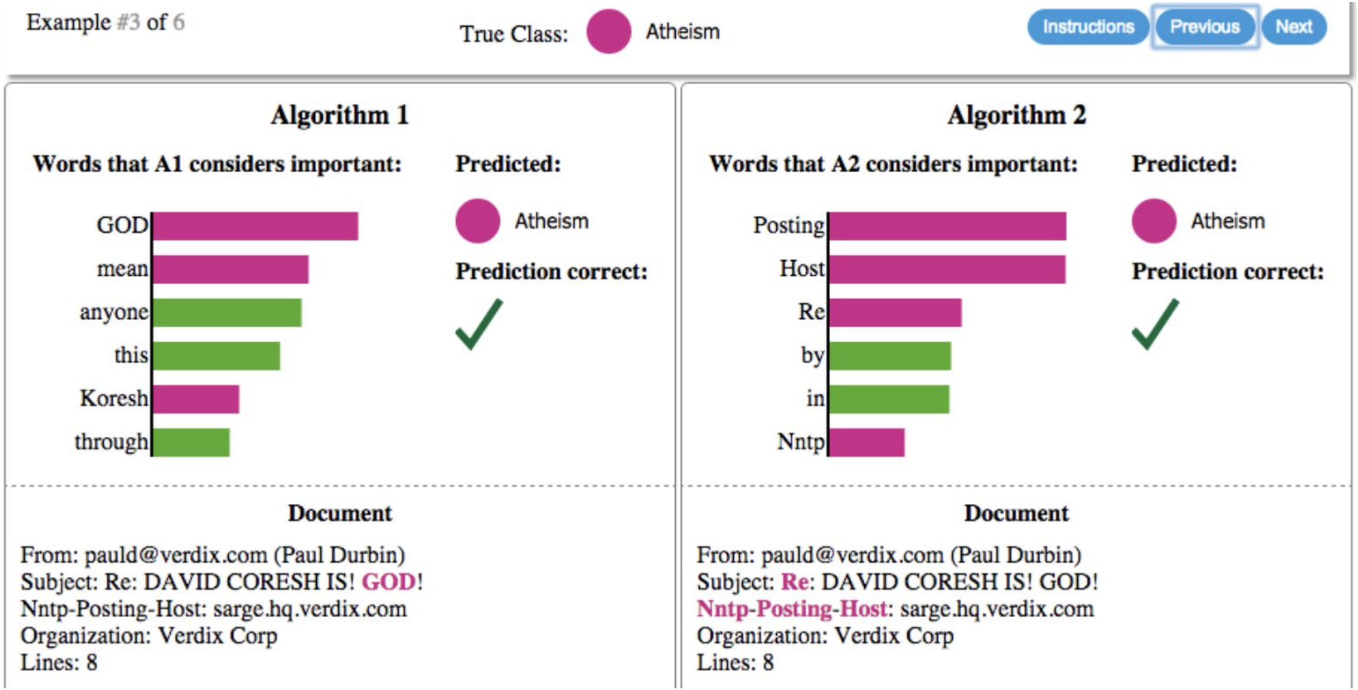
Lecture 4. Surrogate Models. LIME and SHAP

Tatiana Makhalova
MADE, 09.10.2021

Part 1. LIME

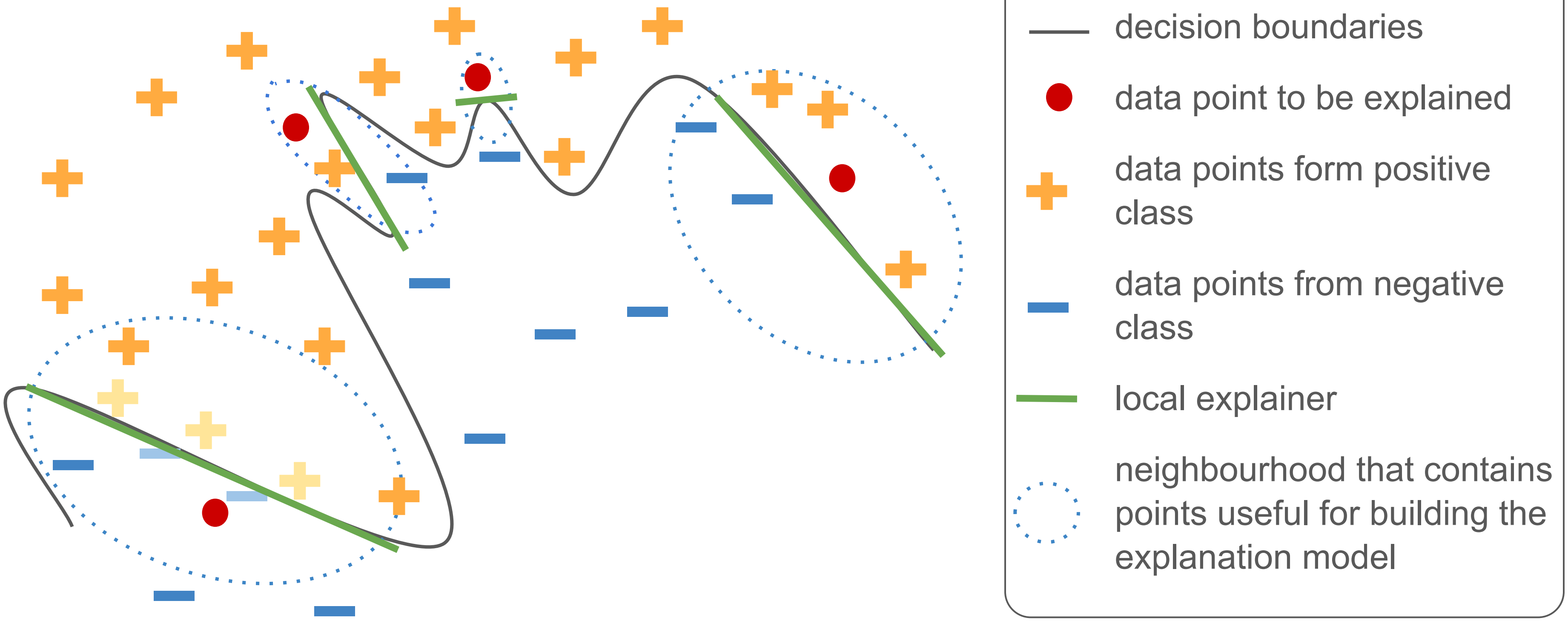
Why do we need more than just accuracy?



