2017

Logistic Ensemble Models

Bob Vanderheyden
Kennesaw State University

Jennifer L. Priestley
Kennesaw State University, jpriestl@kennesaw.edu

Logistic Ensemble Models

Bob Vanderheyden
Department of Statistics and Analytical Sciences
College of Science and Mathematics
Kennesaw State University

Jennifer Priestley, Ph.D.
Department of Statistics and Analytical Sciences
College of Science and Mathematics
Kennesaw State University

***Abstract*—Predictive models that are developed in a regulated industry or a regulated application, like determination of credit worthiness must be interpretable and "rational" (e.g., improvements in basic credit behavior must result in improved credit worthiness scores). Machine Learning technologies provide very good performance with minimal analyst intervention, so they are well suited to a high volume analytic environment but the majority are "black box" tools that provide very limited insight or interpretability into key drivers of model performance or predicted model output values. This paper presents a methodology that blends one of the most popular predictive statistical modeling methods with a core model enhancement strategy, found in machine learning. The resulting prediction methodology provides solid performance, from minimal analyst effort, while providing the interpretability and rationality, required in regulated industries.**

Keywords—logistic regression; ensemble; predictive model; binary classification; quadratic programming

I. Introduction

Logistic regression is a historically successful statistical method for predicting a binary event (i.e., an event with two possible outcomes). The methodology estimates the log-odds of the event based on linear regressors, using the functional form:

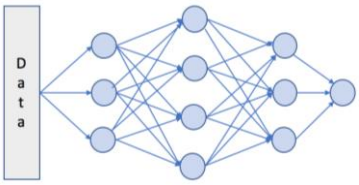
$$\ln\left(\frac{p}{1-p}\right) = \beta x \qquad (1)$$

For model estimation purposes, this functional form is transformed to express the binary event in terms of the logit function:

$$y = \frac{1}{1+e^{-\beta x}} \qquad (2)$$

This same logit function is used as the activation function (represented by arrows in the model below), in neural nets, deep nets and convolutional neural nets.

Fig 1: Multi-Layer Perceptron Structure



One critical concern for any predictive model is its general applicability. Developing a "better" performing model that doesn't generalize to new data, provides limited value for the business or agency that needs to make decisions based on the application of the model. In many situations, models perform within time, but when applied to a future time period, model performance suffers and performance further deteriorates as the time from initial model development increases.

In machine learning, generalizability of modeling techniques that tend to suffer from issues related to over fittting can be mitigated by employing one or more ensemble methodologies, that combine the results of multiple lower performing models to provide a high performing solution that generalizes better than a single model. One example of this technique is the development of random forests which combine the results of multiple decision trees [1].

The current study compares a combination of k less effective logistic regression models (predictors) to a fully developed model (predictor). In specifying the fully developed predictor, techniques that are commonly applied to maximize the performance of a single model (e.g., nonlinear transformations) are applied. The combination of k predictors will be of the form:

$$p = \lambda_1 p_1 + \lambda_2 p_2 + \cdots + \lambda_k p_k \qquad (3)$$

This study examines credit data for 11.8 million prospective business customers. The quarterly data was provided by a major U.S. based credit bureau and span 9 years from 2006 to 2014. It offers an opportunity to not only assess performance of a predictive binary classification model versus the proposed solution within a specified time frame, but also to assess model performance over an extended period of time.

The fundamental hypothesis for the project is:

> A composition of multiple logistic regression classifiers, using no analyst derived attribute transformations or attribute selection steps* will perform as well or better than an optimally developed logistic regression model, in the original time period and performance will be more stable (better), than the optimally developed logistic regression model across an extended time period.

* - no analysis/data manipulation beyond basic cleansing and imputation of missing values

II. Literature Review

The concept of developing "ensembles" of models has been around for more than 30 years. "Stacking" is one of the primary forms of ensemble creation where multiple models, which may be based on different modeling methodologies or use the same methodology, but with different predictive elements. [2]

In 1992, David Wopert published "Stacked Generaliztion" in which he examines the "stacking" of neural net models, to boost generalizability of predictive models [2]. For scenarios where multiple predictors are available, choosing the "best" predictor (e.g., via a voting process, from applying the predictors on a validation sample) may not be generate the best result. Instead, Wolpert performs a heuristic analysis of the viability of using the results from multiple predictors, that he positions as a "generalization" of predictors. Wolpert starts with the most fundamental assessment of generalization of predictive models – the validation sample – and then extends his analysis to generalization of a predictor to other populations/samples. Wolpert not only provides a heuristic foundation supporting the use of stacked models, but includes two key experiments that demonstrate the applicability and value. [2]

