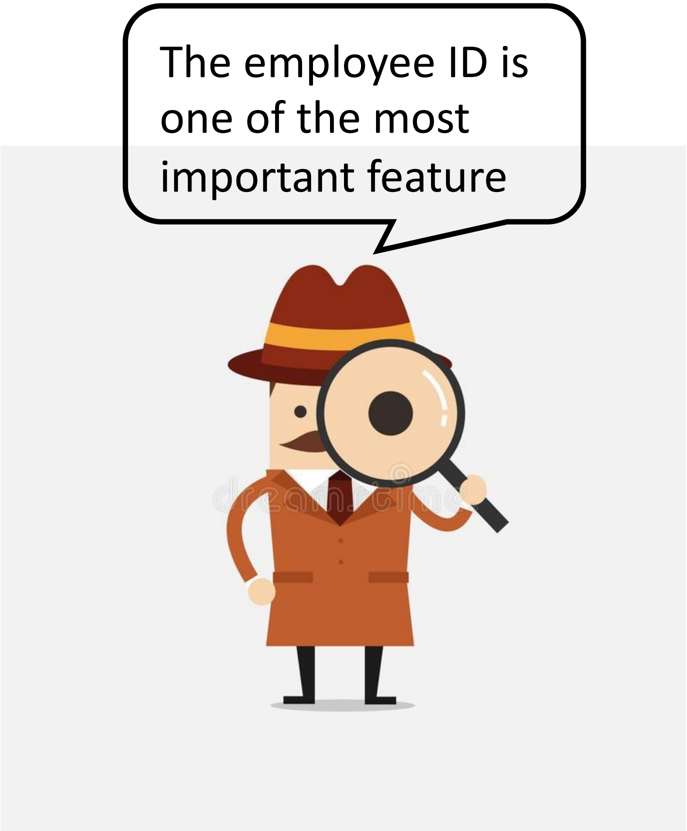
Understanding the model

LOAN APPROVAL:

Think if your collegue ROSSI usually works on the most critical contracts.

employee_ID	other features	target
ROSSI	....	DECLINED
ROSSI	....	DECLINED
BIANCHI	....	APPROVED
BIANCHI	....	APPROVED
ROSSI	....	DECLINED
ROSSI	....	DECLINED
....	....	....





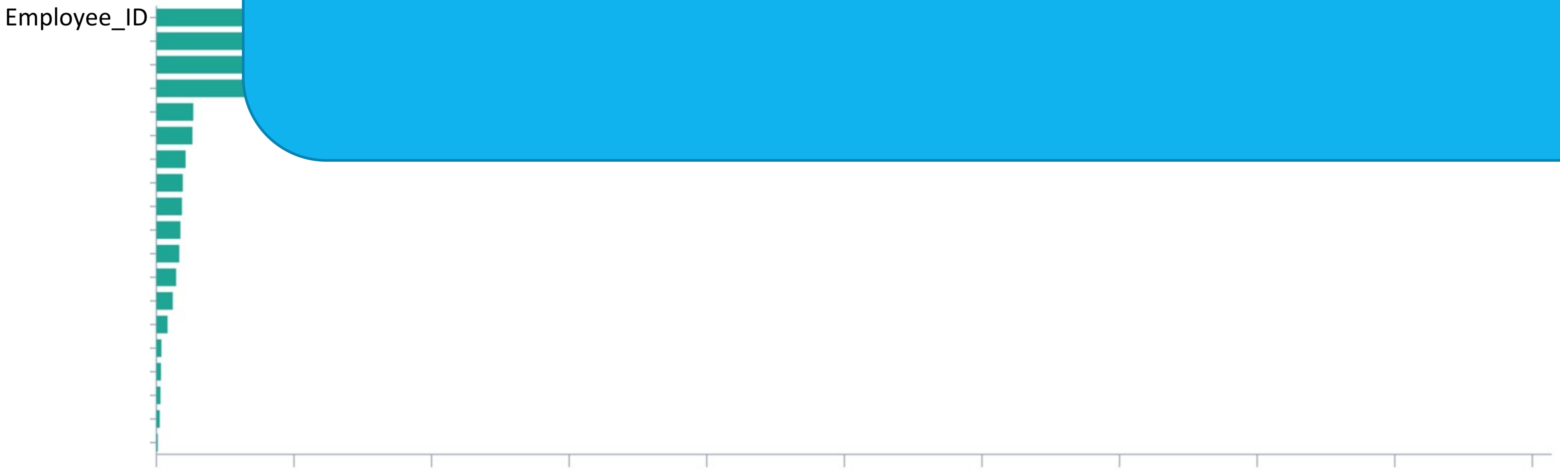
Understanding the model

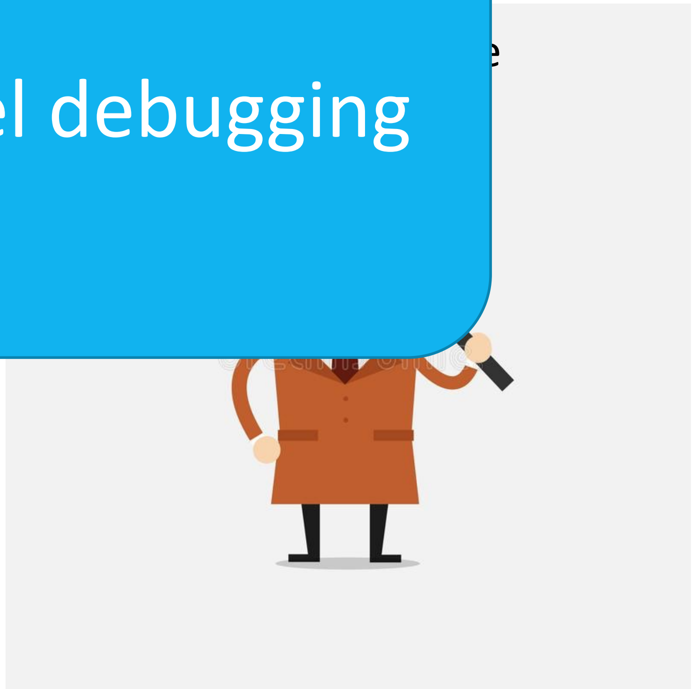
LOAN APPROVAL:

Think if your collegue ROSSI usually works on the most critical contracts.

employee_ID	other features	target

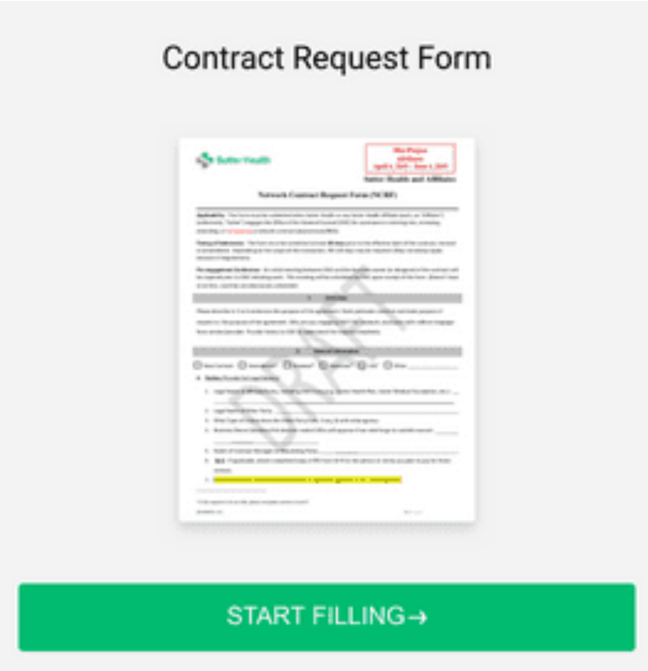
Model understanding facilitates the model debugging





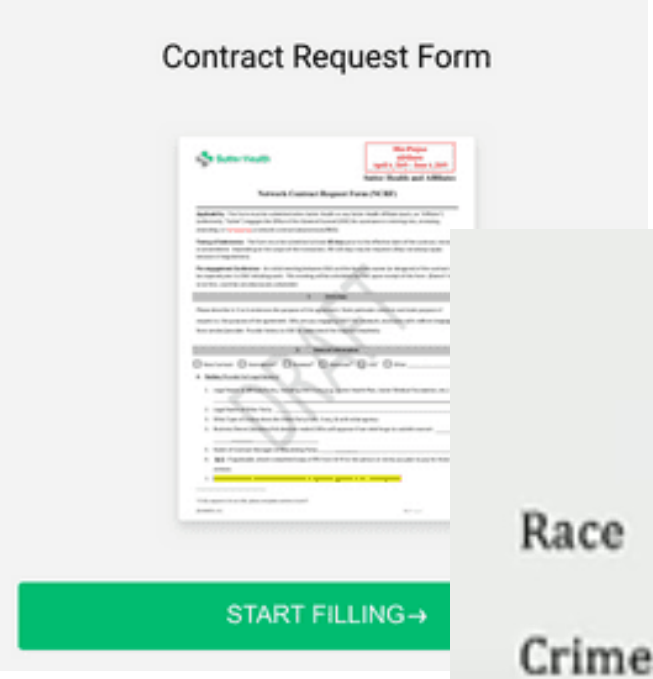
Understanding the model

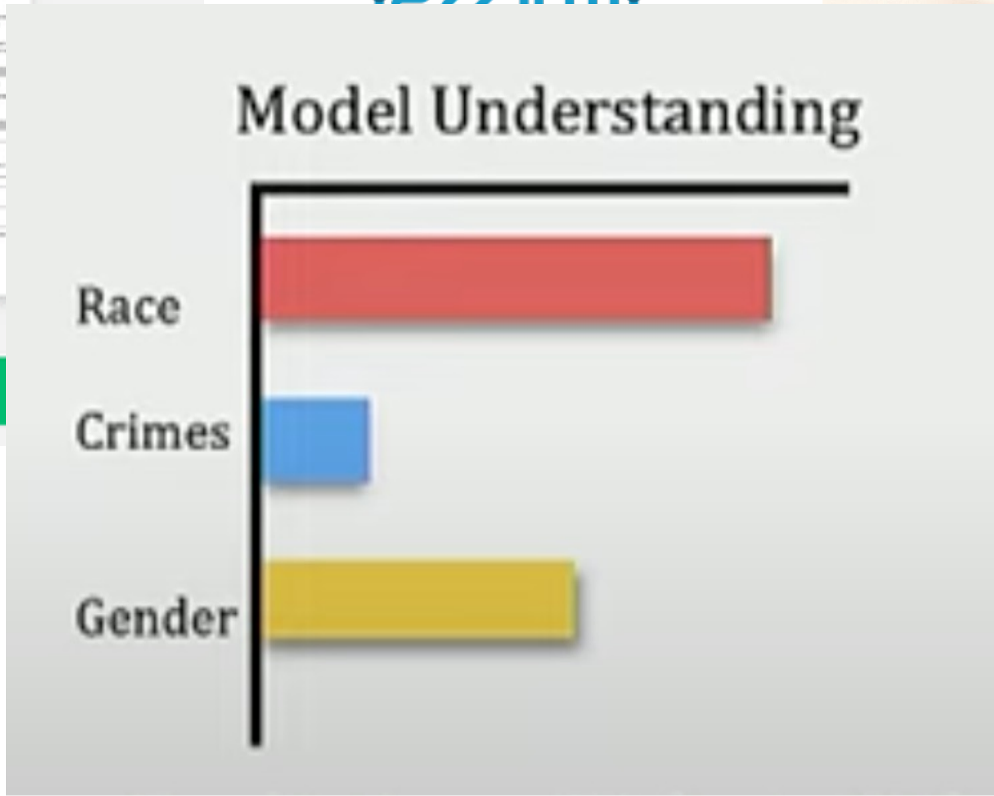
LOAN APPROVAL



Understanding the model

LOAN APPROVAL

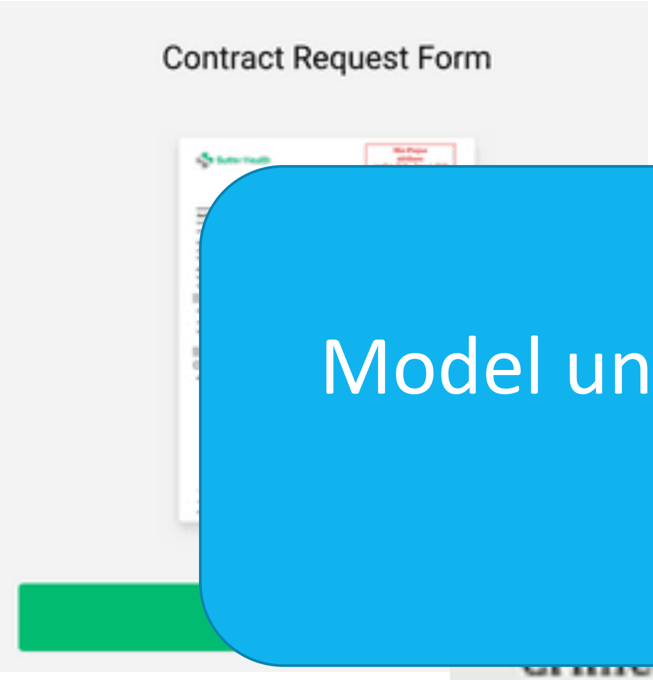


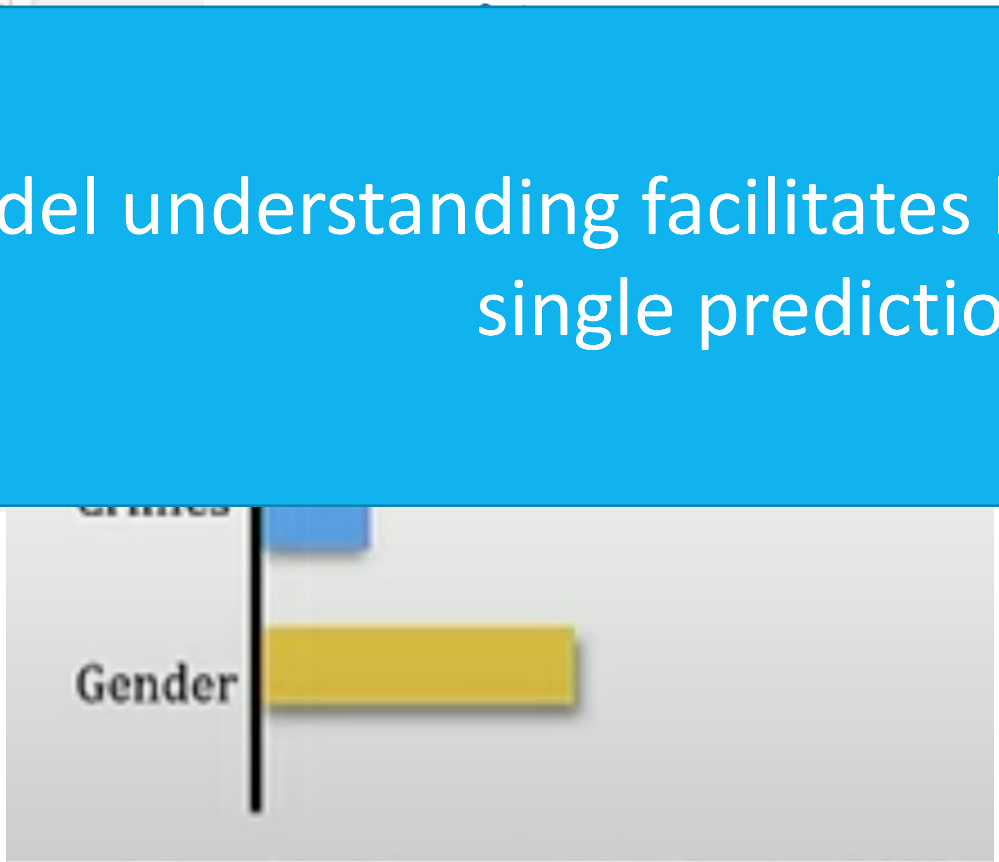


This prediction is biased!!!

