From explanations to feature selection: assessing SHAP values as feature selection mechanism

Wilson E. Marcilio-Jr and Danilo M. Eler
São Paulo State University - Department of Mathematics and Computer Science
Presidente Prudente, São Paulo/Brazil
Email: wilson.marcilio@unesp.br, danilo.eler@unesp.br

***Abstract*—Explainability has become one of the most discussed topics in machine learning research in recent years, and although a lot of methodologies that try to provide explanations to black-box models have been proposed to address such an issue, little discussion has been made on the pre-processing steps involving the pipeline of development of machine learning solutions, such as feature selection. In this work, we evaluate a game-theoretic approach used to explain the output of any machine learning model, SHAP, as a feature selection mechanism. In the experiments, we show that besides being able to explain the decisions of a model, it achieves better results than three commonly used feature selection algorithms.**

I. Introduction

Working with high-dimensional datasets has become a common task for anyone working with data. While offering great opportunities to discover patterns and tendencies, dealing with high-dimensional data can be complicated due to the so-called curse of dimensionality. Essentially, the redundancy of the dataset increases as its dimensionality increases, which can impair the performance of various techniques. To overcome such issues, dimensionality reduction techniques, such as t-SNE [3] or UMAP [4] can be applied to reduce dimensionality while maintaining as much of information as possible. One problem of such algorithms is that they remove the interpretability of the features (if they were interpretable at first) by applying series of non-linear equations and may introduce artifacts that were not perceived in the high-dimensional space. Other approaches to deal with high dimensionality is to use feature selection algorithms, which *select* a subset of variables that can describe the input data while proving good results in prediction [5]. In this case, it is necessary to define a metric (or selection criteria) in which the feature selection will be based [6], for example, the correlation among features. Feature selection methods are commonly divided into three categories: filter, wrapper, and embedded.

One problem with traditional feature selection algorithms is related to their explainability issues. For example, when working with clinical data, how to explain that a few features were simply removed from the provided dataset? Each category of feature selection algorithm has its weakness on how to explain why certain features were chosen without diving into the mathematical formulation. That is, *filter methods* do not leverage a model's characteristic to filter the features; *wrapper methods* do leverage a model's prediction, however, to choose a subset of feature only based on accuracy or another scoring technique has the same problem of trying to choose a model for a task (e.g., in finance or clinical) only based on these metrics; finally, although calculated as a part of the training process, embedded methods have to be incorporated based on each model particularity, which could be difficult and tedious to provide explanations for every different model.

In this work, we provide a methodology and assessment for feature selection based on model agnostic explanations, produced by a novel approach know as SHAP [1]. The approach assigns SHAP values, which are contribution values for a model's output, for each feature of each data point. These SHAP values encode the importance that a model gives for a feature, so that, we use the contribution information of each feature to order the features based on its importance. In this case, selecting a subset of d features based on SHAP values means to select the first d features after ordering them based on the feature contributions to the model's prediction. We validate our methodology on classification and regression tasks upon eight publicly available datasets. Experiments against three common feature selection algorithms show that feature selection based on SHAP values presents the best results. Summarily, the contributions of this work are:

- Assessment of SHAP as feature selection mechanism;
- A library with Python implementation of the methodology[1].

This paper is organized as follows: in Section II we describe the related works; in Section III we provide a brief explanation of SHAP and delineate the methodology to perform feature selection based on it; experiments are provided in Section IV; we discuss the results in Section V; the work is concluded in Section VI.

II. Related Works

As the dimensionality of datasets grows, the redundancy of the data becomes a problem since with too many dimensions, every data point in a dataset appears equidistant from the others [7]. To reduce problems like these, or simply to filter out features that are not useful for a machine learning problem and can introduce artifacts to the dataset while demanding higher time execution, the number of features must be reduced.

Feature selection algorithms are usually classified into three groups: filter, wrapper, and embedded [6], [8], [9]. *Filter*

[1]https://github.com/wilsonjr/SHAP_FSelection

Fig. 1. Explanation generated by SHAP [1] visualized with force plot [2] - x axis: data points; y axis: SHAP values/model's output. **A**. The explanation for one single data point of the dataset – see the arrow coming from a specific index. The SHAP values encode the feature's contribution to the model's prediction according to the base value, i.e., summing all the SHAP values for a data point to the base value will result in the model's prediction. **B**. The explanation for the whole Boston dataset using a force plot. **B**

.

methods select a subset of the most important features from a dataset as a pre-processing step independently from the classifier [10], e.g., one could look the correlation among features to filter out the ones with higher correlation. *Wrapper* methods act like surrogate models where subsets of features are evaluated upon their predictive power [11], [12], for example, features whose changing does not imply significant changes in the model's output could be removed from the subset of *important* features. Finally, embedded methods perform the feature selection during training, which makes them specific to a model itself [12], [13], e.g., one of the most common strategies for network pruning is to remove neurons whose weights are very close to zero [14] since they do not contribute to the model's prediction.

