Interpretable Machine Learning

Shapley Values

Learning goals

- Learn what game theory is
- Understand the concept behind cooperative games
- Understand the Shapley value in game theory

COOPERATIVE GAMES IN GAME THEORY

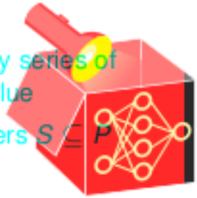
Shapley (1951)

- Game theory is the study of strategic games between players, "game" refers to any series of interactions between actors/agents with gains and losses of quantifiable utility value

Interpretable Machine Learning – 1 / 10

COOPERATIVE GAMES IN GAME THEORY

▸ Shapley (1951)

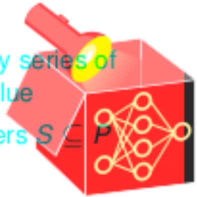


- Game theory is the study of strategic games between players, "game" refers to any series of interactions between actors/agents with gains and losses of quantifiable utility value
- Cooperative games: For all possible players $P = \{1, \ldots, p\}$, each subset of players $S \subseteq P$ forms a coalition – each coalition S achieves a certain payout

Interpretable Machine Learning – 1 / 10

COOPERATIVE GAMES IN GAME THEORY

▸ Shapley (1951)



- Game theory is the study of strategic games between players, "game" refers to any series of interactions between actors/agents with gains and losses of quantifiable utility value
- Cooperative games: For all possible players $P = \{1, \ldots, p\}$, each subset of players $S \subseteq P$ forms a coalition – each coalition S achieves a certain payout
- A value function $v : 2^P \mapsto \mathbb{R}$ maps all $2^{|P|}$ possible coalitions to their payout (or gain)

Interpretable Machine Learning – 1 / 10

COOPERATIVE GAMES IN GAME THEORY

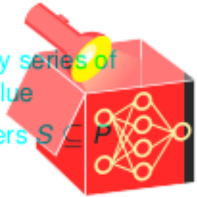
▸ Shapley (1951)

- Game theory is the study of strategic games between players, "game" refers to any series of interactions between actors/agents with gains and losses of quantifiable utility value
- Cooperative games: For all possible players $P = \{1, \dots, p\}$, each subset of players $S \subseteq P$ forms a coalition – each coalition S achieves a certain payout
- A value function $v : 2^P \mapsto \mathbb{R}$ maps all $2^{|P|}$ possible coalitions to their payout (or gain)
- $v(S)$ is the payout of coalition $S \subseteq P$ (payout of empty coalition must be zero: $v(\emptyset) = 0$)

Interpretable Machine Learning – 1 / 10

COOPERATIVE GAMES IN GAME THEORY

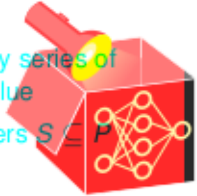
Shapley (1951)



- Game theory is the study of strategic games between players, “game” refers to any series of interactions between actors/agents with gains and losses of quantifiable utility value
- Cooperative games: For all possible players $P = \{1, \ldots, p\}$, each subset of players $S \subseteq P$ forms a coalition – each coalition S achieves a certain payout
- A value function $v : 2^P \mapsto \mathbb{R}$ maps all $2^{|P|}$ possible coalitions to their payout (or gain)
- $v(S)$ is the payout of coalition $S \subseteq P$ (payout of empty coalition must be zero: $v(\emptyset) = 0$)
- As some players contribute more than others, we want to fairly divide the total achievable payout $v(P)$ among the players according to a player's individual contribution

