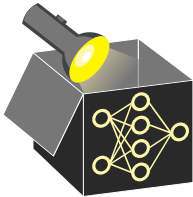


Interpretable Machine Learning

SHAP (SHapley Additive exPlanation) Values



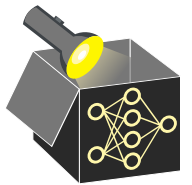
Learning goals

- Recall order- and set-based definitions of Shapley values in ML
- Interpret predictions via additive Shapley decomposition
- Understand SHAP as surrogate-based model
- Understand SHAP properties



Interpretable Machine Learning

Shapley SHAP (SHapley Additive exPlanation)



Learning goals

- Recall order- and set-based definitions of Shapley values in ML
- Interpret predictions via additive Shapley decomposition
- Understand SHAP as surrogate-based model
- Understand SHAP properties

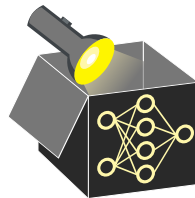


SHAPLEY VALUES IN ML - A SHORT RECAP

Shapley values (order definition): Average over marginal contributions across all permutations of feature indices $\tau \in \Pi$:

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

- For each permutation τ , determine coalition S_j^τ : features before j in τ
- In \hat{f}_S , features not in S are marginalized (e.g., replaced by random imputations)
- Compute marginal contribution of adding j to S_j^τ via the difference above
- Average over all $p!$ permutations (in practice, over $M \ll p!$)

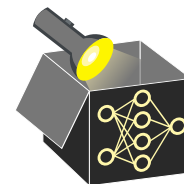


SHAPLEY VALUES IN ML - A SHORT RECAP

Shapley values (order definition): Average over marginal contributions across all permutations of feature indices $\tau \in \Pi$:

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

- For each permutation τ , determine coalition : features before j in τ
- In \hat{f}_S , features not in S are marginalized (e.g., randomly imputed)
- Compute marginal contribution of adding j to via the difference above
- Average over all $p!$ permutations (in practice, over $M \ll p!$)



SHAPLEY VALUES IN ML - A SHORT RECAP

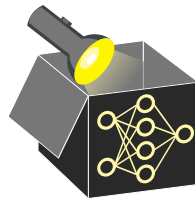
Shapley values (order definition): Average over marginal contributions across all permutations of feature indices $\tau \in \Pi$:

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

- For each permutation τ , determine coalition S_j^τ : features before j in τ
- In \hat{f}_S , features not in S are marginalized (e.g., replaced by random imputations)
- Compute marginal contribution of adding j to S_j^τ via the difference above
- Average over all $p!$ permutations (in practice, over $M \ll p!$)

Alternative (set definition): Average marginal contribution over all subsets, weighted by their relative number of appearances in permutations:

$$\phi_j(\mathbf{x}) = \sum_{S \subseteq \{1, \dots, p\} \setminus \{j\}} \frac{|S|!(p - |S| - 1)!}{p!} \left[\hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) - \hat{f}_S(\mathbf{x}_S) \right].$$



SHAPLEY VALUES IN ML - A SHORT RECAP

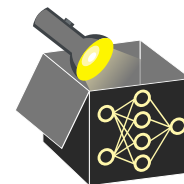
Shapley values (order definition): Average over marginal contributions across all permutations of feature indices $\tau \in \Pi$:

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

- For each permutation τ , determine coalition : features before j in τ
- In \hat{f}_S , features not in S are marginalized (e.g., randomly imputed)
- Compute marginal contribution of adding j to via the difference above
- Average over all $p!$ permutations (in practice, over $M \ll p!$)

Alternative (set definition): Average marginal contribution over all subsets, weighted by their relative number of appearances in permutations:

$$\phi_j(\mathbf{x}) = \sum_{S \subseteq \{1, \dots, p\} \setminus \{j\}} \frac{|S|!(p - |S| - 1)!}{p!} \left[\hat{f}_{S \cup \{j\}}(\mathbf{x}_{S \cup \{j\}}) - \hat{f}_S(\mathbf{x}_S) \right].$$