Explaining individual predictions of competing classifiers trying to determine if a document is about "Christianity" or "Atheism" on the "20 newsgroup" dataset. **Algorithm 1**: SVM trained on its manually clean version, accuracy: train 69.0%, test 88.6% **Algorithm 2**: SVM trained on the original dataset, accuracy: train 57.3%, test 94.0%

Credit: https://arxiv.org/pdf/1602.04938.pdf

Regression-based explanation



This kind of explanation is called **local** since we explain single instances (points) one by one

Local explanation by surrogate models

Let (X, y) be a dataset from $\mathbb{R}^{n \times d}$, i.e., an instance is described in d-dimensional space and f: $\mathbb{R}^d \rightarrow \mathbb{R}$ be a learned black box model, and x in $\mathbb{R}^d$ be an instance that should be explained

Main steps:

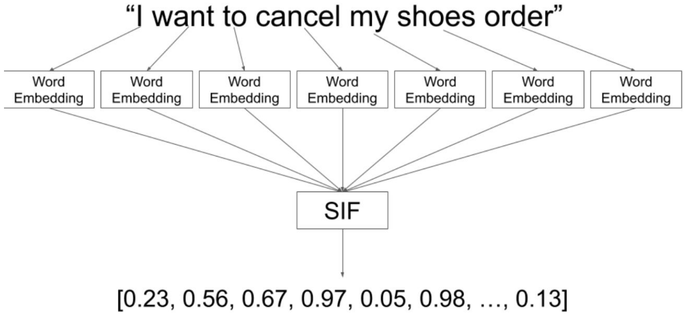
1. **By** introducing local **perturbations** to x generate an new data instances X_{new} in the neighbourhood of x
2. Feed them to the model to **get the target values** $y_{\text{new}} = f(X_{\text{new}})$
3. **Build an explainable model** g on $(X_{\text{new}}, y_{\text{new}})$ and (x, y_x)
4. Use this model g to **explain** $f(x)$

Interpretable space for white box models

Can we use the same space or we need something more?

White model g on images

White model g on word embeddings



Picture source link

Feature importance is meaningless in these cases

We need a space that is **not too large**, such that the features allow for **a clear interpretation**

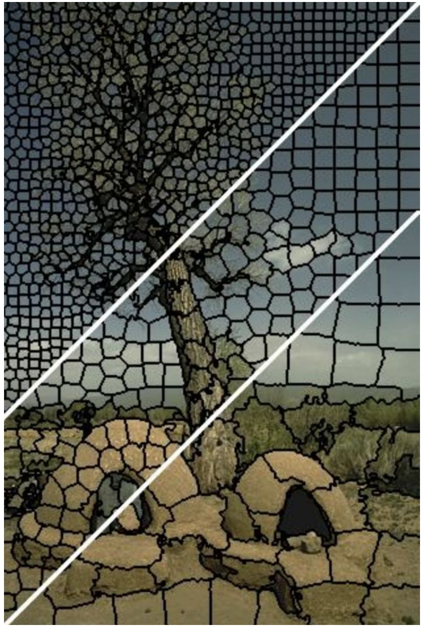
0 Step. Defining an interpretable space for g

Thus, given a black box model f: $\mathbb{R}^d \rightarrow \mathbb{R}$ trained on a dataset (X, y), we use a special transformation to interpretable space $\{0,1\}^{d'}$ and define a white box model g: $\mathbb{R}^{d'} \rightarrow \mathbb{R}$ (or $\{0,1\}^{d'} \rightarrow \mathbb{R}$)

Interpretable space is **specific for each type** of data

0 Step. Defining an interpretable space for g

- Numerical values in tabular data
  - applying a standard scaling: (value - mean) / std
- Categorical values in tabular data
  - one-hot encoding
- Image
  - segmentation (a kind of clustering)
- Texts
  - from the embeddings back to 0-1 vectors



An example of segmentation from https://ivrlwww.epfl.ch/supplementary_material/RK_SLICSuperpixels/index.html

Step 1. Introducing perturbations in tabular data

Let $x = (x_1, \ldots, x_d)$ be an instance described by d features

- for a **continuous** *i*-th attribute
  - *perturbation*: x'_i * std_i + mean_i, where $x'_i \sim N(0,1)$ is a random variable, mean_i and std_i are mean and standard deviation of the *i*-th attribute
  - *interpretable representation*: x'_i
- for a **categorical** *i*-th attribute taking *K* possible values
  - *perturbation*: $x_i \sim \mathrm{Cat}(K, \boldsymbol{\alpha})$, i.e., generalized Bernoulli distribution, where $\boldsymbol{\alpha}$ is a vector of probabilities of each of K values
  - *interpretable representation*: one-hot encoded feature