Wolpert's work is a general discussion of and strong support for combing predictors, but he doesn't arrive at a single "best practice" for combining predictors. Leo Brieman [3] examines simple linear combinations of predictors of the form:

$$v(x) = \sum_k \alpha_k v_k(x) \qquad (4)$$

To estimate the optimal α_k, Brieman minimizes the quadratic cost function.

$$\sum_n (y_n - \sum_k \alpha_k v_k(x))^2 \qquad (5)$$

Unrestricted α_k's, could lead to a combined solution that isn't as effective as one or more of the available predictors. Restricting the parameters to be non-negative and $\sum \alpha_k = 1$, results in what Brieman calls an "interpolating" predictor, that performs at least as well as the best single predictor. Brieman also notes that after optimization is complete, a relatively small number of predictors have a non-zero weight [3]. This sum of squared error for the linear combination of predictors will be used to optimize the ensemble that's produced in this study.

In his work, Brieman examined stacking five different types of predictors. This study focuses on stacking "Subset Regressions" which combines predictors that utilize different sets of predictors, to derive an optimal predictor [3].

III. Data

This study examines the incidence of a business falling two or more months behind on payments of Non-Financial Accounts (NFAs). Much like unsecured credit accounts in the consumer market, businesses may have multiple NFA\s. This type of account was the most prominent account activity for businesses found in the 36 month credit file. In 2006, just under three million of the 11.8 million businesses found in the file had sufficient activity (i.e. sufficient number of fields to provide predictive input for the record; in this case have account behaviors for at least one other account type for the time period) to be considered for inclusion in a model development project.

In the available data files, the initial file, Q1 2006, contains 3.18 million records where the data are not missing for all fields except the account keys. Of these records, almost 2.9 million have sufficient data representation (i.e., they are not "coded" missing/null) to use in a predictive model development effort. On average, each subsequent quarter adds an additional 106K records, to reach 6.88 million viable records by the 4th quarter of 2014. Less than half of the 4th quarter 2014 records have available data for all 36 quarters. Almost 5 million records have missing values for all date except the account keys, for all quarters. This trend of missing data is captured in Fig 2.

Fig 2: Missing Date Rate Across Quarterly Files

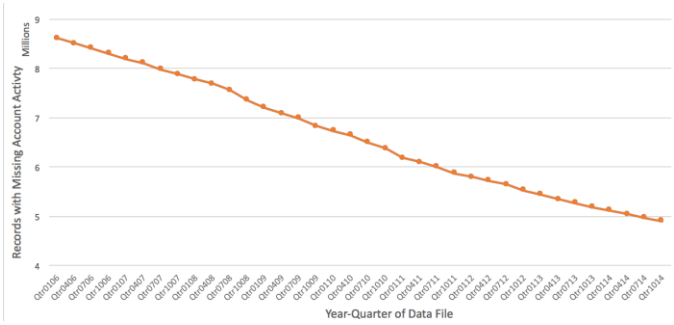


Fig 1. Missing data counts, by quarter. Of the 11.8 million records in the provided data, a large number have missing values for all date, except the account keys.

As is the case with all studies that involve the analysis of real data, cleansing of the data, including imputation of missing values, logic checks and data conversion into structured formats that can be used by the desired analytic method is as critical as the choice of analytic method.

A. *Missing Value Imputation*

Before the data can be analyzed, a considerable amount of recoding and data logic corrections were required. For example, a field may have a coded "missing" value (e.g., "999999999" for account balance fields). In most cases, for example account balance, when the reported number of accounts is 0, imputing with 0 is the most logical choice. One type of coded missing data takes the form of coded missing for all data of an account type. Figure 3, contains an example where NFA data is coded missing for the 1st 3 quarters, for an account. This scenario is taken to mean that no accounts of type NFA were open/active. For these records, accounts of other types (e.g., utility) have account activity.

Fig 3: Example of Coded Missing

Sequential Quarter Number	SiteEFXID	NoNFAcur	NoNFA3mon	NoNFAbalance3mon	NoNFA12mon	NoNFA24mon	totNFAcc
1	661904353	99	99	99	99	99	999999999
2	661904353	99	99	99	99	99	999999999
3	661904353	99	99	99	99	99	999999999

Another type of coded missing due to no account activity is an "activity gap" where the business had one or more quarters of zero active accounts, of the reported type, that are interspersed in a time sequence where they had accounts that had activity. In these cases, the number of accounts is reported as zero, but the account total is coded missing. Figure 4 below contains an example of this coded missing scenario.

