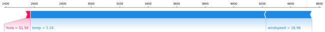
Interpretable Machine Learning

SHAP (SHapley Additive exPlanation) Values



Learning goals

- Get an intuition of additive feature attributions
- Understand the concept of Kernel SHAP
- Ability to interpret SHAP plots
- Global SHAP methods

SHAPLEY VALUES IN ML - A SHORT RECAP

Question: How much does a feature j contribute to the prediction of a single observation.
Idea: Use Shapley values from cooperative game theory

SHAPLEY VALUES IN ML - A SHORT RECAP

Question: How much does a feature j contribute to the prediction of a single observation.
Idea: Use Shapley values from cooperative game theory
Procedure:

- Compare "reduced prediction function" of feature coalition S with $S \cup \{j\}$
- Iterate over possible coalitions to calculate the marginal contribution of feature j to sample $\mathbf{x}$

$$\phi_j = \frac{1}{p!} \sum_{\tau \in \Pi} \underbrace{\hat{f}_{S_j^\tau \cup \{j\}}(\mathbf{x}_{S_j^\tau \cup \{j\}}) - \hat{f}_{S_j^\tau}(\mathbf{x}_{S_j^\tau})}_{\text{marginal contribution of feature } j}$$

SHAPLEY VALUES IN ML - A SHORT RECAP

Question: How much does a feature j contribute to the prediction of a single observation.
Idea: Use Shapley values from cooperative game theory
Procedure:

- Compare "reduced prediction function" of feature coalition S with $S \cup \{j\}$
- Iterate over possible coalitions to calculate the marginal contribution of feature j to sample $\mathbf{x}$

$$\phi_j = \frac{1}{p!} \sum_{\tau \in \Pi} \underbrace{\hat{f}_{S_j^\tau \cup \{j\}}(\mathbf{x}_{S_j^\tau \cup \{j\}}) - \hat{f}_{S_j^\tau}(\mathbf{x}_{S_j^\tau})}_{\text{marginal contribution of feature } j}$$

Remember:

- $\hat{f}$ is the prediction function, p denotes the number of features
- Non-existent features in a coalition are replaced by values of random feature values
- Recall S_j^τ defines the coalition as the set of players before player j in order $\tau = (\tau^{(1)}, \dots, \tau^{(p)})$

$\tau^{(1)}$	...	$\tau^{(\lvert S \rvert)}$	$\tau^{(\lvert S \rvert+1)}$	$\tau^{(\lvert S \rvert+2)}$	...	$\tau^{(p)}$

S_j^τ : Players before player j player j Players after player j

SHAPLEY VALUES IN ML - A SHORT RECAP

Example:

- Train a random forest on bike sharing data only using features humidity (hum), temperature (temp) and windspeed (ws)
- Calculate Shapley value for an observation $\mathbf{x}$ with $\hat{f}(\mathbf{x}) = 2573$
- Mean prediction is $\mathbb{E}(\hat{f}) = 4515$

SHAPLEY VALUES IN ML - A SHORT RECAP

Example:

- Train a random forest on bike sharing data only using features humidity (hum), temperature (temp) and windspeed (ws)
- Calculate Shapley value for an observation $\mathbf{x}$ with $\hat{f}(\mathbf{x}) = 2573$
- Mean prediction is $\mathbb{E}(\hat{f}) = 4515$

Exact Shapley calculation for humidity:

S	$S \cup \{j\}$	$\hat{f}_S$	$\hat{f}_{S \cup \{j\}}$	weight
$\emptyset$	hum	4515	4635	$2/6$
temp	temp, hum	3087	3060	$1/6$
ws	ws, hum	4359	4450	$1/6$
temp, ws	hum, temp, ws	2623	2573	$2/6$

$$\phi_{hum} = \frac{2}{6}(4635 - 4515) + \frac{1}{6}(3060 - 3087) + \frac{1}{6}(4450 - 4359) + \frac{2}{6}(2573 - 2623) = 34$$

FROM SHAPLEY TO SHAP

Example continued: Same calculation can be done for temperature and windspeed:

- $\phi_{temp} = \ldots = -1654$
- $\phi_{ws} = \ldots = -323$

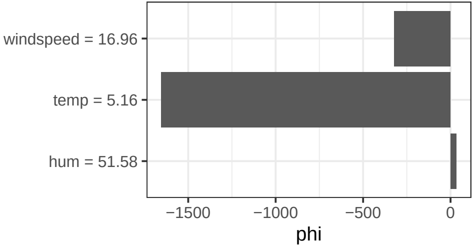
Remember: Shapley values explain the difference between actual and average prediction:

$$2573 - 4515 = 34 - 1654 - 323 = -1942$$
$$\hat{f}(\mathbf{x}) - \mathbb{E}(\hat{f}) = \phi_{hum} + \phi_{temp} + \phi_{ws}$$

$\rightsquigarrow$ can be rewritten to

$$\hat{f}(\mathbf{x}) = \underbrace{\mathbb{E}(\hat{f})}_{\phi_0} + \phi_{hum} + \phi_{temp} + \phi_{ws}$$

Actual prediction: 2572.67 ;
Average prediction: 4515.05



SHAP DEFINITION ▸ Lundberg et al. 2017

Aim: Find an additive combination that explains the prediction of an observation $\mathbf{x}$ by computing the contribution of each feature to the prediction using a (more efficient) estimation procedure.

Definition

- Define simplified (binary) coalition feature space $\mathbf{Z}' \in \{0, 1\}^{K \times p}$ with K rows and p columns
- Rows are referred to as $\mathbf{z}'^{(k)} = \{z_1'^{(k)}, \ldots, z_p'^{(k)}\}$ with $k \in \{1, \ldots, K\}$ (indexes k-th coalition)
- Columns are referred to as $\mathbf{z}_j$ with $j \in \{1, \ldots, p\}$ being the index of the original feature

Example:

Coalition	$\mathbf{z}'^{(k)}$	hum	temp	ws
$\varnothing$	$\mathbf{z}'^{(1)}$	0	0	0
hum	$\mathbf{z}'^{(2)}$	1	0	0
temp	$\mathbf{z}'^{(3)}$	0	1	0
ws	$\mathbf{z}'^{(4)}$	0	0	1
hum, temp	$\mathbf{z}'^{(5)}$	1	1	0
temp, ws	$\mathbf{z}'^{(6)}$	0	1	1
hum, ws	$\mathbf{z}'^{(7)}$	1	0	1
hum, temp, ws	$\mathbf{z}'^{(8)}$	1	1	1

SHAP DEFINITION

▸ Lundberg et al. 2017

Aim: Find an additive combination that explains the prediction of an observation $\mathbf{x}$ by computing the contribution of each feature to the prediction using a (more efficient) estimation procedure.

Definition

- Define simplified (binary) coalition feature space $\mathbf{Z}' \in \{0,1\}^{K \times p}$ with K rows and p columns
- Rows are referred to as $\mathbf{z}'^{(k)} = \{z_1'^{(k)}, \ldots, z_p'^{(k)}\}$ with $k \in \{1, \ldots, K\}$ (indexes k-th coalition)
- Columns are referred to as $\mathbf{z}_j$ with $j \in \{1, \ldots, p\}$ being the index of the original feature

$\mathbf{z}'^{(k)}$: **Coalition** simplified features

$$g\left(\mathbf{z}'^{(k)}\right) = \phi_0 + \sum_{j=1}^{p} \phi_j z_j'^{(k)}$$

SHAP DEFINITION

▸ Lundberg et al. 2017

Aim: Find an additive combination that explains the prediction of an observation **x** by computing the contribution of each feature to the prediction using a (more efficient) estimation procedure.

Definition

- Define simplified (binary) coalition feature space $\mathbf{Z}' \in \{0, 1\}^{K \times p}$ with K rows and p columns
- Rows are referred to as $\mathbf{z}'^{(k)} = \{z_1'^{(k)}, \ldots, z_p'^{(k)}\}$ with $k \in \{1, \ldots, K\}$ (indexes k-th coalition)
- Columns are referred to as $\mathbf{z}_j$ with $j \in \{1, \ldots, p\}$ being the index of the original feature

$\mathbf{z}'^{(k)}$: **Coalition**
simplified features

$$g\left(\mathbf{z}'^{(k)}\right) = \phi_0 + \sum_{j=1}^{p} \phi_j z_j'^{(k)}$$

ϕ_0: **Null Output**
Average Model
Baseline $(\mathbb{E}(\hat{f}))$

SHAP DEFINITION ▸ Lundberg et al. 2017

Aim: Find an additive combination that explains the prediction of an observation **x** by computing the contribution of each feature to the prediction using a (more efficient) estimation procedure.