In the group of *filter* methods, the correlation criteria that use Pearson correlation coefficient [5] and Mutual Information (MI) [15], [16] are usually the most employed feature selection mechanisms. However, the problem with these approaches is that the correlation ranking can only detect linear dependencies between a feature and the model's output [6] while for MI-based ranking, the inter-feature MI is not taken into account [17]. From the perspective of *wrapper* methods, care must be taken since evaluating the 2^m subsets of features is an NP-hard problem. So that, heuristics are usually employed to find sub-optimal subsets such as using tree structures to evaluate different subsets in the Branch and Bound method [18]–[20], or evolutionary algorithms, such as Genetic Algorithms [21]–[24] or Particle Swarm Optimization [25]–[28] to compute solutions with computational feasibility. Finally, embedded methods try to remove the time required for refitting the models as seen in the wrapper methods. In this case, the selection is held during the training process. Such methods use the different incremental estimation of Mutual Information [16], [29], [30] or classifiers' weights [31] to classify features based on importance and perform feature selection. For example, the concept of weights is used to rank features and applied to SVM classifier [30] to perform Recursive Feature Selection. Finally, a multi-layer perceptron network could have its nodes pruned after a penalty is applied to node connections whose weights are closer to zero.

Differently from the methods discussed above, here we proposed to use SHAP, the state-of-the-art method for model-agnostic interpretation, as a feature selection mechanism. The motivation to use such an approach as feature selection is based on the need for model interpretation that has been growing in the past years. In this case, we justify that an important pre-processing of constructing machine learning solutions could also be interpreted, and machine learning practitioners would be able to explain all of their decisions when building machine learning solutions.

III. Background and Methodology

In this section, we provide a brief background on SHAP values and then show how to use them to explain machine learning models based on predictions. Finally, we discuss how to employ SHAP values as a feature selection mechanism.

A. Background

SHAP values [1] is a model addictive explanation approach, in which each prediction is explained by the contribution of the features of the dataset to the model's output. More specifically, SHAP approximate Shapley values [32], a concept from game theory that is the solution for the problem of computing the contribution to a model's prediction of every subset of features given a dataset with m features. While computing the exact solution of Shapley values would be infeasible – due to the exponential nature of the problem – SHAP approximate the solution through special weighted linear regression [1] for any model or throughout different assumptions about feature dependence for ensemble tree models [33].

In linear regression models, the coefficients used to weight the features are used to explain the predictions for all data points, however, it does not account for the heterogeneity of individual data observations. In most of the cases, however, the effect of a feature for a data point could be different from another data point. This is consistent with the fact that local explanations are more accurate than global explanations. This is similar to the idea to approximate global similarities throughout a series of local similarities, as done by non-linear dimensionality reduction methods. SHAP explores and uses the property of local explainability to build surrogate models to black-box machine learning models. In this case, SHAP slightly changes the input and test the changes in prediction, if the model prediction does not change much by slightly changing the input value for a feature, that feature for that particular data point may not be an important predictor.

The sum of the contributions, or SHAP values, of each feature, is equal to the final prediction. In this case, a SHAP value is **not** the difference between the prediction with and without a feature, but it is a contribution of a feature to the difference between the actual prediction and the mean prediction. Fig. 1 shows how SHAP values can be used to provide understanding about the model's functionality for a single data point and the whole Boston dataset after training with XGBoost Regressor [34]. The figure shows negative (in blue) and positive (in red) SHAP values that decrease and increase the model's prediction. These forces balance each other at the actual prediction (output value) of the data instance starting from the average of all predictions (base value). In the explanation for the whole dataset (see Fig. 1**B**), a matrix with the same dimension of the dataset is generated, where every entry i, j contains a SHAP value of feature j for the data point i. Such a matrix is used in the following section in the feature selection methodology.

B. Methodology

To evaluate SHAP as a feature selection approach, we simply use the feature contribution information as delineated in Fig. 2. First, a SHAP values matrix ($E^c_{n\times m}$) is generated for each class (c) of the dataset – the matrix encodes the feature contributions for each data point –, then, the mean of the columns of each matrix is calculated. The vectors of mean SHAP values for each class are summed and ordered in a decreasing way. The first position of the resulting vector contains the most import feature, the second position contains the second most important, and so on. Note that in Fig. 2 blue and red colors indicate SHAP values, however, the semantic meaning is different since red colors depict real numbers and blue colors depict importance.