Fig 4: Example of Coded Missing

Sequential Quarter Number	SiteEFXID	NoNFAcur	NoNFA3mon	NoNFAbalance3mon	NoNFA12mon	NoNFA24mon	totNFAcc
27	759979488	1	1	0	1	1	0
28	759979488	0	1	0	1	1	999999999
29	759979488	1	1	0	1	1	0

B. Missing Value Imputation

Even in cases where non-zero and non-missing coded values are available, logic errors are present. For example, in Figure 4, the quarter 3 maximum balance for the past 3 months (HstNFACmt3mon) is reported to be 25,000, but in the current quarter and the prior quarter, the current month maximum balance values (totNFAcc) are reported to be over 42,000. In this case, using the one of the two "current" values is the logical choice or both of the current month values must be reduced. In addition, the 2nd quarter has a reported value of 65 which matches the current value for the prior quarter. This scenario is prevalent throughout the data files, so it is assumed that "prior n months" refers literally to months prior to current, but not current. Using this insight, 3 month values are logic checked with the current value for prior quarter and updated as appropriate.

Fig 5: Highest Account Balance Logic Error

Sequential Quarter Number	SiteEFXID	totNFAcc	curCrdLmtNFAcur	HstNFACmt3mon
1	620339285	65	7000	65
2	620339285	69174	32000	65
3	620339285	42275	32000	25000

Logic checking the 12 month and 24 month metrics is somewhat easier. Assuming that they use the same base reference for time, the 12 month can be compared to the 3 month values for the current quarter plus 3 prior quarters. The 24 month can be compared to the 12 month for current quarter and 12 month for 1 year prior. For the first, three quarters, the available prior quarters is used. For example, in the second quarter of 2006, the logic check for the 12 month value, uses current 3 month and prior quarter 3 month, sincetwo and three prior quarters are unavailable. When, additional prior quarters, outlined above aren't available, the logic check is restricted to the available quarters. The same logic is applied to the 24 month values for the first four quarters use a similar strategy.

C. Model Target

The current study attempts to predict prospective customers that will have at least one account for which they are 2 or more months behind on payments (including being in default), in the following year. In any given quarter, just under 8% of businesses will have at least one NFA in this status. Almost 62 percent of these businesses will be in the same status for at least one NFA in the prior year. Since it would be irrational for a provider to extend an offer to these accounts, any business in the model development time period, that has been behind by 12 or more months on an NFA will be removed from consideration. Including these accounts in the analysis would also artificially "improve" model performance, to the potential detriment of generalizatibity to prospect businesses, that are the rational target of business. Not only is the decision, to remove current year "bad accounts" (those with at least one account that is two or more months behind on an NFA payment) is consistent with rational business behavior, but it provides a bigger challenge when attempting to predict business behavior and a greater test for the proposed model development process.

Removing the businesses with the 12 month behavior also removes companies with similar 24 month behavior, as well as businesses that are 3 or more months behind in the 12 month period. Related data fields report these behaviors, so this logic, also removes data elements that would provide false high performing model behavior.

IV. METHODOLOGY

This study uses full year data, for a base year to predict the delinquent behavior, in the following year. After removing currently delinquent accounts, just under three percent of the businesses had an account delinquency of two or more months, in the following year. The base year for model development is 2006, to predict behavior in 2007. The models are developed on a training sample for, for 2006, then assessed for generalization, on a validation sample for 2006 as well as for each year 2007 through 2013, predicting behavior in the following year.

Almost 2.9 million records were available for 2006. The data were split 30/70, into training and validation data sets, respectively. With almost 900K records, care must be taken, since almost any affect will meet the traditional 0.05 threshold for a significant p value. For the purpose of model development, the p value threshold is reduced to 0.0001.

The primary goal of the study is to compare the proposed ensemble of logistic regression models to a logistic regresion "Base Model" that is built using standard data transformations. The Ensemble Model is developed without the benefit of recodings or transformations, beyond the coded missing value and logic checks described in the Data section.

A. Base Model

The "Base Model" uses a series of data evaluations, transformations and recodings to build the best possible

performing logistic regression model. There are over 300 data elements, in each of 4 quarters, that are available for model development, including a considerable amount of redundant information. If individual predictor interpretation was not required, a principal component analysis or factor analysis could be used to reduce data without reducing valuable variance that has the potential to be predictive. Instead, variable clustering, using SAS' Proc Varclus [4], is performed, to identify similar underlying critical variance dimensions while allowing the analyst to select representative data elements that are highly associated with the variance dimensions.