Definition

- Define simplified (binary) coalition feature space $\mathbf{Z}' \in \{0, 1\}^{K \times p}$ with K rows and p columns
- Rows are referred to as $\mathbf{z}'^{(k)} = \{z_1'^{(k)}, \ldots, z_p'^{(k)}\}$ with $k \in \{1, \ldots, K\}$ (indexes k-th coalition)
- Columns are referred to as $\mathbf{z}_j$ with $j \in \{1, \ldots, p\}$ being the index of the original feature

$\mathbf{z}'^{(k)}$: **Coalition** simplified features

$$g\left(\mathbf{z}'^{(k)}\right) = \phi_0 + \sum_{j=1}^{p} \phi_j z_j'^{(k)}$$

ϕ_0: **Null Output** Average Model Baseline $(\mathbb{E}(\hat{f}))$

ϕ_j: **Attribution** How much does feature j change the output for coalition k

SHAP DEFINITION

Lundberg et al. 2017

Aim: Find an additive combination that explains the prediction of an observation **x** by computing the contribution of each feature to the prediction using a (more efficient) estimation procedure.

$g(\mathbf{z}'^{(k)})$: **Marginal Contribution**
Contribution of coalition $\mathbf{z}'^{(k)}$ to the prediction

ϕ_j: **Shapley Values**

$$\underbrace{g\left(\mathbf{z}'^{(k)}\right) = \phi_0 + \sum_{j=1}^{p} \phi_j z_j'^{(k)}}_{\textbf{Additive Feature Attribution}}$$

Problem
How do we estimate the Shapley values ϕ_j?

KERNEL SHAP - IN 5 STEPS

Definition: A kernel-based, model-agnostic method to compute Shapley values via local surrogate models (e.g. linear model)

1. Sample coalitions
2. Transfer coalitions into feature space & get predictions by applying ML model
3. Compute weights through kernel
4. Fit a weighted linear model
5. Return Shapley values

KERNEL SHAP - IN 5 STEPS

Step 1: Sample coalitions

- Sample K coalitions from the simplified feature space

$$\mathbf{z}'^{(k)} \in \{0, 1\}^p, \quad k \in \{1, \ldots, K\}$$

- For our simple example, we have in total $2^p = 2^3 = 8$ coalitions (without sampling)

Coalition	$\mathbf{z}'^{(k)}$	hum	temp	ws
$\varnothing$	$\mathbf{z}'^{(1)}$	0	0	0
hum	$\mathbf{z}'^{(2)}$	1	0	0
temp	$\mathbf{z}'^{(3)}$	0	1	0
ws	$\mathbf{z}'^{(4)}$	0	0	1
hum, temp	$\mathbf{z}'^{(5)}$	1	1	0
temp, ws	$\mathbf{z}'^{(6)}$	0	1	1
hum, ws	$\mathbf{z}'^{(7)}$	1	0	1
hum, temp, ws	$\mathbf{z}'^{(8)}$	1	1	1

KERNEL SHAP - IN 5 STEPS

Step 2: Transfer Coalitions into feature space & get predictions by applying ML model

- $\mathbf{z}'^{(k)}$ is 1 if features are are part of the k-th coalition, 0 if they are absent
- To calculate predictions for these coalitions, we need to define a function which maps the binary feature space back to the original feature space

Coalition	$\mathbf{z}'^{(k)}$	hum	temp	ws
$\varnothing$	$\mathbf{z}'^{(1)}$	0	0	0
hum	$\mathbf{z}'^{(2)}$	1	0	0
temp	$\mathbf{z}'^{(3)}$	0	1	0
ws	$\mathbf{z}'^{(4)}$	0	0	1
hum, temp	$\mathbf{z}'^{(5)}$	1	1	0
temp, ws	$\mathbf{z}'^{(6)}$	0	1	1
hum, ws	$\mathbf{z}'^{(7)}$	1	0	1
hum, temp, ws	$\mathbf{z}'^{(8)}$	1	1	1

