Proceedings of the 2019 Winter Simulation Conference
N. Mustafee, K.-H.G. Bae, S. Lazarova-Molnar, M. Rabe, C. Szabo, P. Haas, and Y-J. Son, eds.

IMPROVED FEATURE SELECTION WITH SIMULATION OPTIMIZATION

Sara Shashaani
Kimia Vahdat

Department of Industrial and Systems Engineering
North Carolina State University
111 Lampe Dr.
Raleigh, NC 27601, USA

ABSTRACT

In machine learning problems with a large feature space, non-informative or redundant features can significantly harm the performance of the prediction models, keep the model training costly, and the model interpretability weak. Traditional feature selection methods, preformed often through greedy search, are susceptible to suboptimal solutions, selection bias, and high variability due to noise in the data. Our Sample Average Approximation framework looks for the best subset of features by utilizing bootstraps of the training set, where the random holdout errors are viewed as simulation outputs. We implement the proposed Simulation Optimization with Genetic Algorithms, noting that this framework is generalizable with any other solver on the integer space. Experimentations examine the effect of fixed versus variable and adaptive replication sizes upon estimating the performance of each subset. We provide promising results of higher accuracy and more robustness in solution size and content at the cost of longer computation.

1 INTRODUCTION

In situations where the number of features collected to make a prediction is very large, contrary to what may be understood that more data leads to better predictions, redundant or uninformative features can cause (1) overfitting in the training set, (2) more computationally expensive model building and updating, and (3) less inference or interpretation power on the features (Guyon and Elisseeff 2003). Admittedly, there is no clear way to identify and exclude the uninformative features, especially knowing that, the contribution of features may vary with models used for learning.

Classically, feature selection is done with wrapper methods, filter methods, or embedded methods. In wrapper methods, a search algorithm is "wrapped" around the learning model, while filter methods evaluate the features prior to training the model as a pre-processing step. The embedded methods are specific to tree-based models where the feature selection is part of the model construction (Kuhn and Johnson 2013). Another common approach is focusing on learning models such as Support Vector Machines instead of a direct feature selection mechanism (Chen and Lin 2006). We think a more fundamental question is: instead of changing the training mechanism, can we choose the best feature subset to reach the highest accuracy with any class of training models that is specified? Answering this question means solving a binary minimization problem given a specified training model – sometimes there are reasons to believe that a specific class of models adequately explains the relationship in the data and we seek to extract the best subset of features for it. The binary decision variables in the optimization determine which feature is included in the training, and the objective function measures the deviation of the predicted values from the observed values. Note that, by fixing the learning model we are able to (precisely or imprecisely) find the optimal feature subset and then compare our results with it.

2 METHODOLOGY AND RESULTS

Solving the minimization problem introduced in the previous section requires controlling the error in two directions: (1) are we correct in choosing the best subset? (2) how close are we in estimating the performance of the selected subset? This yields a new framework based on Simulation Optimization with the objective function estimates using Sample Average Approximation (Kleywegt et al. 2002). The idea is simple: average the performance of any given subset on replications of simulated (bootstrapped) training and test sets from the available data. The bootstrapped training sets allow for some characteristics of the test data set to be reproduced or at the least make the feature selection less dependent on one dataset – the original training set. In other words, the selection bias or overfitting will decrease. With SAA those features that do well in most of the bootstrapped training sets survive the search which results in smaller subsets. Furthermore, repeating the experiment for several different original training and test sets reveals more similarity in the selected subsets compared to state-of-the-art feature selection that gives very different subsets with different training and test cases. Specifically, this demonstrates that all typical ways of doing feature selection that involve greedy search would give a wide range of feature subsets (and hence less reliable interpretability and perhaps too much dependence on the training data) while a direct simulation optimization-based approach consistently finds similar features for all the replications.

Our framework, SO based feature selection or SOFS, has three aspects to it:

- The learning **model** used to fit the available data, e.g. linear regression (LM) or random forest (RF).
- The objective **function** or loss function used inside the optimization, e.g. sum of squared error.
- The optimization **method**, e.g. Genetic Algorithms (GA).

Besides incorporating different search methods and algorithms, we investigate the effect of fixed and variable sampling rules in the accuracy and speed of the search. We design sampling rules for the number of bootstraps that depend on an estimate of the optimality gap, and the variability in the estimated performance, and the actual noise that the underlying unknown function has which experiments show differs from subset to subset. Our sampling rules successfully reduce the computation time of the search and within a fixed budget for the optimization, gain better solutions.

Our numerical experiments involve testing SOFS on a variety of datasets and comparing with some commonly used algorithms and packages such as recursive feature elimination (RFE) or an R package on GA based feature selection (GAFS). In short, SOFS leads to higher predictive power and more robust solutions. While these advantages come at the cost of longer operations, the advancements in SO such as adaptive sampling and parallelization can be embedded to reduce the computation.

Our proposed framework here is generic and applicable to any learning model of interest, as well as other optimization routines such as ranking and selection, instead of the Genetic Algorithms that is a slowly converging routine. Our conjecture is improving the accuracy and speed of SOFS by incorporating more spatially-adapted structure induced search leveraging gradient-like heuristics. In addition, we seek to study the parallelization in SO as our future steps.

REFERENCES

Guyon, I., Elisseeff, A. 2003. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 1157-1182.

Kuhn, M. and Johnson, K., 2013. *Applied Predictive Modeling* (Vol. 26). New York: Springer.

Kleywegt, A. J., Shapiro, A., Homem-de-Mello, T. 2002. The Sample Average Approximation Method for Stochastic Discrete Optimization. *SIAM Journal on Optimization, 12*(2), 479-502.

Chen YW., Lin CJ. (2006) Combining SVMs with Various Feature Selection Strategies. In *Feature Extraction: Studies in Fuzziness and Soft Computing*, edited by I. Guyon, M. Nikravesh, S. Gunn, L.A. Zadeh, Vol 207. Springer, Berlin, Heidelberg.