IMPROVED FEATURE SELECTION WITH SIMULATION OPTIMIZATON

Sara Shashaani, Kimia Vahdat

North Carolina State University
Raleigh, NC 27601

Abstract Overview

In many data mining applications where the number of features is large, identifying the most informative features remains a challenge. Non-informative or redundant features can significantly harm the performance of the prediction models while keeping the model training costly, and the model interpretability weak. We develop a framework to search for the best subset of features with genetic algorithms using a Simulation Optimization approach where the random holdout errors are viewed as the simulation replicates. We study the effect of fixed versus variable replication size in the outcome and compare the results with some of the existing feature selection methods on a sample dataset. Investigations on overall accuracy and computation time and in our preliminary results reveal that SO based approaches can provide cost-effective improvements in the predictive power of data mining models.

Keywords

Simulation Optimization, Genetics Algorithms, Feature Selection, Prediction Modeling.

Introduction

The issues of data redundancy (and uninformative-ness) arise with big data. This is the situation in which the number of covariates collected to make a prediction is very large, and contrary to what may be understood that more data leads to better predictions, redundant or uninformative covariates (features) can damage the predictive accuracy. Including such features can also cause (1) overfitting in the training set, (2) more computationally expensive model building and updating, and (3) less inference or interpretation power on the predictors (Guyon and Elisseeff 2003). There is no clear way to identify and exclude the uninformative features and the outcome is often dependent on the algorithms used for learning.

In contrast to feature space dimension reduction methods such as principal component analysis, feature selection techniques keep the original variables and search for a subset of them that provide most of the useful information for prediction. Therefore, feature selection also leads to models with better transparency and interpretability. Feature selection is typically done in three main ways: wrapper methods, filter methods, and embedded methods. In wrapper methods, a search algorithm is ``wrapped" around the model - such as forward selection, while filter methods evaluate the predictors prior to training the model as a pre-processing step. The embedded methods are specific to tree-based algorithms where the feature selection is part of the model construction. However, even for most stable tree-based models such as random forests, the data redundancy can still negatively affect the predictive accuracy.

Significant body of research has dealt with this problem by choosing different learning algorithms; evidently each learning algorithm results in selecting different subsets of the features depending on the assumption that the algorithm makes about the relationship between the predictors to themselves and the predictors to the response variable. We think aside from the fact that the learning algorithms would be performing differently for different datasets, a more fundamental study is to ask the question, given a fixed learning algorithm can we choose the best subset of features

that would lead to a model that provides the highest accuracy in the validation set? By fixing the learning algorithm we will also be able to find the optimal subset and then compare how overlapping our resulting subsets will be to that optimal set. Consequently, we can look at 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?

Due to issues such as selection bias, or in situations where an independent test set is absent, one needs to evaluate the performance of the selected subset using a certain external validation check. More commonly used existing mechanisms such as recursive feature elimination (RFE) or feature selection with genetic algorithms (GAFS) either lack the external validation component or incorporate it at the expense of high computation cost. Some more recent implementations of RFE and GAFS approaches use *k*-fold cross validation. This is not ideal, since in cross-validation *k-2* folds are common between every two simulation runs, causing immense dependency between the runs. Furthermore, neither of these methods actually solves the best subset search as an optimization problem. RFE searches for the best subset size, and GAFS searches for best maximum number of generations.

In this study, we look at a new framework that searches in the space of all features in a bid to maximize the external performance that is structured around out of bag error. The new framework is based on Simulation Optimization (SO) where the objective function is only estimated using the replications of a simulated (bootstrapped) training and testing sets from the available data, to reduce the bias and dependence between the samples that generate the performance estimate. The SO is advantageous in that it assumes little to nothing about the structure of the objective function (in this context, the learning algorithm or the loss function) while a long array of advancements in this field helps to reduce the bias and variance of the outcome models. The out of bag error is calculated with fixed resampled training and test sets; this leverages the common random numbers concept in simulation. In our experiments in this paper we show that our approach can help with some current issues that the feature selection problems have:

- Are highly correlated variables necessarily not important?
- Could seemingly unimportant variables be important in the presence of some other variables?
- Could the redundant variables be detected?

Methodology and Results

Our framework has three aspects to it:

- The learning algorithm or model used to fit the available data e.g. linear regression or random forest.
- The optimization engine, e.g. Genetic Algorithms (GA).
- The objective function used inside the optimization (fitness function in GA) that is the specified loss function to be minimized, e.g. sum of squared error.

Let the dataset at hand W be divided into the learning set M and the validation set V. The learning set is assumed to be the only data available for training the model, and the validation set will be used for the predictive accuracy; through this set we compare the performance of the proposed procedure with the existing ones in selecting the features. The validation process involves comparing the mean squared error and mean absolute error of what the prediction model predicts and the response observations in the validation set. Ultimately, we are interested in the most important features in the dataset. We describe the problem as looking at the expected performance of the selected subset of features on the validation set

$$S^* = \underset{S}{\operatorname{argmin}} \sum_{j \in V} \left(f_{M,A,S}(\boldsymbol{x}_j) - y_j \right)^2,$$

where $f_{M,A,S}(.)$ is the prediction model trained by the subset S of features of the learning set M with the learning algorithm A. The actual performance of subset S is unknown because we do not know the y_j's in the V set. We can estimate its expected value by the concept of Sample Average Approximation (SAA) (Kleywegt et al. 2002) which reduces the stochastic minimization problem into the empirical minimization problem

$$\hat{S}^* = \underset{S}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j \in M_i^c} \left(f_{M_i,A,S}(\boldsymbol{x}_j) - y_j \right)^2 \right),$$

where M_i and M_i^c as resampled training and test sets within the learning set M.