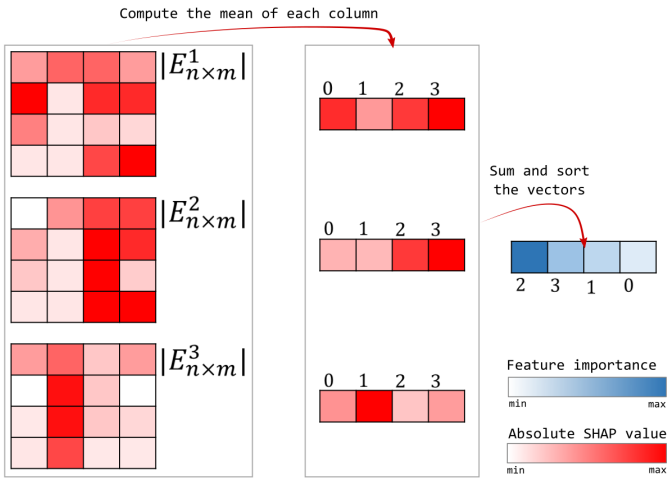


Fig. 2. Calculating the importance of the features based on SHAP contributions. The mean of each feature is retrieved for each SHAP matrix. which corresponds to each class. Then, the resulting vectors are summed. The summed vectors are ordered in a decreasing way to encode feature importance.

We reason that since SHAP can provide means of interpretability of a model's decisions by indicating the importance of the dataset features, a feature selection algorithm based on the most important features according to the absolute SHAP values would provide good results.

IV. Experiments

This section presents the application of SHAP values as a feature selection method. Thus, we tested SHAP against three common feature selection approaches: Mutual Information [35]–[37], Recursive Feature Elimination [31], and ANOVA [38]. The algorithms were evaluated upon eight publicly available datasets, described in Table I, and based on the **Keep Absolute** metric [33], which computes a model score on varying number of features kept for classification/regression. For example, the F1-Score could be used to evaluate a classification model with 10%, 20%, or 30% of the features, which are previously ordered according to their importance – i.e., with 10% of the features, the model would be evaluated with 10% most important features. The remaining of the features are masked, that is, all of its values are changed to their

respective column mean. Fig. 3 shows a schematic view of the metric.

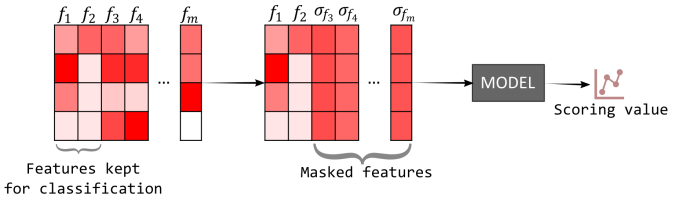


Fig. 3. The Keep Absolute metric. To evaluate how well a feature selection technique can select important features, the model is retrained with d features kept for classification and $m - d$ features masked, where d is the number of features to select and m is the dimensionality of the dataset.

The experiments were performed in a computer with the following configuration: Intel(R) Core(TM) i7-8700 CPU @ 3.20GHz, 32GB RAM, Windows 10 64 bits. Here, we used the same XGBoost [34] Classifier and Regressor (an implementation of gradient boosted decision trees) for performing the tasks with hyper-parameters tuned by grid-search, also indicated in Table I, where α stands for *learning rate*, β stands for *max. depth*, and ω stands for *number of estimators*. Moreover, since we are using boosted trees, we picked the version of SHAP designed to work with tree methods (Tree SHAP), which produce explanations in a reduced amount of time if compared with the model-agnostic version of SHAP.

A. Classification

We used 5-fold cross-validation with F1-score as scoring to evaluate the feature selection methods according to the **Keep Absolute** metric. Table II shows the Area Under the Curve (AUC) for each combination of dataset and feature selection technique. Note that higher values mean that the technique was able to retrieve the best combination of features that would increase the score (F1-score, in this case). We can see that using SHAP as a mechanism to feature selection yielded the higher scores (highlighted in bold), besides being very consistent among the results. Taking the RFE technique as example, it presented the second-best results for *Wine*, *Vertebral Column*, and *Breast Cancer* datasets, however, it also presented the *worst* results for the *Indian Liver Disease* and *Heart Disease* datasets. The good results of the SHAP technique could be explained by the fact that besides informing the importance of the features, as the other techniques do, it adds a certain rigor to the importance of the features by trying to explain why certain decisions were made by the model throughout the feature contributions, that can be different for *each data point*.

To take a closer look at the decision made by the feature selection algorithms, it is possible to look at the curves of F1-scores generated by the **Keep Absolute** metric, i.e., the curves used to generate the values of Table II. These curves are shown in Fig. 4 for each feature selection technique, where the mean of 5-fold cross-validation is indicated by the lines, and the standard deviation is indicated by the areas with the same color.

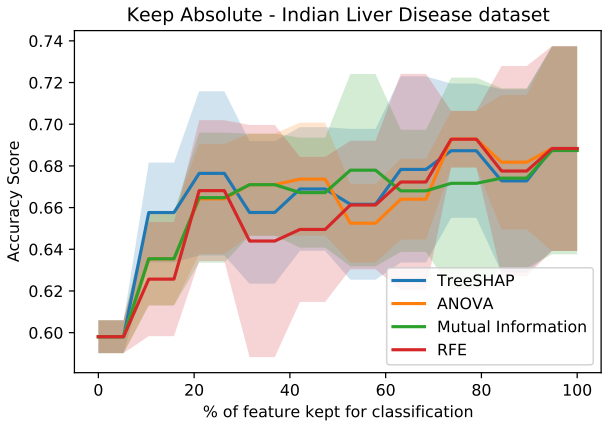
The curves give hints about the importance of ordering chose by the algorithms. While most algorithms present similar results for subsets with only a few features or with nearly all of the features of the datasets, for some parts of the curves, selecting feature with SHAP values is better due to its intrinsical capability to understand the dataset and classifier characteristics – see, for example, the pattern presented between 20% and 50% for curves of Vertebral dataset in Fig. 4d. In this case, selecting and assigning importance to the features with SHAP values has the advantage to capture complex characteristics of the features in a separately way (for each class at a time), then, the joint importance of the feature is likely to be different if the importance was calculated globally. The result is that the ordering for critical positions – the ordering of features that contribute nearly the same for a model – could reveal better importance, as seen in the middle parts of the curves in Fig. 4.