Understanding the model

LOAN APPROVAL

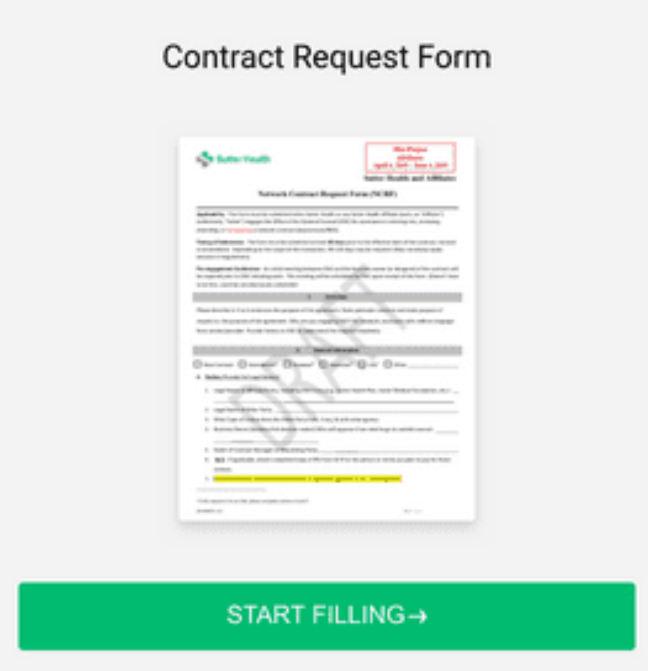




Model understanding facilitates bias detection on a single prediction

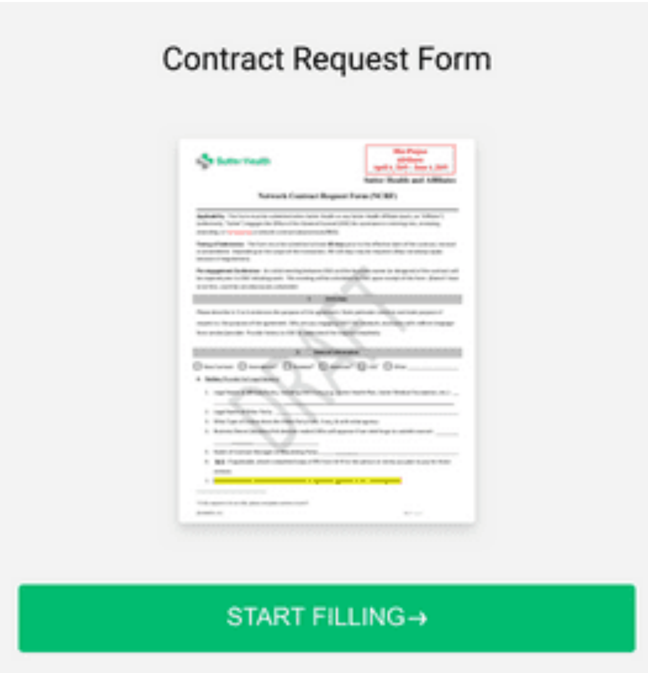
Understanding the model

LOAN APPROVAL IN PRODUCTION



Understanding the model

LOAN APPROVAL IN PRODUCTION



How can I get the loan in N months?

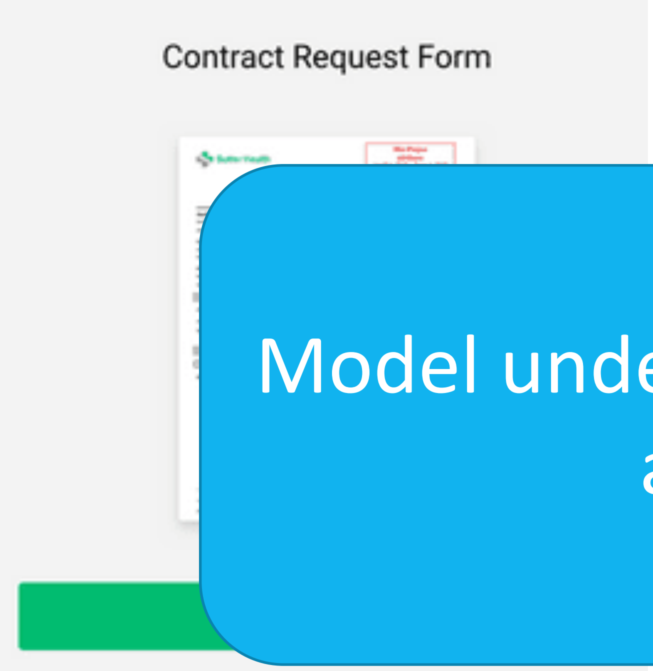
Loan applicant

SUGGESTIONS:

- Pay credit card bills on time for the next 3 months
- Increment your salary by 50K
- ....

Understanding the model

LOAN APPROVAL IN PRODUCTION



How can I get the loan in N months?

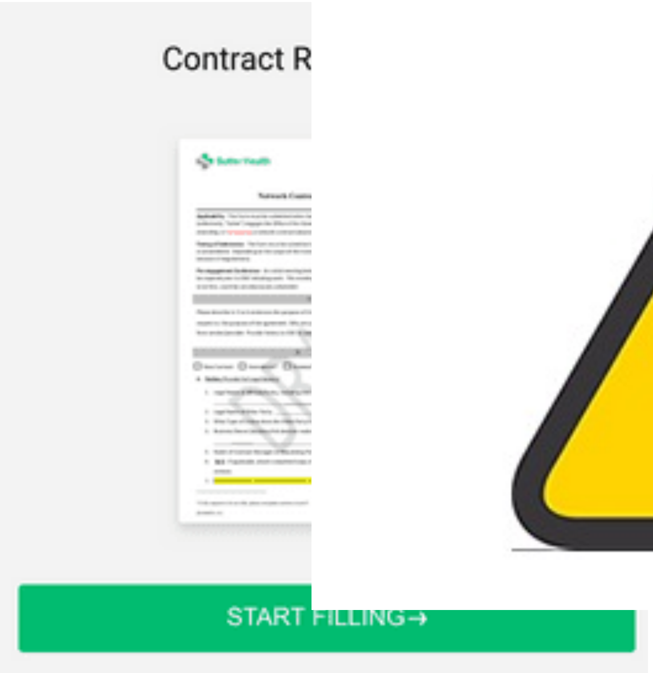
Model understanding provide suggestions to individual affected by the model prediction

SUGGESTIONS:
- Pay credit card bills on time for the next 3 months
- Increment your salary by 50K
- ....

Loan applicant

Understanding the model

LOAN APPRC



That's just racism

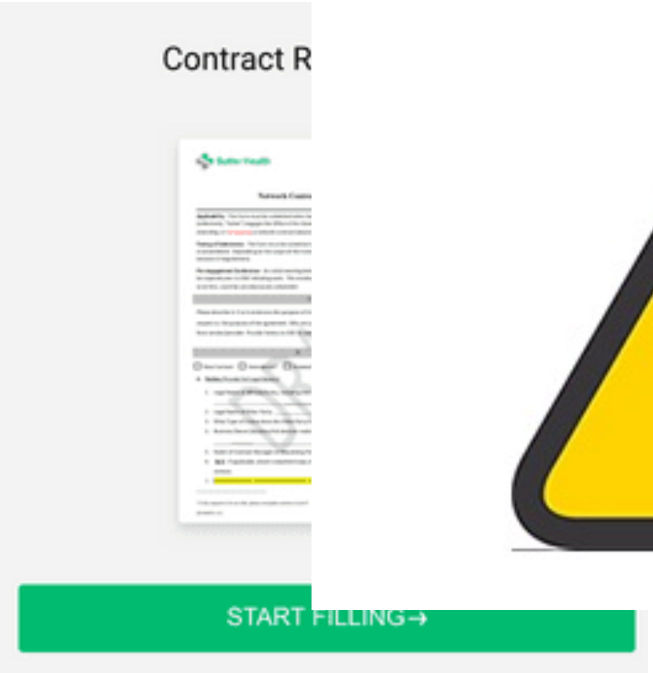
Loan applicant

SUGGESTIONS:

- **CHANGE YOUR RACE**

Understanding the model

LOAN APPRC



We cannot deploy this model

Project Manager

SUGGESTIONS:

- **CHANGE YOUR RACE**

Summary: Why model understanding is usefull/needed?

UTILITY

- Debugging
- Bias Detection
- Suggestions
- If and when to trust a prediction
- Asses suitability for the deployment

STAKEHOLDERS

- End Users
- Decision making
- Regultory systems
- Project manager
- …

Achiving Model Understanding

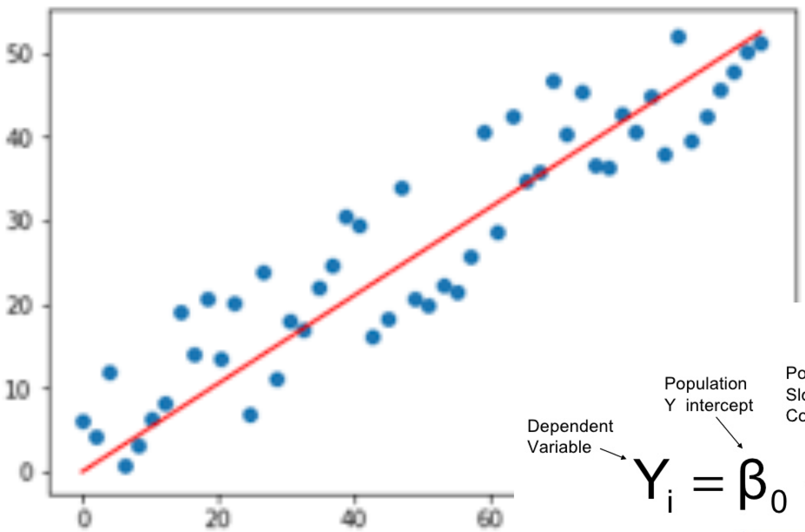
Achiving Model Understanding

Method 1: Build ***inherently interpretable*** prediction models

Achiving Model Understanding

Method 1: Build ***inherently interpretable*** prediction models

Linear regression



Dependent Variable

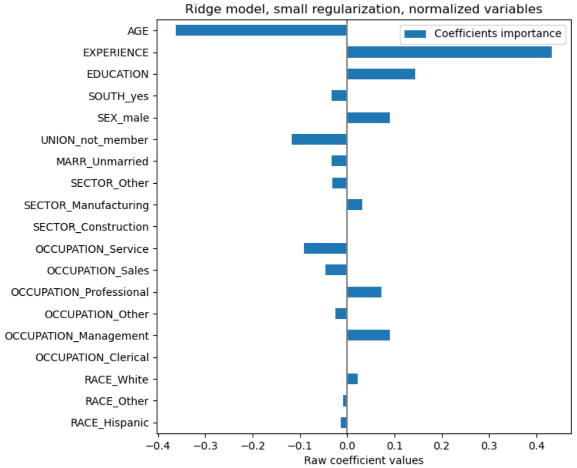
Population Y intercept

Population Slope Coefficient

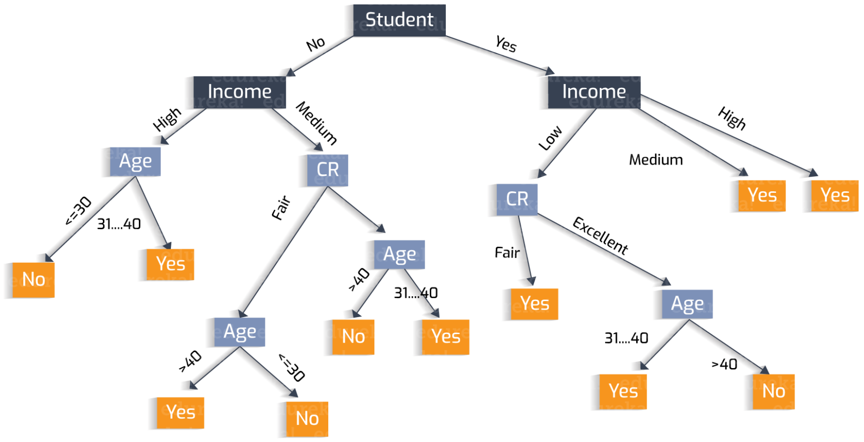
Independent Variable

Random Error term

$$Y_i = \underbrace{\beta_0 + \beta_1 X_i}_{\text{Linear component}} + \underbrace{\varepsilon_i}_{\text{Random Error component}}$$



Tree model



If *Student==Yes:*

if *Income==High:*

then prediction *Yes.*

else if *Income==Medium:*

then prediction *Yes.*

else:

if *CR==Excellent:*

…..