Step 1. Introducing perturbations in images

Let x be an instance described by d pixels

- *preprocessing*:
  1. applying a segmentation algorithm[1]
  2. consider the obtained K segments as clusters
- *perturbation*: randomly choose clusters with p = ⅓
- *interpretable space*: $\{0,1\}^K$, with 1 and 1 meaning that the *k*-th cluster has be selected or not for a given perturbation

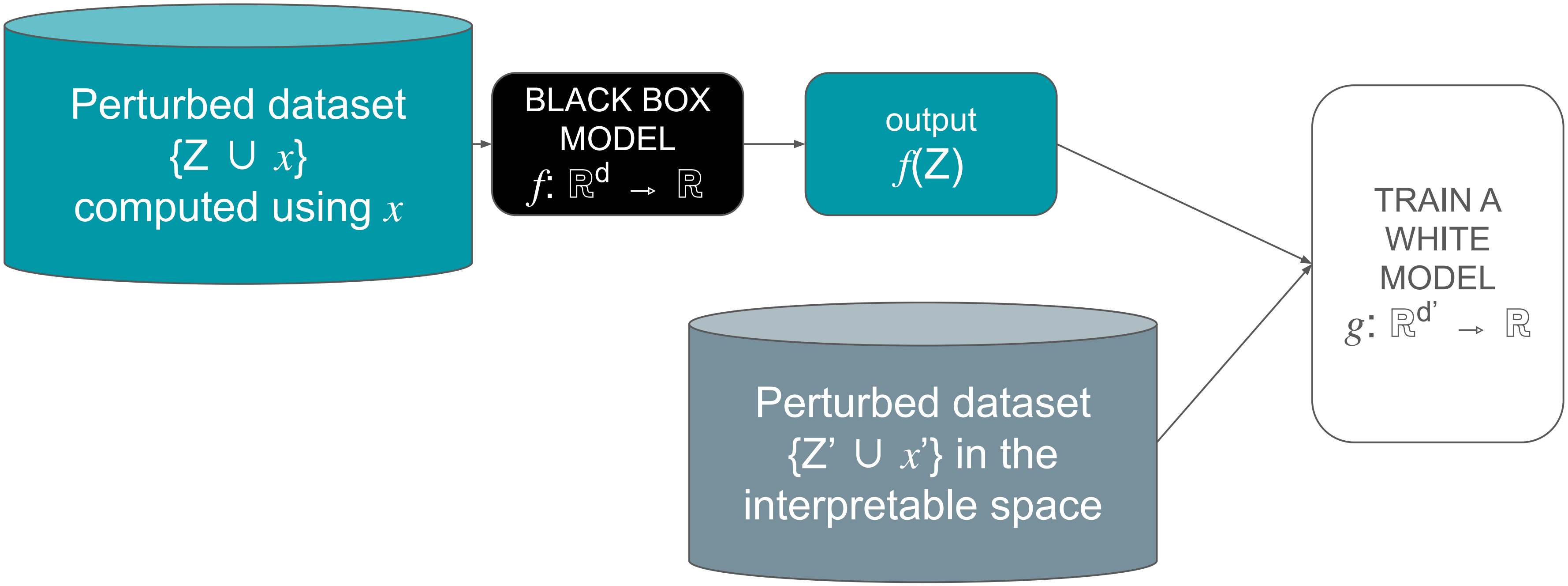
[1] Vedaldi, Andrea, and Stefano Soatto. "Quick shift and kernel methods for mode seeking." *European conference on computer vision*. Springer, Berlin, Heidelberg, 2008.

Step 1. Introducing perturbations in texts

Let $x \in \{1\}^d$ be a vector representing the words appearing in the text instance

- *perturbation*:
  1. select randomly a natural number between $k \in [1, d]$
  2. remove from x k randomly selected words
- *interpretable representation*: a $\{0,1\}^d$ - vector, where 0-values are those that have been removed during perturbation

2. Feed to the black box model



3. Fit an explainable model

The objective is given by $\mathcal{Z}(f, g, \pi_x(z)) = \sum_{z,z'} \boldsymbol{\pi_x(z)}[f(z) - g(z')]^2 + \boldsymbol{\Omega(g)}$

Not all the instances are of the same importance

Let x be the instance to be explained, and z be an perturbed one, then the weight of z is given by $\pi_x(z) = exp(-D(x,z)^2/\sigma^2)$, by default $\sigma = 0.75\sqrt{n}$ and D is the distance (e.g., cosine for texts, L2 for images, etc)

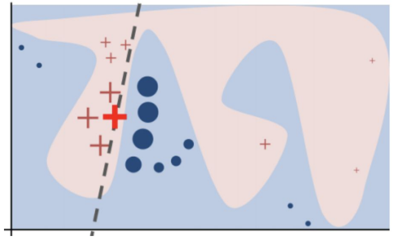


Figure source link

Why do we do weighting: we put more importance to the instances that are close to the x

We need to use a reasonable number of features

We penalise too complex models using $\Omega(g)$, e.g., the height of the tree, the number of non-zero coefficients

Putting all together...

