Risk Scoring Classification Performance Optimization

By

Kathleen Kerwin

Thesis Project

Submitted in partial fulfillment of the

requirements for the degree of

MASTER OF SCIENCE IN PREDICTIVE ANALYTICS

Northwestern University

August 2017

Nathaniel D. Bastian, PhD, First Reader

Don Wedding, PhD, Second Reader

Abstract

RISK SCORING CLASSIFICATION PERFORMANCE OPTIMIZATION

The primary purposes of binary classification is performance optimization since even the slightest prediction improvements can have signification implications for each field application. Finding the most effective separation of classes is the key success indicator in how well each model performs. When modeling prediction fails, the data entry is placed in either a false positive (incorrectly predicted) or negative (incorrectly failed to predict) category. The consequences for failing to correctly categorize can range from wasting resources and missed opportunities to serious harm caused in medical diagnoses. Optimizing classification performance focuses on reducing misclassifications.

Improvements can be identified after applying metrics then in finding ways to increase accuracy and reduce misclassification. Accuracy of classification and AUCROC, area under the Receiver Operating curve, are the most commonly applied metrics, however identifying insights into misclassification requires reviewing Type I and II error metrics. Ensemble methods (bagging, boosting and stacking) are a creative means of resampling and will be utilized to improve the performance of base learners in stacked models.

The German and Australian credit risk scoring datasets were run through 9 diverse algorithms, as well as adding bagging and boosting in ensemble stacked models based on strong and weak learner correlations. The results were ranked based on the overall optimal performance.

The Support Vector Machine (SVM) algorithm ranked best for both datasets with no false positives and negatives. The stacked models were in 16 of the top 17 ranked models, however the ranking of specific models was inconsistent. This study found that the stacking ensemble premise did not have a clear delineation in performance improvement in combining the weakest and strongest learners together, however stacking itself proved very successful. The premise is that combining the weakest learners, least correlated, should be more effective than the strong learners since they have the most to benefit from reweighting for resampling that promotes the weakest and demotes the stronger learners helping to focus on the weakest algorithms fixing their misclassifications.

The conclusion regarding model choice is that no assumptions could be made regarding algorithm performance and that empirical testing is needed for model selection.

The key implication of optimizing classification performance is the reduction of misclassification errors and thereby enabling greater classification accuracy which translates into gaining better insights from the risk score modeling process that can be effectively prescribed back in an organization. One of the most important areas of future work continues to be finding solutions to class balancing which improve the quality of performance metrics that are inherently skewed.

Keywords: *business risk assessment, classification performance optimization, credit risk scoring, risk scoring analytics*

Acknowledgements

First I want to thank my husband who believed in my ability and who supported me graciously through the most difficult times. I also thank my family for the sacrifices they made to help me during my research effort.

I would like to thank Dr. Nathaniel Bastian for being my advisor and suggesting that I include Ensemble and Deep Learning methods in my study and Dr. Donald Wedding, second reader, who reviewed my work for graduate level quality. I would also like to thank Dr. Mary Jo Kranacher for focusing my efforts on Internal Audit for my proof of concept, as well as believing that I just may produce a study that would be of business interest.

Wade Gomes, Dr. Anatole Klyosov and Carol Lindstrom were my references to gain acceptance in the Northwestern University MSPA program. I thank them for their letters of recommendation which enabled me to move forward in my studies.

Contents

List of Figures

CHAPTER 1

Introduction

The primary purpose of this study was to build a new risk scoring process. The credit risk scoring methodology most closely matched the technical requirements and was utilized as a statistical model prototype that quantified the prediction whether an event may or may not occur.

The first credit scoring efforts were created out of necessity and were developed through intuition and experience. Risk scoring was initially achieved manually before being automated through statistical and machine learning techniques which added fact based decision making.

Risk scoring is a method to rate historical behavior enabling a business to make a decision on whether to take an action from criteria that predicts when an event may occur or may not occur. The credit risk example is that a loan will or will not be repaid. A baseline hurdle rate of acceptable risk is linked to business tolerance. In credit risk scoring applicants are rated in order to predict quality of repayment risk.

Business risk can be studied through behavior monitoring identifying inherent risk (like death and car accidents) versus fraud or improper behavior (such as intrusion, delinquency, or account default) that can be controlled or mitigated as in a loan appraisal process. Some risk categories include marketing, credit, collections, operating, insurance, as well as fraud detection and prevention.

Credit scoring methodology is implemented system wide by lending institutions and therefore is an excellent source to research how other systems with similar requirements can be defined and developed. Scoring is becoming of greater importance because of increasing business rivalry locally, nationwide and globally, as well as the ever increasing volume, velocity and variety of data to be managed. Scoring enables better fact based decisions selecting between the favorable to the less or unfavorable which may help improve the ranking of an organization in the competitive struggle.

Lending institutions rely on Credit Risk Analytics to determine the customer's ability to payback credit with the end result affecting the institution's financial bottom line for profit and loss. The modeling and scoring techniques adopted play a significant role in decision making and subsequently the institution's success or failure. If the applicant which is a good risk but is denied, the institution has missed a revenue opportunity which will go unnoticed unless the applicant selection process is monitored and measured. If the bad risk applicant is approved, then there is the possibility of financial loss on the loan, which is more expensive than the former missed opportunity. The failure to add a revenue source is not as expensive as loss foregoing revenue and principal. Losses can only be used against revenue sources for financial and taxation purposes and the tipping point is the break-even point. Another issue is the threshold where approval and denial are separated.

There are several phases of risk scoring in the banking industry. The first rates applicants on the likelihood to repay with a window that is a static point in time with the process predicting on historical data available regarding an applicant's credit history. The second phase is called behavior monitoring which rates changes to the applicant's behavior, after being approved for a loan, using a time series predictive analytics process. A third application for the credit industry is collection monitoring which follows up repayment of defaulted loans. One or all of these long established methods may be useful to new risk scoring methodologies.

Well-researched and implemented analytic techniques can be applied to new risk scoring projects. Critical legal and business issues, regarding which algorithms to use in institutional lending, apply similarly to assessing other risk areas. Regulatory and legal hurdles may determine that more transparent and interpretable algorithms, such a logistic regression and trees, are selected over more advanced machine learning algorithms since the latter are more difficult to understand and interpret. A primary focus of this study is how to improve risk scoring methods by optimizing classification techniques and reducing the risk of incorrect false positives or false negatives, in addition to ascertaining more effective metrics to use given certain circumstances.

By predicting possible or probable events, they can be foreseen, prevented or steps taken to mitigate potential damage by anticipating how the events should be handled. The focus of this study is identifying metrics to improve optimization of a binary classifier, as well as ranking models using these metrics for the purpose of model selection. Optimizing helps to minimize failures to identify events or misclassify events.

Until the requirements for risk factors are identified and refined, no method can ensure project performance improvement. The credit risk scoring application is an excellent business and technical model for research but may give a false impression about the real work needed to start a new risk scoring application. Until the business requirements are written and iteratively tested and modified through to deployment with insight being brought back into the organization and assimilated, the metrics and models will not reach full optimization.

1.1. Technical Risk Scoring Research

Credit risk history extends to the beginning of commerce (Louzada et al., 2016). The stock market in the 1920's and 1930's required credit risk to be more quantified and streamlined which modernized the process traced to Durand (Durand, 1941). Current research uses automated decision models to quantify risks (Thomas et al., 2002). Today the focus is on using predictive analytics to reduce customer defaults through statistical methods (Abdou & Pointon, 2011). New technology enables better discovery of correlations (Koh et al., 2006) and patterns in the data raising the quality of fact based decisions. A premise for predictive modeling is that it can foretell the future from an applicant's behavioral history by monitoring changes over time (Koh et al., 2006).