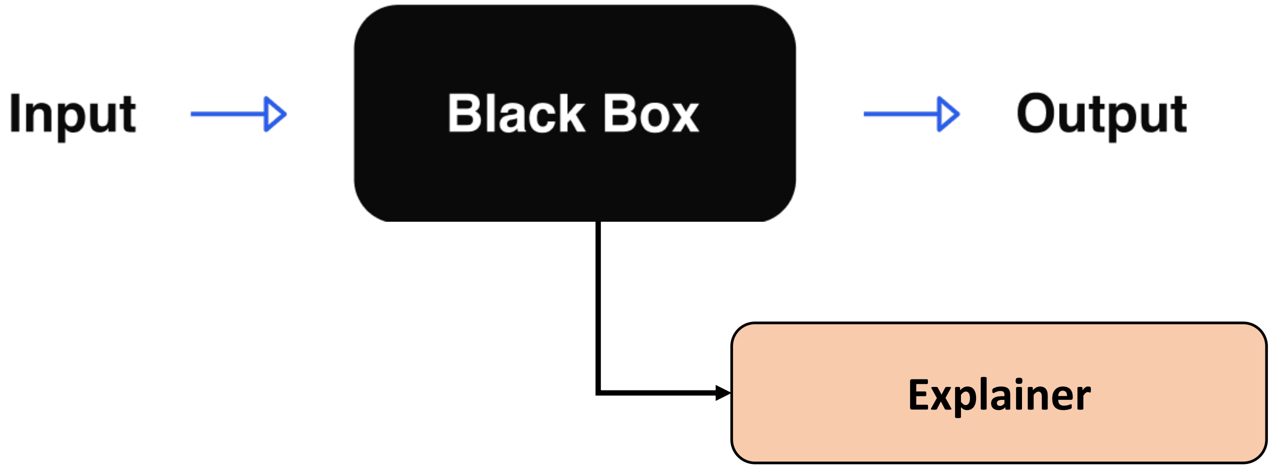
CGn

Achiving Model Understanding

Method 2: ***Explain*** **already-built model in a** ***post-hoc*** **manner**

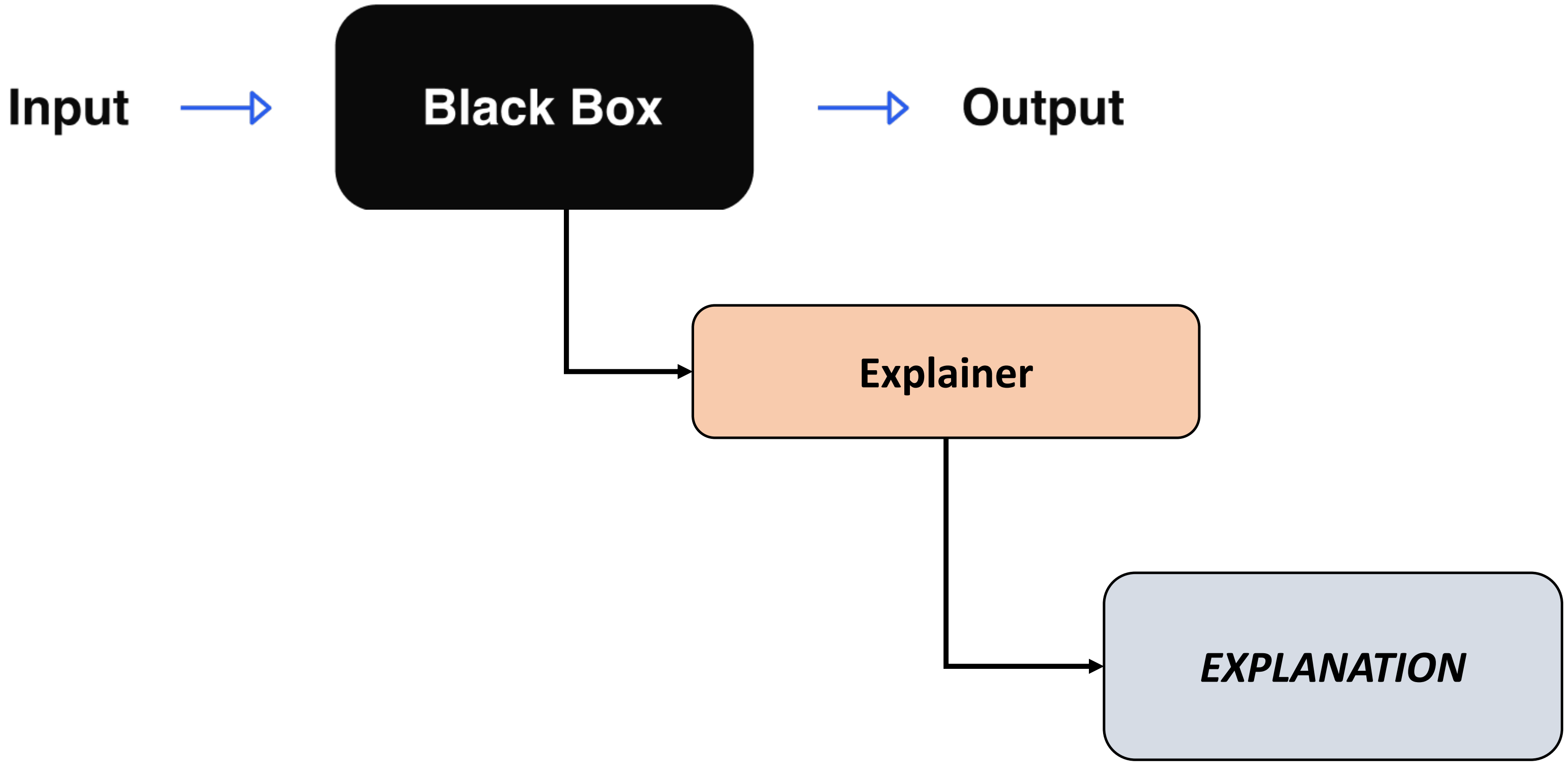
Achiving Model Understanding

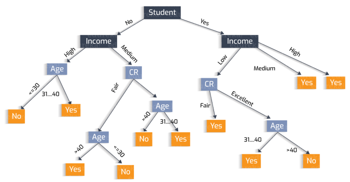
Method 2: ***Explain*** **already-built model in a** ***post-hoc*** **manner**

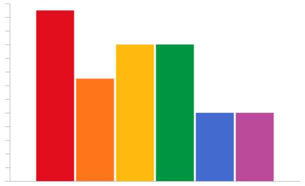


Achiving Model Understanding

Method 2: ***Explain*** already-built model in a ***post-hoc*** manner





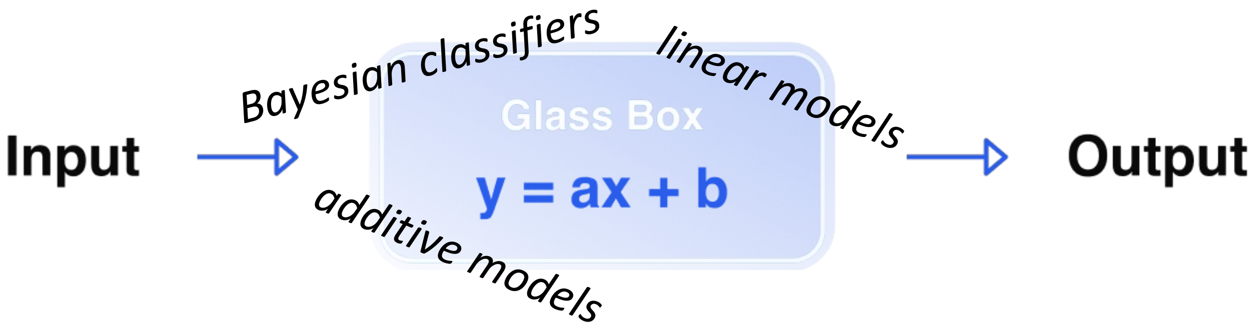


If *Student==Yes:*
  if *Income==High:*
    then prediction *Yes.*
  else if *Income==Medium:*
    then prediction *Yes.*
  else:
    if *CR==Excellent:*
    …..

black box *vs* glass box

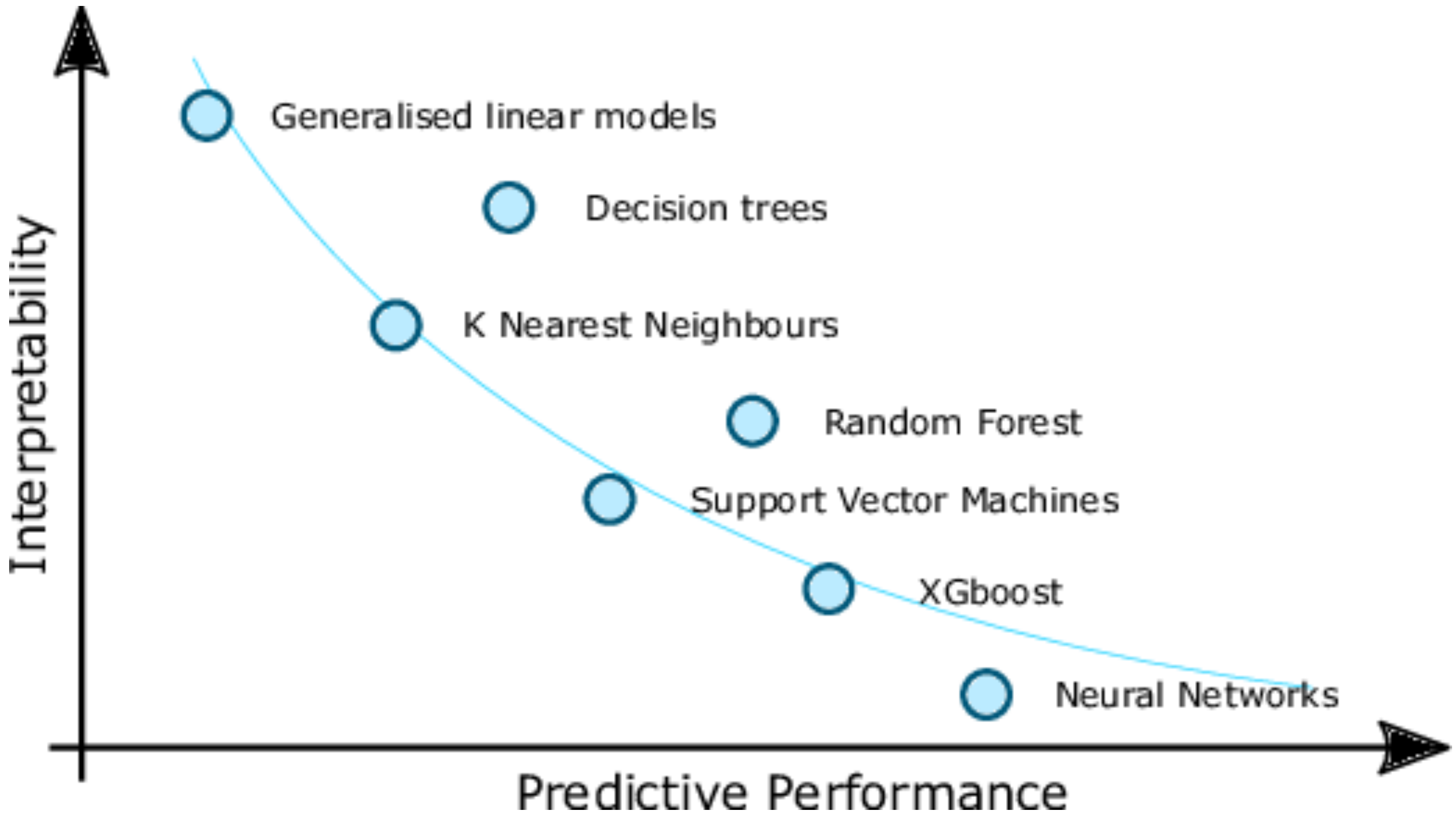
Artificial Intelligence plays a big role in our daily lives. AI is being used everywhere, from our search queries on Google to self-driving vehicles such as Tesla. With the use of deep learning, the models used in these applications have become even more complex. In fact, they are so complex that in many cases we have no idea how these AI models reach their decisions.

Difficult to interpret.
The model structure doesn’t allow explenable reasons for the prediction



Easy to interpret.
The model structure gives explenable reasons for the prediction (ex: the coeffecent of a linear regression)

black box *vs* glass box



Inherently Interpretable Model *vs* Post hoc Explanations

If you can build an easly interpretable model which is **adeguately accurate** for you settings/problem.

DO IT!!!!

If you need a **more complex model** to achive adeguate accuracy, try to use **post hoc explanations**

Interpretation Methods

The various interpretation methods can be roughly differentiated according to their results:

- **Features Importance:** techniques that calculate a score for all the input features for a given model. The scores simply represent the "importance" of each feature.
- **Model Internals:** interpretation of internal components (e.g. parameters, weights). Example are interpretation of intrinsically interpretable models or CNN.
- **Data points**: This category includes all methods that return data points (already existent or newly created) to make a model interpretable. Methods like ***counterfactual explanations*** (similar examples with differences in some features for which the predicted outcome changes in a relevant way) or ***prototypes*** of predicted classes.
- **Intrinsically interpretable model:** interpreting black box models is to approximate them (either globally or locally) with an interpretable model.

Interpretation Methods

The various interpretation methods can be roughly differentiated according to their results:

- **Features Importance:** techniques that calculate a score for all the input features for a given model. The scores simply represent the "importance" of each feature.

- **Model Internals:** interpretation of internal components (e.g. parameters, weights). Example are interpretation of intrinsically interpretable models or CNN.

- **Data points**: This category includes all methods that return data points (already existent or newly created) to make a model interpretable. Methods like ***counterfactual explanations*** (similar examples with differences in some features for which the predicted outcome changes in a relevant way) or ***prototypes*** of predicted classes.

- **Intrinsically interpretable model:** interpreting black box models is to approximate them (either globally or locally) with an interpretable model.

Feature Importance - Introduction

Global

Local

Feature Importance - Introduction

Global → When the method try to explain which is the feature impact on the model predictions (entire model behaviour)

Local → When the method try to explain which is the feature impact on a specific prediction

Feature Importance - Introduction

Global → When the method try to explain which is the feature impact on the model predictions (entire model behaviour)

Local → When the method try to explain which is the feature impact on a specific prediction

---

Model Based →

Model Agnostic →

Feature Importance - Introduction

Global → When the method try to explain which is the feature impact on the model predictions (entire model behaviour)

Local → When the method try to explain which is the feature impact on a specific prediction

---

Model Based → Model-specific interpretation tools are limited to specific model classes. Example: linear regression coefficients or Gini Index

Model Agnostic → Model-agnostic tools can be used on any machine learning model and are applied after the model has been trained

Feature Importance - methods

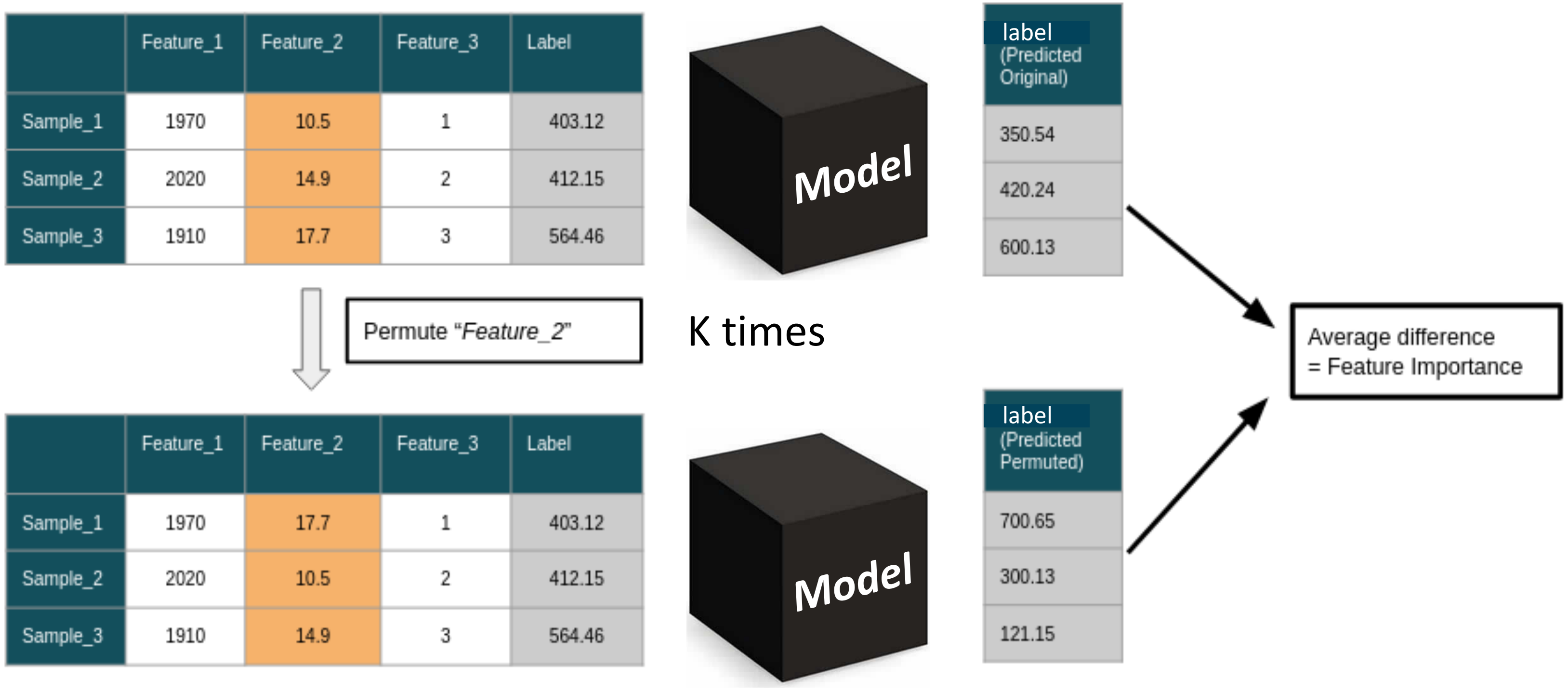
Permutation Importance	Model Agnostic	Global
LIME	Model Agnostic	Local
SHAP	Model Agnostic	Local