COOPERATIVE GAMES IN GAME THEORY

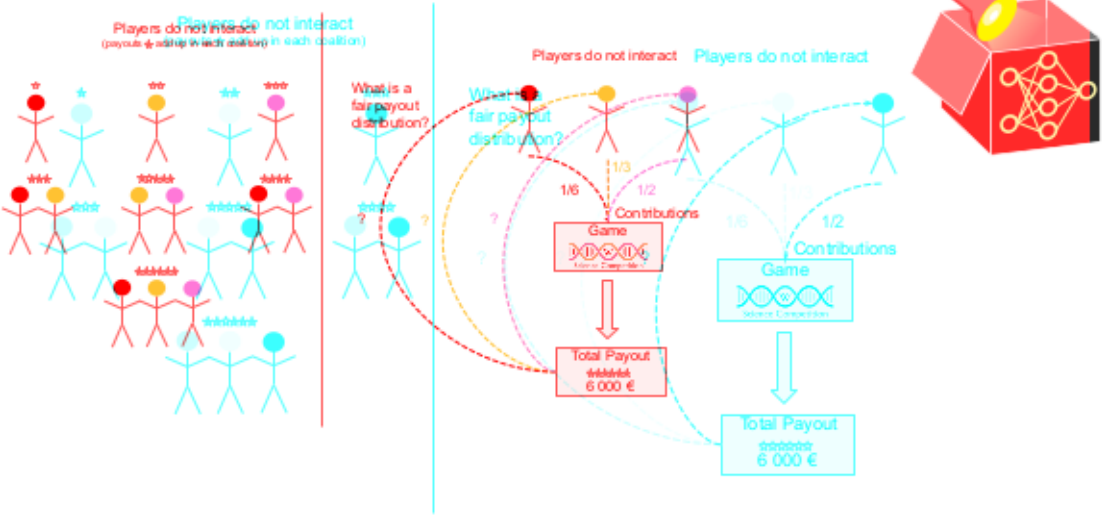
Shapley (1951)



- Game theory is the study of strategic games between players, "game" refers to any series of interactions between actors/agents with gains and losses of quantifiable utility value
- Cooperative games: For all possible players $P = \{1, \ldots, p\}$, each subset of players $S \subseteq P$ forms a coalition – each coalition S achieves a certain payout
- A value function $v : 2^P \mapsto \mathbb{R}$ maps all $2^{|P|}$ possible coalitions to their payout (or gain)
- $v(S)$ is the payout of coalition $S \subseteq P$ (payout of empty coalition must be zero: $v(\emptyset) = 0$)
- As some players contribute more than others, we want to fairly divide the total achievable payout $v(P)$ among the players according to a player's individual contribution
- We call the individual payout per player ϕ_j, $j \in P$ (later: Shapley value)