Another important aspect of credit scoring is measuring expected earnings in connection to the probabilities related to default for an overall risk assessment model (Boyes et al, 1989). This quantification effort is the foundation of a series of investigations required by regulators of lending institutions since each entity needs to justify their lending strategy and tactics as a part of their fiduciary responsibility. The econometric foundation of this research is statistical and operational research using probability modeling employed by a wide number of financial areas that are the basis for forecasting financial risk that includes decisions before and after the credit has been approved (Thomas, 2000).

Updated risk scoring procedures include statistical and artificial intelligence techniques (Wang et al, 2011), however there is no generally accepted model. There is interest in using ensemble techniques (specifically boosting, bagging and stacking) for the purpose of finding improvements to existing models for Accuracy and Type I/II errors. Ensemble methods are characterized as combining hypotheses (multiple learners) rather than applying single hypothesis as in ordinary machine learning.

Multidisciplinary Nature of Machine Learning, SAS Training
https://support.sas.com/resources/papers/proceedings14/SAS313-2014.pdf

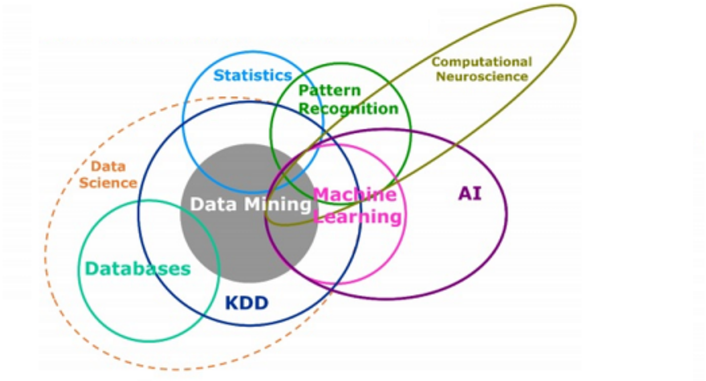


Figure 1.1. Predictive Analytics draws from many dissimilar overlapping disciplines (Hall et al., 2014)

The use of predictive modeling automates the approval process that corrects the bias (Crook 1996) in the previously employed judgmental techniques that relied primarily on subjectivity and provides a legally defensible threshold for approval or denial of applications. Predictive analytics applies only the most significant variables whereas judgmental techniques based on experience may bring in bias (Crook, 1996). Automated credit risk scoring can introduce errors due to misclassification (Abdou & Pointon, 2011) which means that issues with classification analysis are points of failure (Anderson, 2003). Failure to update credit risk specifications and historical data are also limitations in automating the process (Abdou & Pointon, 2011).

There is evidence that the lenders performance and accuracy of risk assessment is improved using credit score prediction (Avery et al, 2000), as well as to borrowers due to the increased efficiency of the process that provides greater credit opportunities. The supposition is that repayment history is the best indicator of future behavior. Although weaknesses are found in how credit bureau scoring data is compiled across bureaus, lenders use other information to make lending decisions. Overall automated systems that create credit scores are better than using judgment alone.

An additional step, after approval, includes a follow-up process that rates the outcome of the predictive model selected, which provides valuable feedback whether improvements or changes are needed grounded on the actual outcome of customer repayments or delinquencies. The highest risk applicants are the ones that provide the greatest return on investment but they also must be carefully selected since the potential losses include both the interest revenue and principal which are mitigated through more accurately identifying the probability of repayment (Hand & Henley, 1996).

The following list of algorithms used in risk scoring are a few examples from an extensive list of research papers on comparative analysis of modeling methods. Logistic Regression has been included in most comparative modeling studies (Crook et al, 2005). Bayesian and ID3 algorithms were compared favoring the Bayesian algorithm (Madyatmadia et al., 2005). Other techniques studied include "weight of evidence measure, regression analysis, discriminant analysis, probit analysis, logistic regression, linear programming, Cox's proportional hazard model, support vector machines, decision trees, neural networks, k-nearest-neighbour, genetic algorithms and genetic programming" (Abdou & Pointon, 2011). Recently Deep Learning is being applied to credit risk scoring taking advantage of leveraging the Deep Belief Network added to the Neural Network algorithm (Luo et al, 2016).

In "A comparative assessment of ensemble learning for credit scoring" (Wang et al, 2011), four base learners (LRA, DT, ANN and SVM) were tested using bagging, boosting and stacked ensemble models. Bagging performed better than boosting with stacking and bagging DT improving more based on classification metrics.

An updated list of modeling techniques adds fuzzy logic (Louzada et al., 2016) with accuracy, sensitivity, specificity, precision and recall, with false positive and false negative rates as a classification metrics. The study included 187 papers of which 51.3% studied proposed new credit scoring methods, 20% were hybrid models, 15% were on combined models, and 13% being neural networks with the rest having less than 10% representation. For credit scoring specifically, the most common techniques are hybrid and combined techniques, support vector machines at 17.6% and neural networks 20.6%, with linear discriminant analysis rarely used at 1.7%. The two most relevant preprocessing techniques of model selection were missing data procedures with the k-fold and holdout methods the most common validation techniques.

There is no consensus or best practices for variable selection, models selected, validation techniques, or cut off points on which predictive analytic processes are optimal with the exception to the application of the confusion matrix or classification table with metrics such as the Receiver Operating Curve (ROC), and the Gini Coefficient which are the widely utilized (Abdou & Pointon, 2011).

1.2. Risk Assessment

Risk scoring is not a goal itself. It is a predictive analytics project that solves a business problem. The credit card dataset used in this study has been refined over a period of time and conforms to the risk associated with credit default and creditworthiness. New risk scoring projects undergo iterative phases of identifying risk factors associated with the business requirements that will be mapped to the response and predictor variables included in the representative dataset to be used.

The initial project requires a thorough risk assessment and analysis that will result in the selection of variables to use in the risk scoring process which are also combined into a risk history variable that is similar to the credit risk history variable. The assessment process identifies, analyzes, and evaluates how to control risks helping the business calculate their risk exposure and tolerance. The dataset attributes are selected to measure the risk factors identified. The insight gained in the assessment process is then added back into the project iteratively updating the variable selection while ranking variable importance to the known risk exposures, as well as updating and calibrating the risk history variable values and risk scorecard. Risk needs to be quantified and managed to make better decisions regarding risk exposures (Koh et al., 2006). The initial risk assessment and evaluation process should include strategies to manage the ’what happens next’ scenario when the potential risk evolves into an event (Siddidi, 2006). Examples in following up risk assessment in lending institutions can provide process road maps for new risk scoring projects. They include foreclosure of loans, write-off principal and interest revenue, repossession, the effects that consumer or business bankruptcy, and court judgments will have on the institution. Prescribing insights back into an organization is then measured through metrics checking for changes and improvements.

1.3. Machine Learning Algorithms

The response variable determines whether regression or classification algorithms are applied which are numeric or categorical respectively. Classification methods measure whether an event may occur or not occur. Regression methods can also be applied to quantify the value of potential loss. This study focuses primarily on the optimization of binary classification methods and uses the credit risk example referencing applicant loan approval based on the binary credit risk variable indicating whether an applicant will default on a loan which is a bad risk or repay a loan representing a good credit risk. Improvement of classification depends on constant recalibration in all areas including whether the strategic goals are being met and reapplied effectively into the organization.