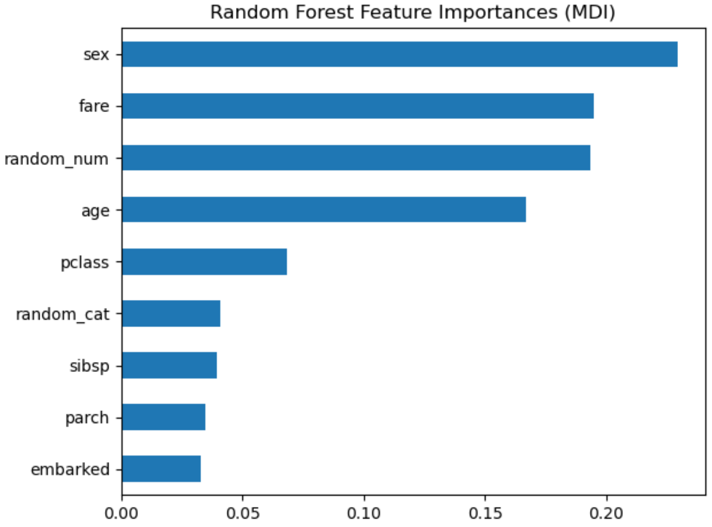
Feature Importance - methods

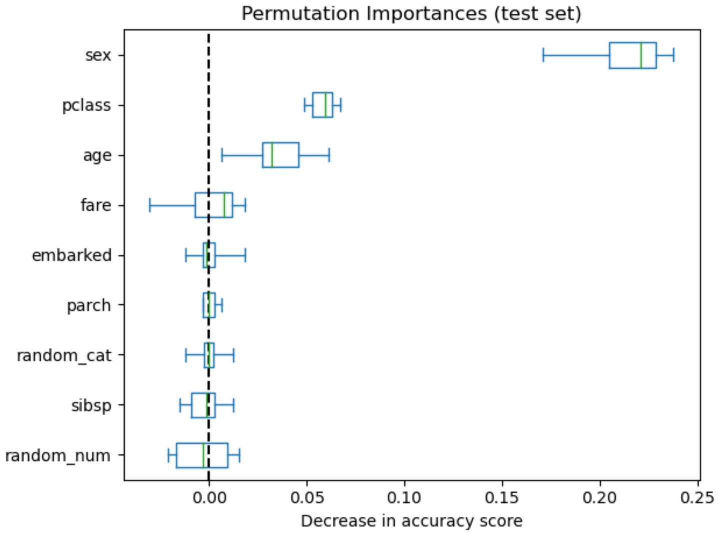
Permutation Importance	Model Agnostic	Global		
LIME	Model Agnostic	Local	*SP-LIME* →	Global
SHAP	Model Agnostic	Local	*aggregating* →	Global

Feature Importance – Permutation Feature Importance

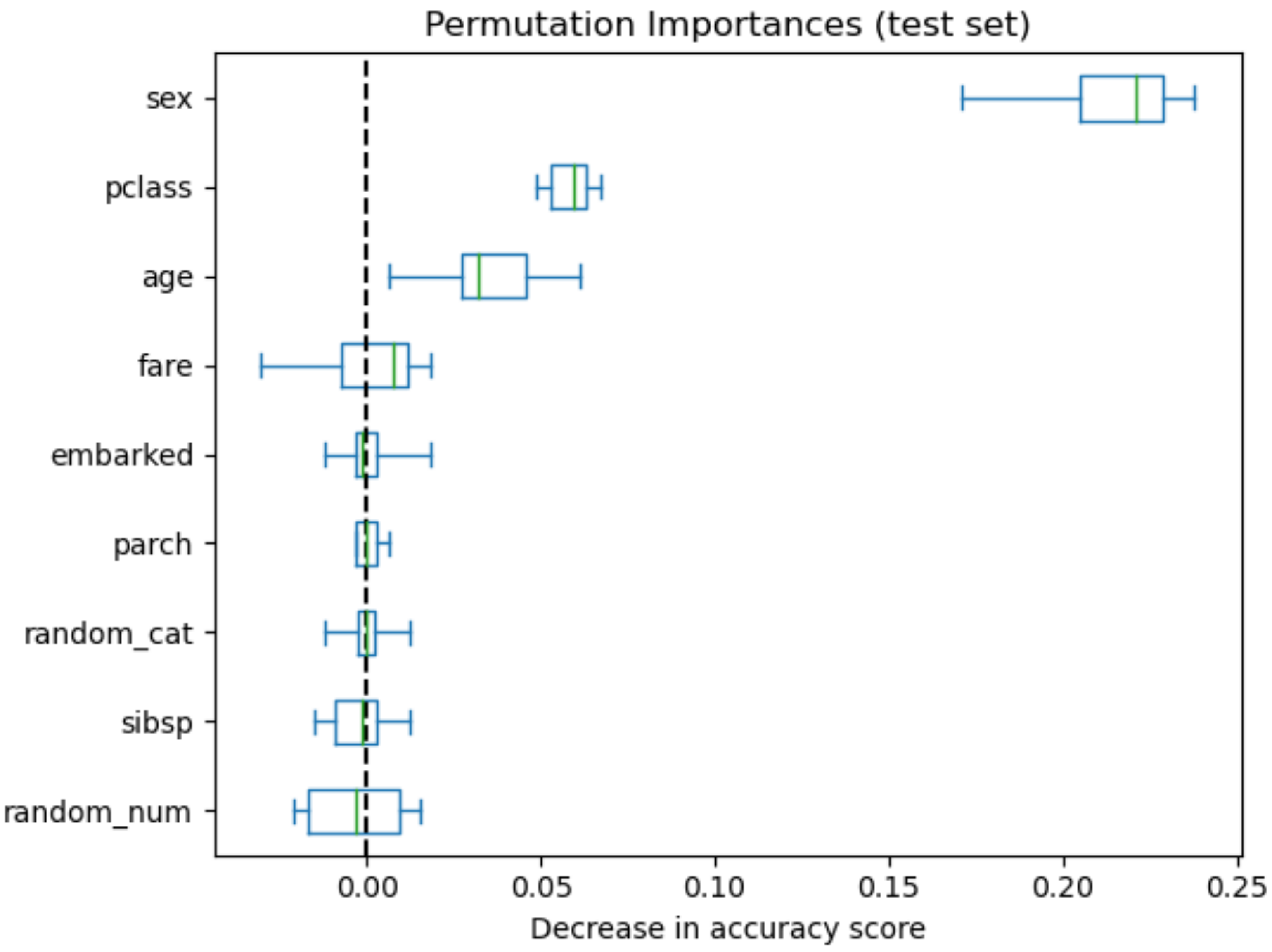
Measure the importance of a feature by calculating the increase in the **model's prediction error** after **permuting** the **feature**. A feature is "important" if shuffling its values increases the model error, because in this case the model relied on the feature for the prediction







Feature Importance – Permutation Feature Importance



Feature Importance – Permutation Feature Importance

Pseudo-code

- Inputs: fitted predictive model m, tabular dataset (training or validation) D.
- Compute the reference score s of the model m on data D (for instance the accuracy for a classifier or the R^2 for a regressor).
- For each feature j (column of D):
  - For each repetition k in $1, \ldots, K$:
    - Randomly shuffle column j of dataset D to generate a corrupted version of the data named $\tilde{D}_{k,j}$.
    - Compute the score $s_{k,j}$ of model m on corrupted data $\tilde{D}_{k,j}$.
  - Compute importance i_j for feature f_j defined as:

$$i_j = s - \frac{1}{K} \sum_{k=1}^{K} s_{k,j}$$

```
from sklearn.inspection import permutation_importance
```

Criticism

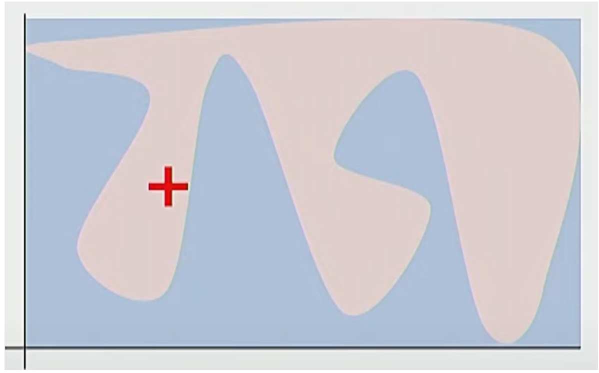
When ***two features are correlated*** and one of the features is permuted, the model will still have access to the feature through its correlated feature. This will result in a lower importance value for both features, where they might *actually* be important.

Feature Importance – LIME

Model Agnostic

Local

LIME = Local Interpretable Model-agnostic Explanations
Try to fit a simple linear model locally



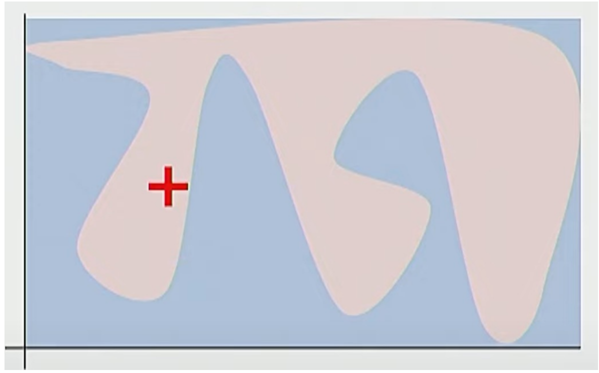
We would like to get features importance for a point x_i:

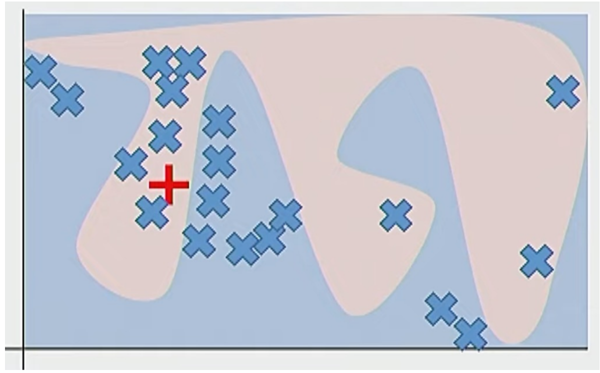
Feature Importance – LIME

Model Agnostic

Local

LIME = Local Interpretable Model-agnostic Explanations
Try to fit a simple linear model locally





We would like to get features importance for a point x_i:

1. Generate random sample points around x_i

Feature Importance – LIME

Model Agnostic

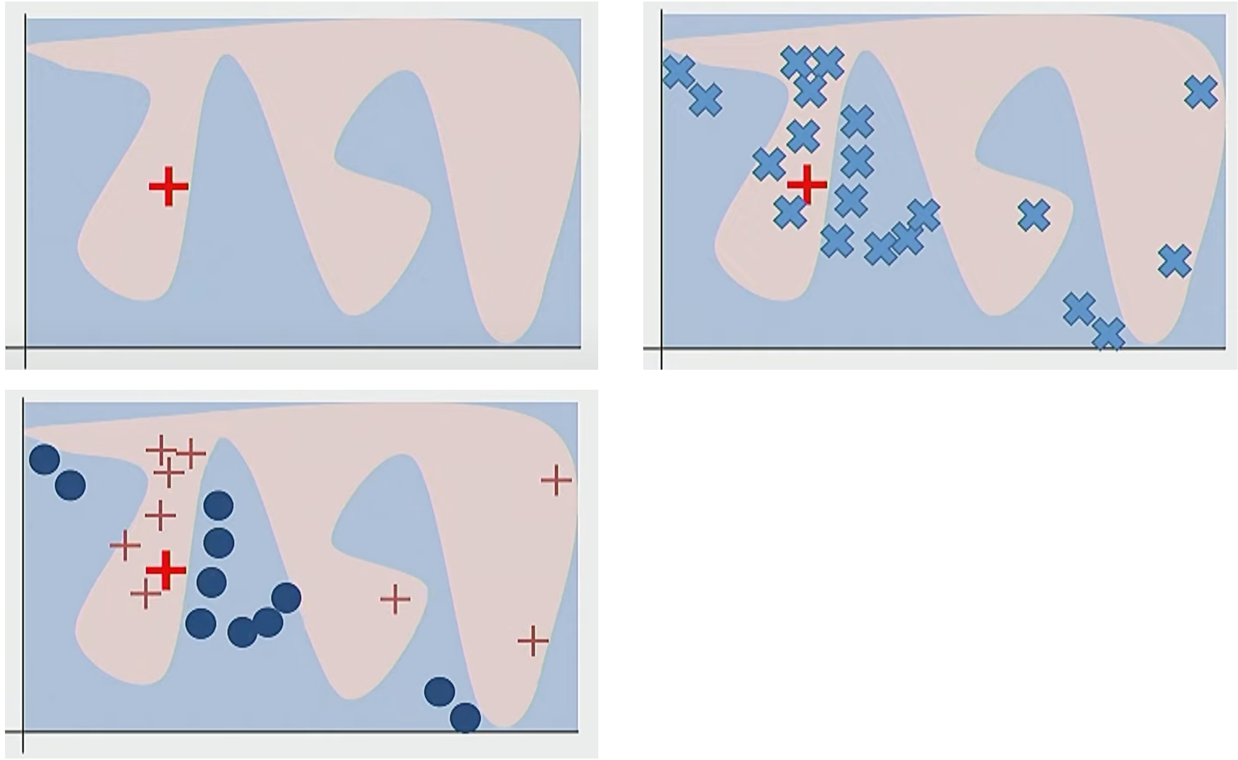
Local

LIME = Local Interpretable Model-agnostic Explanations
Try to fit a simple linear model locally

We would like to get features importance for a point x_i:

1. Generate random sample points around x_i

2. Use Model to predict each generated data point



Feature Importance – LIME

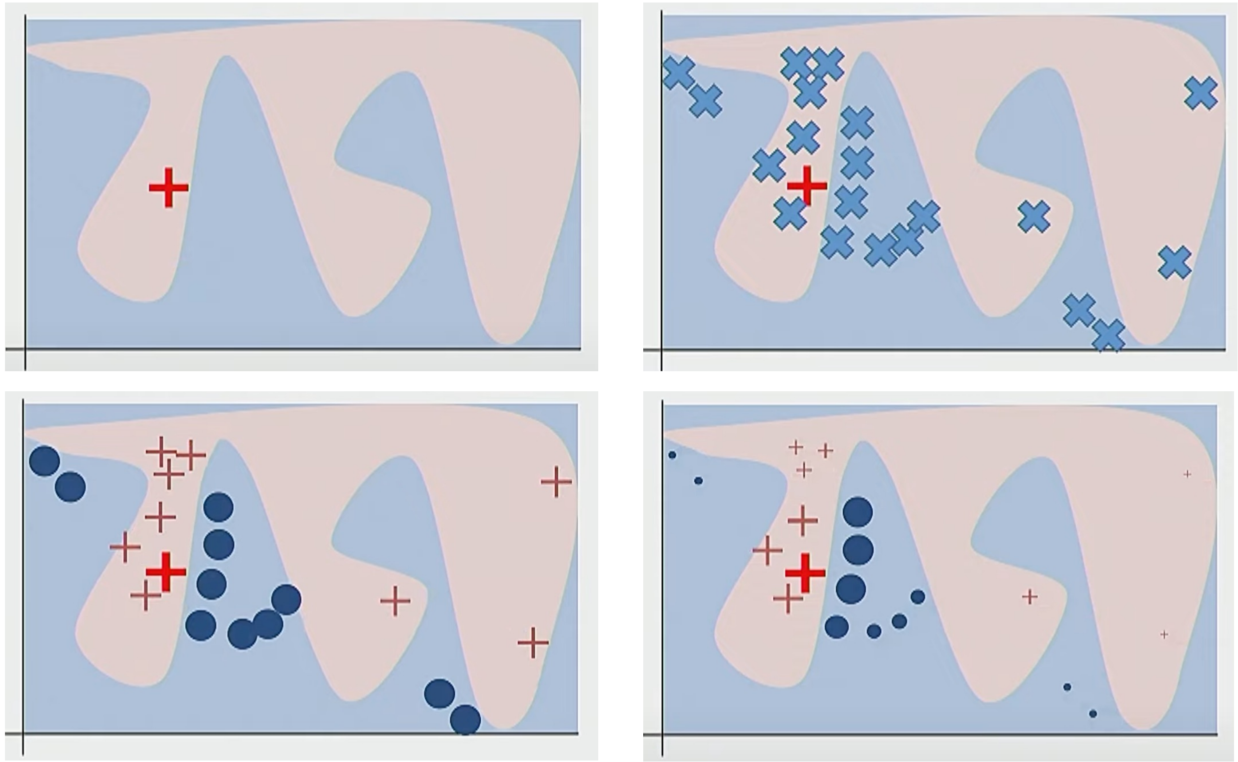
Model Agnostic

Local

LIME = Local Interpretable Model-agnostic Explanations
Try to fit a simple linear model locally

We would like to get features importance for a point x_i:

1. Generate random sample points around x_i
2. Use Model to predict each generated data point
3. Weight samples according to distance from x_i