$\mathbf{x}^{coalition}$	hum	temp	ws
$\mathbf{x}^{\{\varnothing\}}$	$\varnothing$	$\varnothing$	$\varnothing$
$\mathbf{x}^{\{hum\}}$	51.6	$\varnothing$	$\varnothing$
$\mathbf{x}^{\{temp\}}$	$\varnothing$	5.1	$\varnothing$
$\mathbf{x}^{\{ws\}}$	$\varnothing$	$\varnothing$	17.0
$\mathbf{x}^{\{hum,temp\}}$	51.6	5.1	$\varnothing$
$\mathbf{x}^{\{temp,ws\}}$	$\varnothing$	5.1	17.0
$\mathbf{x}^{\{hum,ws\}}$	51.6	$\varnothing$	17.0
$\mathbf{x}^{\{hum,temp,ws\}}$	51.6	5.1	17.0

KERNEL SHAP - IN 5 STEPS

Step 2: Transfer Coalitions into feature space & get predictions by applying ML model

- Define $h_x\left(\mathbf{z}'^{(k)}\right) = \mathbf{z}^{(k)}$ where $h_x : \{0,1\}^p \rightarrow \mathbb{R}^p$ maps 1's to feature values of observation $\mathbf{x}$ for features part of the k-th coalition and 0's to feature values of a randomly sampled observation for features absent in the k-th coalition (feature values are permuted multiple times)
- Predict with ML model on this dataset $\hat{f} : \hat{f}\left(h_x\left(\mathbf{z}'^{(k)}\right)\right)$

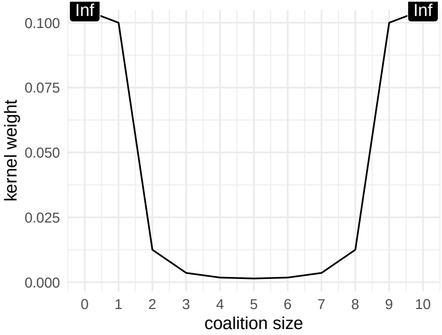
$h_x(\mathbf{z}'^{(k)})$

Coalition	$\mathbf{z}'^{(k)}$	hum	temp	ws	$\mathbf{z}^{(k)}$	hum	temp	ws	$\hat{f}\left(h_x\left(\mathbf{z}'^{(k)}\right)\right)$
$\emptyset$	$\mathbf{z}'^{(1)}$	0	0	0	$\mathbf{z}^{(1)}$	64.3	28.0	14.5	6211
hum	$\mathbf{z}'^{(2)}$	1	0	0	$\mathbf{z}^{(2)}$	51.6	28.0	14.5	5586
temp	$\mathbf{z}'^{(3)}$	0	1	0	$\mathbf{z}^{(3)}$	64.3	5.1	14.5	3295
ws	$\mathbf{z}'^{(4)}$	0	0	1	$\mathbf{z}^{(4)}$	64.3	28.0	17.0	5762
hum, temp	$\mathbf{z}'^{(5)}$	1	1	0	$\mathbf{z}^{(5)}$	51.6	5.1	14.5	2616
temp, ws	$\mathbf{z}'^{(6)}$	0	1	1	$\mathbf{z}^{(6)}$	64.3	5.1	17.0	2900
hum, ws	$\mathbf{z}'^{(7)}$	1	0	1	$\mathbf{z}^{(7)}$	51.6	28.0	17.0	5411
hum, temp, ws	$\mathbf{z}'^{(8)}$	1	1	1	$\mathbf{z}^{(8)}$	51.6	5.1	17.0	2573

KERNEL SHAP - IN 5 STEPS

Step 3: Compute weights through Kernel

Intuition: We learn most about individual features if we can study their effects in isolation or at maximal interaction: Small coalitions (few 1’s) and large coalitions (i.e. many 1’s) get the largest weights