Some classification algorithms that were considered for inclusion are displayed in this graphic (Brownlee, 2015); see figure 1.2.

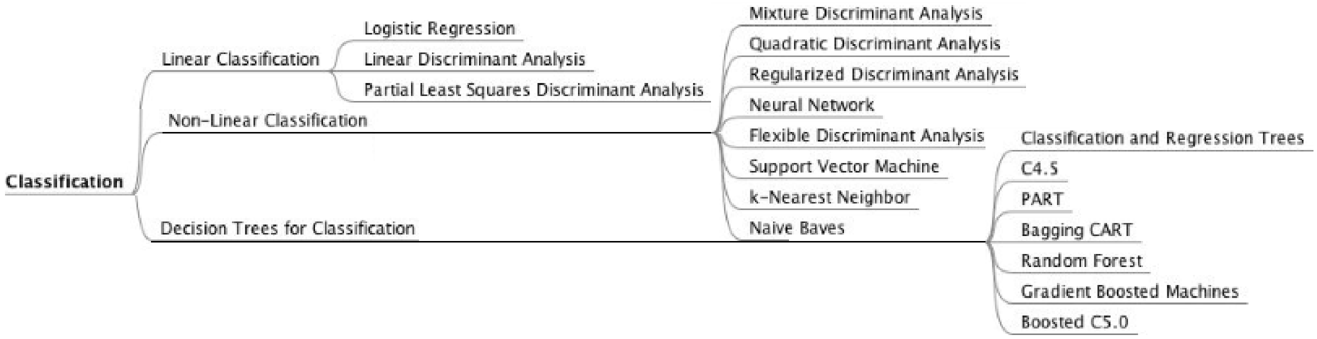

Classification
Linear Classification
Logistic Regression
Linear Discriminant Analysis
Partial Least Squares Discriminant Analysis
Non-Linear Classification
Mixture Discriminant Analysis
Quadratic Discriminant Analysis
Regularized Discriminant Analysis
Neural Network
Flexible Discriminant Analysis
Support Vector Machine
k-Nearest Neighbor
Naive Baves
Decision Trees for Classification
Classification and Regression Trees
C4.5
PART
Bagging CART
Random Forest
Gradient Boosted Machines
Boosted C5.0

Figure 1.2. Classification algorithms

CHAPTER 2

Methodology

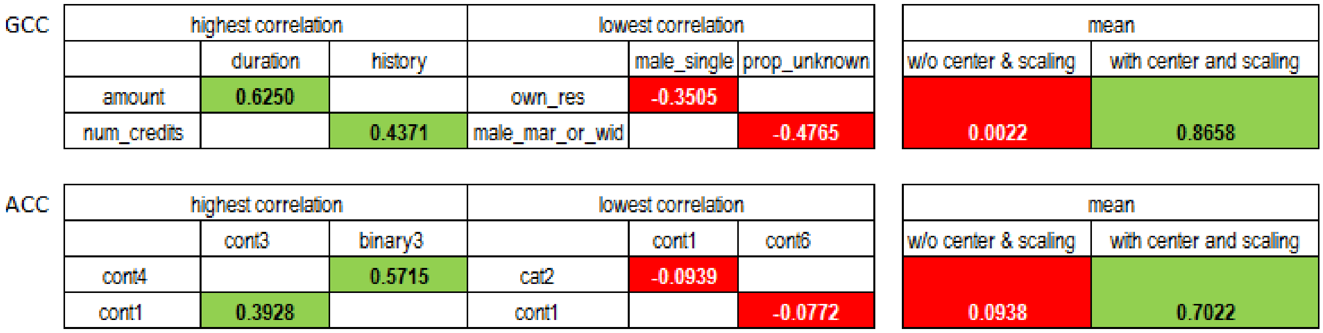
2.1. Data and the Environment

Two datasets with credit card scoring data were used in this study; the German Credit (1000 entries 30/70 split for bad versus good credit) and Australian (690 entries with 307 goods and 383 bads) datasets. The source of the datasets is the University of California at Irvine (UCI) Machine Learning Repository (Lopes et all, 2010). The bad entries were classified as the event having occurred.

There were no missing values in either dataset which had previously been imputed using median values. Each dataset was already in a numeric version having the binary and categorical variables converted to discrete variables. The German Credit age, amount of loan, and duration of loan variables were binned. The Australian Credit variables were not distinquished by names and are identified in order of appearance in the dataset as binary(x), continous(x), and categorical(x) to binary(n), continous(n), and categorical(n). Since context was not provided, binning was not possible because the each bin is based on the meaning and purpose of the variable.

Variable selection was completed in two steps. The first step was a bivariate correlation analysis comparing each predictor to each other eliminating one variable of the pairs if the correlation was greater than 75 percent. No variables in either dataset was eliminated at this stage. The second step used Recursive Feature Elimination (rfe) with the randomForest (rfFuncs) predefined function, from the caret library for variable selection. The rfFuncs function considers all variables in each iteration or resample from which the top three variables were selected. Binary2, continuous4 and continuous7 were selected for the Australian set with checking account balance, credit history and savings account balance selected for the German set; see Appendies B and C for a complete list of the dataset variables.

Centering and scaling of predictors displayed the most interesting EDA relationships. The reason for centering and scaling of data is that the units for the predictors need to be similar for a reasonable interpretation of their association to the response variable. Standardization normalizes the distribution of the predictors ensuring comparability; see figure 2.1.



GCC

	highest correlation		lowest correlation		
	duration	history		male_single	prop_unknown
amount	0.6250		own_res	-0.3505	
num_credits		0.4371	male_mar_or_wid		-0.4765

mean	
w/o center & scaling	with center and scaling
0.0022	0.8658

ACC

	highest correlation		lowest correlation		
	cont3	binary3		cont1	cont6
cont4		0.5715	cat2	-0.0939	
cont1	0.3928		cont1		-0.0772

mean	
w/o center & scaling	with center and scaling
0.0938	0.7022

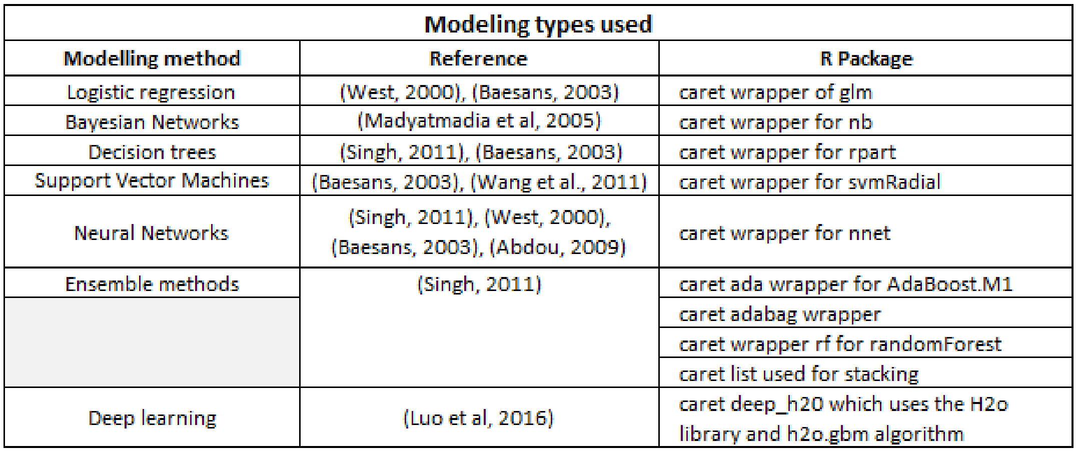
Figure 2.1. High and low correlations with predictor means for each dataset

Note that the standardized mean is higher than without center and scaling for both datasets. Means closer to 1.0 have a normal distribution. The caret library of wrapper functions, introduced later in section 2.4 Technical Issues on page 31, enables preprocessing with centering and scaling. Without including standardization of the predictors, the results of the predictions would not have as reasonable an interpretation.