Feature Importance – LIME

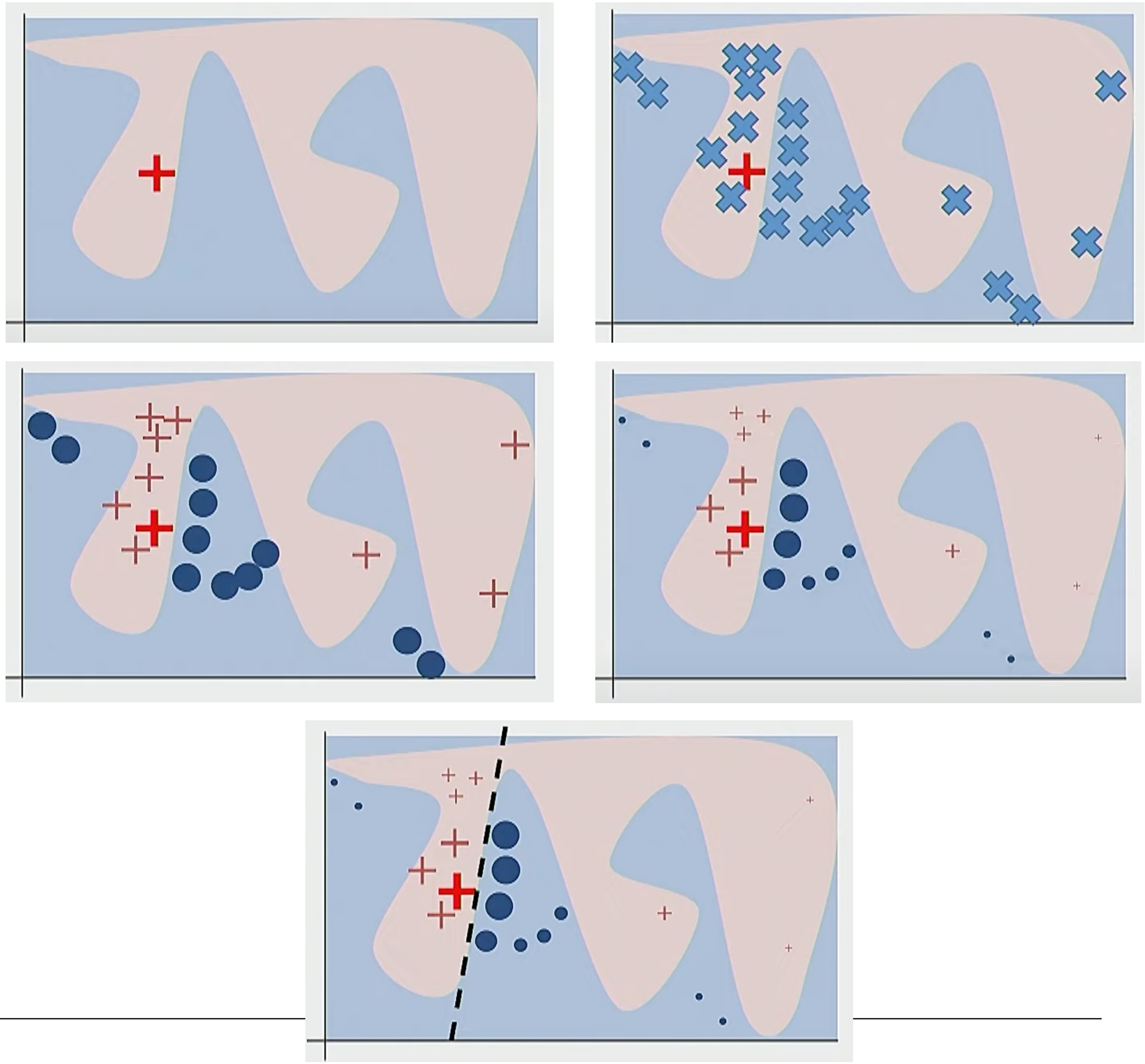
Model Agnostic

Local

LIME = Local Interpretable Model-agnostic Explanations
Try to fit a simple linear model locally

We would like to get features importance for a point x_i:

1. Generate random sample points around x_i
2. Use Model to predict each generated data point
3. Weight samples according to distance from x_i
4. Learn a simple weighted linear model on samples



Feature Importance – LIME

Model Agnostic

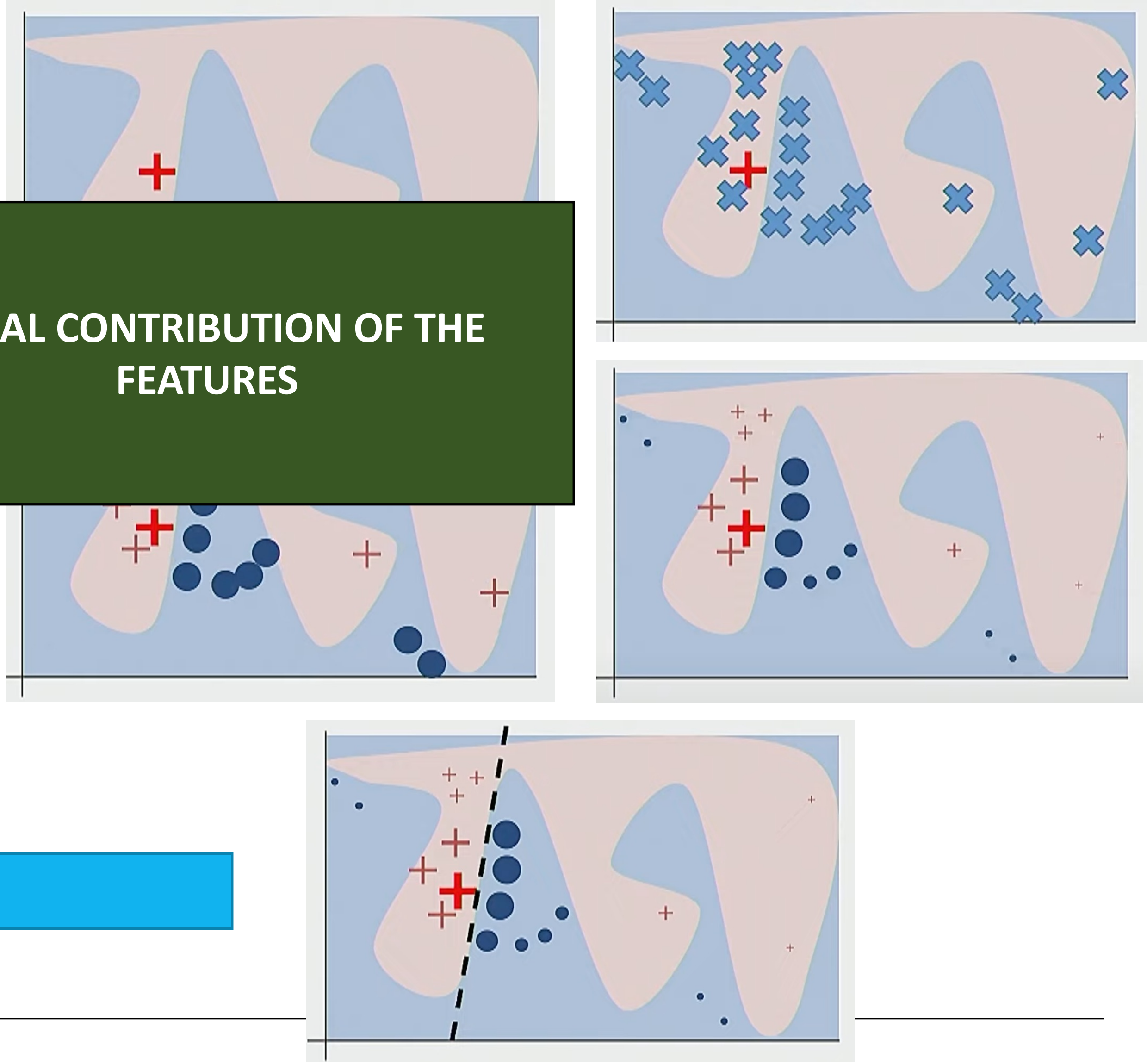
Local

LIME = Local Interpretable Model-agnostic Explanations
Try to fit a simple linear model locally

We would like to get features importance for a poin

1. Generate random sample points around x_i
2. Use Model to predict each generated data point
3. Weight samples according to distance from x_i
4. Learn a simple weighted linear model on samples

LOCAL CONTRIBUTION OF THE FEATURES



Dependent Variable

Population Y intercept

Population Slope Coefficient

Independent Variable

Random Error term

$$Y_i = \beta_0 + \boxed{\beta_1 X_i} + \varepsilon_i$$

$$Ln\left(\frac{P}{1-P}\right) = \beta_0 + \boxed{\beta_1 X_1} + \boxed{\beta_2 X_2} + \ldots + \boxed{\beta_k X_k}$$

Feature Importance – LIME

Model Agnostic

Local

Advantages:

- When using Lasso or short trees, the resulting **explanations are short** (= selective) and possibly contrastive. Therefore, they make **human-friendly explanations**.
- The **fidelity measure** (how well the interpretable model approximates the black box predictions) gives us a good idea of how reliable the interpretable model is in explaining the black box predictions in the neighborhood of the data instance of interest.
- The explanations created with local surrogate models **can use other (interpretable) features than the original model was trained on.**.

Disadvantages:

- The correct **definition** of the **neighborhood** is a very big, **unsolved** problem when using LIME with tabular data.

Feature Importance – SHAP

Model Agnostic

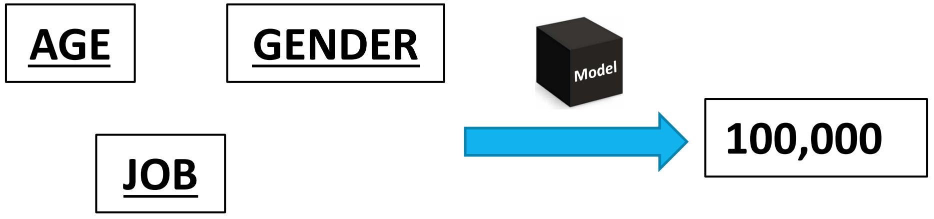
Local

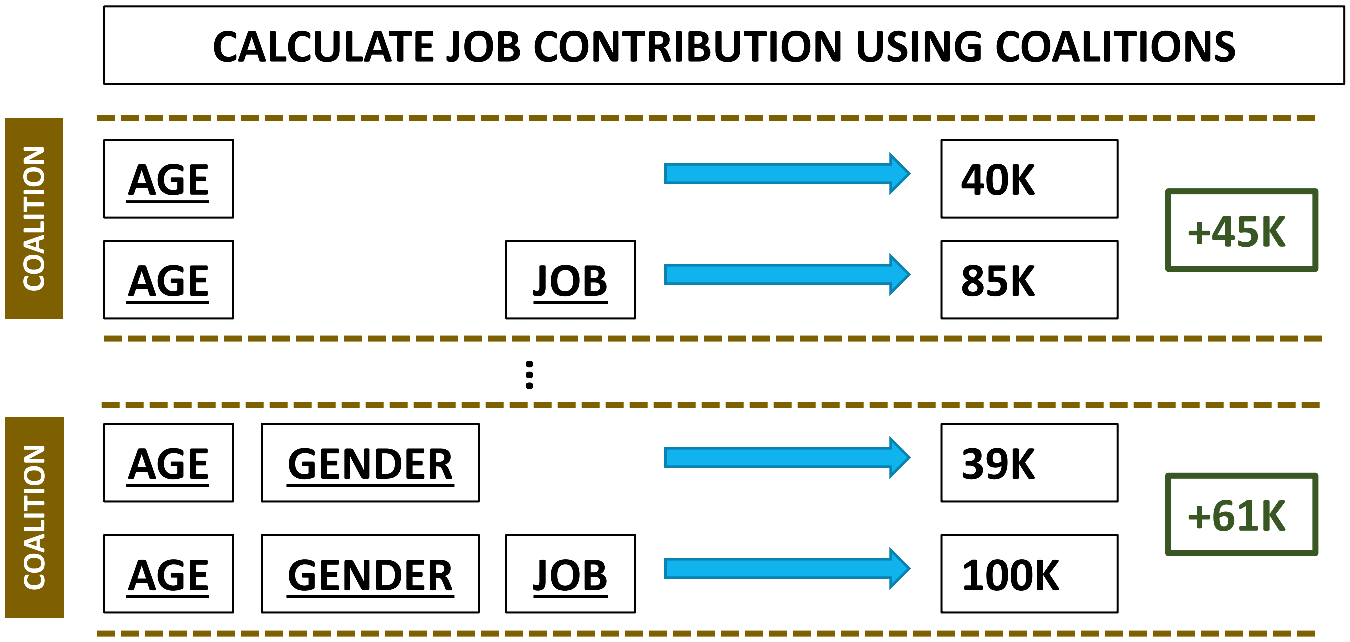
Shapley Values:

Shapley values – a method from coalitional game theory – tells us how to fairly distribute the "payout" among the features. Are based on the idea that the outcome of each possible combination (or coalition) of players should be considered to determine the importance of a single player.

Example:

You have trained a model that predicts the income of a person knowing **age**, **gender** and **job** of the person.





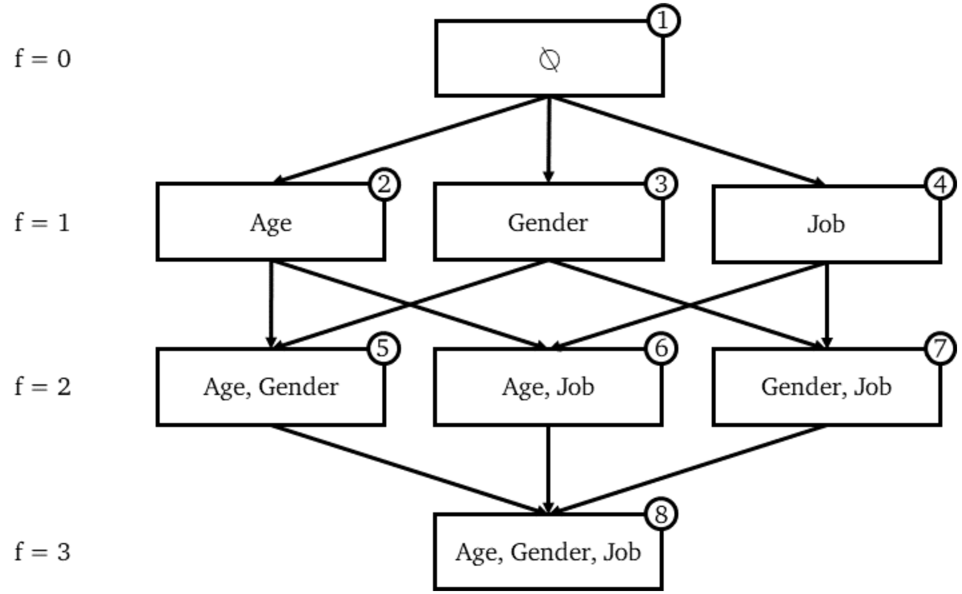
How much has **each feature value contributed** to the **prediction** compared to the average prediction?