SHAPLEY VALUES IN ML - EXAMPLE

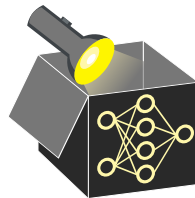
Example (Bike sharing data):

- Train random forest using humidity (hum), temperature (temp), windspeed (ws)
- Consider observation of interest \mathbf{x} with prediction $\hat{f}(\mathbf{x}) = 2573$
- Mean prediction $\mathbb{E}_{\mathbf{x}}[\hat{f}(\mathbf{x})] = 4515$
- Compute exact Shapley value for \mathbf{x} for feature hum:

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	temp, ws, hum	2623	2573	2/6

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

$$\Rightarrow \text{Analogously } \phi_{\text{temp}}(\mathbf{x}) = -1654, \phi_{\text{ws}}(\mathbf{x}) = -322$$



SHAPLEY VALUES IN ML - EXAMPLE

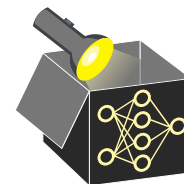
Example (Bike sharing data):

- Train random forest using humidity (hum), temperature (temp), windspeed (ws)
- Consider observation of interest \mathbf{x} with prediction $\hat{f}(\mathbf{x}) = 2573$
- Mean prediction $\mathbb{E}_{\mathbf{x}}[\hat{f}(\mathbf{x})] = 4515$
- Compute exact Shapley value for \mathbf{x} for feature hum:

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	temp, ws, hum	2623	2573	2/6

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

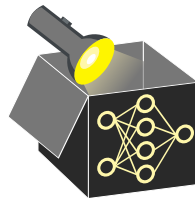
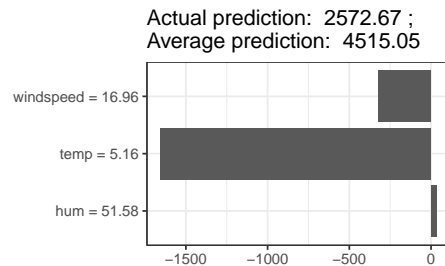
$$\Rightarrow \text{Analogously } \phi_{\text{temp}}(\mathbf{x}) = -1654, \phi_{\text{ws}}(\mathbf{x}) = -322$$



FROM SHAPLEY VALUES TO SHAP

Shapley value interpretation (for x):

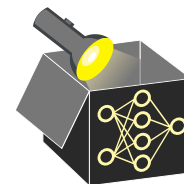
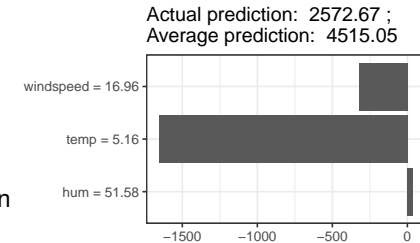
- `hum` (+34) pushes prediction *above* baseline (= average prediction).
- `temp` (−1654) and `ws` (−322) pull prediction *below* baseline.
- Together, they explain full deviation from average prediction.



FROM SHAPLEY VALUES TO SHAP

Shapley value interpretation (for x):

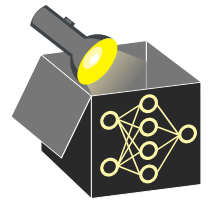
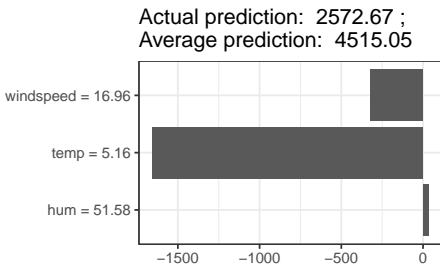
- `hum` (+34) pushes pred. *above* baseline (= average prediction).
- `temp` (−1654) and `ws` (−322) pull prediction *below* baseline.
- Together, they explain full deviation from average prediction.



FROM SHAPLEY VALUES TO SHAP

Shapley value interpretation (for \mathbf{x}):

- hum (+34) pushes prediction *above* baseline (= average prediction).
- temp (−1654) and ws (−322) pull prediction *below* baseline.
- Together, they explain full deviation from average prediction.



