Interpretable Machine Learning
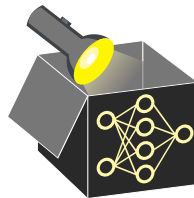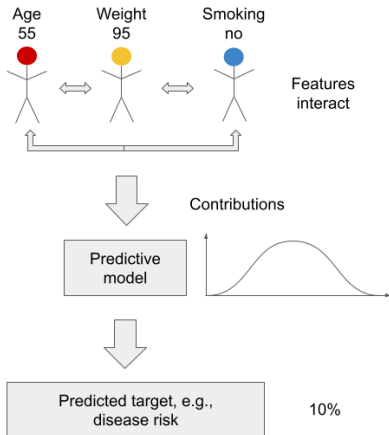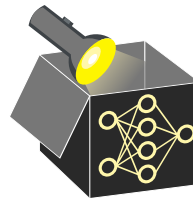
Shapley Values for Local Explanations

Actual prediction: 4514.35
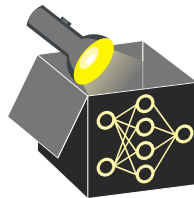Average prediction: 4508.18
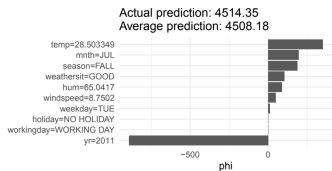
Learning goals

- See model predictions as a cooperative game

- Transfer the Shapley value concept from game theory to machine learning
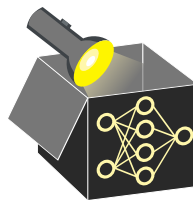
FROM GAME THEORY TO MACHINE LEARNING

FROM GAME THEORY TO MACHINE LEARNING

- Game: Make prediction $\hat{f}(x_1, x_2, \ldots, x_p)$ for a single observation **x**

FROM GAME THEORY TO MACHINE LEARNING

- Game: Make prediction $\hat{f}(x_1, x_2, \ldots, x_p)$ for a single observation **x**
- Players: Features $x_j, j \in \{1, \ldots, p\}$ which cooperate to produce a prediction
  $\rightsquigarrow$ How can we make a prediction with a subset of features without changing the model?
  $\rightsquigarrow$ PD function: $\hat{f}_S(\mathbf{x}_S) := \int_{X_{-s}} \hat{f}(\mathbf{x}_S, X_{-S}) d\mathbb{P}_{X_{-s}}$ ("removing" by marginalizing over $-S$)

FROM GAME THEORY TO MACHINE LEARNING

- Game: Make prediction $\hat{f}(x_1, x_2, \ldots, x_p)$ for a single observation **x**
- Players: Features $x_j, j \in \{1, \ldots, p\}$ which cooperate to produce a prediction
  $\rightsquigarrow$ How can we make a prediction with a subset of features without changing the model?
  $\rightsquigarrow$ PD function: $\hat{f}_S(\mathbf{x}_S) := \int_{X_{-S}} \hat{f}(\mathbf{x}_S, X_{-S}) d\mathbb{P}_{X_{-S}}$ ("removing" by marginalizing over $-S$)
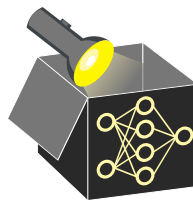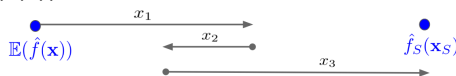- Value function / payout of coalition $S \subseteq P$ for observation **x**:

$$v(S) = \hat{f}_S(\mathbf{x}_S) - \mathbb{E}_{\mathbf{x}}(\hat{f}(\mathbf{x})), \text{ where } \hat{f}_S : \mathcal{X}_S \mapsto \mathcal{Y}$$

$\rightsquigarrow$ subtraction of $\mathbb{E}_{\mathbf{x}}(\hat{f}(\mathbf{x}))$ ensures that v is a value function with $v(\emptyset) = 0$