Feature Importance – SHAP

Model Agnostic

Local

Shaply values

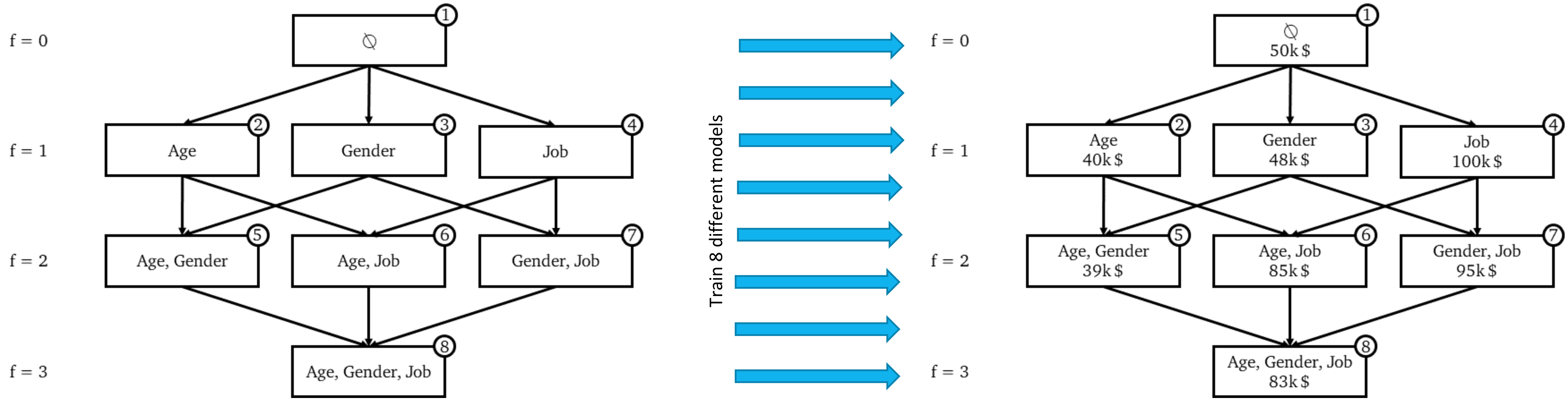


Feature Importance – SHAP

Model Agnostic

Local

Shaply values

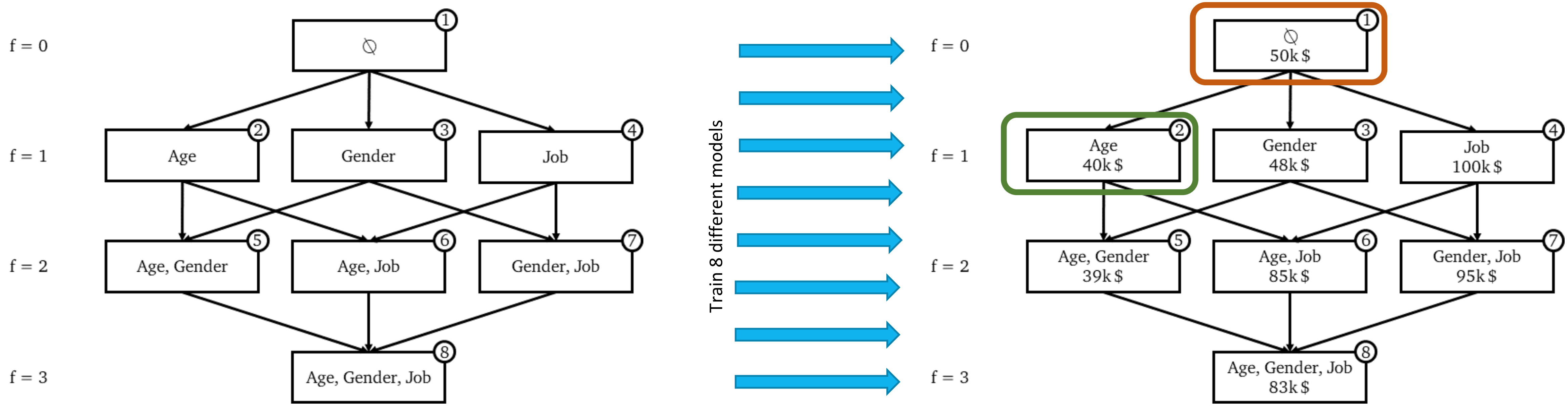


Feature Importance – SHAP

Model Agnostic

Local

Shaply values



Each edge represents the marginal contribution brought by a feature to a model.

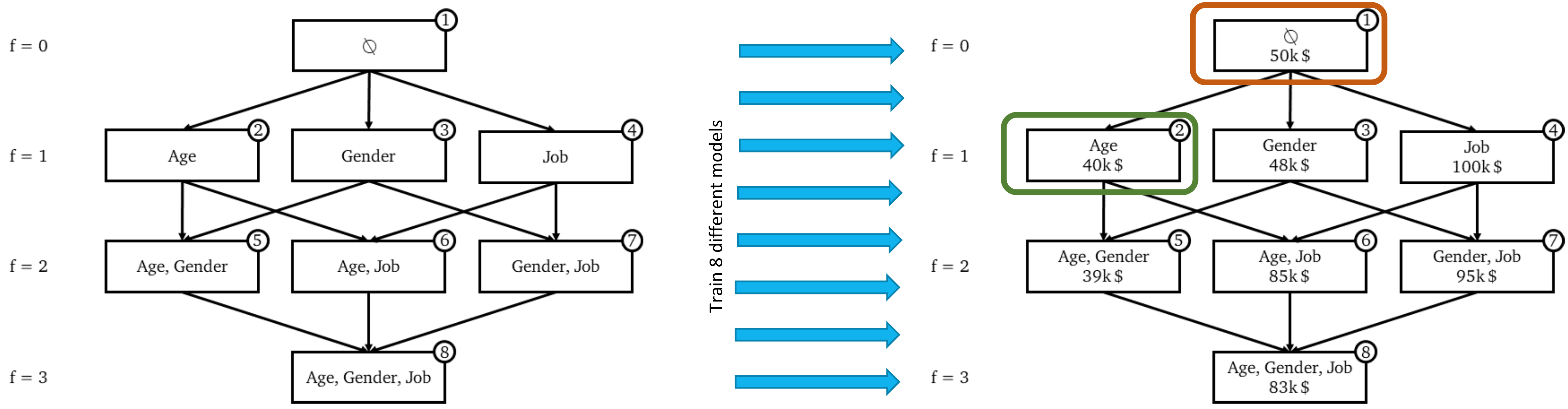
Which is the contribution of feature AGE from (1) to (2)

Feature Importance – SHAP

Shaply values

Model Agnostic

Local



Each edge represents the marginal contribution brought by a feature to a model.

Which is the contribution of feature AGE from (1) to (2)

$$MC_{Age,\{Age\}}(x_0) = \boxed{Predict_{\{Age\}}(x_0)} - \boxed{Predict_{\varnothing}(x_0)} = 40k\$ - 50k\$ = -10k\$$$

$$MC_{Age,\{Age,Gender\}}(x_0)$$

$$MC_{Age,\{Age,Job\}}(x_0)$$

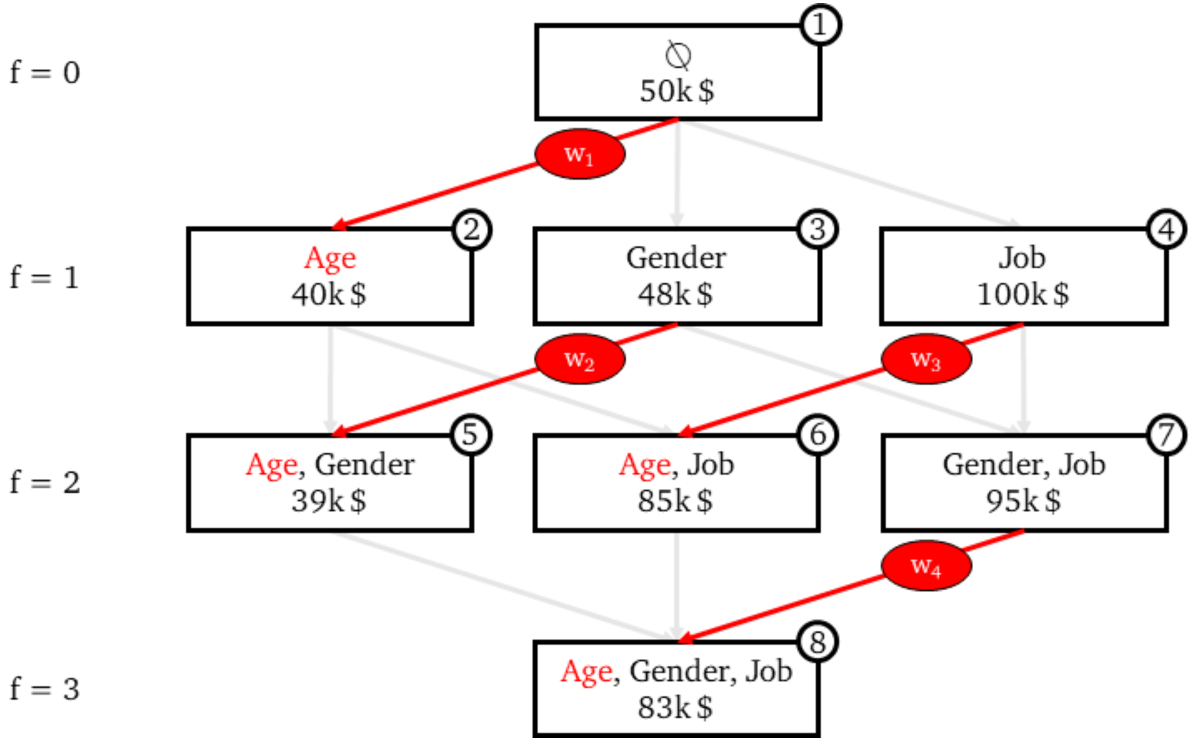
$$MC_{Age,\{Age,Gender,Job\}}(x_0)$$

Feature Importance – SHAP

Model Agnostic

Local

Shaply values



$$
\begin{aligned}
SHAP_{Age}(x_0) = \; & w_1 \times MC_{Age,\{Age\}}(x_0) + \\
& w_2 \times MC_{Age,\{Age,Gender\}}(x_0) + \\
& w_3 \times MC_{Age,\{Age,Job\}}(x_0) + \\
& w_4 \times MC_{Age,\{Age,Gender,Job\}}(x_0)
\end{aligned}
$$

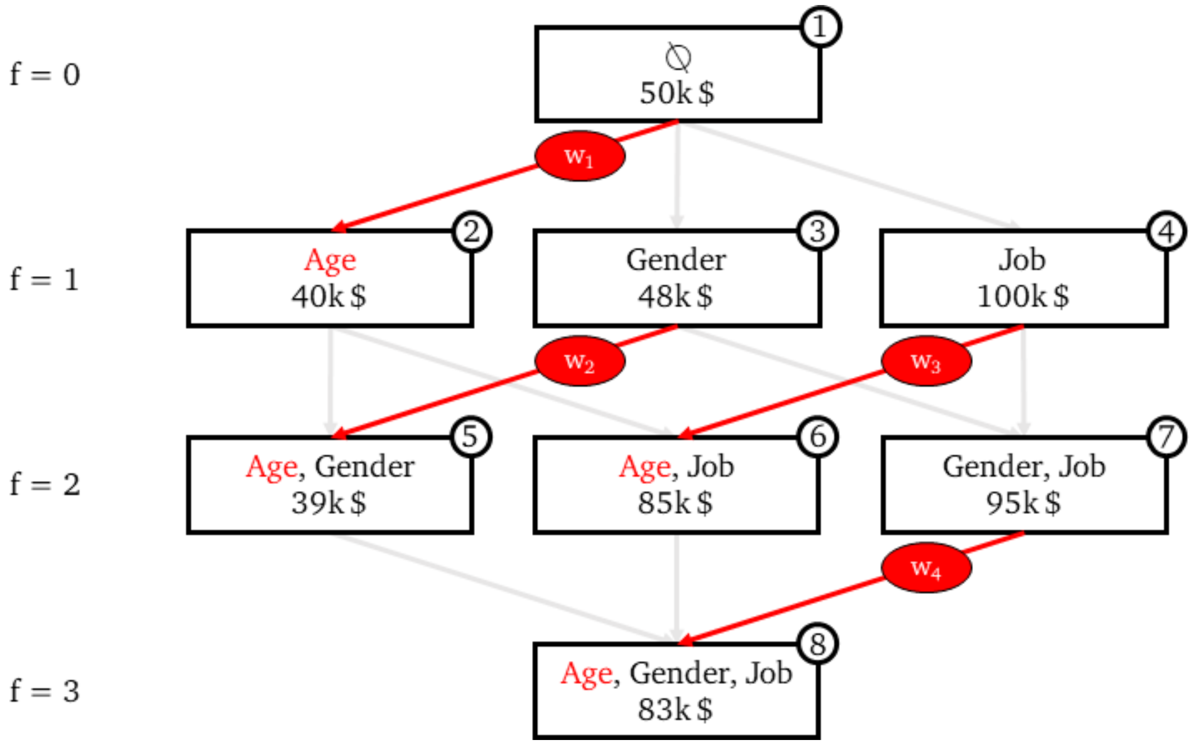
CGnal

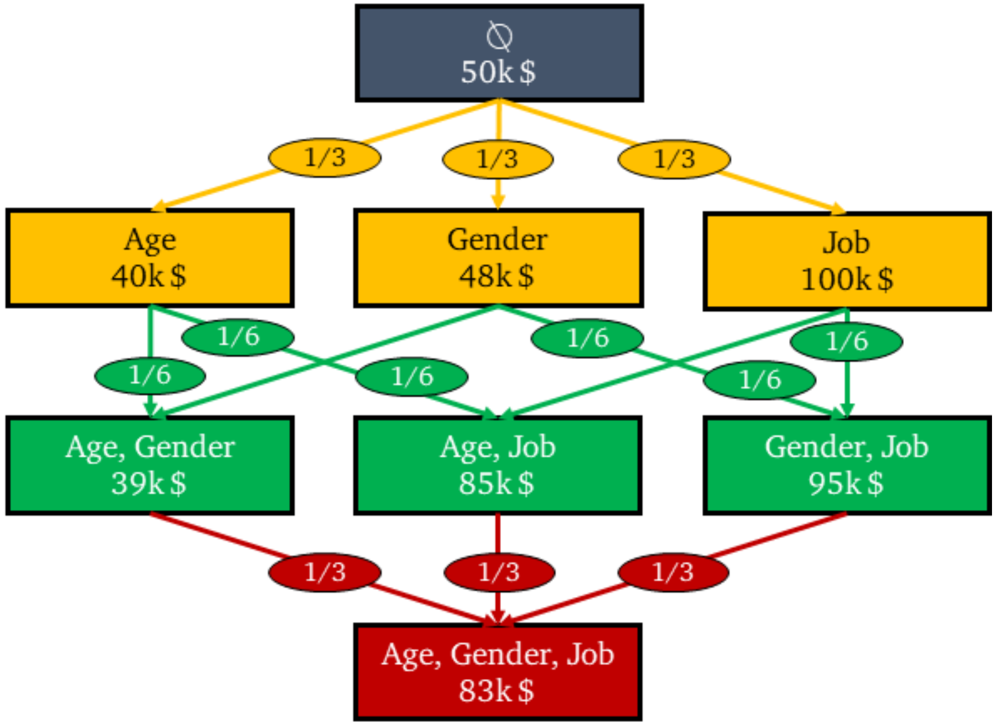
Feature Importance – SHAP

Model Agnostic

Local

Shaply values





$$
\begin{aligned}
SHAP_{Age}(x_0) = \; & w_1 \times MC_{Age,\{Age\}}(x_0) + \\
& w_2 \times MC_{Age,\{Age,Gender\}}(x_0) + \\
& w_3 \times MC_{Age,\{Age,Job\}}(x_0) + \\
& w_4 \times MC_{Age,\{Age,Gender,Job\}}(x_0)
\end{aligned}
$$

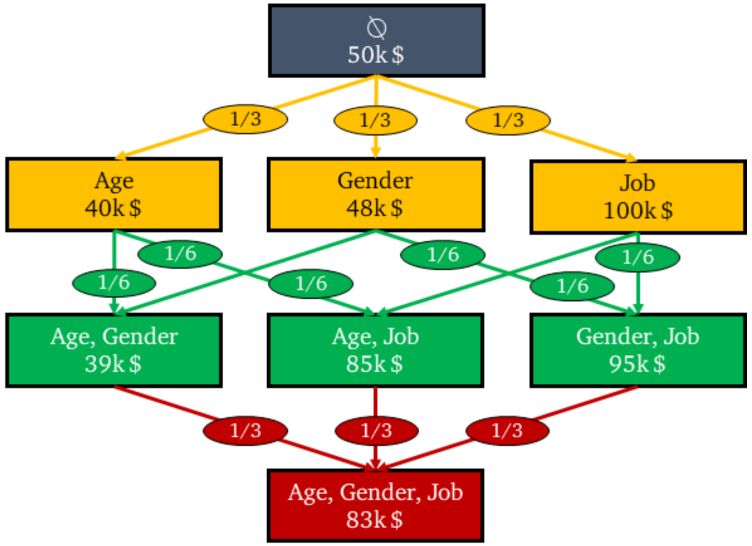
$$
\begin{aligned}
SHAP_{Age}(x_0) = \; & [(1\times\binom{3}{1})]^{-1}\times MC_{Age,\{Age\}}(x_0) \; + \\
& [(2\times\binom{3}{2})]^{-1}\times MC_{Age,\{Age,Gender\}}(x_0) \; + \\
& [(2\times\binom{3}{2})]^{-1}\times MC_{Age,\{Age,Job\}}(x_0) \; + \\
& [(3\times\binom{3}{3})]^{-1}\times MC_{Age,\{Age,Gender,Job\}}(x_0) \; + \\
= \; & \frac{1}{3}\times(-10k\$) + \frac{1}{6}\times(-9k\$) + \frac{1}{6}\times(-15k\$) + \frac{1}{3}\times(-12k\$) \\
= \; & -11.33k\$
\end{aligned}
$$

Feature Importance – SHAP

Model Agnostic

Local

Shaply values



$$
\begin{aligned}
SHAP_{Age}(x_0) = & [(1\times\binom{3}{1})]^{-1}\times MC_{Age,\{Age\}}(x_0) \; + \\
& [(2\times\binom{3}{2})]^{-1}\times MC_{Age,\{Age,Gender\}}(x_0) \; + \\
& [(2\times\binom{3}{2})]^{-1}\times MC_{Age,\{Age,Job\}}(x_0) \; + \\
& [(3\times\binom{3}{3})]^{-1}\times MC_{Age,\{Age,Gender,Job\}}(x_0) \; + \\
= & \frac{1}{3}\times(-10k\$) + \frac{1}{6}\times(-9k\$) + \frac{1}{6}\times(-15k\$) + \frac{1}{3}\times(-12k\$) \\
= & -11.33k\$
\end{aligned}
$$

$$g(z') = \phi_0 + \sum_{j=1}^{M} \phi_j z'_j$$

where g is the explanation model, $z' \in \{0,1\}^M$ is the coalition vector, M is the maximum coalition size and $\phi_j \in \mathbb{R}$ is the feature attribution for a feature j, the Shapley values. What I call “coalition vector” is called “simplified features” in the SHAP paper.