We use GA to search for the best subset, with some pre-defined parameters. Our experiments on a sample dataset with 10 resampled bootstraps are compared to that of RFE which is widely used for the feature selection purpose. To keep the comparison fair, we also use 10 bootstraps for the RFE algorithm available in caret package of the R software. Since our claim is that the advantage of the SO based Feature Selection with we will refer to as SOFS is beyond the learning algorithm, we perform the experiments on a simple case of linear regression, and a nonparametric case of random forest. Table 1 and Table 2 summarize the results of the prediction accuracy of the best subset outcome of our SOFS/GA and RFE in terms of mean absolute error (MAE) and mean squared error (MSE).

From the results we conclude that even though the RFE provides better in-sample measures, it is significantly outperformed by SOFS/GA in terms of out-of-sample measures which are representable of the predictive accuracy. What we observe is showing that SOFS is able to deal with the common problem of overfitting better.

*Table 1. Numerical experiment with **linear regression** as the learning algorithm, with In-Sample (IS) and Out-Of-Sample (OOS) performance metrics.*

Method	# feat	IS MAE	IS MSE	OOS MAE	OOS MSE	Time
RFE (n=10)	45	2.1	8.9	6.1	48.1	98
SOFS/GA (n=10)	26	2.4	12.0	4.6	28.3	980

*Table 2. Numerical experiment with **random forest** as the learning algorithm, with In-Sample (IS) and Out-Of-Sample (OOS) performance metrics.*

Method	# feat	IS MAE	IS MSE	OOS MAE	OOS MSE	Time
RFE (n=10)	13	1.9	8.3	5.8	44.3	206
SOFS/GA (n=10)	29	1.8	8.3	4.3	24.9	1,946

We further compare the size of the best subsets reported by using RFE versus SOFS/GA when we repeat the numerical experiments for 15 different training and test sets that we create from the whole dataset. The boxplots containing the results of all the 15 instances in Figure 1 and Figure 2 suggest that the SOFS/GA is more robust and leads to a smaller subset of features on average and more importantly with a smaller variability. Specifically this demonstrates that a greedy search like RFE, that is the typical way feature selection is done, 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.

Conclusions

We investigate the increased predictive accuracy in machine learning using Simulation Optimization for feature selection where we simulate the future observations that we wish to predict really well by bootstraps (samples) of the available dataset at hand that we refer to as the learning dataset. We validate the results of our SO based approach and the existing procedures on a validation dataset and find significant improvement. Our optimization method, genetic algorithms, respond well in the context of binary variables for the best subset selection. However, the focus is not necessarily the optimization routine. One can look at a suite of optimizers that are specifically designed for binary decision variables. Alternatively, one can use gradient-like heuristics or extract derivatives through neighborhood information in this setting. Additionally, we do not worry about choosing the best learning model, and simply prove our point with the linear least square models and random forests as a nonparametric class of learners. Even though the outcome of feature selection is quite learning-dependent, we believe even within a learning algorithm that may not adequately represent the relationship between variables, choosing the right subset of features could provide better performance.

Our proposed framework here is generic and applicable to any learning algorithm of interest. Our conjecture is that there is a lot of benefit in using SO and common random numbers in this context. We consider parallelization in SO, scenario generation, and more spatially-adapted search as our future steps.

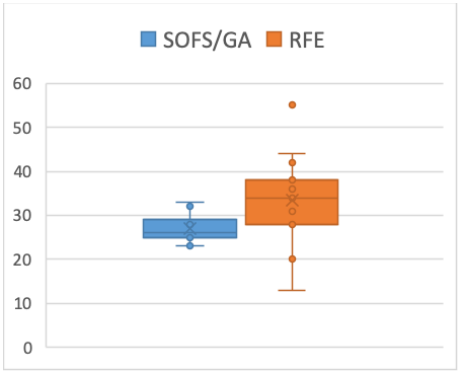

*Figure 1. Best subset size boxplots with **linear regression**.*

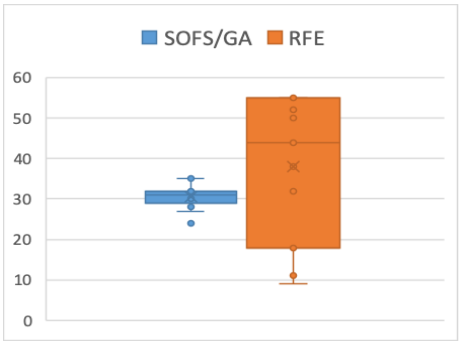

*Figure 2. Best subset size boxplots with **random forest**.*

References

Guyon, I., Elisseeff, A. (2003). An introduction to variable and feature selection. *Journal of Machine Learning Research*, *3*(Mar), 1157-1182.

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.