Each quarter is assessed, to identify variance dimensions and representative data elements. A 60 variable cluster solution, for each quarter, represents over 80% of the variance within a single quarter, so it is chosen for the first clustering step. The 240 remaining data elements for the year are then combined and an additional variable clustering is conducted, to identify the 53 data elements to be used for recode. These data elements represent over 80% of the variance, in the 240 data element set.

At this point two different binning procedures are used:

a) equal size bins, for each data element (but may not be equal record counts for each), using Proc Rank[5], in SAS

b) equally spaced bins, where the range for each bin is equal, but the counts within the bins aren't equal.

For each binned data element, two nonlinear transformations are developed:

a) the odds, for the dependent variable (a binary flag that indicates the business has an NFA that's two or more months late payment)

b) log-odds for the dependent variable.

At this point, there are 371 variables. As was the case with the 300+ variables at the start of the analysis, there is a high degree of information redundancy across the 371 variables. A third variable clustering process reduces the number of data elements to identify 57 data elements that represents over 85% of the variance in the data.

The logistic regression model was estimated using SAS Proc Logistic [6] with backwards elimination. The initial model was developed using a data element p value of 0.0001 (i.e., SLSTAY = 0.0001). From that point, results were examined to identify low contributing data elements as well as data elements that may suffer from multi-collinearity issues. The final Base Model has 22 predictive data elements.

It should be noted that while the project was spread over several months, this model development effort required the equivalent of at least 2 weeks of full time effort.

B. Logistic Ensemble

The fundamental premise of ensembles is the aggregation of the results of multiple sub-optimal models with an application of a "winning strategy" to result in the final prediction. For classification problems, like the binary classification examined in this study, strategies range from simple voting (classify a record on all models then count the yes vs no results and classify accordingly) to more sophisticated strategies like using the model results as inputs to an additional binary classifier – similar to the functionality of a simple neural net. For the purposes of this study, an optimal linear combination of the predictions, based on the sum of squared errors, per Brieman's work [3], for the set of logistic regression models is used.

Similarly, the development of the models that make up the ensemble may utilize different strategies. For random forests, the standard methodology is to use bootstrapping to develop multiple models using the same data elements in the consideration set. This project will utilize the stacking of multiple logistic regression models that are developing using a randomly chosen subset of the available data elements.

One of the goals of this effort is to test to determine, if an ensemble based on minimal analyst intervention can produce results that are on par with the more involved model development process outlined earlier. To achieve this goal, data element samples were drawn from the 300+ raw data points that were available after data cleansing outlined in section 3 above. Due to the goal of minimal intervention by the analyst, models for ensemble consideration will be developed using individual quarter data, with each quart having an equal number of models that predict behavior in the following year.

For each quarter, 40 samples of the data elements were drawn. A random number generator in Microsoft Excel was used to sample 25% of the available data elements. The logistic procedure in SAS, with backwards elimination (again using SLSTAY = 0.0001) was used to develop the 160 models. No additional model/variable assessments were used to improve individual model performance or to reduce model generalization concerns like multi-collinearity.

C. Identifying the Optimal Ensemble

After the 160 models were estimated SAS's Proc LP procedure [7] was used to determine the optimal linear combination of models that maximize prediction. A quadratic program, using the least squares cost function, found in regression analysis was used. To simplify, this step a closed form of the squared error function was calculated by expanding the quadratic function. The resulting quadratic program is then solved.

$$min \sum_{over\ T} (y - \lambda_1 p_1 - \lambda_2 p_2 - \cdots - \lambda_{160} p_{160})^2 \qquad (6a)$$

$$st\ \lambda_1 + \lambda_2 + \cdots + \lambda_{159} + \lambda_{160} = 1 \qquad (6b)$$

$$\lambda_i \geq 0 \qquad (6c)$$

where T is the training set and
p_i is the logistic regression estimate for the ith model

For 160 models, the closed form of the objective function requires over 13,000 coefficients. Solving the quadratic program identified 22 models that contribute to the optimal

combination of values. For others, the $\lambda_i's$ are 0. For each record in the validation set and for each subsequent year, the 22 models with non-zero weights ($\lambda_i's$) are scored and the weights are used to generate the linear combination of models (i.e., the ensemble score).