To better understand the attribution of importance for feature selection based on SHAP values, Fig. 5 shows the feature explanations for one single class of the Vertebral dataset, which consists of patients described by six features derived from the shape and orientation of the pelvis and lumbar spine: `pelvic incidence`, `pelvic tilt`, `lumbar lordosis angle`, `sacral slope`, `pelvic radius`, and `grade of spondylolisthesis`. Such class corresponds to patients with Spondylolisthesis, a disturbance of the spine in which a bone (vertebra) slides forward over the bone below it. In the figure, the SHAP values are represented in the x axis, meaning that values distant from zero impose more influence on the model's prediction – note that the most important features are on top. Since positive SHAP values means that the probability of class prediction is increased, patients presenting higher values for `degree of spondylolisthesis`, `pelvic incidence`, `sacral slope`, and `pelvic tilt` are more likely to present problems in their spine. Besides, see how higher values of degree of spondylolisthesis, which essentially measures how much a vertebra bone slid over a bone below it, contribute to defining this class. This result is also consistent with the literature on Spondylolisthesis, in which pelvic incidence, sacral slope, pelvic tilt are found to be greater in patients with developmental spondylolisthesis [39].

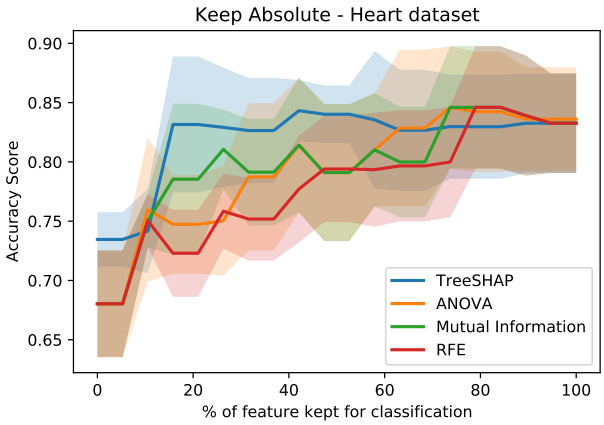
B. Regression

Similarly as performed for the classification datasets of the previous section, we evaluated the same techniques for regression tasks. In this case, we used the Mean Squared Error as score, and the **Keep Absolute** metric was applying as for the classification task, yielding the AUC Table III. Note that the values are negative since we used the negative Mean Squared Error metric of the `sklearn` [40] library, used for minimization of Mean Squared Error.

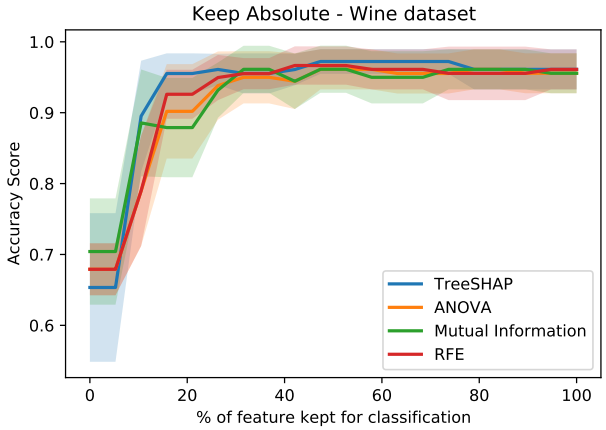
In this case, SHAP as feature selection was not able to provide the best result for *Boston* dataset – although the difference is lower than 0.02. However, it is important to know that sacrificing a bit of accuracy could be beneficial



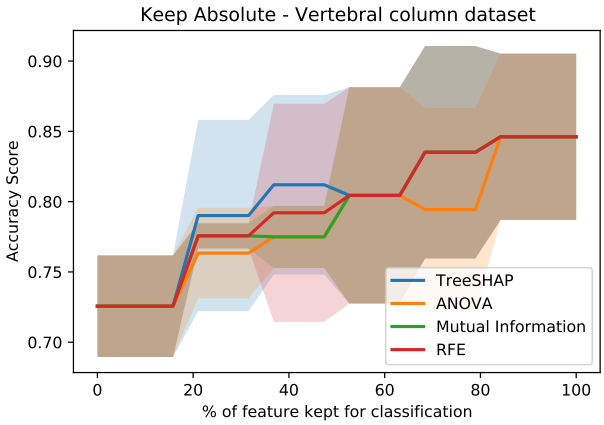
(a) F1-Score curves for varying feature subsets' sizes - Indian Liver Disease dataset.