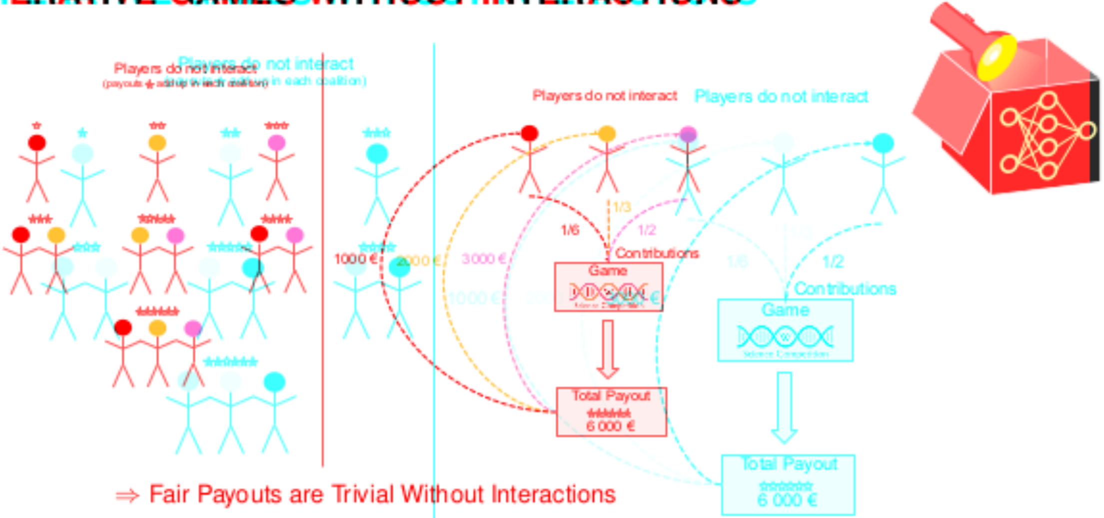
Interpretable Machine Learning – 1 / 10

COOPERATIVE GAMES WITHOUT INTERACTIONS



Interpretable Machine Learning – 2 / 10

COOPERATIVE GAMES WITHOUT INTERACTIONS

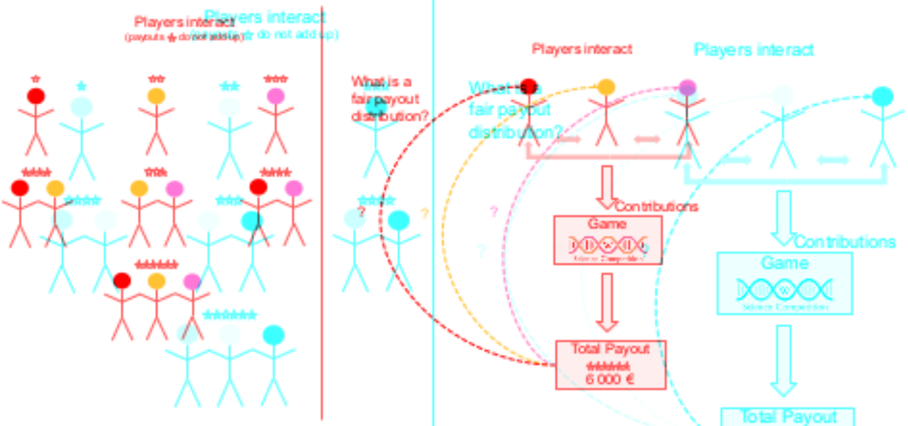


⇒ Fair Payouts are Trivial Without Interactions

⇒ Fair Payouts are Trivial Without Interactions

Interpretable Machine Learning – 2 / 10

COOPERATIVE GAMES WITH INTERACTIONS



⇒ Unclear how to fairly distribute payouts when players interact

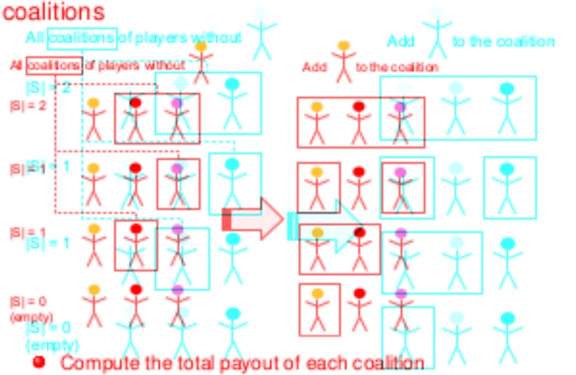
⇒ Unclear how to fairly distribute payouts when players interact

Interpretable Machine Learning – 3 / 10

COOPERATIVE GAMES WITH INTERACTIONS

Question: What is a fair payout for player "yellow"?

Idea: Compute marginal contribution of the player of interest across different coalitions

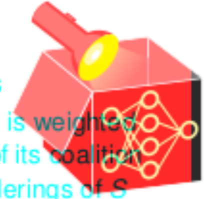


- Compute the total payout of each coalition
- Compute difference in payouts for each coalition with and without player "yellow" (= marginal contribution)
- Average marginal contributions using appropriate weights

- Compute the total payout of each coalition
- Compute difference in payouts for each coalition with and without player "yellow" (= marginal contribution)
- Average marginal contributions using appropriate weights

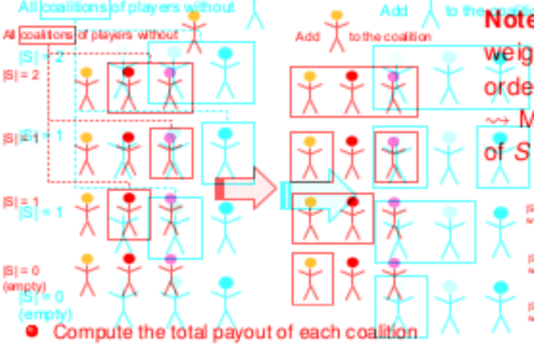
Interpretable Machine Learning – 4 / 10

COOPERATIVE GAMES WITH INTERACTIONS

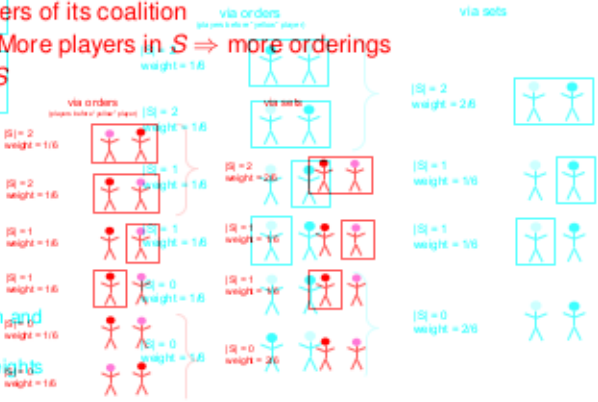


Question: What is a fair payout for player "yellow"?