The development Windows 10 environment included RStudio with R version 3.4.1, as well as accessing the H2o (H2O.ai Team, 2016) client through an R localhost for the Deep Learning modeling. Parallel processing was enabled with the R doParallel library.

2.2. Modeling Methods

The following list of diverse algorithms was selected after conducting a survey of the most efficient risk scoring algorithms from which a single model will be selected based on the best ranking measured by a list of metrics.



Modeling types used

Modelling method	Reference	R Package
Logistic regression	(West, 2000), (Baesans, 2003)	caret wrapper of glm
Bayesian Networks	(Madyatmadia et al, 2005)	caret wrapper for nb
Decision trees	(Singh, 2011), (Baesans, 2003)	caret wrapper for rpart
Support Vector Machines	(Baesans, 2003), (Wang et al., 2011)	caret wrapper for svmRadial
Neural Networks	(Singh, 2011), (West, 2000), (Baesans, 2003), (Abdou, 2009)	caret wrapper for nnet
Ensemble methods	(Singh, 2011)	caret ada wrapper for AdaBoost.M1
		caret adabag wrapper
		caret wrapper rf for randomForest
		caret list used for stacking
Deep learning	(Luo et al, 2016)	caret deep_h20 which uses the H2o library and h2o.gbm algorithm

Figure 2.2. Classification algorithms included in study

The list includes 6 base models (base learners) and 3 ensemble methods (bagging and boosting). Furthermore sixteen stacked models will be developed after assessing the strong and weak correlations of the base learners.

2.3. Model algorithms

2.3.1. Logistic Regression

In a linear regression equation for credit risk, the binary classification response y indicates that a loan defaulted (event = 1 occurred) or was paid back (event = 0 didn't occur).

$$Y = b_0 + b_1 X_1 + b_n X_n + e \tag{2.1}$$

The relationship of the response to the predictors is a discrete relationship that is not useful; see figure 2.3. A way is needed to transform the response y to a probability between

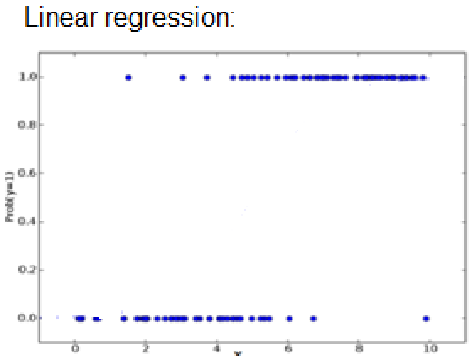


Figure 2.3. Linear regression equation (Ray, 2015a)

0 and 1 while making the relationship between the response and predictors meaningful.

This is accomplished with the logit link function using log transformation converting the y response discrete values of 0 and 1 into event probability results that connect significantly to the predictors; see the transformation in the sigmoid S curve in figure 2.4.

$$Probability{=}(eventoccurred)/(eventdidn'toccur) = p/(1\text{-}p) \tag{2.2}$$

$$Logit(p){=}p/(1\text{-}p){=}e^{(\beta_0+\ \beta_1 X_1)} \tag{2.3}$$

$$Y = e^{(\beta_0+\ \beta_1 X_1)\ /\ (\ 1\ +\ e^{(\beta_0+\ \beta_1 X_1)})} \tag{2.4}$$

The result of the logistic regression equation is to predict the probability whether an applicant will likely default or repay a loan which displays a conversion legend for classification matrix decisions demonstrating how the logistic regression prediction results are compared to the actual results to build a classification matrix (after a threshold percentage has been applied).

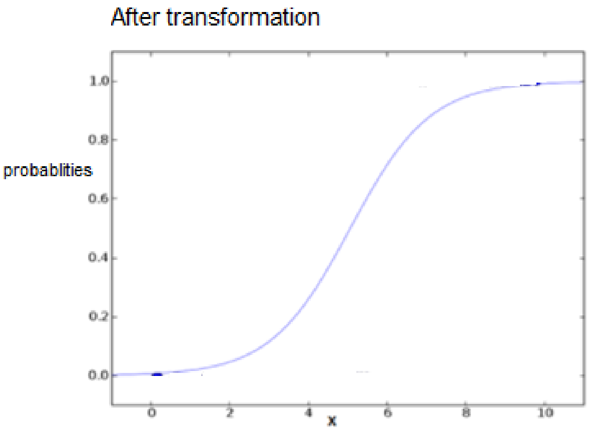


Figure 2.4. Linear regression equation transformed using the logit link

2.3.2. Bayesian Networks

The foundation for Naive Bayes (Cetinkaya-Rundell, 2017) is the Bayes rule named after Thomas Bayes (1702-1761) which provides the logic for conditional probability. The Naive

Bayes algorithm can be built manually and is the ratio of two frequencies. The essence of Bayes rule is:

$$\Pr(A|B) = \frac{\Pr(B|A)\Pr(A)}{\Pr(B)} \tag{2.5}$$

A and B are both events. The probability of Event A is measured by the condition that event B has already occurred; see figure 2.5. The probabilities of the Bayesian network tree are captured in the classification table after stepping through the Bayesian logic; see figure 2.6.

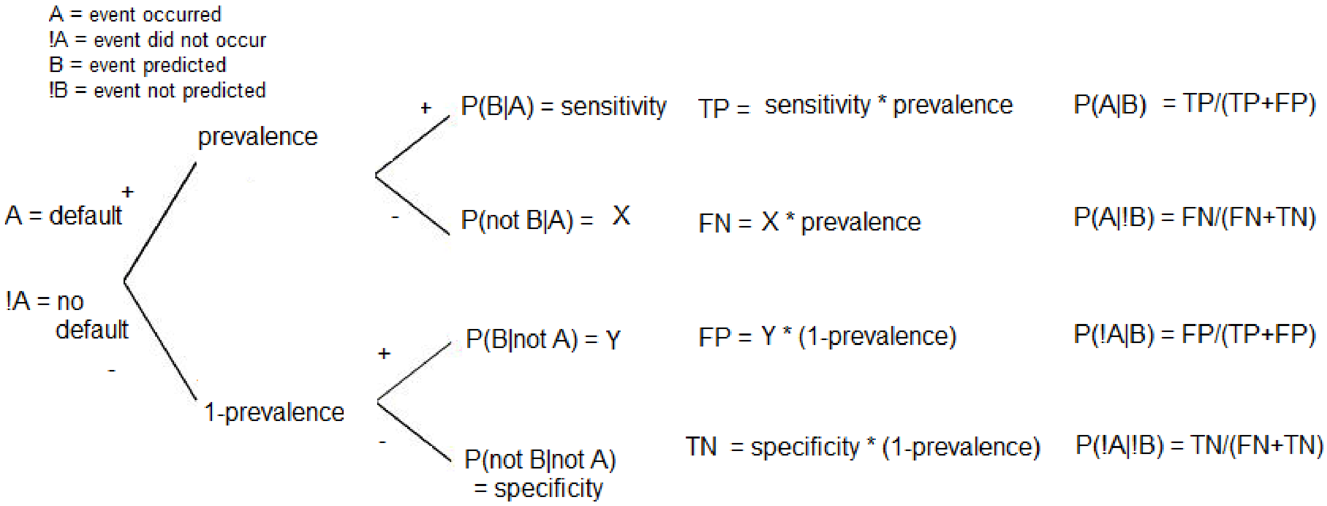


Figure 2.5. Steps of a Bayesian network example; posterior probabilities

A check on whether the logic was followed is that TP and FP, as well as TN and FN should both equal to one. If another test is conducted after the posterior probabilities test, the results from TP and FP are transferred to the next series of steps called prior probabilities since they are based on the previous test.

