On the Tractability of SHAP Explanations

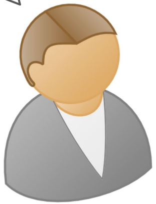
Guy Van den Broeck, Anton Lykov, Maximilian Schleich, Dan Suciu

UCLA Samueli Computer Science

W PAUL G. ALLEN SCHOOL OF COMPUTER SCIENCE & ENGINEERING

Motivation: Explainable AI

But why?



Loan?

“Sorry, can’t do that …”

F() = no loan

Instance **x**
with Features X1, ... ,Xn

Input

Prediction F(**x**) **+ Explanation**

Classifier F

We study:
Computational Complexity of **SHAP Explanations**

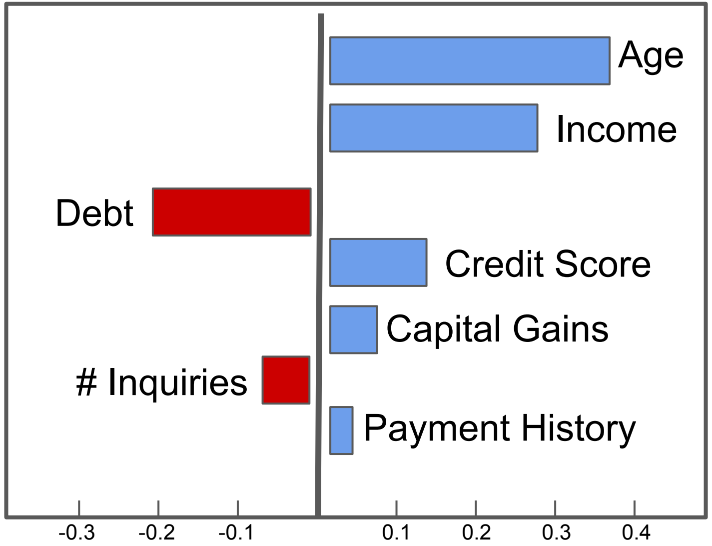
What are SHAP explanations?

Feature-Based Attribution Score

- How much does ith feature influence F(**x**)?
- Based on Shapley values from Game Theory

Benefits

- Model-agnostic
- Intuitive
- Successfully applied in practice



Computing SHAP Explanations

Intuition:

- Assume a total order π of the features
- Compute effect on **E**[F] of presenting one feature at a time following π

Example:

- Assume π = [*X*1,*X*2,...,*Xn*]
- Contribution of X2 w.r.t. π

$$c_\pi(X2) = \mathbf{E}[F \mid X1, X2] - \mathbf{E}[F \mid X1]$$

SHAP-score for X2:

Average contribution of X2 over all possible permutations

$$SHAP_{F,\mathbf{x}}(X2) = \frac{1}{n!}\sum_\pi c_\pi(X2)$$

The Challenge

Various algorithms proposed to compute SHAP explanations:

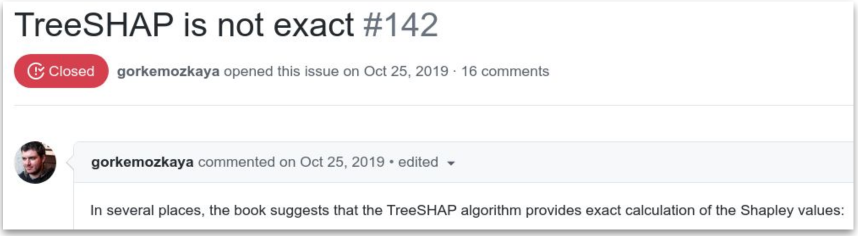
approximately, exactly, efficiently, …, for different machine learning models

There is considerable confusion about the tractability of computing SHAP explanations

- Are the exact algorithms exact, correct, and efficient?
- Are the approximations needed?

<u>Example</u>: TreeSHAP [ICML 2017]

How can we clear this up?



The Main Actors

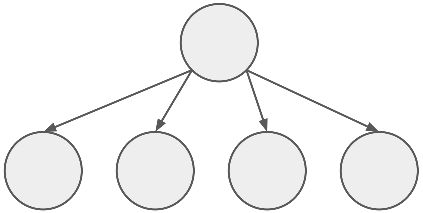
1. The machine learning model class for function F

Linear regression, decision and regression trees, random forests, additive tree ensembles, logistic regression, neural nets with sigmoid activation functions, naive Bayes classifiers, factorization machines, regression circuits, logistic circuits, Boolean functions in d-DNNF, binary decision diagrams, bounded treewidth Boolean functions in CNF, Boolean functions in CNF or DNF, and arbitrary functions

2. The data distribution Pr to compute $\mathbf{E}[F|\mathbf{y}] = \sum_{\mathbf{x}} \Pr(\mathbf{x}|\mathbf{y})\ F(\mathbf{x})$