Idea: Compute marginal contribution of the player of interest across different coalitions



Note: Each marginal contribution is weighted w.r.t. number of possible orders of its coalition
⇝ More players in $S \Rightarrow$ more orderings of S



- Compute the total payout of each coalition
- Compute difference in payouts for each coalition with and without player "yellow" (= marginal contribution)
- Average marginal contributions using appropriate weights

Question: What is a fair payout for player "yellow"?
Idea: Compute marginal contribution of the player of interest across different coalitions
Note: Each marginal contribution is weighted w.r.t. number of possible orders of its coalition ⇝ More players in S ⇒ more orderings of S
Compute the total payout of each coalition
Compute difference in payouts for each coalition with and without player "yellow" (= marginal contribution)
Average marginal contributions using appropriate weights
Interpretable Machine Learning – 4 / 10

SHAPLEY VALUE - SET DEFINITION

This idea refers to the **Shapley value** which assigns a payout value to each player according to its marginal contribution in all possible coalitions.

- Let $v(S \cup \{j\}) - v(S)$ be the marginal contribution of player j to coalition S
  $\rightsquigarrow$ measures how much a player j increases the value of a coalition S

Interpretable Machine Learning – 5 / 10

SHAPLEY VALUE - SET DEFINITION

This idea refers to the **Shapley value** which assigns a payout value to each player according to its marginal contribution in all possible coalitions.

- Let $v(S \cup \{j\}) - v(S)$ be the marginal contribution of player j to coalition S
  $\leadsto$ measures how much a player j increases the value of a coalition S
- Average marginal contributions for all possible coalitions $S \subseteq P \setminus \{j\}$
  $\leadsto$ order of how players join the coalition matters $\Rightarrow$ different weights depending on size of S

SHAPLEY VALUE - SET DEFINITION

This idea refers to the **Shapley value** which assigns a payout value to each player according to its marginal contribution in all possible coalitions.

- Let $v(S \cup \{j\}) - v(S)$ be the marginal contribution of player j to coalition S
  $\rightsquigarrow$ measures how much a player j increases the value of a coalition S
- Average marginal contributions for all possible coalitions $S \subseteq P \setminus \{j\}$
  $\rightsquigarrow$ order of how players join the coalition matters $\Rightarrow$ different weights depending on size of S
- Shapley value via **set definition** (weighting via multinomial coefficient):

$$\phi_j = \sum_{S \subseteq P \setminus \{j\}} \frac{|S|!(|P| - |S| - 1)!}{|P|!} (v(S \cup \{j\}) - v(S))$$

SHAPLEY VALUE - ORDER DEFINITION

The Shapley value was introduced as summation over sets $S \subseteq P \setminus \{j\}$, but it can be equivalently defined as a summation of all orders of players:

$$\phi_j = \frac{1}{|P|!} \sum_{\tau \in \Pi} \left(v(S_j^\tau \cup \{j\}) - v(S_j^\tau) \right)$$

- Π: All possible orders of players (we have $|P|!$ in total)

Interpretable Machine Learning – 6 / 10

SHAPLEY VALUE - ORDER DEFINITION

The Shapley value was introduced as summation over sets $S \subseteq P \setminus \{j\}$, but it can be equivalently defined as a summation of all orders of players:

$$\phi_j = \frac{1}{|P|!} \sum_{\tau \in \Pi} \left(v(S_j^\tau \cup \{j\}) - v(S_j^\tau) \right)$$

- Π: All possible orders of players (we have $|P|!$ in total)
- S_j^τ: Set of players before player j in order $\tau = (\tau^{(1)}, \dots, \tau^{(p)})$ where $\tau^{(i)}$ is i-th element
  $\Rightarrow$ Example: Players $1, 2, 3 \Rightarrow$
  $\Pi = \{(1,2,3), (1,3,2), (2,1,3), (2,3,1), (3,1,2), (3,2,1)\}$
  $\rightsquigarrow$ For order $\tau = (2,1,3)$ and player of interest $j = 3 \Rightarrow S_j^\tau = \{2,1\}$
  $\rightsquigarrow$ For order $\tau = (3,1,2)$ and player of interest $j = 1 \Rightarrow S_j^\tau = \{3\}$
  $\rightsquigarrow$ For order $\tau = (3,1,2)$ and player of interest $j = 3 \Rightarrow S_j^\tau = \emptyset$