2.3.3. Decision Trees - RPART recursive partitioning

Ross Quinlan discovered IDE in 1984 (Quinlan, 1986) which is a top down decision tree. Recursive partition (Lavrenko, 2014) decision trees select the best or significant variables that are chosen for splits based on the response variable then iteratively selects each variable in a divide and conquer manner splitting the data into subsets until no more splits are possible. The weaknesses are that the trees require active pruning of leaves to obtain the best mode and there is also a tendency of over fitting the data.

Entropy and information gain measure uncertainty and confidence respectively selecting the best decisions that don't grow the tree too large or over fit the data. In entropy uncertainty is measured by how pure the node is, that is it can not be split further.

$$Entropy = H(X) = -\sum_{i=1}^{n} p(x_i) \log_b p(x_i) \tag{2.6}$$

The bits needed to measure which subset to make the split is quantified through information gain or Gain (S) from the entropy formula which measures the change from a given set and a class. The entropy formula and confidence level from the information gain determine whether it is better before or after the split. If there is an information gain, the node is retained otherwise it is pruned.

2.3.4. Support Vector Machines

Support Vector Machines seek to find a plane that separates the classes (James et al., 2013) which is a concept that was first introduced by Vladimir Vapnik Ph.D in 1990. Twenty years after it was introduced it is commonly accepted as one of the best classification methods.

The goal is to design a hyperplane to separate a binary classifier labeled as positive or negative points; see figure 2.7.

$$bluehyperplane = b_0+b_1X_1+b_2X_2+...+b_\mathrm{p}X_\mathrm{p} = 0 \tag{2.7}$$

A blue hyperplane that separates the classes is equal to 0 (see blue line in figure 2.7) with the hyperplane in red orthogonal to it. The points on the red hyperplane, to the right side of the blue hyperplane, are greater than or equal to one with the points on the left side being less than or equal to -1 therefore showing the separation of the classes. The best decision rule leaves the maximum margin between the classes using a constrained optimization problem.

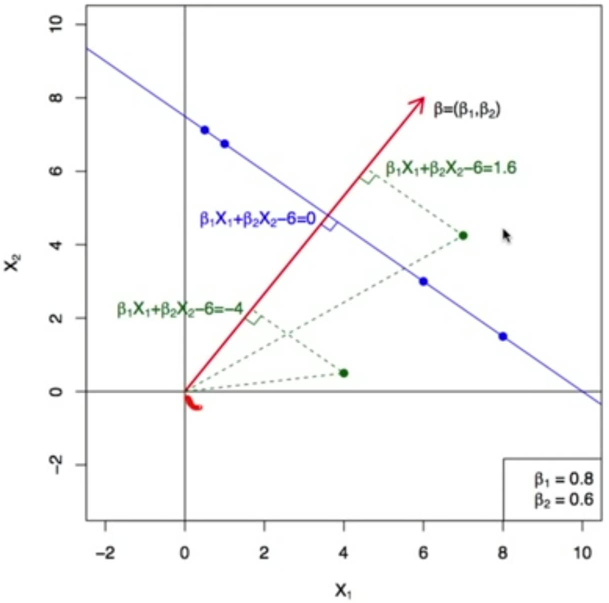


Figure 2.6. Blue and red hyperplanes

2.3.5. Neural Networks

Artificial neural networks are non-linear regression learning algorithms that process information in the same way biological nervous systems work (Stergiou & Siganos ,2017). Supervised learning associates the output pattern with the input pattern whereas in unsupervised learning the out pattern is not introduced into the algorithm (Jha, 2007).

In ANN, the neurons are interconnected solving problems through pattern recognition. A task is learned by providing a dataset for the purpose of recognizing the structure in the data without comparing it to known patterns such as in supervised learning. Unconventional pattern recognition areas such as image, text and speech recognition are uniquely well suited to being solved using ANN's. Neural networks can also solve conventional issues such as classification problems. A series of feature extractions results in a classification tree of attributes. The inputs and outputs match the dimensions of the data

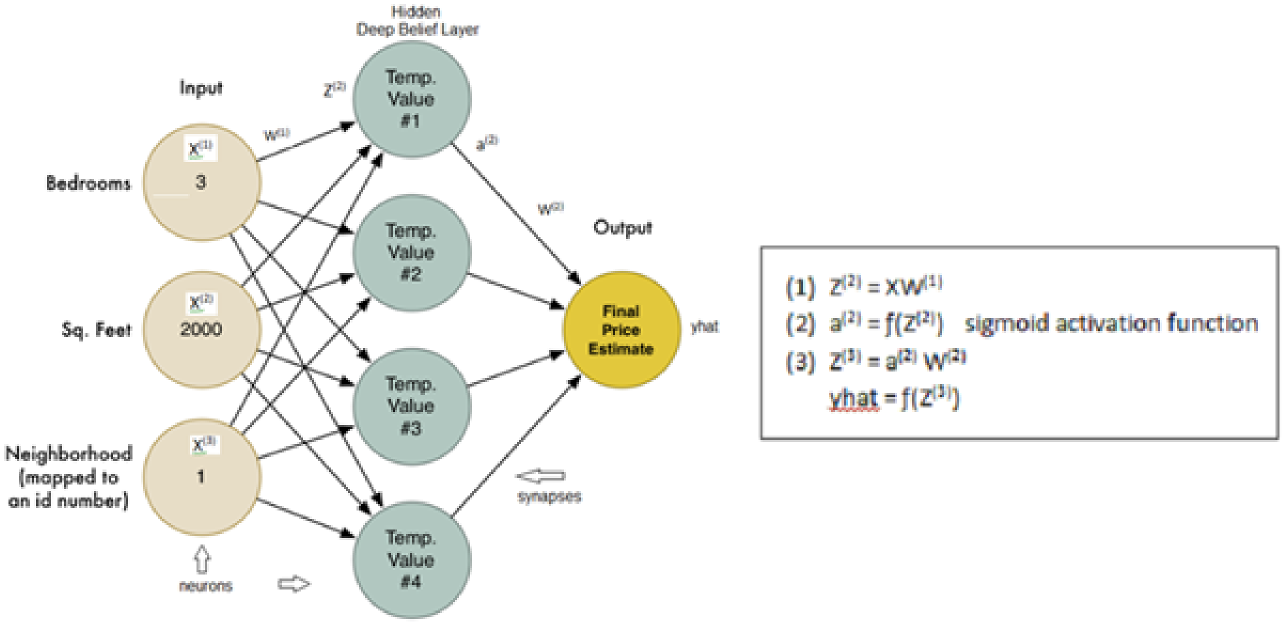


Figure 2.7. Simple neural network (Srivastava, 2014)

(Welch, 2014). The synapses take values from the input then multiply them by weights to get an output value. The neurons add together the output from all the synapses. The input data is scaled (normalized) so that there is a standard relationship between the input and output variables. Vector and matrix multiplication is used to speed up the process with multiple inputs handled at the same time. This is the first pass through without training the model. Next the cost function is applied to minimize the errors.

$$e = (y - yhat) \tag{2.8}$$

$$cost = \sum \frac{1}{2}(y - yhat)^2 \tag{2.9}$$

The final model finds the point where the cost is minimized. This process is called gradient descent. ANN's tend to overfit the training data which makes them less useful for predictive analytics (Srivastava, 2014); see model result output charts.