FROM GAME THEORY TO MACHINE LEARNING

- Game: Make prediction $\hat{f}(x_1, x_2, \ldots, x_p)$ for a single observation **x**
- Players: Features $x_j, j \in \{1, \ldots, p\}$ which cooperate to produce a prediction
  $\rightsquigarrow$ How can we make a prediction with a subset of features without changing the model?
  $\rightsquigarrow$ PD function: $\hat{f}_S(\mathbf{x}_S) := \int_{X_{-S}} \hat{f}(\mathbf{x}_S, X_{-S}) d\mathbb{P}_{X_{-S}}$ ("removing" by marginalizing over $-S$)
- Value function / payout of coalition $S \subseteq P$ for observation **x**:

$$v(S) = \hat{f}_S(\mathbf{x}_S) - \mathbb{E}_{\mathbf{x}}(\hat{f}(\mathbf{x})), \text{ where } \hat{f}_S : \mathcal{X}_S \mapsto \mathcal{Y}$$

$\rightsquigarrow$ subtraction of $\mathbb{E}_{\mathbf{x}}(\hat{f}(\mathbf{x}))$ ensures that v is a value function with $v(\emptyset) = 0$
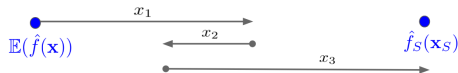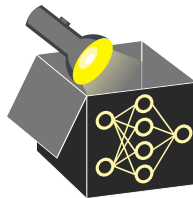

- Marginal contribution: $v(S \cup \{j\}) - v(S) = \hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) - \hat{f}_S(\mathbf{x}_S)$
  $\rightsquigarrow \mathbb{E}_{\mathbf{x}}(\hat{f}(\mathbf{x}))$ cancels out due to the subtraction of value functions

# SHAPLEY VALUE - DEFINITION ▸ Shapley (1953) ▸ Strumbelj et al. (2014)

Shapley value ϕ_j of feature j for observation **x** via **order definition**:

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

- Interpretation: Feature x_j contributed ϕ_j to difference between $\hat{f}(\mathbf{x})$ and average prediction
  $\rightsquigarrow$ Note: Marginal contributions and Shapley values can be negative
- For exact computation of $\phi_j(\mathbf{x})$, the PD function $\hat{f}_S(\mathbf{x}_S) = \frac{1}{n} \sum_{i=1}^{n} \hat{f}(\mathbf{x}_S, \mathbf{x}_{-S}^{(i)})$ for any set of features S can be used which yields

$$\phi_j(\mathbf{x}) = \frac{1}{|P|! \cdot n} \sum_{\tau \in \Pi} \sum_{i=1}^{n} \hat{f}(\mathbf{x}_{S_j^\tau \cup \{j\}}, \mathbf{x}_{-\{S_j^\tau \cup \{j\}\}}^{(i)}) - \hat{f}(\mathbf{x}_{S_j^\tau}, \mathbf{x}_{-S_j^\tau}^{(i)})$$

  $\rightsquigarrow$ Note: $\hat{f}_S$ marginalizes over all other features $-S$ using all observations
  $i = 1, \ldots, n$
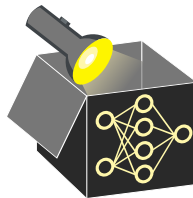
ESTIMATION: A PRACTICAL PROBLEM

- Exact Shapley value computation is problematic for high-dimensional feature spaces

  $\rightsquigarrow$ For 10 features, there are already $|P|! = 10! \approx 3.6$ million possible orders of features
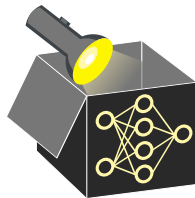
ESTIMATION: A PRACTICAL PROBLEM

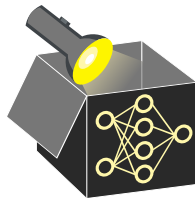- Exact Shapley value computation is problematic for high-dimensional feature spaces

  $\rightsquigarrow$ For 10 features, there are already $|P|! = 10! \approx 3.6$ million possible orders of features

- Additional problem due to estimation of the marginal prediction $\hat{f}_{S_j^\tau}$: Averaging over the entire data set for each coalition S_j^τ introduced by τ can be very expensive for large data sets