SHAPLEY VALUE - ORDER DEFINITION

The Shapley value was introduced as summation over sets $S \subseteq P \setminus \{j\}$, but it can be equivalently defined as a summation of all orders of players:

$$\phi_j = \frac{1}{|P|!} \sum_{\tau \in \Pi} (v(S_j^\tau \cup \{j\}) - v(S_j^\tau))$$

- Π: All possible orders of players (we have $|P|!$ in total)
- S_j^τ: Set of players before player j in order $\tau = (\tau^{(1)}, \dots, \tau^{(p)})$ where $\tau^{(i)}$ is i-th element
  $\Rightarrow$ Example: Players $1, 2, 3 \Rightarrow$
  $\Pi = \{(1,2,3), (1,3,2), (2,1,3), (2,3,1), (3,1,2), (3,2,1)\}$
  $\rightsquigarrow$ For order $\tau = (2, 1, 3)$ and player of interest $j = 3 \Rightarrow S_j^\tau = \{2, 1\}$
  $\rightsquigarrow$ For order $\tau = (3, 1, 2)$ and player of interest $j = 1 \Rightarrow S_j^\tau = \{3\}$
  $\rightsquigarrow$ For order $\tau = (3, 1, 2)$ and player of interest $j = 3 \Rightarrow S_j^\tau = \emptyset$
- Order definition: Marginal contribution of orders that yield set $S = \{1, 2\}$ is summed twice
  $\rightsquigarrow$ In set definition, it has the weight $\frac{2!(3-2-1)!}{3!} = \frac{2 \cdot 0!}{6} = \frac{2}{6}$

SHAPLEY VALUE - COMMENTS ON ORDER DEFINITION

- Order and set definition are equivalent
- Reason: The number of orders which yield the same coalition S is $|S|!(|P|-|S|-1)!$
  - $\Rightarrow$ There are $|S|!$ possible orders of players within coalition S
  - $\Rightarrow$ There are $(|P|-|S|-1)!$ possible orders of players without S and j

$$\underbrace{\overbrace{\tau^{(1)} \;\; \ldots \;\; \tau^{(|S|)}}^{|S|!\text{ permutations}}}_{\text{Players before player } j} \;\; \underbrace{\tau^{(|S|+1)}}_{\text{player } j} \;\; \underbrace{\overbrace{\tau^{(|S|+2)} \;\; \ldots \;\; \tau^{(p)}}^{(|P|-|S|-1)!\text{ permutations}}}_{\text{Players after player } j}$$

SHAPLEY VALUE - COMMENTS ON ORDER DEFINITION

- Order and set definition are equivalent
- Reason: The number of orders which yield the same coalition S is $|S|!(|P| - |S| - 1)!$
  $\Rightarrow$ There are $|S|!$ possible orders of players within coalition S
  $\Rightarrow$ There are $(|P| - |S| - 1)!$ possible orders of players without S and j

$$\underbrace{\overbrace{\tau^{(1)} \;\big|\; \dots \;\big|\; \tau^{(|S|)}}^{|S|!\text{ permutations}}}_{\text{Players before player } j} \;\big|\; \underbrace{\tau^{(|S|+1)}}_{\text{player } j} \;\big|\; \underbrace{\overbrace{\tau^{(|S|+2)} \;\big|\; \dots \;\big|\; \tau^{(p)}}^{(|P|-|S|-1)!\text{ permutations}}}_{\text{Players after player } j}$$

- Relevance of the order definition: Approximate Shapley values by sampling permutations
  $\rightsquigarrow$ randomly sample a fixed number of M permutations and average them:

$$\phi_j = \frac{1}{M} \sum_{\tau \in \Pi_M} \left(v(S_j^\tau \cup \{j\}) - v(S_j^\tau) \right)$$

where $\Pi_M \subset \Pi$ is a random subset of Π containing only M orders of players