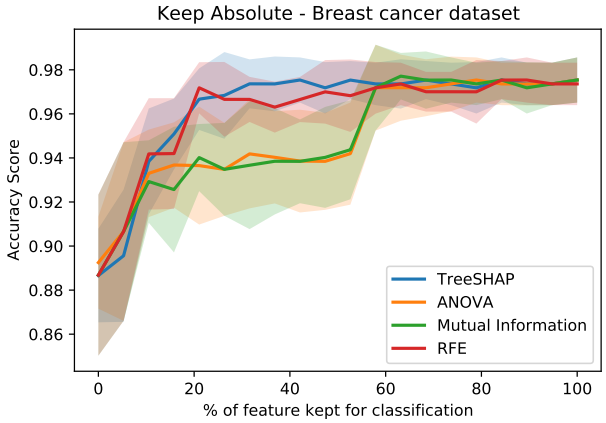
(b) F1-Score curves for varying feature subsets' sizes - Heart disease dataset.



(c) F1-Score curves for varying feature subsets' sizes - Wine dataset.



(d) F1-Score curves for varying feature subsets' sizes - Vertebral column dataset.



(e) F1-Score curves for varying feature subsets' sizes - Breast cancer dataset.

Fig. 4. F1-Scores (mean and standard deviation) in a 5-fold cross-validation setting when different subset sizes are selected from all features of each dataset.

TABLE I
DESCRIPTION OF THE DATASETS USED IN EXPERIMENTATION.

Dataset	# data points	Dimensionality	Task	Params
Indian Liver Disease	583	10	Classification	$\alpha : 0.05; \beta : 3, \omega : 1000$
Heart Disease	303	13	Classification	$\alpha : 0.05; \beta : 2, \omega : 100$
Wine	178	13	Classification	$\alpha : 0.05; \beta : 3, \omega : 1000$
Vertebral Column	310	6	Classification	$\alpha : 0.05; \beta : 2, \omega : 100$
Breast Cancer	569	30	Classification	$\alpha : 0.05; \beta : 3, \omega : 1000$
Boston	506	13	Regression	$\alpha : 0.05; \beta : 10, \omega : 1000$
Diabetes	442	10	Regression	$\alpha : 0.05; \beta : 2, \omega : 100$
NHANESI [33]	9932	19	Regression	$\alpha : 0.05; \beta : 3, \omega : 1000$

TABLE II
AREA UNDER THE CURVE (AUC) FOR EACH ONE OF THE CURVES OF FIG. 4.

Technique	Indian L.	Heart disease	Wine	Vertebral C.	Breast cancer
Tree SHAP	**0.665840**	**0.819581**	**0.935631**	**0.801384**	**0.963747**
ANOVA	0.663182	0.793923	0.918693	0.784878	0.950952
Mutual Inf.	0.662554	0.798282	0.923603	0.793256	0.950482
RFE	0.658528	0.779519	0.924560	0.795961	0.961747

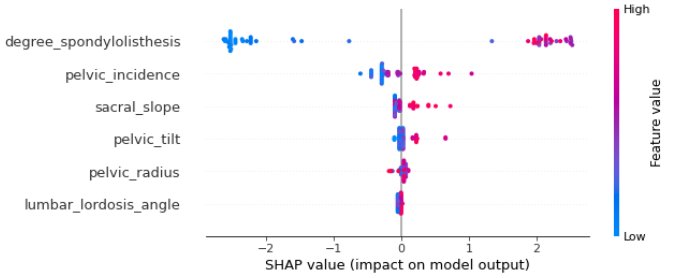


Fig. 5. SHAP values for the class of patients with Spondylolisthesis of the Vertebral dataset. Colors encode feature values and the x axis shows information about SHAP values. For this class, in particular, higher values of the feature "degree of Spondylolisthesis" are determinant to increase the probability of classification.

TABLE III
AREA UNDER THE CURVE (AUC) FOR THE REGRESSION TASK. VALUES CLOSER TO ZERO ENCODE BETTER RESULTS.

Technique	Diabetes	Boston	NHANESI
Tree SHAP	**-46.424751**	-3.460678	**-10.916105**
ANOVA	-48.673970	**-3.442648**	-13.058894
Mutual Inf.	-51.315229	-4.396497	-11.577051
RFE	-47.739208	-3.536853	-10.990975

in situations where interpretability could be more useful than model fitness, such as in medical applications [33]. Further that, Fig. 6 shows that a percentage of features can be chosen to provide results as good as if all the features were used, i.e., selecting features with SHAP values where the percentage is between approx. 42% and 80% would provide better results than if selecting these percentages of features with the ANOVA technique. For instance, inspecting Fig. 6, we can notice that ANOVA received greater AUC due to its higher values when selecting approx. between 15% and 20% of the features, however, it corresponds to lower scores when using SHAP values for selecting more than 40% of the features.