Feature Importance – SHAP

Model Agnostic

Local

```
pip install shap
```

KernelSHAP: an alternative, kernel-based estimation approach for Shapley values inspired by LIME.

```
from shap import KernelExplainer
```

TreeSHAP: an efficient estimation approach for tree-based models like ***decision trees***, ***random forest*** and ***gradient boosted trees***.

```
from shap import TreeExplainer
```

Feature Importance – SHAP

Model Agnostic

Local

```
pip install shap
```

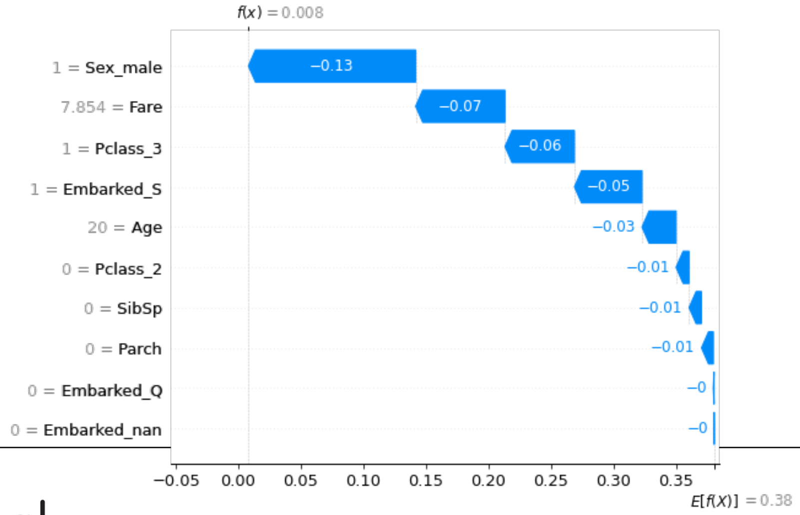
KernelSHAP: an alternative, kernel-based estimation approach for Shapley values inspired by LIME.

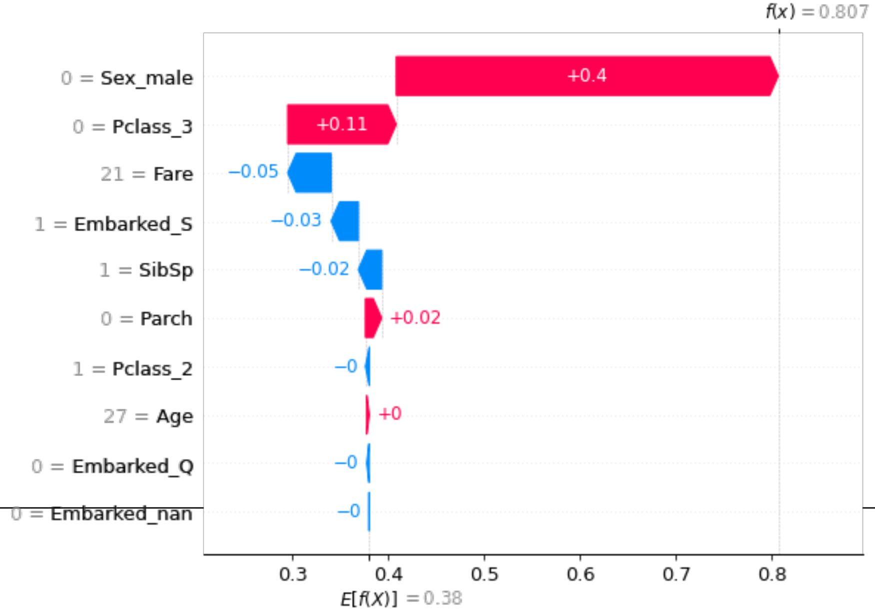
```
from shap import KernelExplainer
```

TreeSHAP: an efficient estimation approach for tree-based models like ***decision trees***, ***random forest*** and ***gradient boosted trees***.

```
from shap import TreeExplainer
```

SHAP VALUES VISUALIZATION





$$\sum_{j=1}^{M} \phi_j z'_j$$

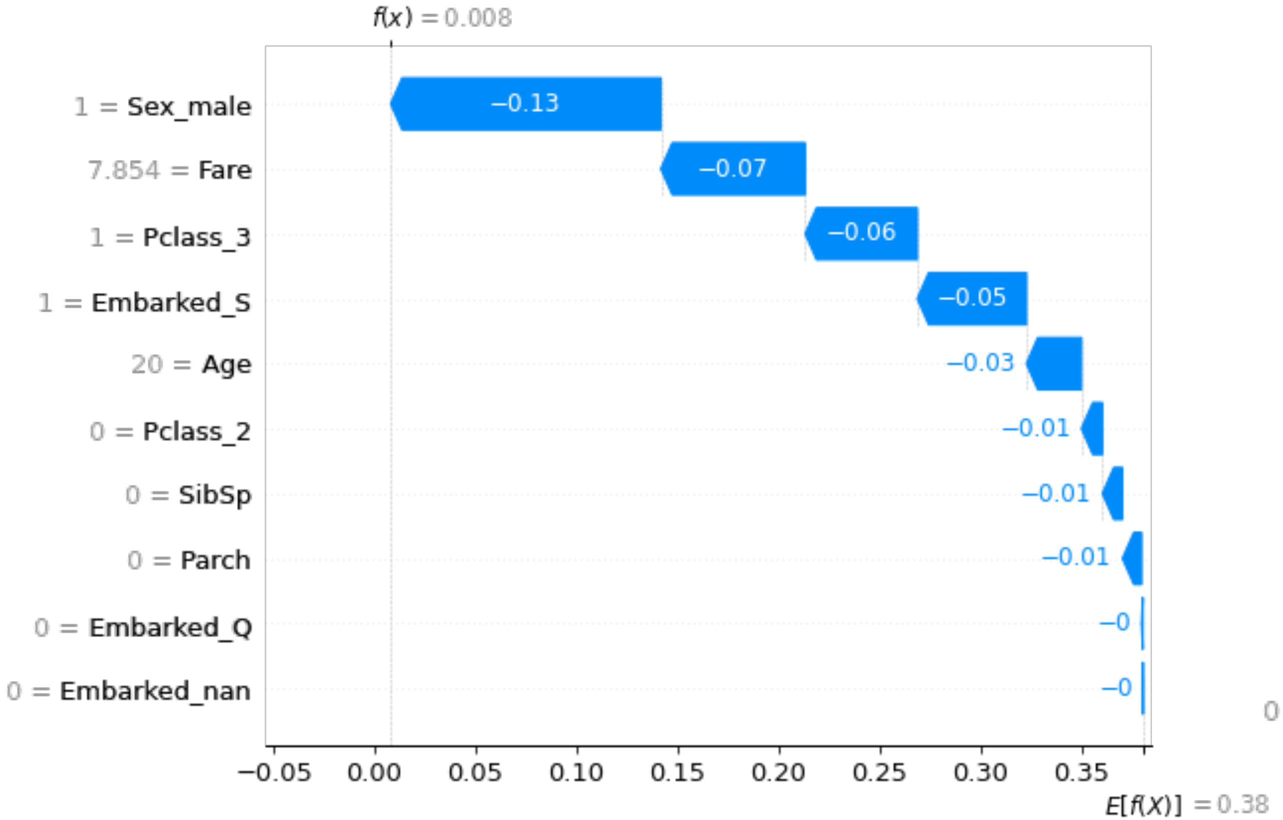
Feature Importance – SHAP

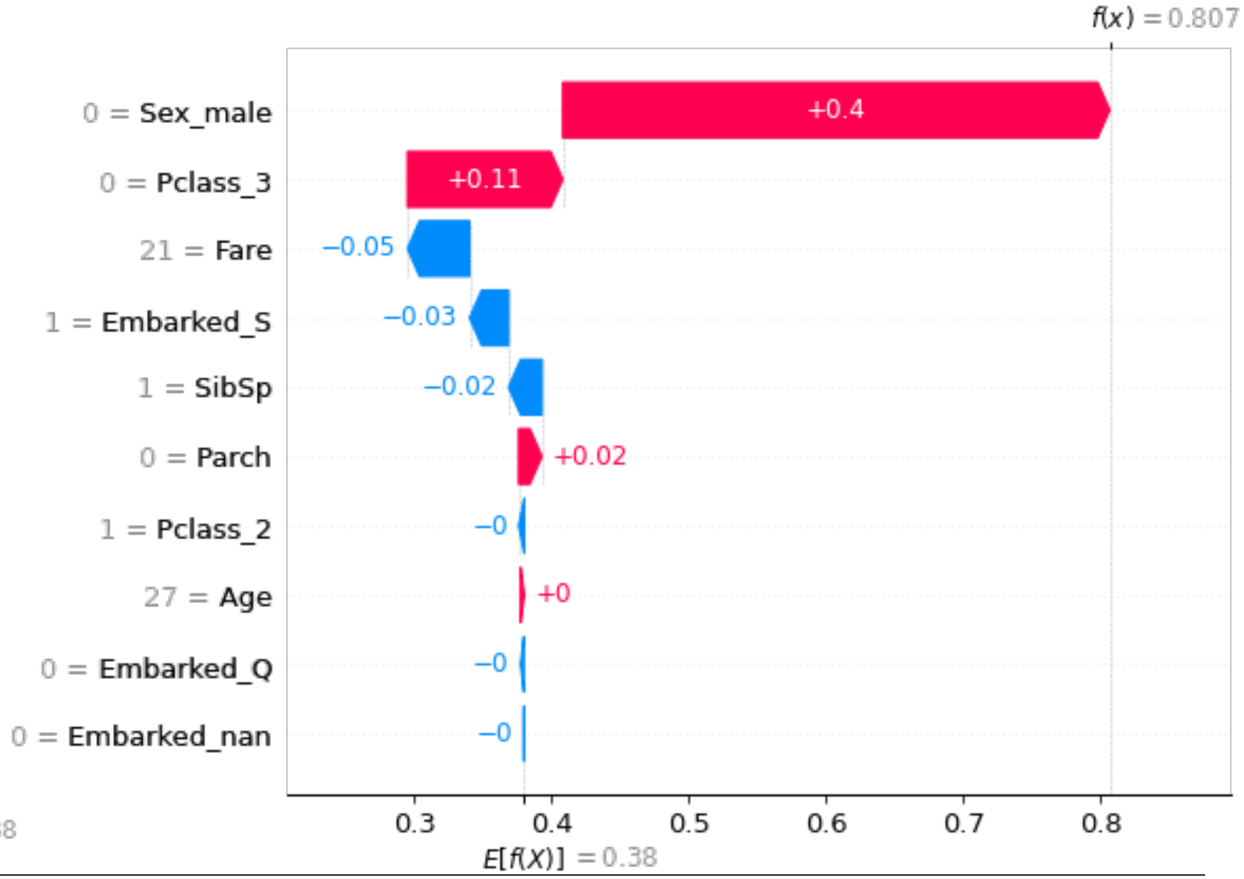
Model Agnostic

Local

$$\sum_{j=1}^{M} \phi_j z'_j$$

SHAP explanation force plots for two different samples from the titanic dataset:





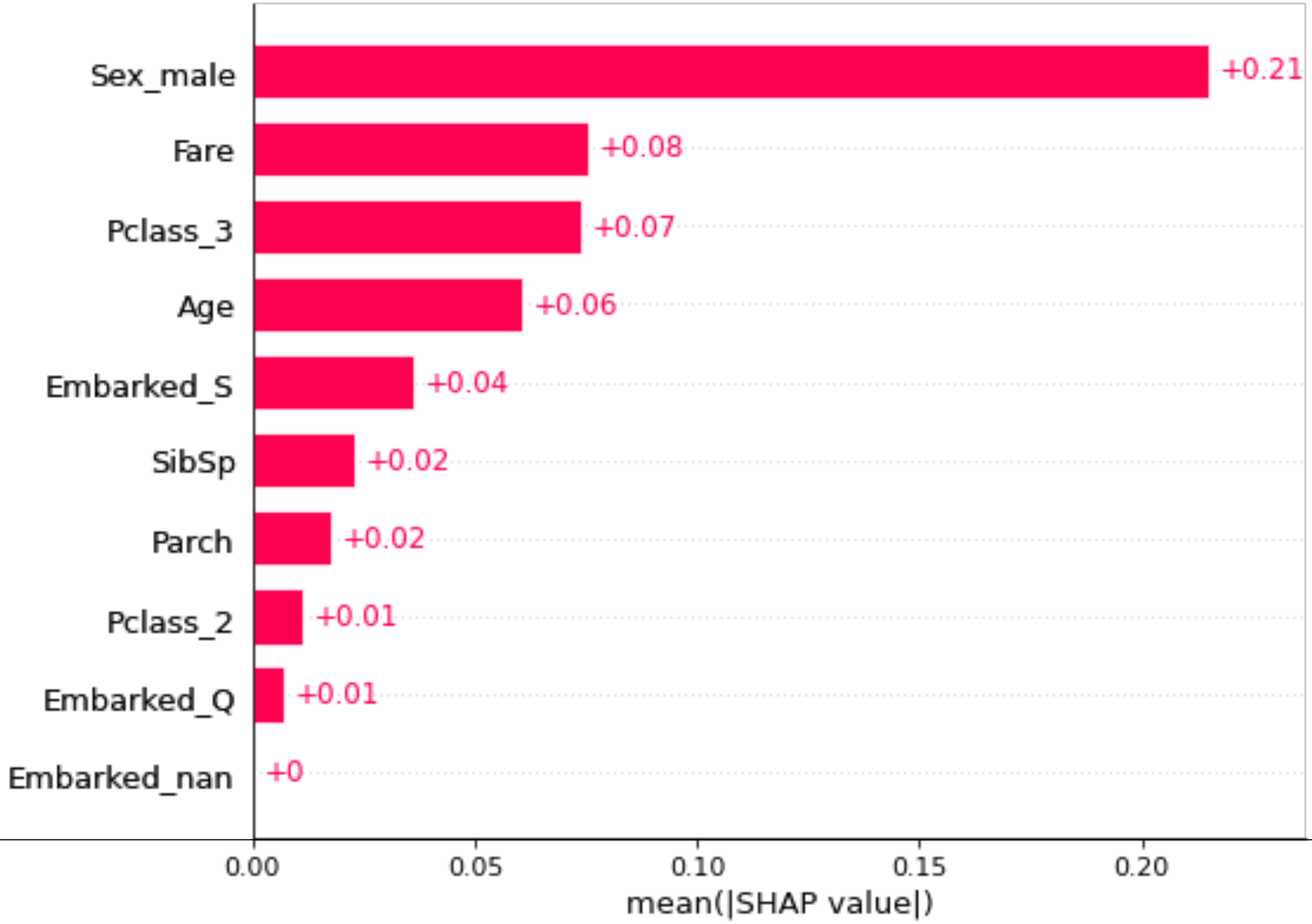
Feature Importance – SHAP

Model Agnostic | Local → *aggregating* → Global

The idea behind SHAP feature importance is simple: Features with large absolute Shapley values are important. Since we want the global importance, we average the absolute Shapley values per feature *j* across the data:

$$I_j = \frac{1}{n}\sum_{i=1}^{n}|\phi_j^{(i)}|$$

Where *n* is the number of samples

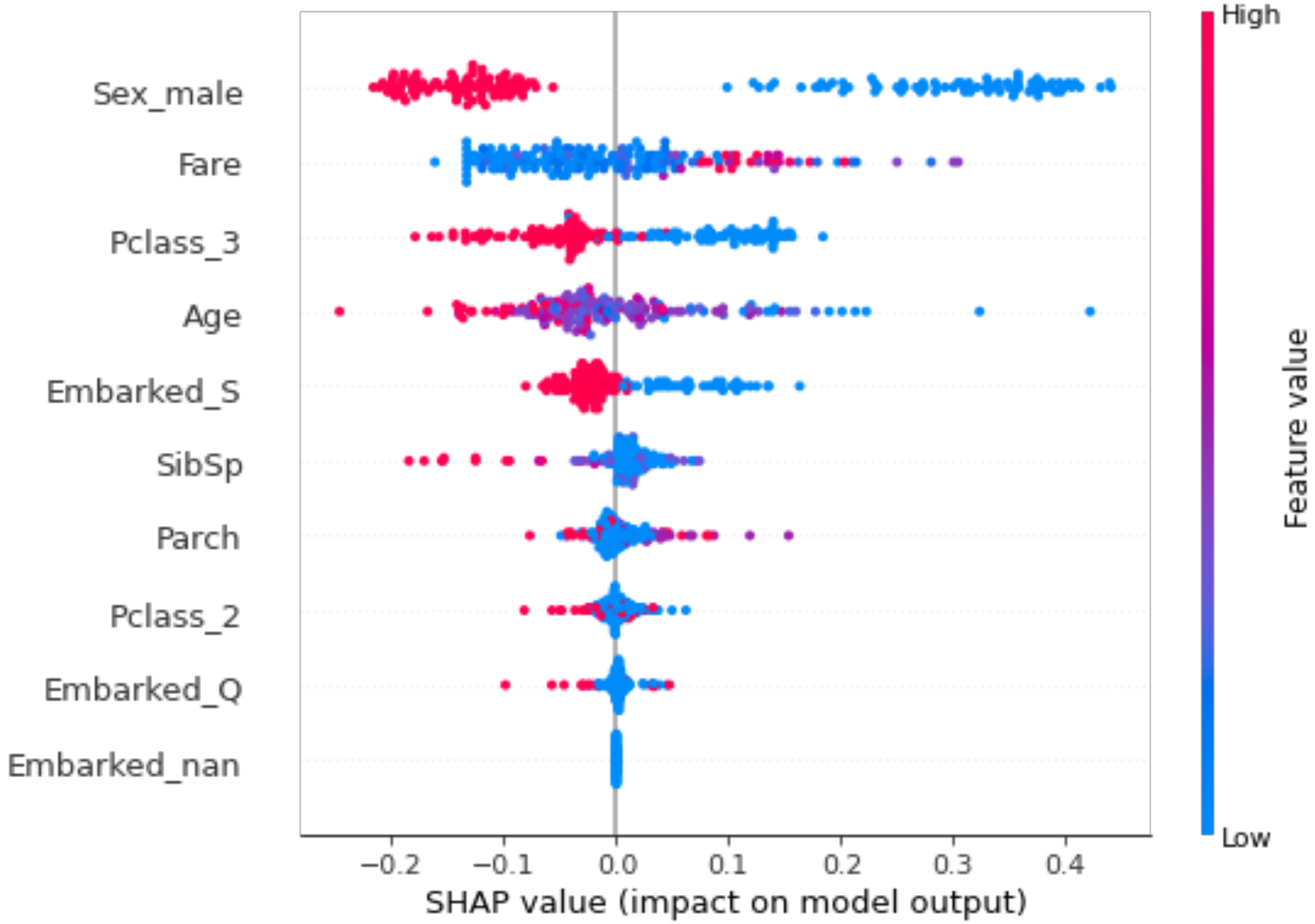


Feature Importance – SHAP

Model Agnostic | Local *aggregating* → Global

SHAP Summary Plot

The summary plot combines feature importance with feature effects.



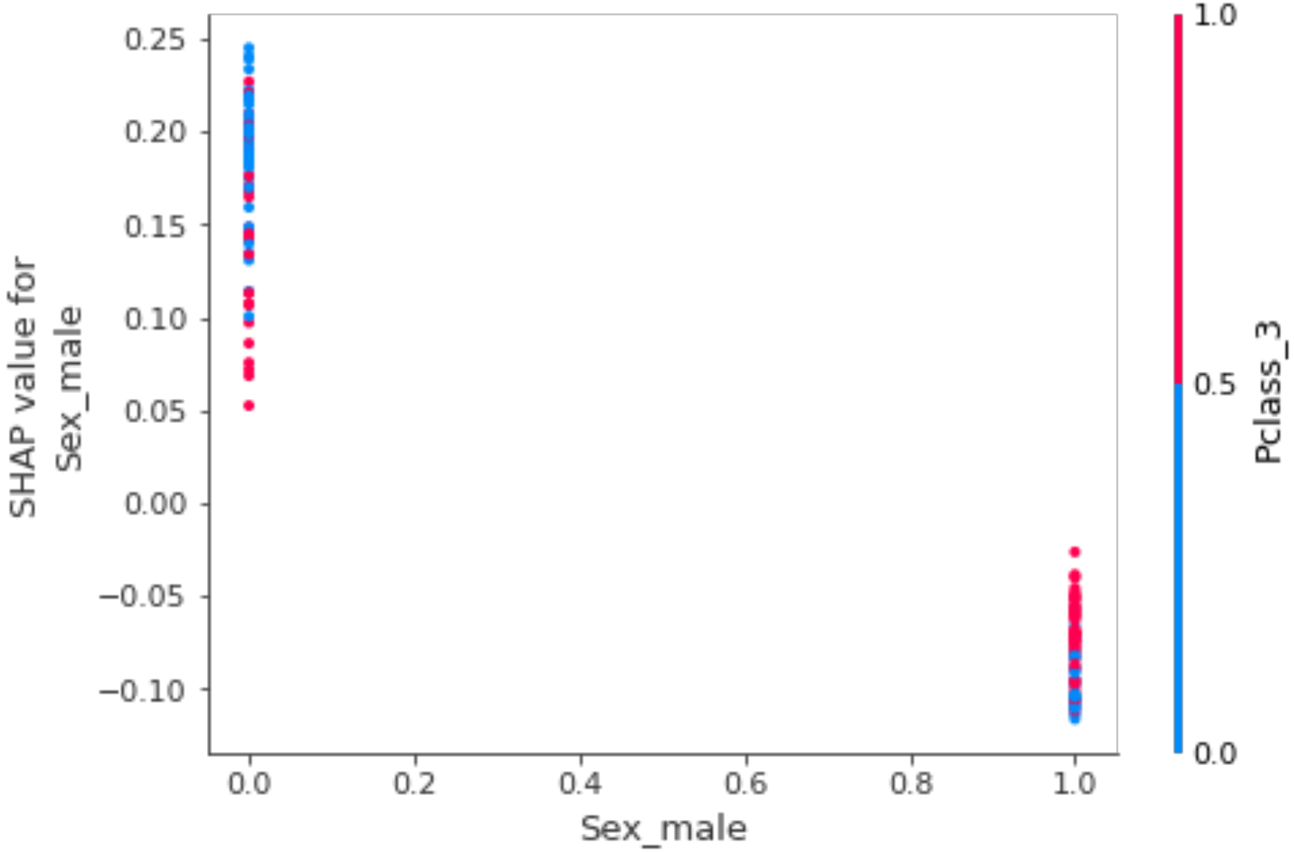
Feature Importance – SHAP

Model Agnostic | Local → *aggregating* → Global

SHAP – dependence plot

The dependecy plot combines feature importance with feature value.

It is possible to visualize also interaction



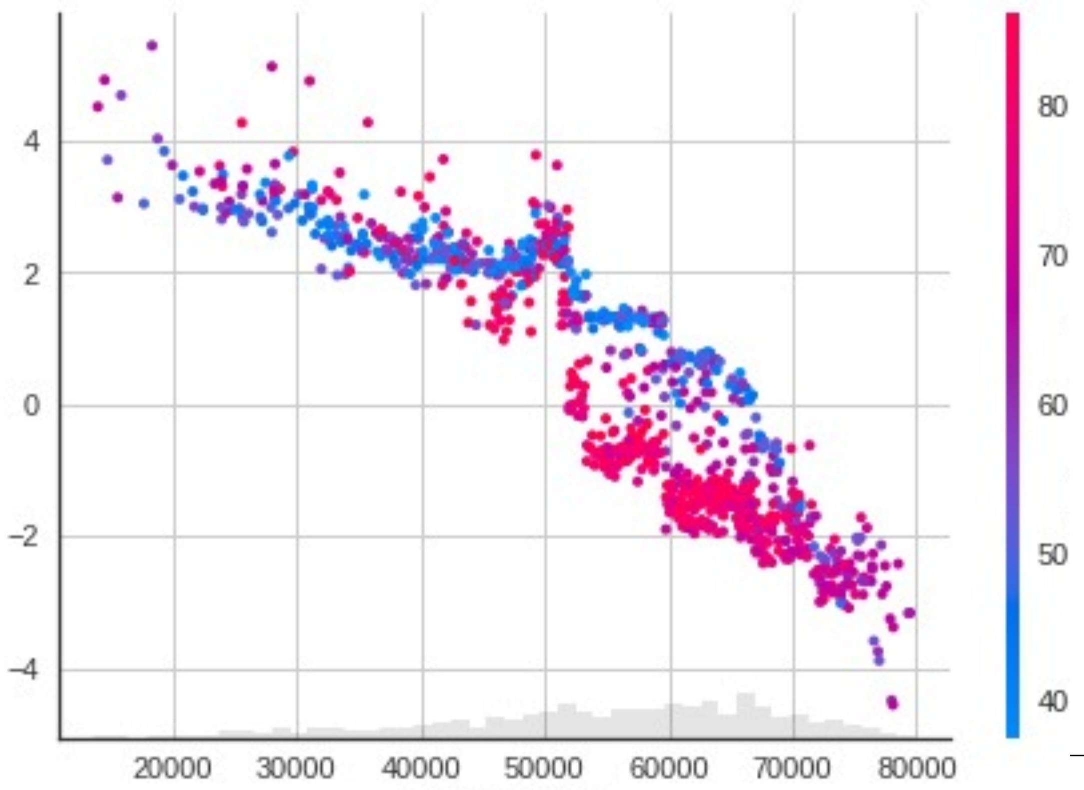
Feature Importance – SHAP

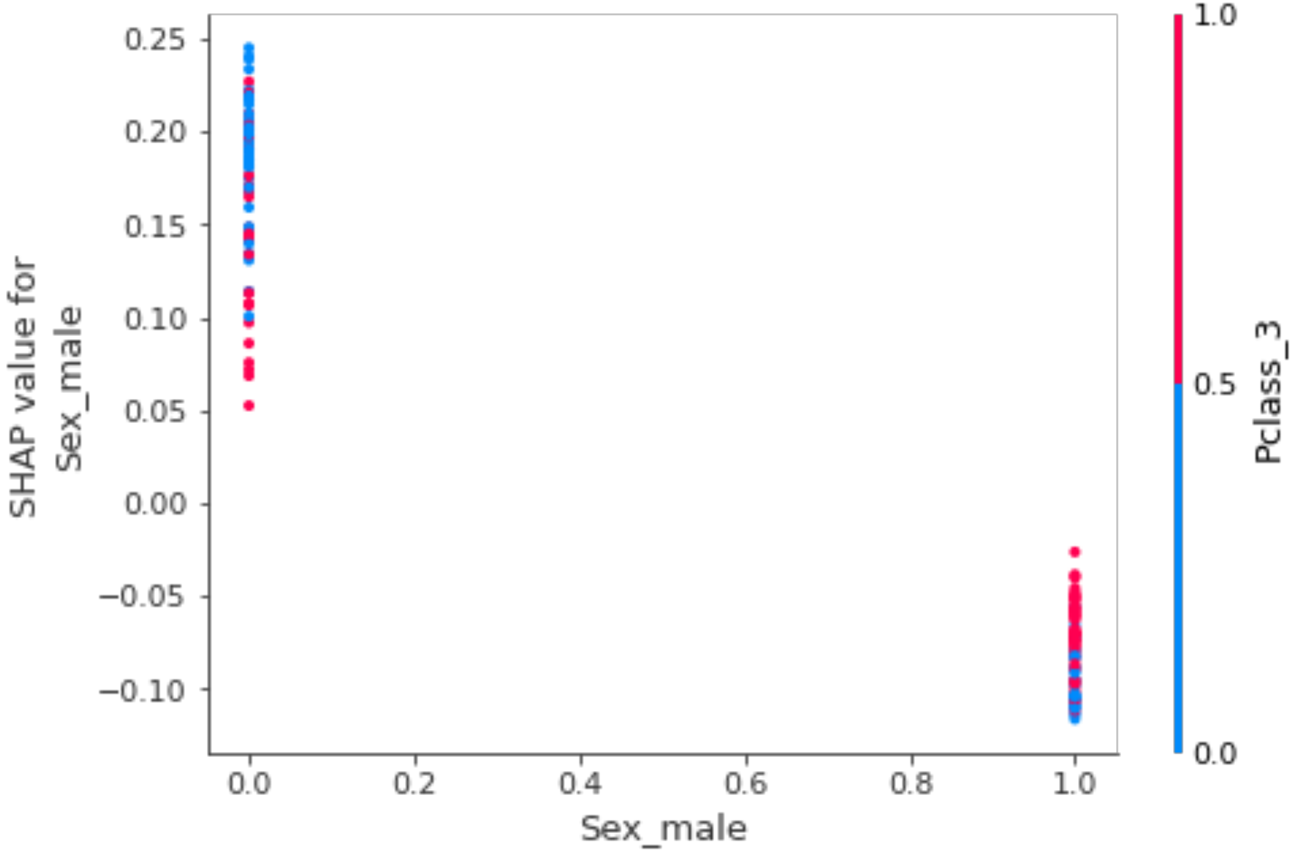
Model Agnostic | Local | *aggregating* → | Global

SHAP – dependence plot

The dependecy plot combines feature importance with feature value.

It is possible to visualize also interaction





CGnal

Feature Importance – SHAP

Model Agnostic

Local

Advantages:

- SHAP has a solid **theoretical foundation** in game theory
- SHAP **connects LIME and Shapley values.**
- SHAP has a **fast implementation for tree-based models.**

Disadvantages:

- **KernelSHAP ignores feature dependence.**
- **TreeSHAP can produce unintuitive feature attributions.**

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

https://github.com/marcotcr/lime

https://towardsdatascience.com/shap-explained-the-way-i-wish-someone-explained-it-to-me-ab81cc69ef30