SHAPLEY VALUE - COMMENTS ON ORDER DEFINITION
Order and set definition are equivalent
Reason: The number of orders which yield the same coalition S is $|S|!(|P| - |S| - 1)!$
$\Rightarrow$ There are $|S|!$ possible orders of players within coalition S
$\Rightarrow$ There are $(|P| - |S| - 1)!$ possible orders of players without S and j
Relevance of the order definition: Approximate Shapley values by sampling permutations
$\rightsquigarrow$ randomly sample a fixed number of M permutations and average them:
where $\Pi_M \subset \Pi$ is a random subset of Π containing only M orders of players
Interpretable Machine Learning – 7 / 10

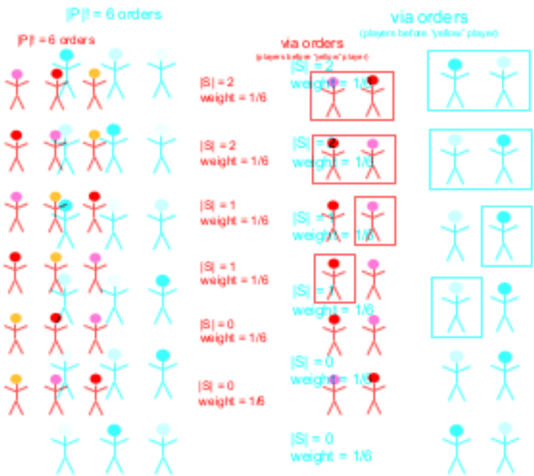
WEIGHTS FOR MARGINAL CONTRIBUTION - ILLUSTRATION

WEIGHTS FOR MARGINAL CONTRIBUTION - ILLUSTRATION

WEIGHTS FOR MARGINAL CONTRIBUTION - ILLUSTRATION

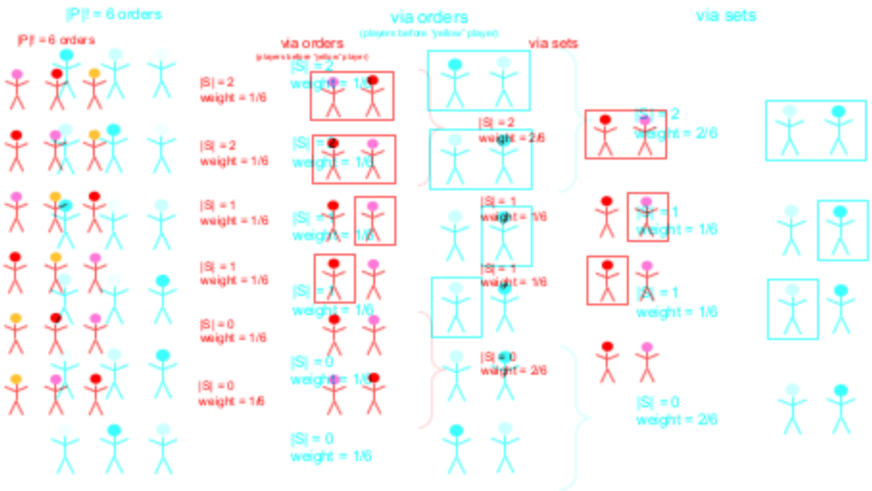
Interpretable Machine Learning – 8 / 10

WEIGHTS FOR MARGINAL CONTRIBUTION - ILLUSTRATION



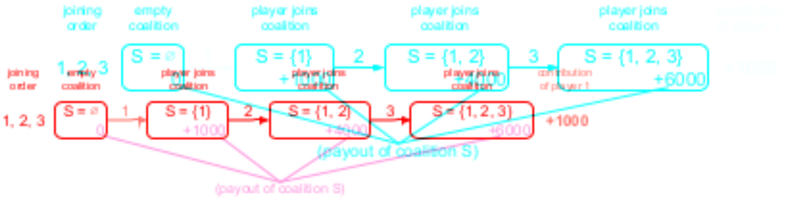
Interpretable Machine Learning – 8 / 10

WEIGHTS FOR MARGINAL CONTRIBUTION - ILLUSTRATION



SHAPLEY VALUES - ILLUSTRATION

- Shapley value of player j is the marginal contribution to the value when it enters any coalition
- Produce all possible joining orders of player coalitions