2.3.6. Deep Learning

Deep Learning uses non-linear regression modeled on artificial neural networks. It is an unsupervised, feature learning, self-taught learning representation with the output not present in the algorithm (Ng, 2016). Andrew Ng noted that researchers have found that experiments isolating brain functions and sensory skills which show that the brain may have one way to learn. This analysis was used in training a learning algorithm. The relation to the neural network is that the hidden belief layer (see figure 2.8 above) is enhanced in multiple layers in a progressive recursive function to gain learning patterns successively from the data. For example, training to recognize images starts by inputting pixels, finding the edges of the pattern in the first hidden layer, finding larger object parts using the information from the previous edge finding pattern, then find the whole picture image with the previous patterns before using vector and matrix calculation to find the yhat output. These calculations are computationally intensive and require parallel, distributed, or cloud based processing. The results indicate that Deep Learning is out pacing any other previously discovered algorithms by improving the accuracy rate and conversely reducing the misclassification rate. The deep learning algorithm, gbm_h2o, uses an ensemble method combining the gbm stochastic gradient ensemble method as the recursive routine managing the deep belief learning functionality.

2.3.7. Ensemble Methods

2.3.7.1. Bagging, Random Forests and Adabag. Bagging is short for bootstrap aggregation. The bootstrap method divides the dataset randomly into subsets with replacement which will have data entries duplicated in more than one model. A full dataset trains a base learner (Wang et al, 2011). One of the subsets is held out for validation purposes. Each subset model is then trained and measured with the final model being a majority vote measured in the metrics indicated (such as accuracy or AUCROC for example) of all the models reducing variance error. Bagging is useful if learned classifiers are affected by small changes in the dataset (Breiman, 1994).

Although bagging improves performance, it has a tendency to overfit with too large trees (Kraus, 2014). It shouldn't be used with too few trees as parameters because every input row gets used too many times. The advantages are that less variable selection is needed and few parameters need to be tuned, as well as being useful with smaller datasets.

2.3.7.2. Boosting. Weak classifiers, with higher error rates that are no better than random sampling (less than 50%), are combined by reweighting then voting is applied to make a strong classifier. The dataset is divided into subsets with replacement. Each new subset trains previously identified weaker classifiers weighted higher and stronger classifiers weighted lower. Based on the tree number selected, additional trees are built but now the higher weighted samples are forced to learn better until an equilibrium is achieved where all the learners have higher than the 50% requirement rate (Ray, 2015). Training on multiple training sets with replacement ensures that patterns of overfitting do not occur (Daumé III, 2012).

Combining, blending or stacking ensembles use "before the fact" techniques (LeDell, 2015).

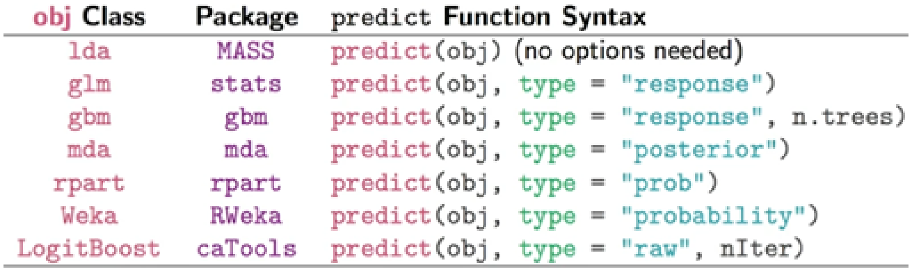
2.3.7.3. Stacking. The least correlated models can be combined to improve the classification categorization. The theory is to use a majority voting method to increase accuracy using a combination of bagging and boosting voting. The strengths and weaknesses in each model are weighted which promotes the weak classifiers and demote the stronger classifiers in each resampling round which focuses the algorithm to fix misclassifications. Only models that improve the overall accuracy are included reducing measurement errors (Ray, 2015) with blending favoring models that are the least correlated. The combined model should therefore outscore the individual model metrics (Zhu, 2001) since the ensemble method complements their opposite strengths (Lee and Jung, 1999/2000).

The modeling phase first evaluates within then between models. Greater accuracy often comes with increased complexity and loss of transparency. It would be helpful to answer the question, can enough information be gained during the validation and evaluation process that developers can provide stakeholders so they have greater confidence in selecting more complex models?

2.4. Technical Issues

Starting the comparative analysis process became somewhat complicated in the R environment because many packages have the same function name but had unique input and output criteria that became a challenge. This is a natural outcome of an open source environment that enables many contributions from multiple sources without having an overall design road map or process to ensure uniform requirements for submission.

Specifically the main problem was the inconsistent syntax for the algorithm inputs. Figure 2.9 displays several R library versions of the predict function, each of which expects different input types and variables. One of the ways to get around the problem is to read in the required library into Rstudio right before the specific predict function is needed which enables the developer to know the version that is currently in the R environment,



obj Class	Package	predict Function Syntax
lda	MASS	predict(obj) (no options needed)
glm	stats	predict(obj, type = "response")
gbm	gbm	predict(obj, type = "response", n.trees)
mda	mda	predict(obj, type = "posterior")
rpart	rpart	predict(obj, type = "prob")
Weka	RWeka	predict(obj, type = "probability")
LogitBoost	caTools	predict(obj, type = "raw", nIter)

Figure 2.8. Classification algorithm function syntax comparison

however with multiple predict algorithms being read into memory for different functions requires an ongoing road map of which function is being used at any given time. The problem was solved at the Pfizer Global Research and Development Nonclinical Statistics Department in 2005 while addressing problems their modeling teams encountered during the development process. The package they created is called 'caret' (Classification And REgression Training), an R package maintained by Max Kuhn. It has a unified interface for predictive models, streamlines tuning and increases efficiency with options to use parallel or distributed processing (Kuhn, 2014). The Applied Predictive Modeling textbook was written to explain how to use the library (Kuhn, Johnson, 2013). The first caret package was published in the Cran R repository in October 2007.

The solution was to write a wrapper around algorithms. The wrapper design created conformity of input types and parameters. Tuning and optimization parameters were added to support improvements. The architecture was written to accept the original library algorithm names which helped users have confidence in the new wrapper functions. The caret package supports a long list of well-known library algorithms including all the model types selected for this study. Therefore all models were built using the caret wrapper functions. By using a single source library it enabled greater comparability of model results since each model was treated in a similar fashion.

CHAPTER 3

Evaluation Criteria and Metrics Determining Goodness of Fit

Developing the testing strategy is directly tied to the data structure. If the goal of predictive analytics is to find the best and most accurate model, predefining and understanding the metrics that measure these qualities is an effective way to determine the goodness of fit of each model selected for testing. The development strategy should identify issues that may need an alternative testing path.

