Using SHAP Post-Model Explainability as a Model-Agnostic Feature Selection Technique

•••

Joshua Gottlieb, Martin Sichali Chunsen, and Alexander Yoo

Advised by Dr. Krishna Bathula

Seidenberg Annual Research Day 2025

Feature Selection and SHAP Overview

Feature selection techniques in data science help reduce dimensionality of datasets, remove sparsity, and eliminate noisy patterns, thereby improving generalization.

SHAP is a machine learning explainability technique that is used to evaluate predictions for black-box models. SHAP is an additive model, so each feature is given a SHAP value, similar to the coefficients for a linear model. These coefficients can be used to produce feature rankings, allowing SHAP to be used for feature selection.

Previous work using SHAP as a feature selection technique failed to provide good methods for choosing feature subsets without expensive iterative processes. We aimed to solve this problem with two novel selection algorithms, MAX and SUM, which are designed for simplicity and grounded in intuitive feature selection goals. We tested our algorithms across 5 model types and 10 datasets, to ensure robustness of the technique.

MAX and SUM Algorithms for Feature Selection

The MAX strategy frames features selection in the context of *relative feature importance* by only keeping features that are as strong as a proportion of the strongest feature.

Algorithm 1 MAX SHAP Psuedocode

Sort the feature strengths I_j in descending order: $I_j \geq I_{j+1}$
Choose $M = \max(I_j) = I_0$
Define a proportion constant ρ in [0, 1].
Initialize an index $j = 0$.
while $I_j \geq \rho \cdot M$ **do**
  Increment j
end while
return $[f_0, \ldots, f_j]$, the selected sorted features.

The SUM strategy frames feature selection in the context of *total data explanation* by only keeping features with summed importance equal to a proportion of the total summed importances.

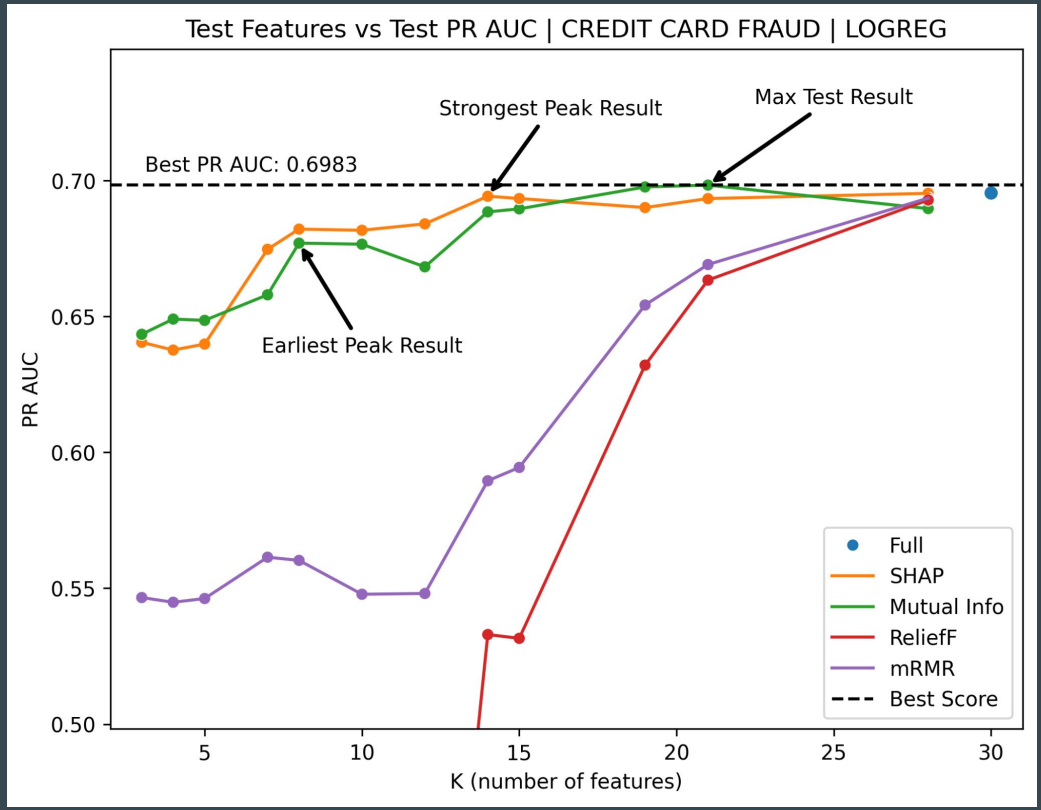
Algorithm 2 SUM SHAP Pseudocode

Sort the feature strengths I_j in descending order: $I_j \geq I_{j+1}$
Calculate the total sum $S = \sum I_j$
Define a proportion constant r in [0, 1].
Initialize a running sum s and an index $j = 0$.
while $s < r \cdot S$ **do**
  Add I_j to s and increment j
end while
return $[f_0, \ldots, f_j]$, the selected sorted features.

Frameworks for Selecting the Best Feature Subsets from Rankings

Four frameworks were considered for choice of best feature subset.

- Minimum Overfitting
- Maximum Test Result
- Strongest Peak Result
- Earliest Peak Result



Performance Counts and Mean Features Kept by Technique

Counts and Mean Features Kept by Framework, Global Counts								
Selection Type	Min Overfit Count	Min Overfit Mean K%	Max Test Count	Max Test Mean K%	Strongest Peak Count	Strongest Peak Mean K%	Earliest Peak Count	Earliest Peak Mean K%
mRMR	5	21.36	5	75.76	7	74.85	4	72.46
Mutual Information	9	37.51	10	54.59	11	54.05	12	46.28
ReliefF	14	20.18	2	90.00	0	0.00	2	49.85
SHAP	10	16.42	24	46.21	25	43.60	26	39.38

Across 50 dataset-model combinations, SHAP performed the best for 3 out of 4 frameworks. SHAP was often the best or second best selection technique for all dataset-model combinations. SHAP consistently keeps the least amount of features on average, indicating that SHAP rankings are superior in quality.

MAX and SUM Threshold Counts Across Datasets and Models

SHAP MAX and SUM Threshold Counts by Framework

Selection Type	Min Overfit Count	Max Test Count	Strongest Peak Count	Earliest Peak Count
Max = 0.01	2	15	10	10
Max = 0.05	4	10	8	3
Max = 0.1	7	6	6	5
Max = 0.15	3	2	5	7
Max = 0.25	4	1	1	2
Max = 0.5	14	4	3	3
Sum = 0.5	6	0	0	0
Sum = 0.6	4	0	1	4
Sum = 0.7	1	2	2	2
Sum = 0.8	2	4	7	8
Sum = 0.9	1	1	2	1
Sum = 0.95	2	5	5	5

- MAX with threshold 0.01 showed the best test performance across 3 out of 4 frameworks.
- MAX with threshold 0.5 best minimized overfitting.
- SUM performs worse compared to MAX on most dataset-model combinations at all thresholds.

Conclusion

Our experiments show that SHAP is an effective feature selection technique. SHAP feature subsets exhibit equivalent or superior performance to feature subsets selected by common filter-based techniques, while resulting in greater feature reduction.

The MAX and SUM algorithms we created are effective in choosing the number of features in a simple, intuitive, and non-iterative manner. The MAX strategy was especially effective and naturally filters out weak features by directly enforcing feature relevance. MAX thresholds between 0.01 and 0.1 are ideal for most datasets.

Future improvements include updating the SUM algorithm to filter out weak features to increase its effectiveness and the creation of automatic or semi-automatic modes for our MAX and SUM algorithms that require less human input.