SHAPLEY VALUES - ILLUSTRATION

- Shapley value of player j is the marginal contribution to the value when it enters any coalition
- Produce all possible joining orders of player coalitions
- Measure and average the difference in payout after player 1 enters the coalition

joining order	empty coalition	player joins coalition	player joins coalition	player joins coalition	contribution of player 1
1, 2, 3	S = ∅ 0	1 → S = {1} +1000	2 → S = {1, 2} +4000	3 → S = {1, 2, 3} +6000	+1000
1, 3, 2	S = ∅ 0	1 → S = {1} +1000	3 → S = {1, 3} +4000	2 → S = {1, 2, 3} +6000	+1000
2, 1, 3	S = ∅ 0	2 → S = {2} +2000	1 → S = {1, 2} +4000	3 → S = {1, 2, 3} +6000	+2000
2, 3, 1	S = ∅ 0	2 → S = {2} +2000	3 → S = {2, 3} +3000	1 → S = {1, 2, 3} +6000	+3000
3, 1, 2	S = ∅ 0	3 → S = {3} +3000	1 → S = {1, 3} +4000	2 → S = {1, 2, 3} +6000	+1000
3, 2, 1	S = ∅ 0	3 → S = {3} +3000	2 → S = {2, 3} +3000	1 → S = {1, 2, 3} +6000	+3000

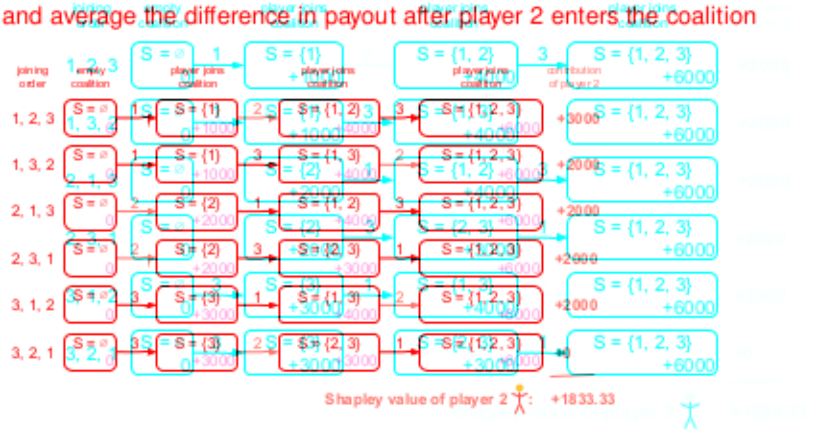
Shapley value of player 1: +1833.33

Interpretable Machine Learning – 9 / 10

SHAPLEY VALUES - ILLUSTRATION

- Shapley value of player j is the marginal contribution to the value when it enters any coalition
- Produce all possible joining orders of player coalitions
- Measure and average the difference in payout after player 2 enters the coalition

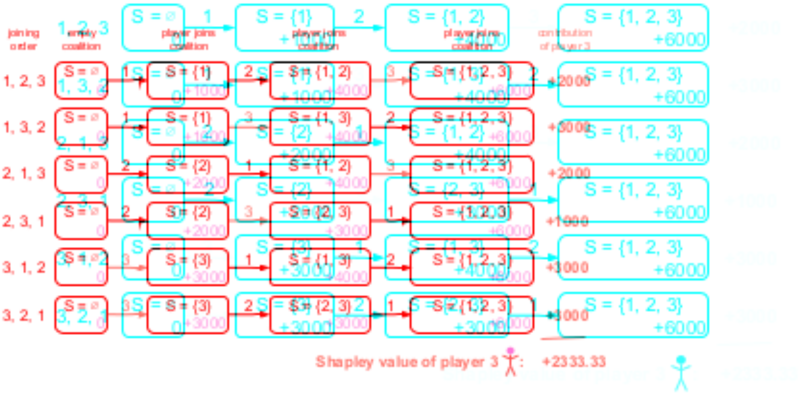
- Shapley value of player j is the marginal contribution to the value when it enters any coalition
- Produce all possible joining orders of player coalitions
- Measure and average the difference in payout after player 2 enters the coalition