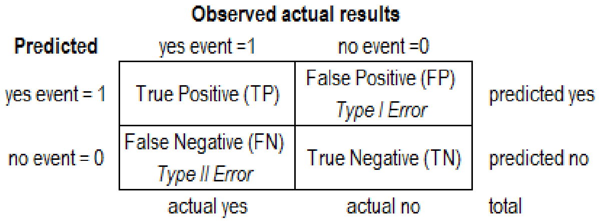
Performance evaluates the model goodness of fit to the structure and patterns in the data that generalizes to a new data source. The fundamental question is which are the best metrics to measure the performance of a binary classifier? The most commonly used metrics are Accuracy and AUCROC. When comparing models for selection, which of these metrics is better? Each is constructed from a different premise measuring distinct aspects of the same details (Shahram, 2016). The primary determinant in deciding whether to choose a metric is based on the business purpose, the structure of the data, and the balance of the classes. AUCROC is not as useful with imbalanced classes. There is no absolute ranking of the usefulness of metrics as much as a need to understand their composition and benefits based on the structure of the data. No single metric can help select a model (Brownlee, 2014). Another key issue to address is imbalance in the classifier mainly because the test strategy takes a dissimilar path depending on the degree of imbalance. The choice of metrics is also dependent on this knowledge.

3.1. Which metrics to use?

There are threshold and non-threshold based metrics represented by classification and the Receiver Operating Curve (ROC) respectively. The initial response observations and the results of the modeling and prediction functions build a classification table of categories after a threshold has been applied to the prediction results using a single optimal cutoff threshold. The Receiver Operating Curve uses the entire threshold range to build the ROC curve (Shadram, 2016) which is literally represented by as many classification tables as there are threshold values. Many factors can influence classification results including the available data, data variables, data cleansing, and model algorithm selected (Jeatrakul et al., 2001).

3.2. Classification Matrix

The classification matrix includes the actual observations in the historical data represented by whether a customer repays obligations translating into a credit rating paired with a prediction whether the event may occur. A credit risk binary classifier labels the non-event of a good credit risk as 0 or in the case of default, the event occurred, the credit risk response variable gets a bad rating value of 1.

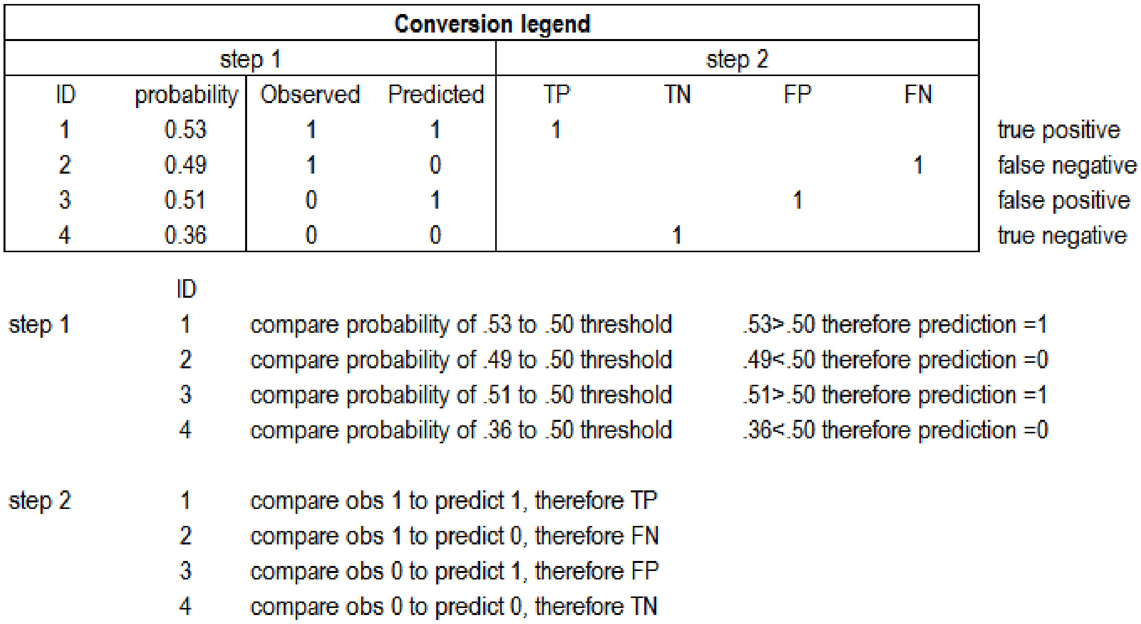


	Observed actual results		
Predicted	yes event =1	no event =0	
yes event = 1	True Positive (TP)	False Positive (FP) *Type I Error*	predicted yes
no event = 0	False Negative (FN) *Type II Error*	True Negative (TN)	predicted no
	actual yes	actual no	total

Figure 3.1. Classification Matrix (aka confusion matrix)

Predictions are created when the model is trained and finds patterns in the dataset which are indicators of good or bad ratings; see figure 3.1. The prediction is run on new test data with the algorithm returning results as the probabilities of creditworthy ratings. If the model patterns generalize well for the new data then the information is useful for analysis.

In step 1, the probability results from the predict function are compared to the hurdle rate which is 50 percent in this case. If the probability > .50, the prediction is considered a positive prediction and given a value of 1. If less than the hurdle rate, its value is 0 or a negative prediction. In step 2, the observed value and newly created predict value are compared. If the observed is 1 and it's prediction is 1 then the category it is placed in is the true positive or TP class.



Conversion legend

step 1				step 2				
ID	probability	Observed	Predicted	TP	TN	FP	FN	
1	0.53	1	1	1				true positive
2	0.49	1	0				1	false negative
3	0.51	0	1			1		false positive
4	0.36	0	0		1			true negative

	ID		
step 1	1	compare probability of .53 to .50 threshold	.53>.50 therefore prediction =1
	2	compare probability of .49 to .50 threshold	.49<.50 therefore prediction =0
	3	compare probability of .51 to .50 threshold	.51>.50 therefore prediction =1
	4	compare probability of .36 to .50 threshold	.36<.50 therefore prediction =0
step 2	1	compare obs 1 to predict 1, therefore TP	
	2	compare obs 1 to predict 0, therefore FN	
	3	compare obs 0 to predict 1, therefore FP	
	4	compare obs 0 to predict 0, therefore TN	

Figure 3.2. Conversion legend for classification matrix decisions

Correct ratings:

TP Customer defaulted, correctly predicted default; actual 1, pred 1

TN Customer didn't default, correctly predicted no default; actual 0, pred 0

Incorrect ratings:

FP Customer didn't default but incorrectly predicted to default; actual 0, pred 1

FN Customer defaulted but was incorrectly predicted no default; actual 1, pred 0

Type I Error - (FP/(FP+TN) or FP/actual no)

Medicine - Patient doesn't have cancer but is incorrectly predicted to have cancer. Unnecessary operation may occur without a second opinion. Real harm.

Credit - Customer didn't default but incorrectly predicted to default. Loan was not given with potential loss of interest revenue.

Fraud - Fraud did not occur but was incorrectly predicted to occur. If not properly handled, there is a legal risk of reputation damage to the organization and to the accused if treated as an actual fraud.

Type II Error - (FN/(FN+TP) or FN/actual yes)

Medicine - Patient has cancer but the test incorrectly predicts no cancer. Without a second opinion, a patient may not get needed treatment. Real harm.

Credit - Customer defaults but was incorrectly predicted not to default. Loan was given with the potential loss of the principal and interest revenue.

Fraud - Fraud did occur but the test incorrectly predicts no fraud. Fraud undetected.