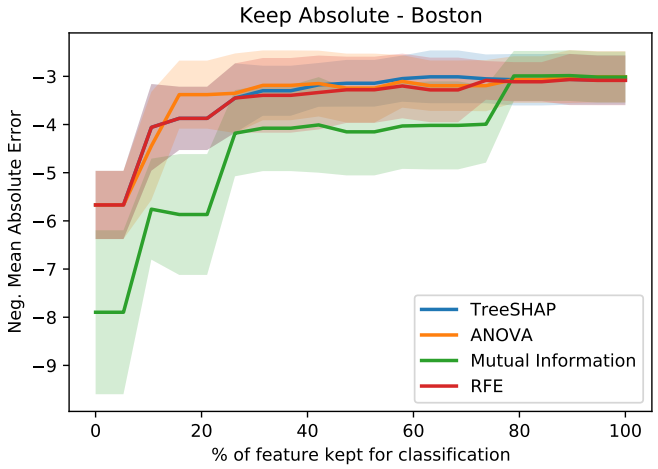


Fig. 6. Curves of Keep Absolute metric for regression applied to the *Boston* dataset. The Neg. Mean Squared Error was used for evaluating the techniques. Although ANOVA presented the best AUC (see Table III), selecting using SHAP yielded the best results for only 40% of features kept for regression.

C. Run-time execution

In this section, we aim to investigate the run-time execution of the feature selection algorithms on the classification and regression tasks discussed in the previous sections. Table IV shows that the *ANOVA* algorithm is scalable according to the number of data points and dimensionality of the datasets – bold numbers indicate the best results. *Mutual Information* and *RFE* algorithms follow the complexity of the dataset (size and dimensionality) by increasing the run-time execution, where *RFE* takes an unreasonable amount of time. Lastly, *TreeSHAP* shows a relationship with the depth of the tree boosted algorithms, that is, it takes longer run-time execution for deeper models.

Table V also shows such a relationship between the depth of the boosted tree models and the *TreeSHAP*'s run-time execution. That is, although the NHANESI dataset has a higher

TABLE IV
RUN-TIME EXECUTION (IN MILLISECONDS) FOR FEATURE SELECTION ON THE CLASSIFICATION TASK.

Technique	Liver	Heart	Wine	Vertebral	Breast
TreeSHAP	232.374	16.955	274.294	45.878	132.669
ANOVA	**2.010**	**0.0**	**0.0**	**0.996**	**1.011**
Mutual Inf.	21.926	19.947	17.952	9.947	54.817
RFE	16614.61	2570.12	44214.8	1118.04	132937.5

dimensionality and greater number of data points, the run-time execution of the Boston dataset took a considerably greater amount of time. Such a characteristic could be explained by the depth of the models trained for each dataset, for instance, depth three for NHANESI and depth ten for Boston.

TABLE V
RUN-TIME EXECUTION (IN MILLISECONDS) FOR FEATURE SELECTION ON THE CLASSIFICATION TASK.

Technique	Boston	Diabetes	NHANESI
TreeSHAP	10894.861	2565.136	62.809
ANOVA	**1.010**	**0.999**	**4.959**
Mutual Inf.	25.896	19.917	403.947
RFE	251591.223	21728.887	33322.885

V. DISCUSSIONS

Using SHAP values for model explainability has proven to be a useful tool for discovering patterns in data and to interpret model decisions in earlier works [33], [41]–[43]. Interestingly, as shown in this work, the concept of importance given to features by the absolute of SHAP values can be extrapolated to be used as a feature selection mechanism. Although many other feature selection methods are present in the literature, to the best of our knowledge, none of them take great rigor as SHAP. So that, feature selection – as a widely used pre-processing step in machine learning – could be benefited from explainable characteristics. We believe that feature selection based on SHAP values will turn to be a common approach for machine learning practitioners. For the matter of classification, SHAP values could be defined as a wrapper method since its definition consists of a surrogate model.

Although other model agnostic explainability approaches, such as LIME [44] could also be used similarly as we showed in this work, we chose SHAP due to its mathematical guarantees [1] and other aspects that have demonstrated its compatibility to the human thinking [1], [33].

Finally, the main difficulty of applying SHAP is the execution time needed for computing explanations for KernelSHAP, which is the model-agnostic approach to compute SHAP values that use weighted linear regression – notice that we used the Tree SHAP approach since boosted trees were used for classification and regression. The Kernel SHAP approach takes a quadratic amount of time in both dimensionality and size of the dataset, which could be prohibitively for even moderate datasets. This problem could be decreased by selecting features based on correlation before feeding SHAP.

VI. CONCLUSION

Model explainability has been a very discussed topic due to its necessity in risk applications where ethical issues can be a problem for the adoption of black-box machine learning solutions. Although a lot of strategies to explain complex models have been proposed in the literature in recent years, few works focus their attention on the pre-processing steps of the machine learning pipeline.