KERNEL SHAP - IN 5 STEPS

Step 3: Compute weights through Kernel ▸ see shapley_kernel_proof.pdf

Intuition: We learn most about individual features if we can study their effects in isolation or at maximal interaction: Small coalitions (few 1's) and large coalitions (i.e. many 1's) get the largest weights

$$\pi_x\left(\mathbf{z}'^{(k)}\right) = \frac{(p-1)}{\binom{p}{|\mathbf{z}'^{(k)}|}|\mathbf{z}'^{(k)}|\left(p - |\mathbf{z}'^{(k)}|\right)}$$

p: Number of features in $\mathbf{x}$

$\pi_x(\mathbf{z}'^{(k)})$: kernel weight for coalition $\mathbf{z}'^{(k)}$

$|\mathbf{z}'^{(k)}|$: coalition size / sum of 1s in $\mathbf{z}'^{(k)}$

KERNEL SHAP - IN 5 STEPS

Step 3: Compute weights through Kernel

Purpose: to include this knowledge in the local surrogate model (linear regression), we calculate weights for each coalition which are the observations of the linear regression

$$\pi_x(\mathbf{z}') = \frac{(p-1)}{\binom{p}{|\mathbf{z}'|}|\mathbf{z}'|(p-|\mathbf{z}'|)} \rightsquigarrow \pi_x(\mathbf{z}' = (1,0,0)) = \frac{(3-1)}{\binom{3}{1}1(3-1)} = \frac{1}{3}$$

Coalition	$\mathbf{z}'^{(k)}$	hum	temp	ws	weight
$\varnothing$	$\mathbf{z}'^{(1)}$	0	0	0	∞
hum	$\mathbf{z}'^{(2)}$	1	0	0	0.33
temp	$\mathbf{z}'^{(3)}$	0	1	0	0.33
ws	$\mathbf{z}'^{(4)}$	0	0	1	0.33
hum, temp	$\mathbf{z}'^{(5)}$	1	1	0	0.33
temp, ws	$\mathbf{z}'^{(6)}$	0	1	1	0.33
hum, ws	$\mathbf{z}'^{(7)}$	1	0	1	0.33
hum, temp, ws	$\mathbf{z}'^{(8)}$	1	1	1	∞

KERNEL SHAP - IN 5 STEPS

Step 3: Compute weights through Kernel

Purpose: to include this knowledge in the local surrogate model (linear regression), we calculate weights for each coalition which are the observations of the linear regression

Coalition	$\mathbf{z}'^{(k)}$	hum	temp	ws	weight
$\emptyset$	$\mathbf{z}'^{(1)}$	0	0	0	∞
hum	$\mathbf{z}'^{(2)}$	1	0	0	0.33
temp	$\mathbf{z}'^{(3)}$	0	1	0	0.33
ws	$\mathbf{z}'^{(4)}$	0	0	1	0.33
hum, temp	$\mathbf{z}'^{(5)}$	1	1	0	0.33
temp, ws	$\mathbf{z}'^{(6)}$	0	1	1	0.33
hum, ws	$\mathbf{z}'^{(7)}$	1	0	1	0.33
hum, temp, ws	$\mathbf{z}'^{(8)}$	1	1	1	∞

$\rightsquigarrow$ weights for empty and full set are infinity and not used as observations for the linear regression
$\rightsquigarrow$ instead constraints are used such that properties (local accuracy and missingness) are satisfied

KERNEL SHAP - IN 5 STEPS

Step 4: Fit a weighted linear model

Aim: Estimate a weighted linear model with Shapley values being the coefficients ϕ_j

$$g\left(\mathbf{z}'^{(k)}\right) = \phi_0 + \sum_{j=1}^{p} \phi_j z_j'^{(k)}$$

and minimize by WLS using the weights π_x of step 3

$$L\left(\hat{f}, g, \pi_x\right) = \sum_{k=1}^{K} \left[\hat{f}\left(h_x\left(\mathbf{z}'^{(k)}\right)\right) - g\left(\mathbf{z}'^{(k)}\right)\right]^2 \pi_x\left(\mathbf{z}'^{(k)}\right)$$

with $\phi_0 = \mathbb{E}(\hat{f})$ and $\phi_p = \hat{f}(x) - \sum_{j=0}^{p-1} \phi_j$ we receive a $p-1$ dimensional linear regression problem