V. Findings

A. Base Model Performance

The Base Model has relatively good performance. The 22 predictive data elements produced a model that has a percent concordance of 85.2 and a KS statistic of 54:

Fig 6: Base Model Performance

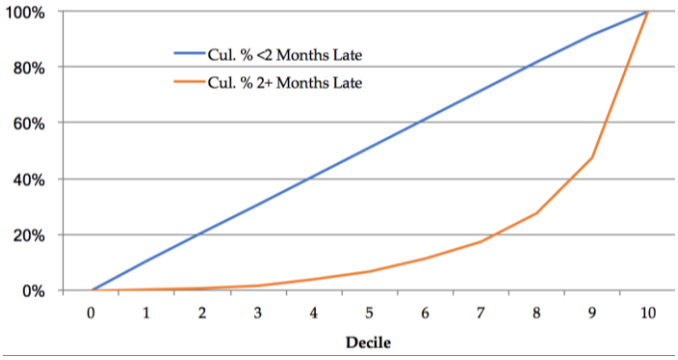


Fig 5. Base Model percent of "bad accounts" by model score decile in the model ordered validation sample vs the percent of "good accounts"

Somewhat surprisingly, the Base Model had very impressive performance when applied to the additional years. These additional years of data represent not only a lag forward in time (one type of generalization challenge), but also due to the increasing number of available records, for scoring, the additionaly years of data represented an assessment of generalization on new prospective accounts. Performance not only didn't degrade but had a modest increasing trend, in terms of the KS statistic:

Fig 7: Base Model Performance Over Time

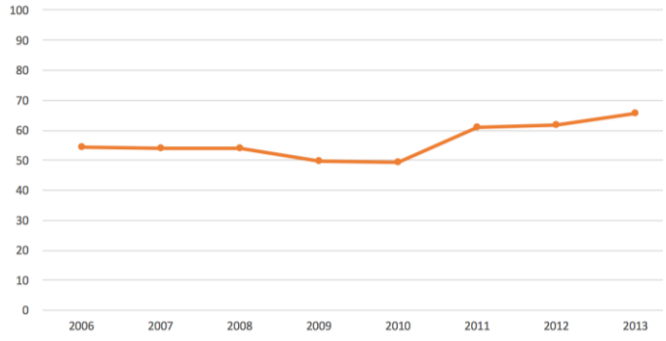


Fig 6. Base Model KS statistic for application of the model to independent samples, over time.

This model development effort required the equivalent of 3 days to complete. Most of the time (roughly 2 days) was waiting for the near automated development of the 160 predictors that make up the ensemble.

B. Ensemble Model Performance

The ensemble of quarterly logistic regression models had comparable performance. While percent concordance isn't available, the KS statistic can be calculated for the ensemble. When applied to the 2006 validation set, the Ensemble Model performed on par with the Base Model and had a KS statistic of almost 58:

Fig 8: Ensemble Model Performance

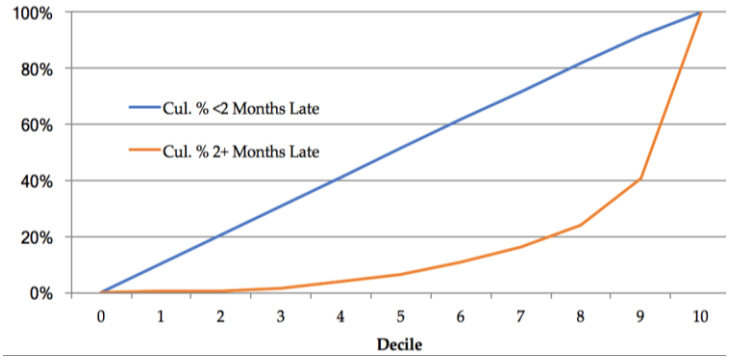


Fig 7. Ensemble Model percent of "bad accounts" by decile in the model ordered validation sample vs the percent of "good accounts"

While higher performance in the same time period is good, the most important assessment is the application over time. As can be seen in the chart below, Ensemble Model continued to outperform the Base Model in each of the available years of data.

Fig 9: Ensemble Model Performance

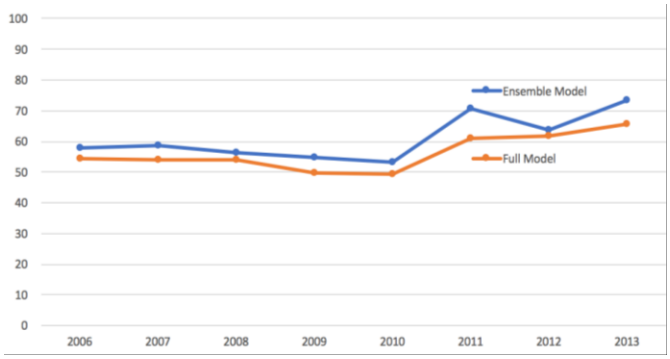


Fig 8. Base Model KS statistic for application of the model to independent samples, over time.