LIME provides an explanation by linear local surrogate model

The explanation is obtained as follows:

$$\xi(x) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \Omega(g)$$

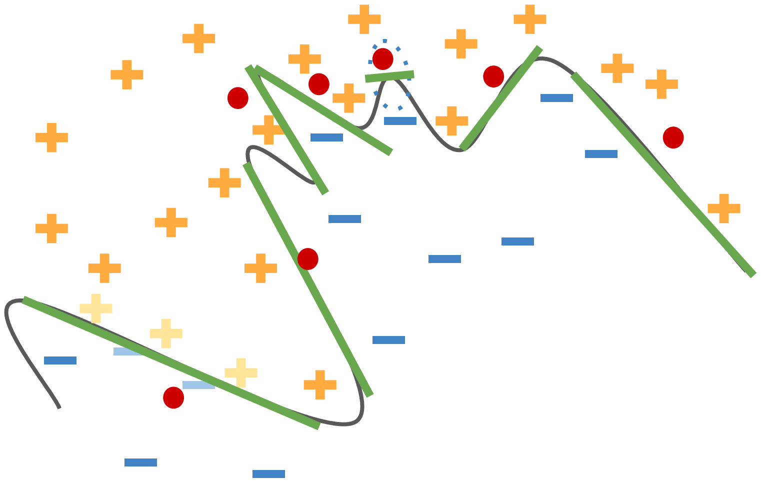
Specificities:

- interpretable representation
- importance of perturbed instances (weighted loss)
- regularization of the complexity of the model

How to trust our model?

With LIME we obtain explanation for a single prediction. Can we explain the whole model?

Yes! By selecting the most diverse instances



This kind of explanation is called **local** since we explain single instances (points) one by one

SP-LIME

Let $\mathscr{W}$ be an explanation matrix of size $|X| \times d'$, i.e., all instances are represented in the interpretable space

	a_1	a_2	a_3	a_4	a_5	a_6
$g(x'_1)$	$\lvert w_{11} \rvert$	$\lvert w_{12} \rvert$				
$g(x'_2)$		$\lvert w_{22} \rvert$	$\lvert w_{23} \rvert$	$w_{24}\rvert$		
$g(x'_3)$		$\lvert w_{32} \rvert$	$\lvert w_{33} \rvert$	$w_{34}\rvert$		
$g(x'_4)$					$\lvert w_{45} \rvert$	$\lvert w_{46} \rvert$

Submodular pick

Interpretable features	a_1	a_2	a_3	a_4	a_5	a_6
$g(x_1)$	$\|w_{11}\|$	$\|w_{12}\|$				
$g(x_2)$		$\|w_{22}\|$	$\|w_{23}\|$	$\|w_{24}\|$		
$g(x_3)$		$\|w_{32}\|$	$\|w_{33}\|$	$\|w_{34}\|$		
$g(x_4)$					$\|w_{45}\|$	$\|w_{46}\|$

Feature importance	I_1	I_2	I_3	I_4	I_5	I_6
$I_j = \sqrt{\sum_{i=1}^{n} \|w_{ij}\|}$	$\|w_{11}\|$	$\|w_{12}\|$ $\|w_{22}\|$ $\|w_{32}\|$	$\|w_{23}\|$ $\|w_{33}\|$	$\|w_{24}\|$ $\|w_{34}\|$	$\|w_{45}\|$	$\|w_{46}\|$

Submodular pick to get

1. Feature importance: which features are the most important for explanation

$$I_j = \sqrt{\sum_{i=1}^{n} |w_{ij}|}$$

2. Maximize the coverage for set of instances V

$$c(V, \mathcal{W}, I) = \sum_{j=1}^{d'} \mathbb{1}_{[\exists i \in V : w_{ij} > 0]} I_j$$

3. Select gradually features that maximize the coverage

$$V \leftarrow V \cup \arg\max_i i(V \cup \{i\}, \mathcal{W}, I)$$

Additional materials

Ribeiro, Marco Tulio, Sameer Singh, and Carlos Guestrin. "Why should i trust you?" Explaining the predictions of any classifier." *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*. 2016

Sebastian Gruber, Christoph Molnar, “LIME and sampling”, Limitations of Interpretable Machine Learning Methods, student seminar

Christoph Molnar Interpretable, “Local Surrogate (LIME)”, Machine Learning, A Guide for Making Black Box Models Explainable,

EXAMPLES (from the LIME package):

- Tabular data
- Texts (2-class and multiclass)
- Images
- Submodular pick

Part 2. SHAP

LIME. Quick recap

$$\xi(x) = \arg\min_{g \in G} \underbrace{\mathcal{L}(f, \underbrace{\boxed{g}}_{\text{linear model}}, \underbrace{\boxed{\pi_x}}_{\text{kernal}})}_{\text{weighted MSE errors}} + \underbrace{\boxed{\Omega(g)}}_{\text{penalty term}}$$

$$\mathcal{L}(f, g, \pi_x) = \sum_{z,z'} \pi_x(z)[f(z) - g(z')]^2$$

$$\pi_x(z) = \exp\left(\frac{-D(x,z)^2}{\sigma^2}\right)$$

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

$x = \boldsymbol{h}_x(x')$ converts a binary vector x' of interpretable inputs into the original input space

In LIME ϕ_i are chosen heuristically. Can we do better?

Yes, using Shapley values with nice theoretical properties.