KERNEL SHAP - IN 5 STEPS

Step 4: Fit a weighted linear model

Aim: Estimate a weighted linear model with Shapley values being the coefficients ϕ_j

$$g\left(\mathbf{z}'^{(k)}\right) = \phi_0 + \sum_{j=1}^{p} \phi_j z_j'^{(k)} \rightsquigarrow g\left(\mathbf{z}'^{(k)}\right) = 4515 + 34 \cdot z_1'^{(k)} - 1654 \cdot z_2'^{(k)} - 323 \cdot z_3'^{(k)}$$

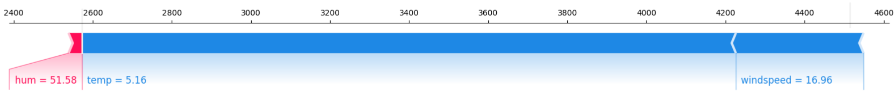
$\mathbf{z}'^{(k)}$	hum	temp	ws	weight	$\hat{f}$
$\mathbf{z}'^{(2)}$	1	0	0	0.33	4635
$\mathbf{z}'^{(3)}$	0	1	0	0.33	3087
$\mathbf{z}'^{(4)}$	0	0	1	0.33	4359
$\mathbf{z}'^{(5)}$	1	1	0	0.33	3060
$\mathbf{z}'^{(6)}$	0	1	1	0.33	2623
$\mathbf{z}'^{(7)}$	1	0	1	0.33	4450
	input				*output*

KERNEL SHAP - IN 5 STEPS

Step 5: Return SHAP values

Intuition: Estimated Kernel SHAP values are equivalent to Shapley values

$$g(\mathbf{z}'^{(8)}) = \hat{f}(h_x(\mathbf{z}'^{(8)})) = 4515 + 34 \cdot 1 - 1654 \cdot 1 - 323 \cdot 1 = \underbrace{\mathbb{E}(\hat{f})}_{\phi_0} + \phi_{hum} + \phi_{temp} + \phi_{ws} = \hat{f}(\mathbf{x}) = 2573$$



PROPERTIES

Local Accuracy

$$f(\mathbf{x}) = g\left(\mathbf{x}'\right) = \phi_0 + \sum_{j=1}^{p} \phi_j x_j'$$

Intuition: If the coalition includes all features ($\mathbf{x}' \in \{1\}^p$), the attributions ϕ_j and the null output ϕ_0 sum up to the original model output $f(\mathbf{x})$

Local accuracy corresponds to the **axiom of efficiency** in Shapley game theory

PROPERTIES

Local Accuracy

$$f(\mathbf{x}) = g\left(\mathbf{x}'\right) = \phi_0 + \sum_{j=1}^{p} \phi_j x_j'$$

Missingness

$$x_j' = 0 \Longrightarrow \phi_j = 0$$

Intution: A missing feature gets an attribution of zero

PROPERTIES

Local Accuracy

$$f(\mathbf{x}) = g\left(\mathbf{x}'\right) = \phi_0 + \sum_{j=1}^{p} \phi_j x_j'$$

Missingness

$$x_j' = 0 \Longrightarrow \phi_j = 0$$

Consistency
$\hat{f}_x\left(\mathbf{z}'^{(k)}\right) = \hat{f}\left(h_x\left(\mathbf{z}'^{(k)}\right)\right)$ and $\mathbf{z}'^{(k)}_{-j}$ denote setting $z_j'^{(k)} = 0$. For any two models $\hat{f}$ and $\hat{f}'$, if

$$\hat{f}_x'\left(\mathbf{z}'^{(k)}\right) - \hat{f}_x'\left(\mathbf{z}'^{(k)}_{-j}\right) \geq \hat{f}_x\left(\mathbf{z}'^{(k)}\right) - \hat{f}_x\left(\mathbf{z}'^{(k)}_{-j}\right)$$

for all inputs $\mathbf{z}'^{(k)} \in \{0, 1\}^p$, then

$$\phi_j\left(\hat{f}', \mathbf{x}\right) \geq \phi_j(\hat{f}, \mathbf{x})$$

PROPERTIES

Local Accuracy

$$f(\mathbf{x}) = g\left(\mathbf{x}'\right) = \phi_0 + \sum_{j=1}^{p} \phi_j x_j'$$