C. Coefficient Interpetation

For a single logistic regression model, an analyst interprets a coefficient in terms of change in log odds, for a unit change in the data value. In its simplest terms,

$$\ln\left(\frac{p}{1-p}\right) = \beta(x+1) \qquad (7)$$

$$\frac{p}{1-p} = e^{\beta x}\, e^{\beta} \qquad (8)$$

So change x by 1 and the odds change by a multiple of e^{β}.

Given the nature of the ensemble, calculating change in odds isn't viable. Instead, analysts can lean on multivariate calculus and derive the rate of change with respect to a given data element. For n predictors in the optimal ensemble:

$$\frac{\partial p}{\partial x_i} = \lambda_1 \frac{\partial p_1}{\partial x_1} + \lambda_2 \frac{\partial p_2}{\partial x_2} + \cdots + \lambda_n \frac{\partial p_n}{\partial x_n} \qquad (9)$$

$$= \lambda_1 \frac{\partial p_1}{\partial x_i} + \lambda_2 \frac{\partial p_2}{\partial x_i} + \cdots + \lambda_n \frac{\partial p_n}{\partial x_i} \qquad (10)$$

$$= \lambda_1 \frac{\beta_{1i} e^{-\beta_1 x}}{(1+e^{-\beta_1 x})^2} + \cdots + \lambda_n \frac{\beta_{ni} e^{-\beta_n x}}{(1+e^{-\beta_n x})^2} \qquad (11)$$

$$= \lambda_1 p_1^2 \beta_{1i} e^{-\beta_1 x} + \cdots + \lambda_n p_n^2 \beta_{ni} e^{-\beta_n x} \qquad (12)$$

While this calculation is more complicated than what is typically done, in interpreting logistic regression coefficients, the resulting insight is a direct change in model score or credit score (if the model scores are converted to credit score metrics) for a change in identified parameter for a specific business behavioral profile. For general reporting purposes (e.g., reports to regulators), the mean or median value for the data element could be used.

D. Conclusions

The current study represents a successful enhancement of a powerful machine learning enhancement process, blended with a traditional statistical predictive methodology. The Base Model, performed very well, but given the 3 separate variable clustering processes that were used to reduce the number of data elements, may have reduced the available variance (i.e, "explanatory power" of the predictive data elements) by more than 50%, since roughly 15% of variance is removed in each of the variable clustering steps.

The Ensemble Model didn't have the benefit of the nonlinear transformations, but was able to utilize all of the initial variance that was available in the data, for the year. In addition to the opportunity to "operationalize" the Ensemble Model development process, and the reduced time to develop, this increase in available variance, to contribute predictive power is a very compelling argument for employing the proposed ensemble process.

REFERENCES

[1] Jehad Ali, Rehanullah Khan, Nasir Ahmad, Imran Maqsood. "Random Forests and Decision Trees." *IJCSI International Journal of Computer Science Issues, Vol. 9, Issue 5, No 3*, pp 272-278, September 2012

[2] Wolpert, David H.. "Stacked Generalization." *Neural Networks* 5, 1992, pp 241-259, 1992

[3] Breiman, Leo. "Stacked Regression." *Machine learning 24*, July 1, 1996, pp 49-64

[4] "SAS VARCLUS Procedure" SAS, April 17, 2017, https://support.sas.com/documentation/cdl/en/statug/63033/HTML/default/viewer.htm#statug_varclus_sect004.htm

[5] "SAS Rank Procedure" SAS, April 17, 2017, http://support.sas.com/documentation/cdl/en/proc/70377/HTML/default/viewer.htm#p0le3p5ngj1zlbn1mh3tistq9t76.htm

[6] "SAS Logistic Procedure" SAS, April 17, 2017, https://support.sas.com/documentation/cdl/en/statug/63347/HTML/default/viewer.htm#statug_logistic_sect004.htm

[7] "SAS LP Procedure" SAS, April 17, 2017, http://support.sas.com/documentation/cdl/en/ormplpug/64004/HTML/default/viewer.htm - ormplpug_lp_sect009.htm