Interpretable Machine Learning – 9 / 10

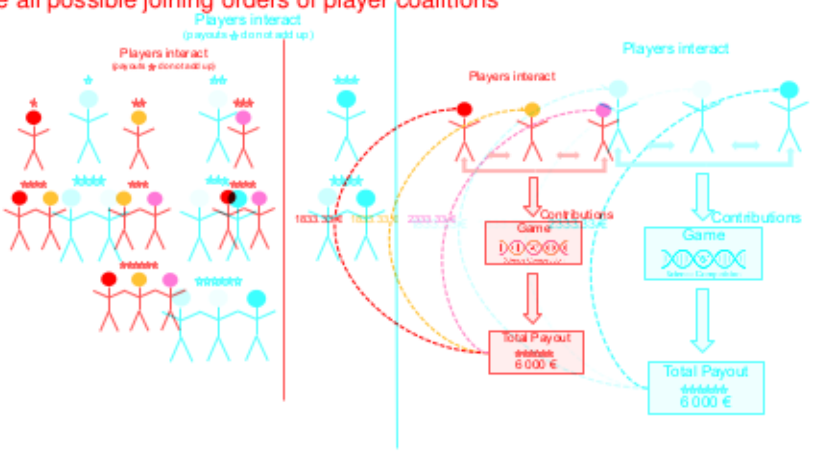
SHAPLEY VALUES - ILLUSTRATION

- Shapley value of player j is the marginal contribution to the value when it enters any coalition
- Produce all possible joining orders of player coalitions
- Measure and average the difference in payout after player 3 enters the coalition



SHAPLEY VALUES - ILLUSTRATION

- Shapley value of player j is the marginal contribution to the value when it enters any coalition
- Produce all possible joining orders of player coalitions



AXIOMS OF FAIR PAYOUTS

Why is this a fair payout solution?

One possibility to define fair payouts are the following axioms for a given value function v:

- **Efficiency**: Player contributions add up to the total payout of the game: $\sum_{j=1}^{p} \phi_j = v(P)$

Interpretable Machine Learning – 10 / 10

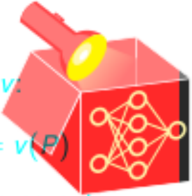
AXIOMS OF FAIR PAYOUTS

Why is this a fair payout solution?

One possibility to define fair payouts are the following axioms for a given value function v:

- **Efficiency**: Player contributions add up to the total payout of the game: $\sum_{j=1}^{p} \phi_j = v(P)$
- **Symmetry**: Players $j, k \in P$ who contribute the same to any coalition get the same payout:
  If $v(S \cup \{j\}) = v(S \cup \{k\})$ for all $S \subseteq P \setminus \{j, k\}$, then $\phi_j = \phi_k$

AXIOMS OF FAIR PAYOUTS



Why is this a fair payout solution?

One possibility to define fair payouts are the following axioms for a given value function v:

- **Efficiency**: Player contributions add up to the total payout of the game: $\sum_{j=1}^{p} \phi_j = v(P)$
- **Symmetry**: Players $j, k \in P$ who contribute the same to any coalition get the same payout:
  If $v(S \cup \{j\}) = v(S \cup \{k\})$ for all $S \subseteq P \setminus \{j, k\}$, then $\phi_j = \phi_k$
- **Dummy/Null Player**: Payout is 0 for players who don't contribute to the value of any coalition:
  If $v(S \cup \{j\}) = v(S) \quad \forall \quad S \subseteq P \setminus \{j\}$, then $\phi_j = 0$

Interpretable Machine Learning – 10 / 10

AXIOMS OF FAIR PAYOUTS

Why is this a fair payout solution?

One possibility to define fair payouts are the following axioms for a given value function v:

- **Efficiency**: Player contributions add up to the total payout of the game: $\sum_{j=1}^{p} \phi_j = v(P)$
- **Symmetry**: Players $j, k \in P$ who contribute the same to any coalition get the same payout:
  If $v(S \cup \{j\}) = v(S \cup \{k\})$ for all $S \subseteq P \setminus \{j, k\}$, then $\phi_j = \phi_k$
- **Dummy/Null Player**: Payout is 0 for players who don't contribute to the value of any coalition:
  If $v(S \cup \{j\}) = v(S) \quad \forall \quad S \subseteq P \setminus \{j\}$, then $\phi_j = 0$
- **Additivity**: For a game v with combined payouts $v(S) = v_1(S) + v_2(S)$, the payout is the sum of payouts: $\phi_{j,v} = \phi_{j,v_1} + \phi_{j,v_2}$

Interpretable Machine Learning – 10 / 10