ESTIMATION: A PRACTICAL PROBLEM

- Exact Shapley value computation is problematic for high-dimensional feature spaces

  $\leadsto$ For 10 features, there are already $|P|! = 10! \approx 3.6$ million possible orders of features

- Additional problem due to estimation of the marginal prediction $\hat{f}_{S_j^\tau}$: Averaging over the entire data set for each coalition S_j^τ introduced by τ can be very expensive for large data sets

- Solution to both problems is sampling: Instead of averaging over $|P|! \cdot n$ terms, we approximate it using a limited amount of M random samples of τ to build coalitions S_j^τ
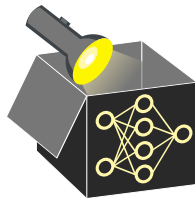
ESTIMATION: A PRACTICAL PROBLEM

- Exact Shapley value computation is problematic for high-dimensional feature spaces

  $\rightsquigarrow$ For 10 features, there are already $|P|! = 10! \approx 3.6$ million possible orders of features

- Additional problem due to estimation of the marginal prediction $\hat{f}_{S_j^\tau}$: Averaging over the entire data set for each coalition S_j^τ introduced by τ can be very expensive for large data sets

- Solution to both problems is sampling: Instead of averaging over $|P|! \cdot n$ terms, we approximate it using a limited amount of M random samples of τ to build coalitions S_j^τ

- M is a tradeoff between accuracy of the Shapley value and computational costs

  $\rightsquigarrow$ The higher M, the closer to the exact Shapley values, but the more costly the computation
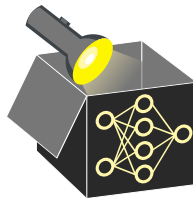
APPROXIMATION ALGORITHM

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

❶ For $m = 1, \ldots, M$ **do**:

APPROXIMATION ALGORITHM ▸ Strumbelj et al. (2014)

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

1. For $m = 1, \ldots, M$ **do**:
   1. Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$
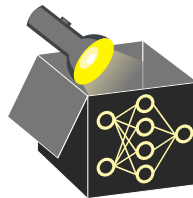
APPROXIMATION ALGORITHM ▶ Strumbelj et al. (2014)

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

1. For $m = 1, \ldots, M$ **do**:
   1. Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$
   2. Determine coalition $S_m := S_j^\tau$, i.e., the set of feat. before feat. j in order τ
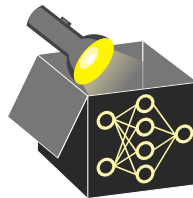
APPROXIMATION ALGORITHM ▸ Strumbelj et al. (2014)

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

1. For $m = 1, \ldots, M$ **do**:
   1. Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$
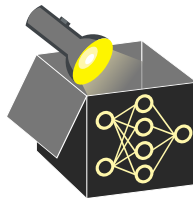   2. Determine coalition $S_m := S_j^\tau$, i.e., the set of feat. before feat. j in order τ
   3. Select random data point $\mathbf{z}^{(m)} \in \mathcal{D}$

APPROXIMATION ALGORITHM ▸ Strumbelj et al. (2014)

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

1. For $m = 1, \ldots, M$ **do**:

   1. Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$
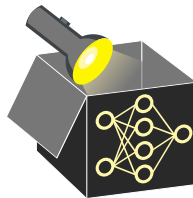   2. Determine coalition $S_m := S_j^\tau$, i.e., the set of feat. before feat. j in order τ
   3. Select random data point $\mathbf{z}^{(m)} \in \mathcal{D}$
   4. Construct two artificial obs. by replacing feature values from **x** with $\mathbf{z}^{(m)}$:

APPROXIMATION ALGORITHM ▸ Strumbelj et al. (2014)

