From Shapley Values to Explainable AI

Kyle Vedder - GRASP Game Theory Seminar

Video of this talk can be found at
https://www.youtube.com/watch?v=4RkhsIz14Yc

Background Definitions

- Permutation - order **does** matter
- Combination - order **does not** matter
- Power Set - Set of all subsets of a given set
    - Every combination of a given set or its subsets

Introduction to Shapley Values

Shapley Values

- Developed in the 1950s by Lloyd Shapley
- Helped win him the 2012 Nobel Prize in Economics along with Alvin Roth

Source: Wikipedia “Lloyd Shapley”

Farmer Example

- Fixed group of farmers work together to grow wheat
- Collaboration causes better (or worse) total yield of wheat than working individually
- **How do you assign “credit” to each farmer?**

Farmer Example - “Credit”

- Sum over each farmer’s credit is total wheat
  - “Efficiency”
- Farmer who contributes nothing get none
  - “Dummy”
- Equal contributions get equal credit
  - “Symmetry”

Farmer Example - “Credit” cont.

- If two harvests involving the same farmers merge, then the joint harvest credit is the sum of the farmer’s individual harvest credits
  - “Linearity”

Shapley Values
are these “credit” values

Shapley Values are these “credit” values

Any “credit” systems that uphold these properties must be Shapley Values!

Farmer Example - Permutations

- Let F be the set of farmers $\{1,2,...,p\}$
- Assume we have model v: $S_\sigma \longmapsto \mathbb{R}$
  - Input: $S \subseteq F$, σ is ordering of S
  - Output: Real value corresponding to wheat output
    - Farmers added in the order of σ
    - Different σ might change output

Farmer Example - Permutations

- To compute credit of farmer $i \in \{1,2,...p\}$:
    - For every permutation of farmers that aren't *i*, sum the difference between wheat output with *i* and without *i* (marginal value of *i*)

Farmer Example - Permutations

$$\underbrace{\phi_i}_{\text{Farmer } i \text{ Credit}} = \underbrace{\frac{1}{|F|!}}_{\text{Average over \# permutations}} \sum_{S \subseteq F \setminus \{i\}} \sum_{\sigma} \left(v\left(S_\sigma \cup \{i\}\right) - v\left(S_\sigma\right)\right)$$

Farmer Example - Permutations

$$\underbrace{\phi_i}_{\text{Farmer } i \text{ Credit}} = \frac{1}{\underbrace{|F|!}_{\text{Normalize by \# permutations}}} \underbrace{\sum_{S \subseteq F \setminus \{i\}} \sum_{\sigma} \left(v\left(S_\sigma \cup \{i\}\right) - v\left(S_\sigma\right) \right)}_{\text{\#P Hard}}$$

“As hard as the counting problems associated with NP hard problems”
e.g. #SAT, exact Bayes net inference, matrix permanent

Glove Game Example

- Goal is to form maximal pairs of gloves
  - P1 has *L*, P2 has *L*, P3 has *R*
- $v(S) = 1$ if S is {1,3}, {2,3}, {1,2,3}, 0 otherwise

Order R	Marginal Contribution of P1
$1, 2, 3$	$v(\{1\}) - v(\varnothing) = 0 - 0 = 0$
$1, 3, 2$	$v(\{1\}) - v(\varnothing) = 0 - 0 = 0$
$2, 1, 3$	$v(\{1, 2\}) - v(\{2\}) = 0 - 0 = 0$
$2, 3, 1$	$v(\{1, 2, 3\}) - v(\{2, 3\}) = 1 - 1 = 0$
$3, 1, 2$	$v(\{1, 3\}) - v(\{3\}) = 1 - 0 = 1$
$3, 2, 1$	$v(\{1, 2, 3\}) - v(\{2, 3\}) = 1 - 1 = 0$

Thus:
- $\phi_1 = 1/6$
- $\phi_2 = 1/6$
- $\phi_3 = 2/3$

From: https://en.wikipedia.org/wiki/Shapley_value#Glove_game

Glove Game Example

- Goal is to form maximal pairs of gloves
  - P1 has *L*, P2 has *L*, P3 has *R*
- $v(S) = 1$ if S is {1,3}, {2,3}, {1,2,3}, 0 otherwise

Order R	Marginal Contribution of P1
$1, 2, 3$	$v(\{1\}) - v(\varnothing) = 0 - 0 = 0$
$1, 3, 2$	$v(\{1\}) - v(\varnothing) = 0 - 0 = 0$
$2, 1, 3$	$v(\{1, 2\}) - v(\{2\}) = 0 - 0 = 0$
$2, 3, 1$	$v(\{1, 2, 3\}) - v(\{2, 3\}) = 1 - 1 = 0$
$3, 1, 2$	$v(\{1, 3\}) - v(\{3\}) = 1 - 0 = 1$
$3, 2, 1$	$v(\{1, 2, 3\}) - v(\{2, 3\}) = 1 - 1 = 0$

From: https://en.wikipedia.org/wiki/Shapley_value#Glove_game

Thus:

- $\phi_1 = \frac{1}{6}$
- $\phi_2 = \frac{1}{6}$
- $\phi_3 = \frac{2}{3}$

- Efficiency
- Dummy
- Symmetry
- Linearity

Farmer Example - Combinations

- Let F be the set of farmers $\{1,2,...,p\}$
- Assume we have model $f: S \longmapsto \mathbb{R}$
  - Input: $S \subseteq F$
  - Output: Real value corresponding to wheat output
    - Averaged over result for every ordering of farmers
    - Hides some combinatorics
    - If order doesn’t matter, can save computation

Farmer Example - Combinations

- To compute credit of farmer $i \in \{1,2,\ldots p\}$:
  - For every <u>permutation</u> of farmers that aren't *i*, sum the difference between wheat output with *i* and without *i* (marginal value of *i*)
  - Implement via *f* evaluated on every <u>combination</u> of farmers; let *f* handle the ordering

Farmer Example - Combinations

$$\underbrace{\phi_i}_{\text{Farmer } i \text{ Credit}} = \underbrace{\sum_{S \subseteq F \setminus \{i\}}}_{\text{Powerset of Farmers without Farmer } i} \underbrace{\frac{|S|!(|F| - |S| - 1)!}{|F|!}}_{\text{Combinatorial Normalization}} \underbrace{\left[f(x_{S \cup \{i\}}) - f(x_S)\right]}_{f \text{ evaluated with } S \text{ and } i\text{, minus } f \text{ evaluated with } S}$$

Farmer Example - Combinations

ways to permute proceeding farmers

ways to permute succeeding farmers

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

Normalize by # permutations

Unordered subset (combination)

Handles ordering of S

Benefits of Shapley Values

- Efficiency
- Dummy
- Symmetry
- Linearity

Problems with Computing SVs

- Computation is #P Hard
  - Very expensive in practice
- Must be able to include/exclude farmers and get a meaningful real value
  - How do we do this for non-artificial games?

From Shapley Values to SHAP Values

SHAP Values

- SHapley Additive exPlanations
  - Introduced by Lundberg et. al. 2017
- Tackles the problems of SV computation
  - Data table to bypass full definition of *v*
  - Data structures to speed computation
- Extends SVs to apply to general ML

SHAP From Tables

T =

F1	F2	…	F*p*	Yield (y)
1	4	…	7	27
1	4	…	1	9
0	0	…	3	8

N

$\mathbb{R}^p$

$\mathbb{R}$

SHAP From Tables

- f: partial assignment of p features $\mapsto \mathbb{R}$
  - Assignment means feature *value*
  - Need to handle the unassigned features

SHAP From Tables - Example

- $f(\{F1 = 1\}) = ?$

F1	F2	...	F*p*	Yield (y)
1	4	...	7	27
1	4	...	1	9
0	0	...	3	8

SHAP From Tables - Nominal

- $f(\{F1 = 1\})$ = avg(T | F1 = 1, F2 = **0**, …, Fp = **0**)

F1	**F2**	**…**	**F*p***	**Yield (y)**
1	4	…	7	27
1	4	…	1	9
0	0	…	3	8

SHAP From Tables - Nominal

- Might not have table entries
- **Not** guaranteed to uphold **any** Shapley properties

SHAP From Tables - Marginal

- $f(\{F1 = 1\})$ = avg(T | F1 = 1)

F1	**F2**	**...**	**F*p***	**Yield (y)**
1	4	...	7	27
1	4	...	1	9
0	0	...	3	8

SHAP From Tables - Marginal

- Proposed by Lundberg et. al. 2017
- Impacted by sparsity
  - Conditional dist. might differ from full dist.
  - Distribution can collapse with real-world (noisy) data
- Upholds Efficiency and Symmetry, not Dummy and Linearity
  - Sundararajan et. al. 2019

SHAP From Tables - Interventional

- $f(\{F1 = 1\})$ = avg(T | *do*(F1 = 1))

F1	**F2**	**…**	**F*p***	**Yield (y)**
1	4	…	7	27
1	4	…	1	9
1	0	…	3	8

SHAP From Tables - Interventional

- Proposed by Janzing et. al. 2019
- *do* notation by Judea Pearl
  - Breaks feature correlations
  - Implies that the model is causal
- Upholds Efficiency, Dummy, Linearity, not Symmetry