In this work, we proposed and analyzed SHAP as a feature selection mechanism. SHAP is a model agnostic approach that assigns the importance of features based on their contribution to the model's output. Here, we used these contributions to order features according to their importance and used them as a feature selection strategy. In our experimentation, SHAP demonstrated to be superior to other common feature selection mechanisms, which can be a useful approach when developing machine learning solutions with interpretability in mind.

As future works, we plan to further analyze SHAP to compare the ordering imposed by the feature selection methods to inspect where are the critical parts that decrease or increase a model's score.

ACKNOWLEDGMENT

This work was supported by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brazil (CAPES) - grant #88887.487331/2020-00, and by the Fundação de Amparo à Pesquisa do Estudo de São Paulo - grant #2018/17881-3.

REFERENCES

[1] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 4765–4774.

[2] S. M. Lundberg, B. Nair, M. S. Vavilala, M. Horibe, M. J. Eisses, T. Adams, D. E. Liston, D. K.-W. Low, S.-F. Newman, J. Kim *et al.*, "Explainable machine-learning predictions for the prevention of hypoxaemia during surgery," *Nature Biomedical Engineering*, vol. 2, no. 10, p. 749, 2018.

[3] L. J. P. Maaten and G. E. Hinton, "Visualizing high-dimensional data using t-sne," *Journal of Machine Learning Research*, vol. 9, pp. 2579––2605, 2008.

[4] L. McInnes, J. Healy, and J. Melville, "UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction," *ArXiv e-prints*, Feb. 2018.

[5] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, no. null, p. 1157–1182, Mar. 2003.

[6] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers Electrical Engineering*, vol. 40, no. 1, pp. 16 – 28, 2014, 40th-year commemorative issue.

[7] R. Bellman, *Adaptive Control Processes*. Princeton University Press, 1961.

[8] V. Bolón-Canedo, N. Sánchez-Maroño, A. Alonso-Betanzos, J. Benítez, and F. Herrera, "A review of microarray datasets and applied feature selection methods," *Information Sciences*, vol. 282, pp. 111 – 135, 2014.

[9] J. C. Ang, A. Mirzal, H. Haron, and H. N. A. Hamed, "Supervised, unsupervised, and semi-supervised feature selection: A review on gene selection," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 13, no. 5, pp. 971–989, 2016.

[10] K. Benabdeslem and M. Hindawi, "Constrained laplacian score for semi-supervised feature selection," in *Machine Learning and Knowledge Discovery in Databases*, D. Gunopulos, T. Hofmann, D. Malerba, and M. Vazirgiannis, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011, pp. 204–218.

[11] J. Yang, H. Xu, and P. Jia, "Effective search for genetic-based machine learning systems via estimation of distribution algorithms and embedded feature reduction techniques," *Neurocomputing*, vol. 113, pp. 105 – 121, 2013.

[12] R. Sheikhpour, M. Sarram, and R. Sheikhpour, "Particle swarm optimization for bandwidth determination and feature selection of kernel density estimation based classifiers in diagnosis of breast cancer," *Applied Soft Computing*, vol. 40, pp. 113 – 131, 2016.

[13] X. Zhang, G. Wu, Z. Dong, and C. Crawford, "Embedded feature-selection support vector machine for driving pattern recognition," *Journal of the Franklin Institute*, vol. 352, no. 2, pp. 669 – 685, 2015.

[14] R. Garcia, A. X. Falcão, A. Telea, B. C. da Silva, J. Tørresen, and J. L. D. Comba, "A methodology for neural network architectural tuning using activation occurrence maps," *2019 International Joint Conference on Neural Networks (IJCNN)*, pp. 1–10, 2019.

[15] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Transactions on Neural Networks*, vol. 5, no. 4, pp. 537–550, July 1994.

[16] N. Kwak and C.-H. Choi, "Input feature selection for classification problems," *Trans. Neur. Netw.*, vol. 13, no. 1, p. 143–159, Jan. 2002. [Online]. Available: https://doi.org/10.1109/72.977291

[17] G. Forman, "An extensive empirical study of feature selection metrics for text classification," *J. Mach. Learn. Res.*, vol. 3, pp. 1289–1305, 2003.

[18] R. Kohavi and G. H. John, "Wrappers for feature subset selection," *Artificial Intelligence*, vol. 97, no. 1, pp. 273 – 324, 1997, relevance.

[19] J. Reunanen, "Overfitting in making comparisons between variable selection methods," *J. Mach. Learn. Res.*, vol. 3, no. null, p. 1371–1382, Mar. 2003.

[20] S. Nakariyakul and D. P. Casasent, "An improvement on floating search algorithms for feature subset selection," *Pattern Recognition*, vol. 42, no. 9, pp. 1932 – 1940, 2009.

[21] O. Cordón, S. Damas, and J. Santamaría, "Feature-based image registration by means of the chc evolutionary algorithm," *Image and Vision Computing*, vol. 24, no. 5, pp. 525 – 533, 2006.