Additive feature attribution model

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

- **feature attribution**: the quantity of interest of the model for each feature
- **additive**: summing the interest of single features results in the actual interest of the model

PROPERTIES:

Local accuracy The explanation model $g(x')$ matches the original model $f(x)$ when $x = h_x(x')$

Missingness A feature $x_i' = 0$ has not attributed impact, i.e., $x_i' = 0 \Longrightarrow \phi_i = 0$

Consistency Let $f_x(z') = f(h_x(z'))$ and $z' \setminus i$ denote setting $z_i' = 0$. For any two models f and f', if

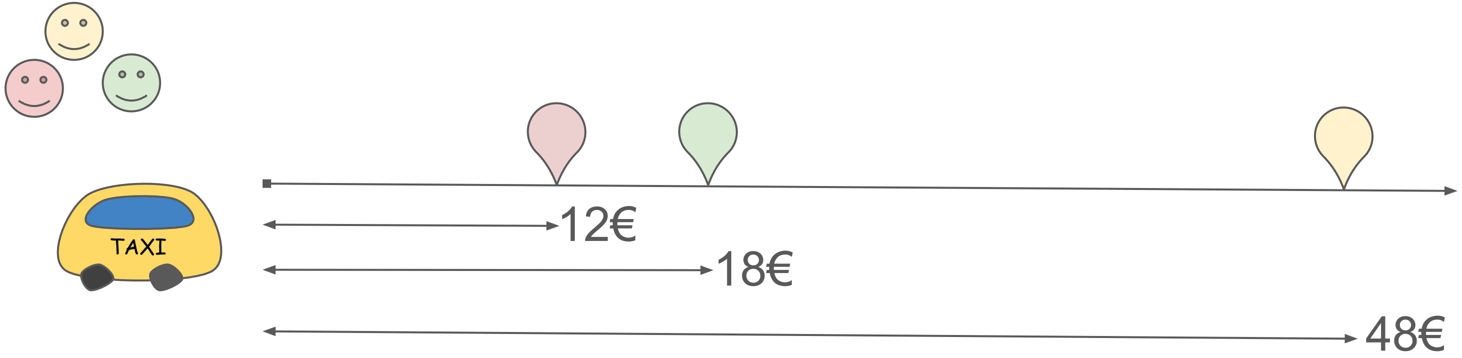
$f_x'(z') - f_x'(z' \setminus i) \geq f_x(z') - f_x(z' \setminus i)$ for all inputs $z_i' \in \{0,1\}^M$ then $\phi_i(f', x) \geq \phi_i(f, x)$.

Coalition game. Example

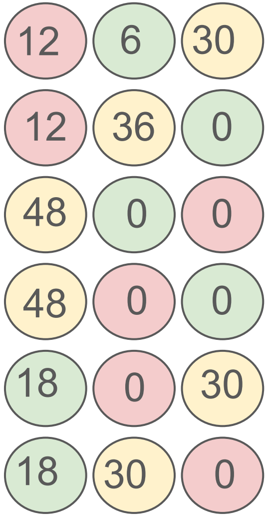
Three players take a taxi

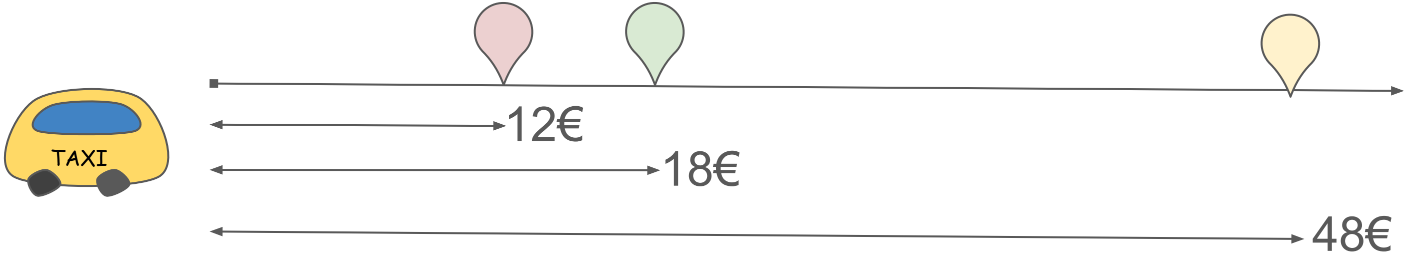
The value function v is the taxi driver's income. The players can make coalition to save some money

What is the **fair** price that each should pay if they take the taxi together?



Payoffs in the coalition



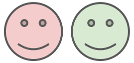


What is a fair contribution for each player?