Fully-factorized distributions

Graphical models (naive Bayes)



Empirical data distribution

Summary of our contributions

SHAP is ***tractable*** on:

Distribution Pr	Predictive model F
Fully-factorized	Linear regression
	Decision and regression trees
	Random forests, additive tree ensembles
	Factorization machines, regression circuits
	Boolean functions in d-DNNF, BDDs
	Bounded treewidth Boolean functions in CNF

SHAP is ***intractable*** on:

Distribution Pr	Predictive model F
Fully-factorized	Logistic regression
	Neural Nets with sigmoid activation functions
	NB classifiers, logistic circuits
	Boolean funcs in CNF or DNF
Naive Bayes, Bayes Nets, Factor Graphs, Probabilistic Circuits, etc.	All classes of functions*
Empirical	Any (empirical) function

*That contain some function F' that depends only on one of the features

Fully-factorized distributions

Key result:

For any classifier F, the following problems have the same complexity:

- Computing **SHAP** explanations of F
- Computing the expectation **E** of F

Expectations **E** are efficient to compute for

- linear regression
- decision trees, random forests, additive tree ensembles
- Boolean functions in d-DNNF form, bounded-treewidth CNF
- … and more

therefore

SHAP explanations are efficient to compute on those same models!

Fully-factorized distributions

Key result:

For any classifier F, the following problems have the same complexity:

- Computing **SHAP** explanations of F
- Computing the expectation **E** of F

We prove that expectations **E** are #P-hard to compute for

- logistic regression
- naive Bayes classifiers
- neural networks with sigmoid activations
- Boolean functions in CNF or DNF

therefore

SHAP explanations are #P-hard to compute on those same models!

Intuition: Expectation of Logistic Regression

Consider the number partitioning problem for {1,2,3,2}

- {1,3} and {2,2} partition the set into subsets with the same sum
- Counting such partitions is #P-hard

Consider the logistic regression model:

F(**X**) = sigmoid(1000 X1 + 2000 X2 + 3000 X3 + 2000 X4 - 4500)

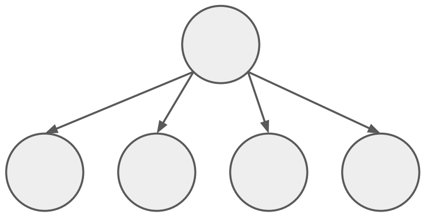
- **x** = [1,1,0,1] and **x'** = [0,0,1,0] correspond to non-partitions: F(**x**) ≈ 1 and F(**x'**) ≈ 0
- Under a uniform distribution **E**[F] ≈ 0.5
- **x** = [1,0,1,0] and **x'** = [0,1,0,1] correspond to partitions: F(**x**) = F(**x'**) ≈ 0
- Missing probability mass 0.5 - **E**[F] tells us how many partitions there are
- Computing **E**[F] is #P-hard

Going Beyond Fully-Factorized Distributions

<u>Idea</u>: the real world is not fully-factorized: features depend on each other

Consider the simplest case:

1. Simplest possible classifier: F(**X**) = X1
2. Simplest tractable distribution: naive Bayes



SHAP explanations are NP-hard to compute.

SHAP explanations are NP-hard to compute for all probabilistic graphical models, even all tractable probabilistic models, even on simple function classes

Trivial function classes do not make **SHAP** tractable...

Empirical Distributions

Idea: Properties of distributions are often estimated on sampled data.
Perhaps the empirical data distribution is easier to work with?

The # of possible worlds is limited by the number of rows (samples) in data

Computing **SHAP** is #P-hard in the size of the empirical distribution.

The problem that TreeSHAP is trying to solve efficiently is in fact #P-hard

Proof sketch

- Associate a PP2CNF logical sentence **Φ** with the data matrix
- Computing **E[Φ]** under a quasi-symmetric distribution is #P-hard (Provan and Ball, 1983)
- SHAP(F, X) ≡ **E[Φ]**

Summary of Contributions

	Distribution Pr		
Predictive Model F	Fully Factorized	Naive-Bayes	Empirical
Linear regression Regression circuits Factorization machines	**Tractable**	**Intractable**	**Intractable**
Decision Tree Random Forest,Boosted Tree	**Tractable**	**Intractable**	**Intractable**
Boolean functions in d-DNNF, BDD, Bounded treewidth CNF	**Tractable**	**Intractable**	**Intractable**
Logistic regression Logistic circuits, Naive Bayes	**Intractable**	**Intractable**	**Intractable**
Neural Networks with sigmoid activation	**Intractable**	**Intractable**	**Intractable**

- Proved connections between SHAP and the expectation of classifiers
- … and more theoretical insights of independent interest

Thank you!