Shapley-based additive decomposition of prediction for \mathbf{x} gives insights on how features shift prediction from baseline $\mathbb{E}(\hat{f})$:

$$\underbrace{\hat{f}(\mathbf{x})}_{\text{actual prediction}} = \underbrace{\phi_0}_{\mathbb{E}_{\mathbf{x}}[\hat{f}(\mathbf{x})]} + \sum_{j \in \{\text{hum}, \text{temp}, \text{ws}\}} \phi_j(\mathbf{x})$$
$$2573 = 4515 + (34 - 1654 - 322) = 4515 - 1942$$

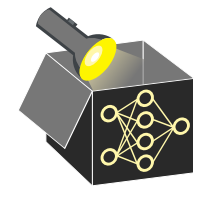
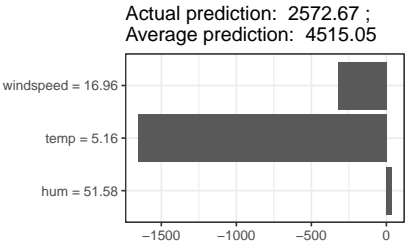
↪ Like a LM evaluated at \mathbf{x} : global intercept ϕ_0 plus per-feature contributions $\phi_j(\mathbf{x})$.

SHAP Motivation: Can we efficiently estimate this Shapley-based additive decomposition of $\hat{f}(\mathbf{x})$ using a surrogate model (while preserving Shapley axioms)?

FROM SHAPLEY VALUES TO SHAP

Shapley value interpretation (for \mathbf{x}):

- hum (+34) pushes pred. *above* baseline (= average prediction).
- temp (−1654) and ws (−322) pull prediction *below* baseline.
- Together, they explain full deviation from average prediction.



Shapley-based additive decomposition of prediction for \mathbf{x} gives insights on how features shift prediction from baseline $\mathbb{E}(\hat{f})$:

$$\underbrace{\hat{f}(\mathbf{x})}_{\text{actual prediction}} = \underbrace{\phi_0}_{\mathbb{E}_{\mathbf{x}}[\hat{f}(\mathbf{x})]} + \sum_{j \in \{\text{hum}, \text{temp}, \text{ws}\}} \phi_j(\mathbf{x})$$
$$2573 = 4515 + (34 - 1654 - 322) = 4515 - 1942$$

↪ Like a LM evaluated at \mathbf{x} : global intercept ϕ_0 plus per-feature contrihs $\phi_j(\mathbf{x})$.

SHAP Motivation: Can we efficiently estimate this Shapley-based additive decomp. of $\hat{f}(\mathbf{x})$ via a surrogate model (while preserving Shapley axioms)?

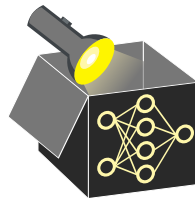
SHAP FRAMEWORK

► Lundberg et al. 2017

SHAP expresses the prediction of \mathbf{x} as a sum of contributions from each feature:

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

- $\mathbf{z}' \in \{0, 1\}^p$: simplified binary input referring to a coalition (coalition vector)
- $z'_j = 1$: feature j is "present" \Rightarrow use x_j in model evaluation
- $z'_j = 0$: feature j is "absent"
 \Rightarrow influence of x_j is removed via marginalization over a reference distribution



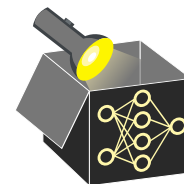
SHAP FRAMEWORK

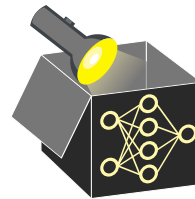
► LUNDBERG_2017

SHAP expresses the prediction of \mathbf{x} as a sum of contribs from each feature:

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

- $\mathbf{z}' \in \{0, 1\}^p$: simplified binary input referring to a coalition (coal. vector)
- $z'_j = 1$: feature j is "present" \Rightarrow use x_j in model evaluation
- $z'_j = 0$: feature j is "absent"
 \Rightarrow influence of x_j is removed via marginalization over a reference distrib.