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

1. For $m = 1, \ldots, M$ **do**:
   1. Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$
   2. Determine coalition $S_m := S_j^\tau$, i.e., the set of feat. before feat. j in order τ
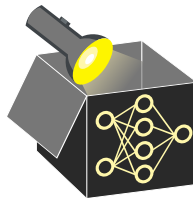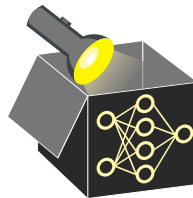   3. Select random data point $\mathbf{z}^{(m)} \in \mathcal{D}$
   4. Construct two artificial obs. by replacing feature values from **x** with $\mathbf{z}^{(m)}$:
      - $\mathbf{x}_{+j}^{(m)} = (\underbrace{x_{\tau^{(1)}}, \ldots, x_{\tau^{(|S_m|-1)}}, x_j}_{\mathbf{x}_{S_m \cup \{j\}}}, \underbrace{z_{\tau^{(|S_m|+1)}}^{(m)}, \ldots, z_{\tau^{(p)}}^{(m)}}_{\mathbf{z}_{-\{S_m \cup \{j\}\}}^{(m)}})$ takes features

      $S_m \cup \{j\}$ from **x**

APPROXIMATION ALGORITHM ▸ Strumbelj et al. (2014)

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

1 For $m = 1, \ldots, M$ **do**:

   1 Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$

   2 Determine coalition $S_m := S_j^\tau$, i.e., the set of feat. before feat. j in order τ

   3 Select random data point $\mathbf{z}^{(m)} \in \mathcal{D}$

   4 Construct two artificial obs. by replacing feature values from **x** with $\mathbf{z}^{(m)}$:

     • $\mathbf{x}_{+j}^{(m)} = (\underbrace{x_{\tau^{(1)}}, \ldots, x_{\tau^{(|S_m|-1)}}, x_j}_{\mathbf{x}_{S_m \cup \{j\}}}, \underbrace{z_{\tau^{(|S_m|+1)}}^{(m)}, \ldots, z_{\tau^{(p)}}^{(m)}}_{\mathbf{z}_{-\{S_m \cup \{j\}\}}^{(m)}})$ takes features

     $S_m \cup \{j\}$ from **x**

     • $\mathbf{x}_{-j}^{(m)} = (\underbrace{x_{\tau^{(1)}}, \ldots, x_{\tau^{(|S_m|-1)}}}_{\mathbf{x}_{S_m}}, \underbrace{z_j^{(m)}, z_{\tau^{(|S_m|+1)}}^{(m)}, \ldots, z_{\tau^{(p)}}^{(m)}}_{\mathbf{z}_{-S_m}^{(m)}})$ takes features

     S_m from **x**

APPROXIMATION ALGORITHM ▸ Strumbelj et al. (2014)

Estimation of ϕ_j for observation $\mathbf{x}$ of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:
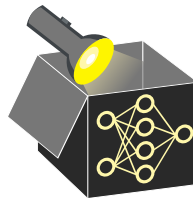
1. For $m = 1, \ldots, M$ **do**:
   1. Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$
   2. Determine coalition $S_m := S_j^\tau$, i.e., the set of feat. before feat. j in order τ
   3. Select random data point $\mathbf{z}^{(m)} \in \mathcal{D}$
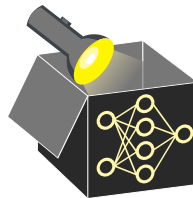   4. Construct two artificial obs. by replacing feature values from $\mathbf{x}$ with $\mathbf{z}^{(m)}$:
      - $\mathbf{x}_{+j}^{(m)} = (\underbrace{x_{\tau^{(1)}}, \ldots, x_{\tau(|S_m|-1)}, x_j}_{\mathbf{x}_{S_m \cup \{j\}}}, \underbrace{z_{\tau(|S_m|+1)}^{(m)}, \ldots, z_{\tau^{(p)}}^{(m)}}_{\mathbf{z}_{-\{S_m \cup \{j\}\}}^{(m)}})$ takes features
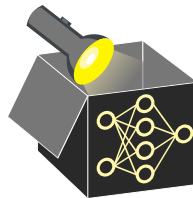
        $S_m \cup \{j\}$ from $\mathbf{x}$