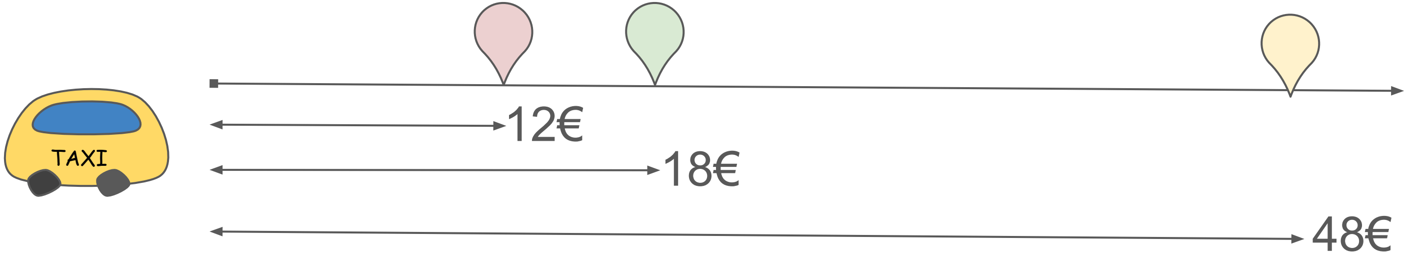
average values over the permutations

ϕ $\boxed{4} + \boxed{7} + \boxed{37} = 48$

Value function

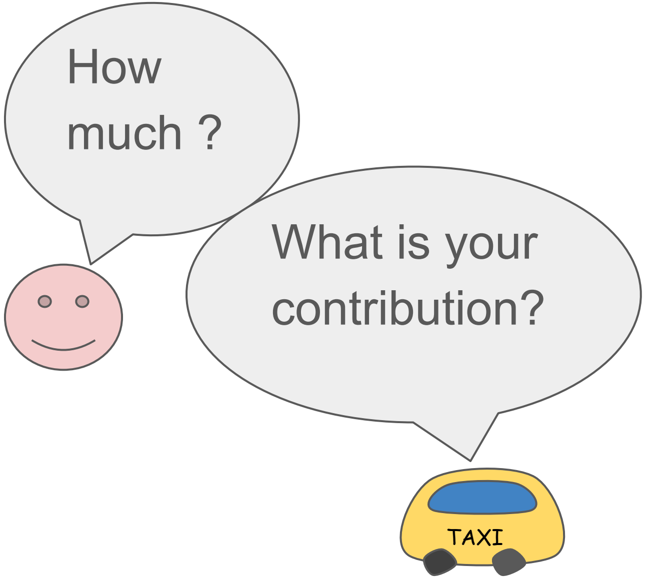
coalition*	v
	0
	12€
	18€
	48€
	18€
	48€
	48€
	48€

* these are the sets, there is not order between the instances



How the players may pay for the ride?

Individual contribution (alternative computing)

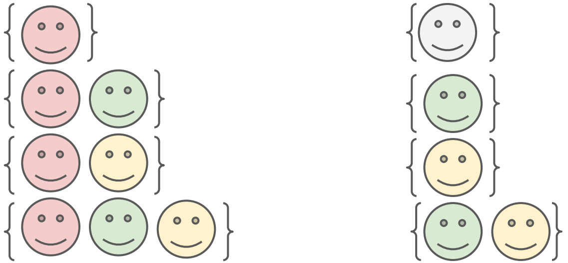


Coalitions with r: $S \cup \{r\}$

Coalitions without r: S

Difference in value functions

$v(S \cup \{r\}) - v(S)$

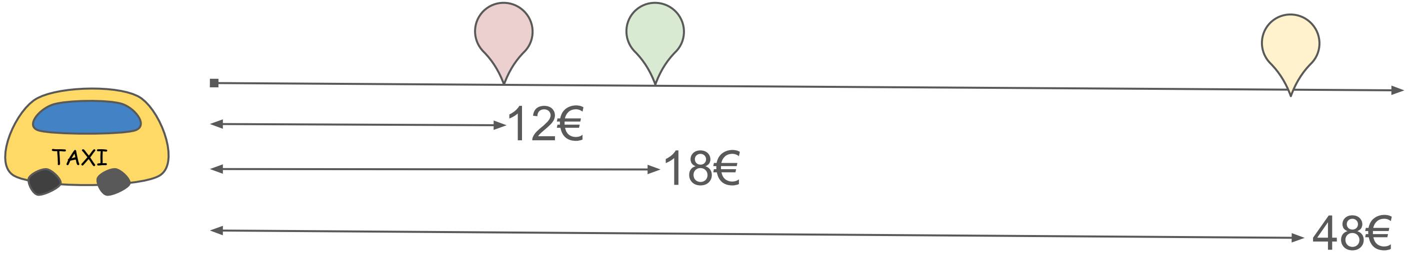


12€ - 0 = 12€

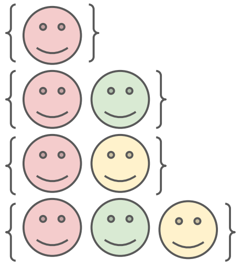
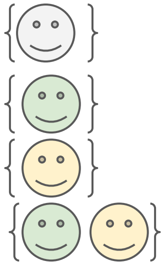
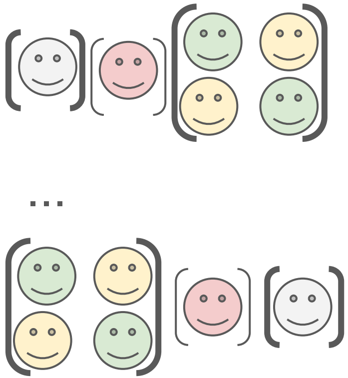
18€ - 18€ = 0

48€ - 48€ = 0

48€ - 48€ = 0



Individual contribution (alternative computing)

Coalitions with r: $S \cup \{r\}$	Coalitions without r: S	Difference in value functions $v(S \cup \{r\}) - v(S)$	How many permutation? $\lvert S\rvert!(\lvert F\rvert-\lvert S\rvert-1)!$	Weights $\frac{\lvert S\rvert!(\lvert F\rvert-\lvert S\rvert-1)!}{\lvert F\rvert!}$
		12€ - 0 = 12€ 18€ - 18€ = 0 48€ - 48€ = 0 48€ - 48€ = 0		2 / 3! = 1/3

The number of permutation for a coalition $S \cup \{r\}$: $\lvert S\rvert!(\lvert F\rvert - \lvert S\rvert-1)!$, where F is the set of the players, the total number of permutations is $\lvert F\rvert!$

More on permutations

Suppose we have 5 elements, i.e., F = {1, 2, 3, 4, 5}. We study a variable **1.** Let's find a number of permutations for the coalition **S = {2, 3}.** The rest is {4,5}

Thus, we search the permutation of the following form (2,3) (1) (4,5). It's obvious, that we have only 4 such kind of permutations, namely,

2, 3, 1, 4, 5
2, 3, 1, 5, 4
3, 2, 1, 4, 5
3, 2, 1, 5, 4

|S| = 2, |F| = 5, |F|-|S|-1 = 2 for $\frac{|S|!(|F|-|S|-1)!}{|F|!}$

Shapley regression values

Let F be a set of features, f_S is a linear model trained on a feature set S, and x_S represents if the input where the feature values from S are retained

Then Shapley regression values are given as follows:

$$\phi_i = \sum_{S \subseteq F \setminus \{i\}} \frac{|S|!(|F|-|S|-1)!}{|F|!} [f_{S \cup \{i\}}(x_{S \cup \{i\}}) - f_S(x_S)]$$

Example of the interpretable space

$f(x_1, x_2, x_3, x_4, x_5)$ is a function obtained by a black box model

Let $S = \{x_2, x_3\}$, then $z' = \{0, 1, 1, 0, 0\}$.