SHAP expresses the prediction of \mathbf{x} as a sum of contributions from each feature:

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

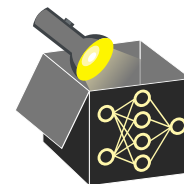
- $\mathbf{z}' \in \{0, 1\}^p$: simplified binary input referring to a coalition (coalition vector)
- $z'_j = 1$: feature j is "present" \Rightarrow use x_j in model evaluation
- $z'_j = 0$: feature j is "absent"
 \Rightarrow influence of x_j is removed via marginalization over a reference distribution

SHAP as a theoretical framework: Fit a surrogate model $g(\mathbf{z}')$ satisfying Shapley axioms and recovering $\hat{f}(\mathbf{x})$ when all features are "present":

$$\hat{f}(\mathbf{x}) = g(\mathbf{1}) = \phi_0 + \sum_{j=1}^p \phi_j$$

Evaluation of $g(\mathbf{z}')$: Let $S = \{j : z'_j = 1\}$ be the active coalition. Then:

- $g(\mathbf{z}') \approx \mathbb{E}[\hat{f}(\mathbf{X}) \mid \mathbf{X}_S = \mathbf{x}_S]$ (conditional expectation)
- $g(\mathbf{z}') \approx \mathbb{E}_{\mathbf{x}_{-S}}[\hat{f}(\mathbf{x}_S, \mathbf{x}_{-S})]$ (marginal expectation, i.e., PD function)
- *Note: Practical implementations (e.g., KernelSHAP) use the marginal expectation, approximated via random imputations from background data.*



SHAP expresses the prediction of \mathbf{x} as a sum of contribs from each feature:

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

- $\mathbf{z}' \in \{0, 1\}^p$: simplified binary input referring to a coalition (coal. vector)
- $z'_j = 1$: feature j is "present" \Rightarrow use x_j in model evaluation
- $z'_j = 0$: feature j is "absent"
 \Rightarrow influence of x_j is removed via marginalization over a reference distrib.

SHAP as a theoretical framework: Fit a surrogate model $g(\mathbf{z}')$ satisfying Shapley axioms and recovering $\hat{f}(\mathbf{x})$ when all features are "present":

$$\hat{f}(\mathbf{x}) = g(\mathbf{1}) = \phi_0 + \sum_{j=1}^p \phi_j$$

Evaluation of $g(\mathbf{z}')$: Let $S = \{j : z'_j = 1\}$ be the active coalition. Then:

- $g(\mathbf{z}') \approx \mathbb{E}[\hat{f}(\mathbf{X}) \mid \mathbf{X}_S = \mathbf{x}_S]$ (conditional expectation)
- $g(\mathbf{z}') \approx \mathbb{E}_{\mathbf{x}_{-S}}[\hat{f}(\mathbf{x}_S, \mathbf{x}_{-S})]$ (marginal expectation, i.e., PD function)
- *Note: Practical implementations (e.g., KernelSHAP) use the marginal expectation, approximated via random imputations from background data.*

SHAP FRAMEWORK

► Lundberg et al. 2017

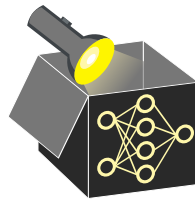
SHAP defines an additive surrogate $g(\mathbf{z}')$ over a binary simplified input $\mathbf{z}' \in \{0, 1\}^p$:

$\mathbf{z}'^{(k)}$: **coalition vector**
subset of features

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

ϕ_0 : **baseline** $\mathbb{E}[\hat{f}(\mathbf{X})]$

ϕ_j : **feature attribution**
marginal effect of j in
coalition



SHAP FRAMEWORK

► LUNDBERG_2017

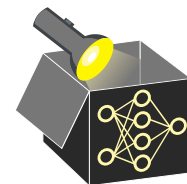
SHAP defines an additive surrogate $g(\mathbf{z}')$ over a binary input $\mathbf{z}' \in \{0, 1\}^p$:

$\mathbf{z}'^{(k)}$: **coalition vector**
subset of features

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

ϕ_0 : **baseline** $\mathbb{E}[\hat{f}()]$

ϕ_j : **feature attribution**
marginal effect of j in
coalition

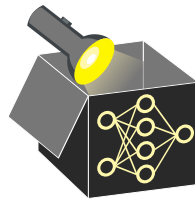


SHAP defines an additive surrogate $g(\mathbf{z}')$ over a binary simplified input $\mathbf{z}' \in \{0, 1\}^p$:

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

$g(\mathbf{z}'^{(k)})$: approx. prediction for coalition

ϕ_j : Shapley value



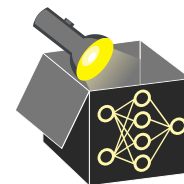
Next: How do we estimate the Shapley values ϕ_j efficiently?