      - $\mathbf{x}_{-j}^{(m)} = (\underbrace{x_{\tau^{(1)}}, \ldots, x_{\tau(|S_m|-1)}}_{\mathbf{x}_{S_m}}, \underbrace{z_j^{(m)}, z_{\tau(|S_m|+1)}^{(m)}, \ldots, z_{\tau^{(p)}}^{(m)}}_{\mathbf{z}_{-S_m}^{(m)}})$ takes features

        S_m from $\mathbf{x}$

   5. Compute difference $\phi_j^m = \hat{f}(\mathbf{x}_{+j}^{(m)}) - \hat{f}(\mathbf{x}_{-j}^{(m)})$

      $\rightsquigarrow \hat{f}_{S_m}(\mathbf{x}_{S_m})$ is approximated by $\hat{f}(\mathbf{x}_{-j}^{(m)})$ and $\hat{f}_{S_m \cup \{j\}}(\mathbf{x}_{S_m \cup \{j\}})$ by $\hat{f}(\mathbf{x}_{+j}^{(m)})$ over M iters.

APPROXIMATION ALGORITHM ▸ Strumbelj et al. (2014)

Estimation of ϕ_j for observation **x** of model $\hat{f}$ fitted on data $\mathcal{D}$ using sample size M:

1. For $m = 1, \ldots, M$ **do**:
   1. Select random order / perm. of feature indices $\tau = (\tau^{(1)}, \ldots, \tau^{(p)}) \in \Pi$
   2. Determine coalition $S_m := S_j^\tau$, i.e., the set of feat. before feat. j in order τ
   3. Select random data point $\mathbf{z}^{(m)} \in \mathcal{D}$
   4. Construct two artificial obs. by replacing feature values from **x** with $\mathbf{z}^{(m)}$:
      - $\mathbf{x}_{+j}^{(m)} = (\underbrace{x_{\tau^{(1)}}, \ldots, x_{\tau(|S_m|-1)}, x_j}_{\mathbf{x}_{S_m \cup \{j\}}}, \underbrace{z_{\tau(|S_m|+1)}^{(m)}, \ldots, z_{\tau^{(p)}}^{(m)}}_{\mathbf{z}_{-\{S_m \cup \{j\}\}}^{(m)}})$ takes features

        $S_m \cup \{j\}$ from **x**

      - $\mathbf{x}_{-j}^{(m)} = (\underbrace{x_{\tau^{(1)}}, \ldots, x_{\tau(|S_m|-1)}}_{\mathbf{x}_{S_m}}, \underbrace{z_j^{(m)}, z_{\tau(|S_m|+1)}^{(m)}, \ldots, z_{\tau^{(p)}}^{(m)}}_{\mathbf{z}_{-S_m}^{(m)}})$ takes features

        S_m from **x**

   5. Compute difference $\phi_j^m = \hat{f}(\mathbf{x}_{+j}^{(m)}) - \hat{f}(\mathbf{x}_{-j}^{(m)})$
      $\rightsquigarrow \hat{f}_{S_m}(\mathbf{x}_{S_m})$ is approximated by $\hat{f}(\mathbf{x}_{-j}^{(m)})$ and $\hat{f}_{S_m \cup \{j\}}(\mathbf{x}_{S_m \cup \{j\}})$ by $\hat{f}(\mathbf{x}_{+j}^{(m)})$
      over M iters.
2. Compute Shapley value $\phi_j = \frac{1}{M} \sum_{m=1}^{M} \phi_j^m$
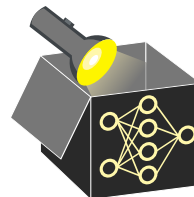
SHAPLEY VALUE APPROXIMATION - ILLUSTRATION