Missingness

$$x_j' = 0 \Longrightarrow \phi_j = 0$$

Consistency

$$\hat{f}_x'\left(\mathbf{z}'^{(k)}\right) - \hat{f}_x'\left(\mathbf{z}_{-j}'^{(k)}\right) \geq \hat{f}_x\left(\mathbf{z}'^{(k)}\right) - \hat{f}_x\left(\mathbf{z}_{-j}'^{(k)}\right) \Longrightarrow \phi_j\left(\hat{f}', \mathbf{x}\right) \geq \phi_j(\hat{f}, \mathbf{x})$$

Intution: If a model changes so that the marginal contribution of a feature value increases or stays the same, the Shapley value also increases or stays the same

From **consistency** the Shapley **axioms of additivity, dummy and symmetry** follow

GLOBAL SHAP ▸ Lundberg et al. 2018

Idea:

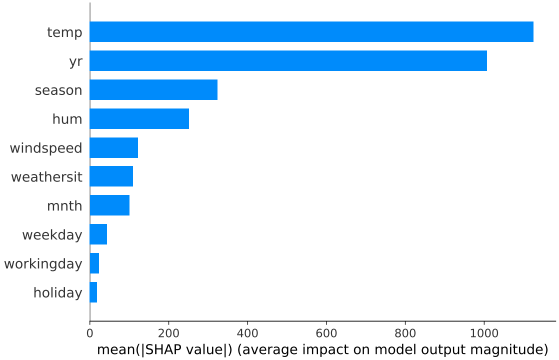
- Run SHAP for every observation and thereby get a matrix of Shapley values
- The matrix has one row per data observation and one column per feature
- We can interpret the model globally by analyzing the Shapley values in this matrix

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \phi_{13} & \dots & \phi_{1p} \\ \phi_{21} & \phi_{22} & \phi_{23} & \dots & \phi_{2p} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{n1} & \phi_{n2} & \phi_{n3} & \dots & \phi_{np} \end{bmatrix}$$

FEATURE IMPORTANCE

Idea: Average the absolute Shapley values of each feature over all observations. This corresponds to calculating averages column by column in Φ

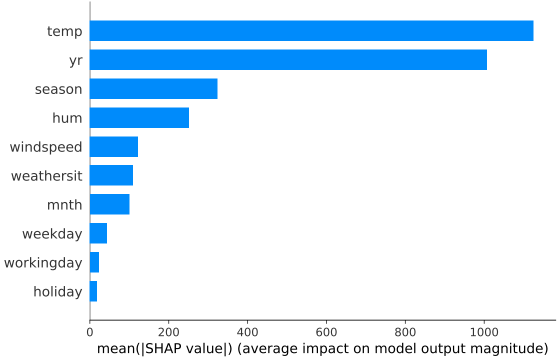
$$I_j = \frac{1}{n} \sum_{i=1}^{n} \left| \phi_j^{(i)} \right|$$



FEATURE IMPORTANCE

Interpretation:

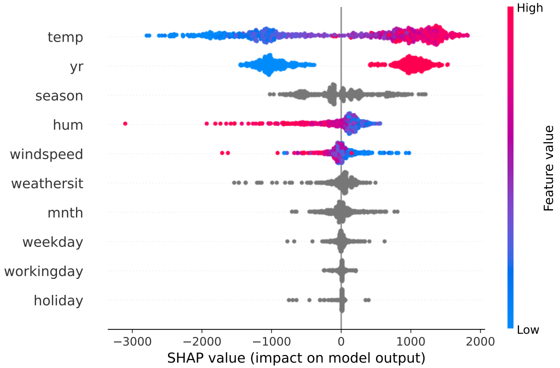
- The features temperature and year have by far the highest influence on the model's prediction
- Compared to Shapley values, no effect direction is provided, but instead a feature ranking similar to PFI
- However, Shapley FI is based on the model's predictions only while PFI is based on the model's performance (loss)