SHAP defines an additive surrogate $g(\mathbf{z}')$ over a binary input $\mathbf{z}' \in \{0, 1\}^p$:

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

$g(\mathbf{z}'^{(k)})$: approx. prediction for coalition

ϕ_j : Shapley value



Next: How do we estimate the Shapley values ϕ_j efficiently?

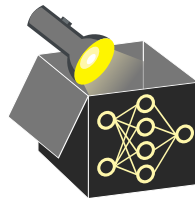
PROPERTIES

Local Accuracy

$$\hat{f}(\mathbf{x}) = g(\mathbf{z}') = \phi_0 + \sum_{j=1}^p \phi_j z'_j$$

Intuition: If the coalition includes all features $(\mathbf{z}' = (z'_1, \dots, z'_p)^\top = (1, \dots, 1)^\top)$, the attributions ϕ_j and the baseline ϕ_0 sum up to the original model output $\hat{f}(\mathbf{x})$

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



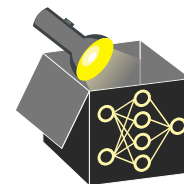
PROPERTIES

Local Accuracy

$$\hat{f}(\mathbf{x}) = g(\mathbf{z}') = \phi_0 + \sum_{j=1}^p \phi_j z'_j$$

Intuition: If coalition includes all features $(\mathbf{z}' = (z'_1, \dots, z'_p)^\top = (1, \dots, 1)^\top)$, the attributions ϕ_j and the baseline ϕ_0 sum up to the original model output $\hat{f}(\mathbf{x})$

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



PROPERTIES

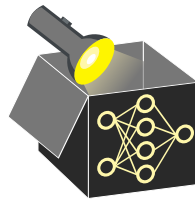
Local Accuracy

$$\hat{f}(\mathbf{x}) = g(\mathbf{z}') = \phi_0 + \sum_{j=1}^p \phi_j z'_j$$

Missingness

$$z'_j = 0 \implies \phi_j = 0$$

Intuition: A "missing" feature (whose value is imputed) gets an attribution of zero



PROPERTIES

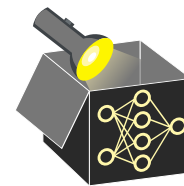
Local Accuracy

$$\hat{f}(\mathbf{x}) = g(\mathbf{z}') = \phi_0 + \sum_{j=1}^p \phi_j z'_j$$

Missingness

$$z'_j = 0 \implies \phi_j = 0$$

Intuition: A "missing" feature (whose value is imputed) gets zero attribution



PROPERTIES

Local Accuracy

$$\hat{f}(\mathbf{x}) = g(\mathbf{z}') = \phi_0 + \sum_{j=1}^p \phi_j z'_j$$

Missingness

$$z'_j = 0 \implies \phi_j = 0$$

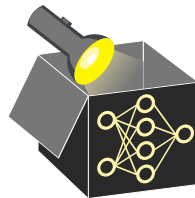
Consistency (Let $\mathbf{z}'_{-j}^{(k)}$ refer to $z_j^{(k)} = 0$)

For any two models \hat{f} and \hat{f}' , if for all inputs $\mathbf{z}'^{(k)} \in \{0, 1\}^p$

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

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



PROPERTIES

Local Accuracy

$$\hat{f}(\mathbf{x}) = g(\mathbf{z}') = \phi_0 + \sum_{j=1}^p \phi_j z'_j$$

Missingness

$$z'_j = 0 \implies \phi_j = 0$$

Consistency (Let $\mathbf{z}'_{-j}^{(k)}$ refer to $z_j^{(k)} = 0$)

For any two models \hat{f} and \hat{f}' , if for all inputs $\mathbf{z}'^{(k)} \in \{0, 1\}^p$

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

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

Consistency implies Shapley's axioms of **additivity, dummy, symmetry**.