Definition

x: obs. of interest

x with feature values in S_m (other are replaced)

$$\phi_j(\mathbf{x}) = \frac{1}{M} \sum_{m=1}^{M} \left[\hat{f}(\mathbf{x}_{+j}{}^{(m)}) - \hat{f}(\mathbf{x}_{-j}{}^{(m)}) \right]$$

x with feature values in $S_m \cup \{j\}$

	Temperature	Humidity	Windspeed	Year
x	10.66	56	11	2012
x_{+j}	10.66	56	$random : z_{windspeed}^{(m)}$	2012
x_{-j}	10.66	56	$random : z_{windspeed}^{(m)}$	$random : z_{year}^{(m)}$

j

SHAPLEY VALUE APPROXIMATION - ILLUSTRATION

Definition

Contribution of feature j to coalition S_m

$$\phi_j(\mathbf{x}) = \frac{1}{M} \sum_{m=1}^{M} \underbrace{\left[\hat{f}(\mathbf{x}_{+j}^{(m)}) - \hat{f}(\mathbf{x}_{-j}^{(m)}) \right]}_{:= \Delta(j, S_m)}$$
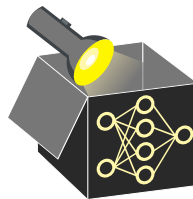
- $\Delta(j, S_m) = \hat{f}(\mathbf{x}_{+j}^{(m)}) - \hat{f}(\mathbf{x}_{-j}^{(m)})$ is the marginal contribution of feature j to coalition S_m
- Here: Feature *year* contributes +700 bike rentals if it joins coalition $S_m = \{temp, hum\}$

	Temperature	Humidity	Windspeed	Year	Count		
x	10.66	56	11	2012			
x_{+j}	10.66	56	$random : z_{windspeed}^{(m)}$	2012	5600		
x_{-j}	10.66	56	$random : z_{windspeed}^{(m)}$	$random : z_{year}^{(m)}$	4900	700	

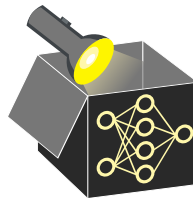j $\hat{f}$ $\Delta(j, S_m)$ marginal contribution

SHAPLEY VALUE APPROXIMATION - ILLUSTRATION

Definition

average the contributions of feature j

$$\phi_j(\mathbf{x}) = \frac{1}{M} \sum_{m=1}^{M} \left[\hat{f}(\mathbf{x}_{+j}^{(m)}) - \hat{f}(\mathbf{x}_{-j}^{(m)}) \right]$$

- Compute marginal contribution of feature j towards the prediction across all randomly drawn feature coalitions $S_1, \ldots, S_m$
- Average all M marginal contributions of feature j
- Shapley value ϕ_j is the payout of feature j, i.e., how much feature *year* contributed to the overall prediction in bicycle counts of a specific observation **x**
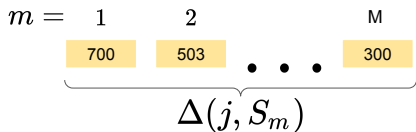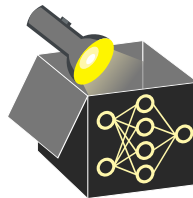
REVISITED: AXIOMS FOR FAIR ATTRIBUTIONS

We take the general axioms for Shapley Values and apply it to predictions:

- **Efficiency**: Shapley values add up to the (centered) prediction:
  $\sum_{j=1}^{p} \phi_j = \hat{f}(\mathbf{x}) - \mathbb{E}_{\mathbf{x}}(\hat{f}(X))$

REVISITED: AXIOMS FOR FAIR ATTRIBUTIONS

We take the general axioms for Shapley Values and apply it to predictions:
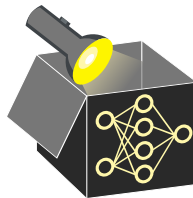
- **Efficiency**: Shapley values add up to the (centered) prediction:
  $\sum_{j=1}^{p} \phi_j = \hat{f}(\mathbf{x}) - \mathbb{E}_{\mathbf{x}}(\hat{f}(X))$

- **Symmetry**: Two features j and k that contribute the same to the prediction get the same payout
  $\rightsquigarrow$ interaction effects between features are fairly divided
  $\hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) = \hat{f}_{S \cup \{k\}}(\mathbf{x}_{S \cup \{k\}})$ for all $S \subseteq P \setminus \{j, k\}$ then $\phi_j = \phi_k$