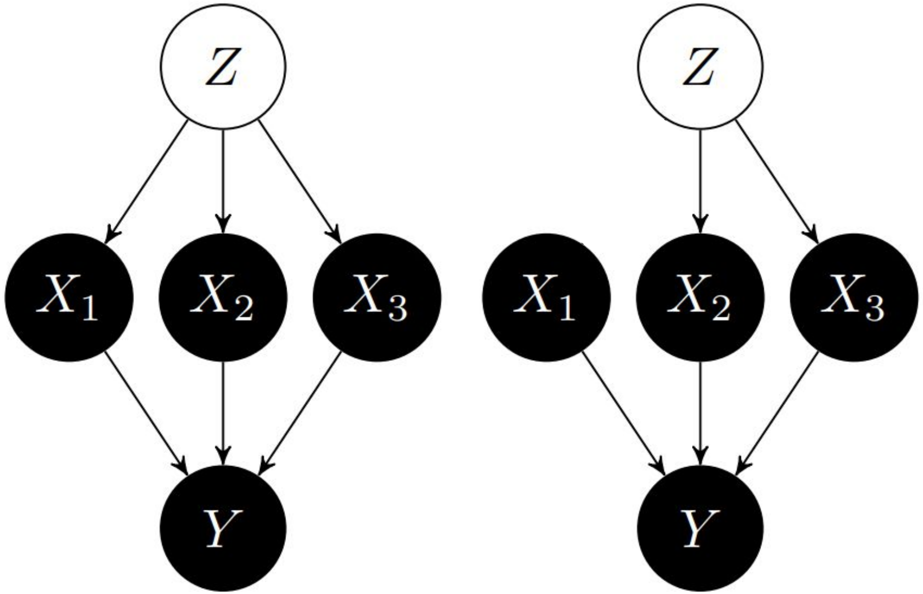
do notation

- Assumes model is causal
- Y | X1 = v

  ≠

  Y | *do*(X1 = v)

From Janzing et. al. 2019



Y | X1 = v

Y | *do*(X1 = v)

SHAP From Tables - Interventional

- Proposed by Janzing et. al. 2019
- *do* notation by Judea Pearl
  - Breaks feature correlations
  - Implies that the model is causal
- Upholds Efficiency, Dummy, Linearity, not Symmetry

Applying SHAP

Applying SHAP

T =

F1	**F2**	**...**	**F*p***	**y**
1	4	...	7	27
1	4	...	1	9
0	0	...	3	8

N

$\mathbb{R}^p$ $\mathbb{R}$

Applying SHAP

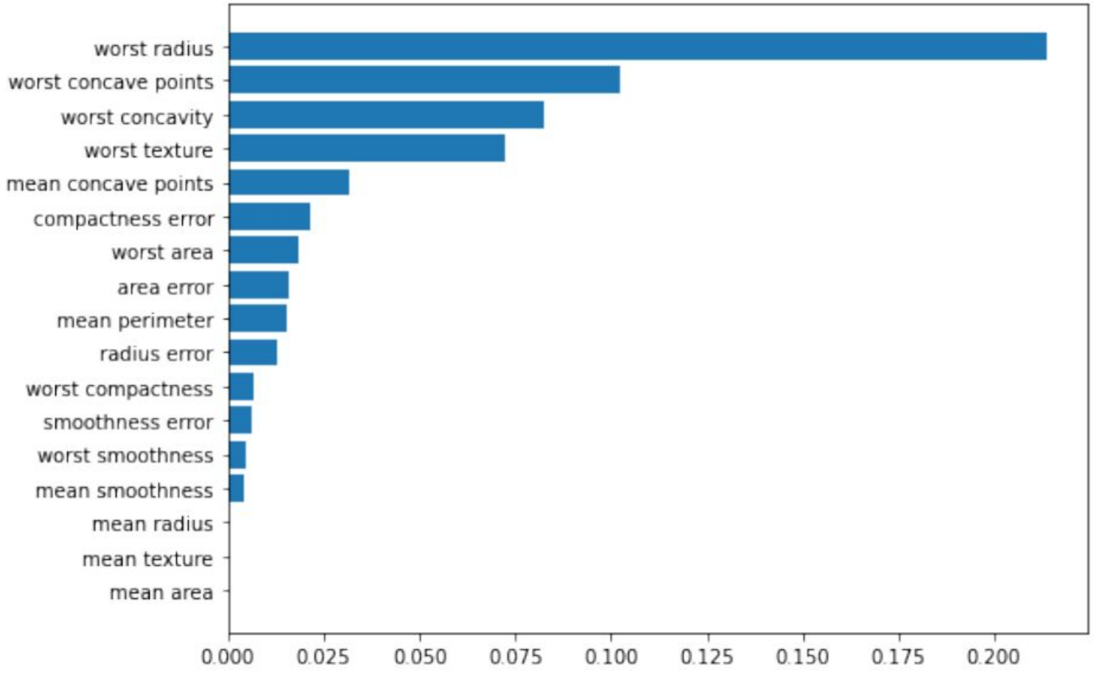
$$\Phi =$$

F1	F2	...	Fp	y
$\varphi_{1,1}$	$\varphi_{1,2}$	...	$\varphi_{1,p}$	12.33
$\varphi_{2,1}$	$\varphi_{2,2}$	...	$\varphi_{2,p}$	-5.66
$\varphi_{3,1}$	$\varphi_{3,2}$	...	$\varphi_{3,p}$	-6.66

N

$\mathbb{R}^p$

Applying SHAP



Sum of |φ| for each feature

Making SHAP Tractable

Tractability

- Enabled more flexible definitions of *f*
    - Forgo some guarantees for flexible definition
- Need to fix runtime

Data Structures for faster SHAP

- TreeSHAP
  - Lundberg et. al. 2020
- Supports Marginal
  - Impl. supports interventional
- Open source impl.