The input of the function f, i.e., $z_S = h_x(z')$ is $(\cdot, x_2, x_3, \cdot, \cdot)$

Simplified computing of Shapley values

Let S be a set of non-zero indices of z', i.e., $h_x(z') = z_S$, where z_S has missing values for features not in S, and

In SHAP as $f(z_S) = f(h_x(z'))$ one uses the conditional expectation, i.e.,

$$f(h_x(z')) = \underbrace{E[f(z)|z_S]}_{\text{expectation}} = E_{z_{\overline{S}}|z_S}[f(z)] \approx \underbrace{E_{z_{\overline{S}}}[f(z)]}_{\text{assuming feature independence*}} \approx \underbrace{f([z_S, E[z_{\overline{S}}]])}_{\text{assuming model linearity}}$$

expectation

assuming **feature independence***

assuming **model linearity**

*Instead of considering the conditional distribution and making the assumption on independence we may consider the interventional distribution $E[f(z)|do(Z_S=z_S)]$ and obtain the same results, see for details Janzing, D., Minorics, L., & Blöbaum, Feature relevance quantification in explainable AI: A causal problem. In *International Conference on Artificial Intelligence and Statistics* (pp. 2907-2916). PMLR.

Additive feature attribution model

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

- **feature attribution**: the quantity of interest of the model for each feature
- **additive**: summing the interest of single features results in the actual interest of the model

PROPERTIES:

Local accuracy The explanation model $g(x')$ matches the original model $f(x)$ when $x = h_x(x')$

Missingness A feature $x_i' = 0$ has not attributed impact, i.e., $x_i' = 0 \Longrightarrow \phi_i = 0$

Consistency Let $f_x(z') = f(h_x(z'))$ and $z' \setminus i$ denote setting $z_i' = 0$. For any two models f and f', if

$f'_x(z') - f'_x(z' \setminus i) \geq f_x(z') - f_x(z' \setminus i)$ for all inputs $z' \in \{0,1\}^M$ then $\phi_i(f', x) \geq \phi_i(f, x)$.

Additive feature attribution method

Additive feature attribution (AFA) methods have an explanation model that is a linear function of binary variables:

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

Theorem. Only one possible AFA explanation model g that satisfies properties of local accuracy, missingness, and consistency:

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - z' - 1)!}{M!} [f_x(z') - f_x(z' \setminus i)]$$

where $|z'|$ is the number of non-zeros entries in z', and $z' \subseteq x'$ represents all z' vectors where the non-zeros entries are a subset of the non-zeros entries in x'

Kernel SHAP (LIME + Shapley values)

$$\xi(x) = \arg\min_{g \in G} \mathcal{L}(f, g, \pi_x) + \cancel{\Omega(g)}$$

0

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

$$\mathcal{L}(f, g, \pi_x) = \sum_{z' \in Z} \pi_x(z')[f(h_x^{-1}(z')) - g(z')]^2$$

$$\pi_x(z) = \exp\left(\frac{-D(x,z)^2}{\sigma^2}\right)$$

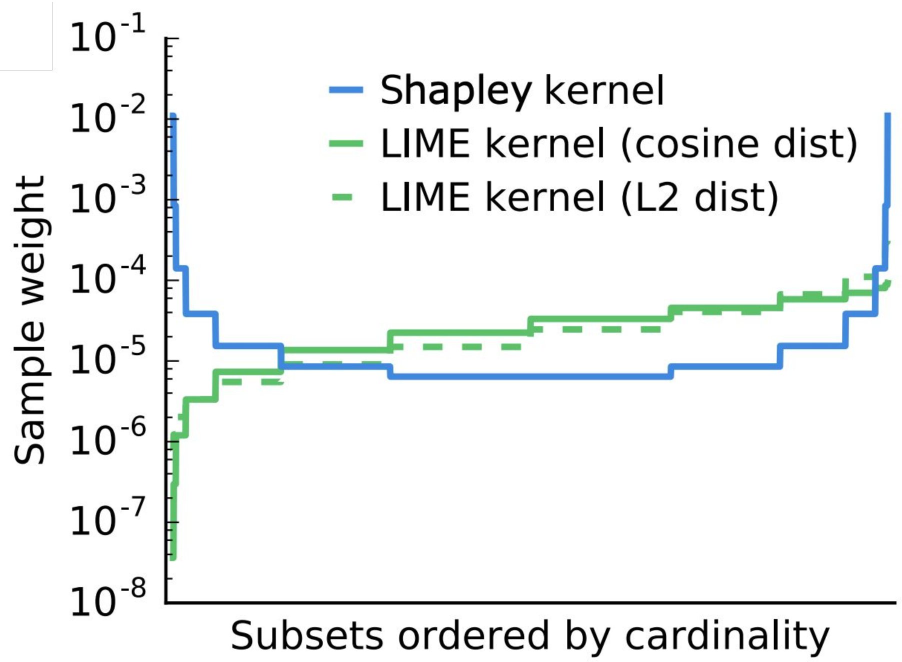
$$\boxed{\pi_x(z') = \frac{(M-1)}{(M\ choose\ |z'|)|z'|(M-|z'|)}}$$