SUMMARY PLOT

Combines feature importance with feature effects

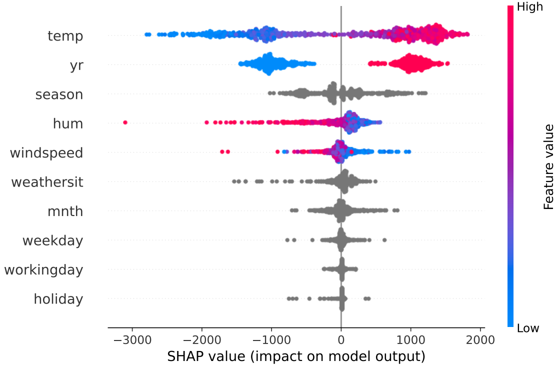
- Each point is a Shapley value for a feature and an observation
- The color represents the value of the feature from low to high
- Overlapping points are jittered in y-axis direction



SUMMARY PLOT

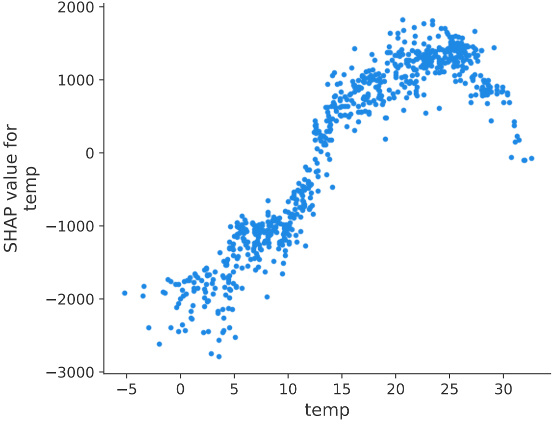
Interpretation:

- Low temperatures have a negative impact while high temperatures lead to more bike rentals
- Year: two point clouds for 2011 and 2012 (other categorical features are gray)
- A high humidity has a huge, negative impact on the bike rental, while low humidity has a rather minor positive impact on bike rentals



DEPENDENCE PLOT

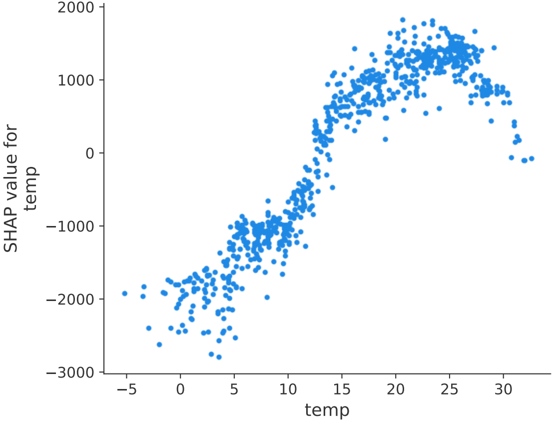
- Visualize the marginal contribution of a feature similar to the PDP
- Plot a point with the feature value on the x-axis and the corresponding Shapley value on the y-axis



DEPENDENCE PLOT

Interpretation:

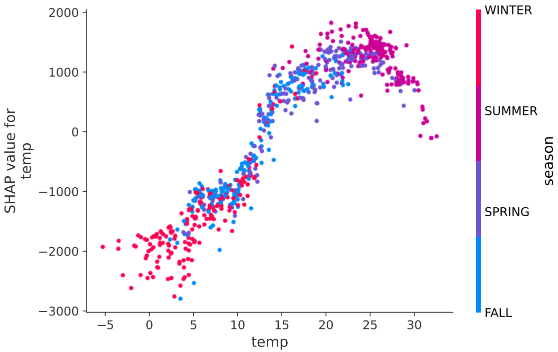
- Increasing temperatures induce increasing bike rentals until 25°C
- If it gets too hot, the bike rentals decrease



DEPENDENCE PLOT

Interpretation:

- We can colour the observations by a second feature to detect interactions
- Visibly the temperatures interaction with the season is very strong



DISCUSSION

Advantages

- All the advantages of Shapley values
- Unify the field of interpretable machine learning in the class of additive feature attribution methods
- Has a fast implementation for tree-based models
- Various global interpretation methods

Disadvantages

- Disadvantages of Shapley values also apply to SHAP
- KernelSHAP is slow (TreeSHAP can be used as a faster alternative for tree-based models ▸ Lundberg et al 2018 – and for an intuitive explanation ▸ see Sukumar: TreeSHAP)
- KernelSHAP ignores feature dependence