[22] S. Oreski and G. Oreski, "Genetic algorithm-based heuristic for feature selection in credit risk assessment," *Expert Systems with Applications*, vol. 41, no. 4, Part 2, pp. 2052 – 2064, 2014.

[23] P. Ghamisi and J. A. Benediktsson, "Feature selection based on hybridization of genetic algorithm and particle swarm optimization," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 2, pp. 309–313, 2015.

[24] B. Wutzl, K. Leibnitz, F. Rattay, M. Kronbichler, M. Murata, and S. M. Golaszewski, "Genetic algorithms for feature selection when classifying severe chronic disorders of consciousness," *PLOS ONE*, vol. 14, no. 7, pp. 1–16, 07 2019. [Online]. Available: https://doi.org/10.1371/journal.pone.0219683

[25] C.-J. Tu, L.-Y. Chuang, J.-Y. Chang, and C.-H. Yang, "Feature selection using pso-svm," in *IMECS*, 2006.

[26] S. Tabakhi and P. Moradi, "Relevance-redundancy feature selection based on ant colony optimization," *Pattern Recogn.*, vol. 48, no. 9, p. 2798–2811, 2015.

[27] C. Qiu, "A novel multi-swarm particle swarm optimization for feature selection," *Genet Program Evolvable*, pp. 503–529, 2019.

[28] C. H. Z. Y. e. a. Wang, G., "Multiple parameter control for ant colony optimization applied to feature selection problem," *Neural Comput & Applic*, no. 26, pp. 1693–1708, 2015.

[29] C. Ding and H. Peng, "Minimum redundancy feature selection from microarray gene expression data," in *Computational Systems Bioinformatics. CSB2003. Proceedings of the 2003 IEEE Bioinformatics Conference. CSB2003*, Aug 2003, pp. 523–528.

[30] P. A. Mundra and J. C. Rajapakse, "Svm-rfe with mrmr filter for gene selection," *IEEE Transactions on NanoBioscience*, vol. 9, no. 1, pp. 31–37, 2010.

[31] I. Guyon, J. Weston, S. Barnhill, V. Vapnik, and N. Cristianini, "Gene selection for cancer classification using support vector machines," in *Machine Learning*, p. 2002.

[32] L. S. S. Shapley, "A valuue for n-person games," in *Contributions to the Theory of Games*, 1953, pp. 307–317.

[33] S. M. Lundberg, G. Erion, H. Chen, A. DeGrave, J. M. Prutkin, B. Nair, R. Katz, J. Himmelfarb, N. Bansal, and S.-I. Lee, "From local explanations to global understanding with explainable ai for trees," *Nature Machine Intelligence*, vol. 2, no. 1, pp. 2522–5839, 2020.

[34] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, ser. KDD '16. New York, NY, USA: Association for Computing Machinery, 2016, p. 785–794. [Online]. Available: https://doi.org/10.1145/2939672.2939785

[35] L. F. Kozachenko and N. N. Leonenko, "Sample estimate of the entropy of a random vector," *Probl. Peredachi Inf.*, pp. 9–16, 1987.

[36] A. Kraskov, H. Stögbauer, and P. Grassberger, "Estimating mutual information." *Physical review. E, Statistical, nonlinear, and soft matter physics*, vol. 69 6 Pt 2, p. 066138, 2004.

[37] B. C. Ross, "Mutual information between discrete and continuous data sets," *PLoS ONE*, vol. 9, no. 2, p. e87357, Feb. 2014. [Online]. Available: http://dx.plos.org/10.1371/journal.pone.0087357

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg *et al.*, "Scikit-learn: Machine learning in python," *Journal of machine learning research*, vol. 12, no. Oct, pp. 2825–2830, 2011.

[39] H. Labelle, P. Roussouly, E. Berthonnaud, J. Dimnet, and M. Obrien, "The importance of spino-pelvic balance in l5–s1 developmental spondylolisthesis: A review of pertinent radiologic measurements," *Spine*, vol. 30, pp. S27–S34, 2005.

[40] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[41] K. E. Mokhtari, B. P. Higdon, and A. Başar, "Interpreting financial time series with shap values," in *Proceedings of the 29th Annual International Conference on Computer Science and Software Engineering*, ser. CASCON '19. USA: IBM Corp., 2019, p. 166–172.

[42] Batunacun, R. Wieland, T. Lakes, and C. Nendel, "Using shap to interpret xgboost predictions of grassland degradation in xilingol, china," *Geoscientific Model Development Discussions*, vol. 2020, pp. 1–28, 2020. [Online]. Available: https://gmd.copernicus.org/preprints/gmd-2020-59/

[43] A. B. Parsa, A. Movahedi, H. Taghipour, S. Derrible, and A. K. Mohammadian, "Toward safer highways, application of xgboost and shap for real-time accident detection and feature analysis," *Accident Analysis Prevention*, vol. 136, p. 105405, 2020.

[44] M. T. Ribeiro, S. Singh, and C. Guestrin, ""why should I trust you?": Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, 2016, pp. 1135–1144.