REVISITED: AXIOMS FOR FAIR ATTRIBUTIONS

We take the general axioms for Shapley Values and apply it to predictions:

- **Efficiency**: Shapley values add up to the (centered) prediction:
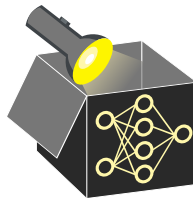 $\sum_{j=1}^{p} \phi_j = \hat{f}(\mathbf{x}) - \mathbb{E}_{\mathbf{x}}(\hat{f}(X))$

- **Symmetry**: Two features j and k that contribute the same to the prediction get the same payout
  $\rightsquigarrow$ interaction effects between features are fairly divided
  $\hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) = \hat{f}_{S \cup \{k\}}(\mathbf{x}_{S \cup \{k\}})$ for all $S \subseteq P \setminus \{j, k\}$ then $\phi_j = \phi_k$

- **Dummy / Null Player**: Shapley value of a feature that does not influence the prediction is zero $\rightsquigarrow$ if a feature was not selected by the model (e.g., tree or LASSO), its Shapley value is zero
  $\hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) = \hat{f}_{S}(\mathbf{x}_{S})$ for all $S \subseteq P$ then $\phi_j = 0$

REVISITED: AXIOMS FOR FAIR ATTRIBUTIONS

We take the general axioms for Shapley Values and apply it to predictions:

- **Efficiency**: Shapley values add up to the (centered) prediction:
  $\sum_{j=1}^{p} \phi_j = \hat{f}(\mathbf{x}) - \mathbb{E}_{\mathbf{x}}(\hat{f}(X))$

- **Symmetry**: Two features j and k that contribute the same to the prediction get the same payout
 $\rightsquigarrow$ interaction effects between features are fairly divided
  $\hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) = \hat{f}_{S \cup \{k\}}(\mathbf{x}_{S \cup \{k\}})$ for all $S \subseteq P \setminus \{j, k\}$ then $\phi_j = \phi_k$

- **Dummy** / **Null Player**: Shapley value of a feature that does not influence the prediction is zero $\rightsquigarrow$ if a feature was not selected by the model (e.g., tree or LASSO), its Shapley value is zero
  $\hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) = \hat{f}_{S}(\mathbf{x}_{S})$ for all $S \subseteq P$ then $\phi_j = 0$

- **Additivity**: For a prediction with combined payouts, the payout is the sum of payouts: $\phi_j(v_1) + \phi_j(v_2) \rightsquigarrow$ Shapley values for model ensembles can be combined
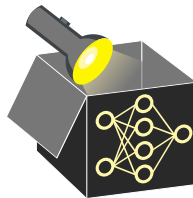
BIKE SHARING DATASET
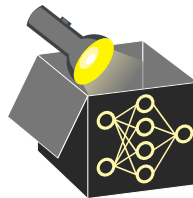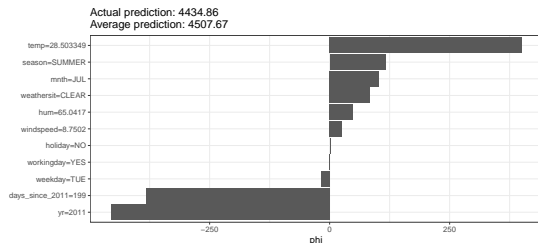
- Shapley values of observation $i = 200$ from the bike sharing data
- Difference between model prediction of this observation and the average prediction of the data is fairly distributed among the features (i.e., $4434 - 4507 \approx -73$)
- Feature value temp = 28.5 has the most positive effect, with a contribution (increase of prediction) of about +400

ADVANTAGES AND DISADVANTAGES

Advantages:

- **Solid theoretical foundation** in game theory
- Prediction is **fairly distributed** among the feature values ⤳ easy to interpret for a user
- **Contrastive explanations** that compare the prediction with the average prediction

Disadvantages:

- Without sampling, Shapley values need a lot of computing time to inspect all possible coalitions
- Like many other IML methods, Shapley values suffer from the inclusion of unrealistic data observations when features are correlated