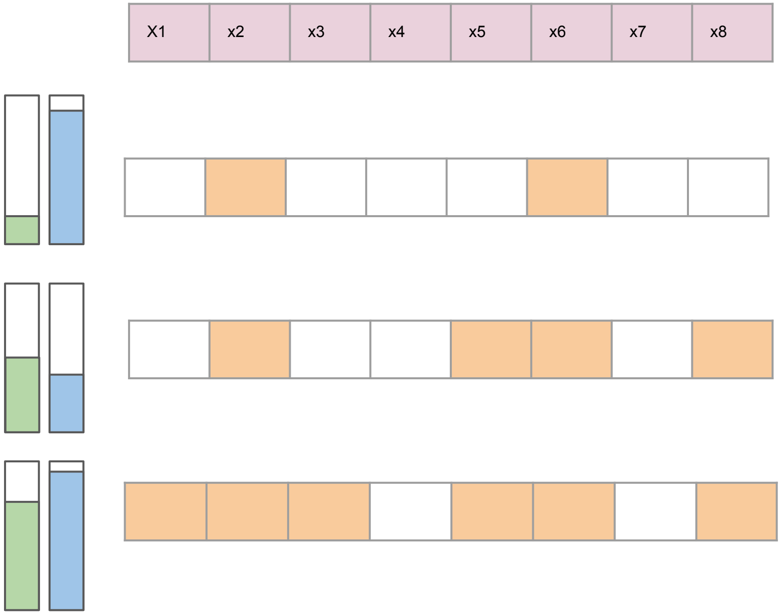
Proof for the kernel

Remark: $\pi_x(z') = \infty$ when $|z'| \in \{0,M\}$, which enforces $\phi_0 = f_x(\varnothing)$ and $f(x) = \sum_{0,\dots,M} \phi_i$, but they are eliminated in practice

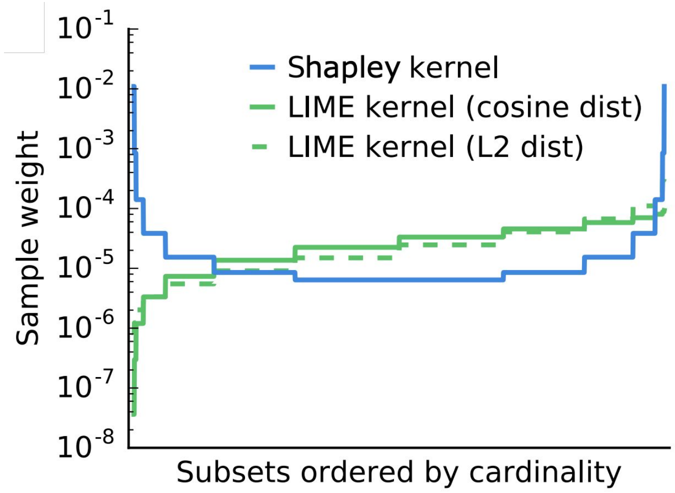
Kernels

SHAP values to measure feature importance





Kernel. The intuition behind



Let x = {3,4,2,4,5,3}. We need to compute explain the response of the model by SHAP and LIME. for Consider

For LIME z' = {0,0,0,1,1,0} in means that we introduce perturbation in 4 features, thus a new instance z will not be very similar to the original one

For SHAP it means that, we fix only x_4 and x_5 and averaged over other variables, thus me learn about the features' isolated main effect on the prediction. If on the opposite, z' = {1,1,1,0,0,1} learn about this features' total effect (main effect plus feature interactions).

If a coalition consists of half the features, we learn little about an individual features contribution

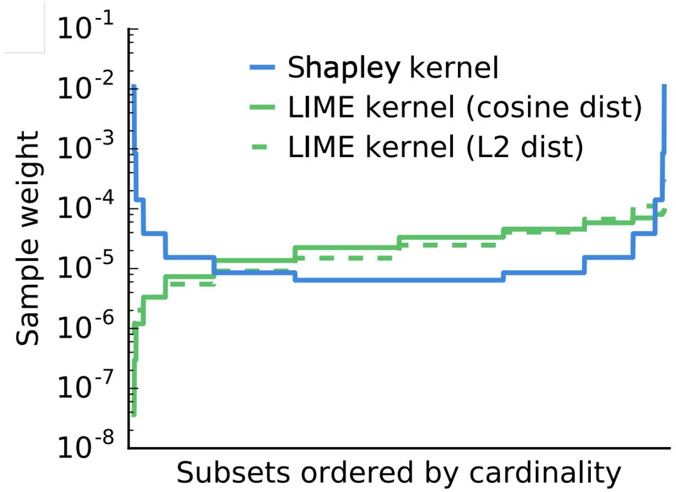
Complexity of the sampling

For an instance given in an M-dimensional space, the importance of the i-th feature is given by

$$\phi_i(f, x) = \sum_{z' \subseteq x'} \frac{|z'|!(M - z' - 1)!}{M!} [f_x(z') - f_x(z' \setminus i)]$$

Thus, we need to generate $O(2^{|M|})$ perturbation.

However, we may use a fixed number of perturbations which are simpled w.r.t. the kernel weight



SHapley Additive exPlanation Values. Summary

- additive
- good properties
- show how f relies on features
- Kernel SHAP model agnostic, potentially of high complexity

Model specific algorithms:

- TreeSHAP
- LinearSHAP
- GradientSHAP
- DeepSHAP