Accuracy measures the diagonal of TP (observed event occurred and was predicted correctly) and TN (observed event did not occur and was predicted correctly). The false positive and false negatives contribute to the misclassification rate. When real scenarios are applied to the misclassification rate, it becomes apparent the damage that can occur by not reducing these rates as much as possible. The severity of the risk is very specific to the application.

The goal for selecting the best machine learning algorithm is not just to maximize classification but also to minimizing misclassification that could cause harm, damage or loss. To achieve this level of refinement requires increases in optimization and improvements in tuning of models. A key factor in model selection and interpretation is the tolerance level for misclassification. The field of medicine has a very low tolerance since Type I and II errors both have possible human toll. The credit industry is harmed in Type I errors but Type II errors are categorized as a missed opportunity. Marketing may have a higher tolerance since business decisions regarding cutoff values often are based on the marketing budget which may limit project scope rather than the need to maximize the classification metrics. The cost due to fraud focuses on both Type I and II errors while mitigating damage by establishing a thorough review process for all FP classifications, as well as having a policy that understands that predictive analytics does not identify fraud rather finds patterns in the data that may confirm indicators for further investigation. Ratios created from the table build metrics which explain the details of the model predictions.

Accuracy ((TP+TN)/Total) measures the percentage of positive predictions correctly classified. Issues arise with imbalanced classes where the frequency of the minority class is not equally represented, as well as the fact that neither the false positives nor false negatives are included in the ratio. It measures a minimum requirement. It should not be used with imbalanced classes. In this case, using the Kappa statistic is more relevant (Shahram, 2016), as well as precision and recall which are more discriminating. The Accuracy Paradox (Descoins, 2013) points out that the ratio can be misleading if $\text{TN} < \text{FN}$ and $\text{TP} < \text{FP}$.

Misclassification ((FP+FN)/Total) is the percentage of negative predictions incorrectly classified.

Sensitivity, Recall, TPR (TP/(TP+FN) or TP/actual yes) calculates the proportion of positives correctly predicted. The preference is to have a low value. The TPR, true positive rate, is used as the y axis of the ROC curve. It does not include TN's. When TN's are not meaningful, use precision and recall. Use it when the TP benefit is high. If low then FN are high.

Specificity, TNR (TN/(FP+TN) or TN/actual no) calculates the proportion of negatives correctly predicted. The preference is to have a high value. Without TP and FN, the ratio is less useful for imbalanced classes.

FPR (FP/(FP+TN) or FP/(actual no) aka (1 - Specificity)) is the false positive rate or the positive predictions incorrectly classified. FPR is used in the x axis of the ROC curve.

Specificity = TN/(FP + TN) = 1 - FPR

Specificity + FPR = 1 + (- FPR + FPR) (FPR cancels out)

Specificity + FPR = 1

Specificity + FPR - Specificity (specificity cancels out) = 1 - Specificity

FPR = 1 - Specificity

FNR (FN/(TP+FN)) or FN/(actual yes) is the false negative rate or the negative predictions incorrectly classified.

Precision, PPV (TP/ (FP+TP) or TP/(predicted yes)) measures the percentage of positive predictions over the total of correctly predicted or when correct how often. Use when the cost of a FP is high. If low then FP are high. It is a measure of exactness. It does not include TN's. When TN's are not meaningful, use precision and recall.

Troubleshooting - use TPR, TNR, FPR and FNR that independently test positive and negative classes (López el al, 2013); see Appendix Summary of classification metrics for details.

If avoiding FP, find the result of false negatives (FN) while not using metrics with false negatives.

If avoiding FN, find the result of false positives (FP) while not using metrics with false positives.

The end goal of using metrics is to thoroughly interpret and improve classification while reducing misclassification rates. Optimization enhancements can be made by changing the variables in feature selection, optimizing model tuning to select the best model, improving the quality and increasing the volume of data, as well as changing the cutoff thresholds.

3.3. Receiver Operating Curve and AUCROC

Receiver operating curve (ROC) uses TPR (Sensitivity) and FPR (1-Specificity) metrics as the y and x axes of the ROC curve. The area under the curve (AUCROC) quantifies the ROC curve performance and summaries Accuracy for the entire test result outcomes which represent the threshold range (Fawcett, 2005).

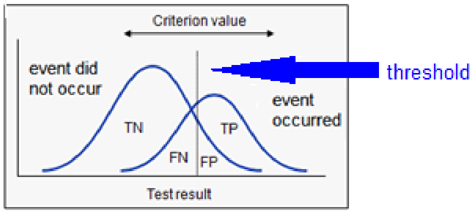
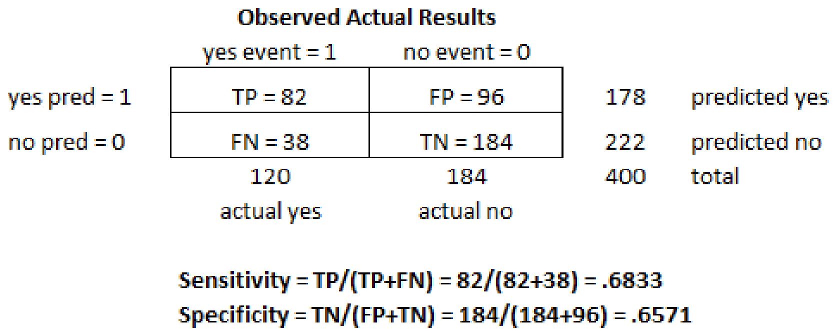


Figure 3.3. Visualizing ordered obs. and predicted probabilities (medcalc)

The area under the curve quantifies all the changes that occur along the threshold range. It is normally shown as a smooth curve but which actually represents each point based on the classification table range of possibilities; see figure 3.3. Figure 3.4 displays an example where 82 applicants were predicted to default.



Observed Actual Results

	yes event = 1	no event = 0		
yes pred = 1	TP = 82	FP = 96	178	predicted yes
no pred = 0	FN = 38	TN = 184	222	predicted no
	120	184	400	total
	actual yes	actual no		

Sensitivity = TP/(TP+FN) = 82/(82+38) = .6833

Specificity = TN/(FP+TN) = 184/(184+96) = .6571

Figure 3.4. Classification matrix with 82 defaulters in test set of 400 records

What is the point on the ROC curve? Given a Sensitivity (TPR) of .68 and a FPR value of .34, the point on the curve is (FPR=.34, TPR=.68). The optimal cutoff is where Sensitivity is relatively equal to Specificity. The diagonal line from (0,0) to (0,1) is the 50 percent threshold representing random selection. Every point on the ROC curve is plotted based on a FDR (1-Specificity) and TPR (Sensitivity) pair; (x=FDR, y=TPR); see figure 3.5. The optimal cutoff threshold is where Sensitivity and Specificity are relatively equal.

The following gauges the Area under the curve, the integral of x and y (Cornell, 2003):

AUCROC >= 0.9 outstanding discrimination or over fit

0.8 <= AUCROC < 0.9 excellent discrimination

0.7 <= AUCROC < 0.8 acceptable discrimination

0.5 = AUCROC, no discriminatory power or equal to a random case

AUCROC < 0.5 is worse than random case

The AUCROC is not considered one of the most discriminating metrics (Wu, Lee, 2014). Medical researchers have noted that the AUCROC is insensitive to the introduction of strong markers in diagnostic testing.

Several other metrics have been proposed including the Gini index. Researchers in medicine and science use the classification matrix and ROC curve in their work long

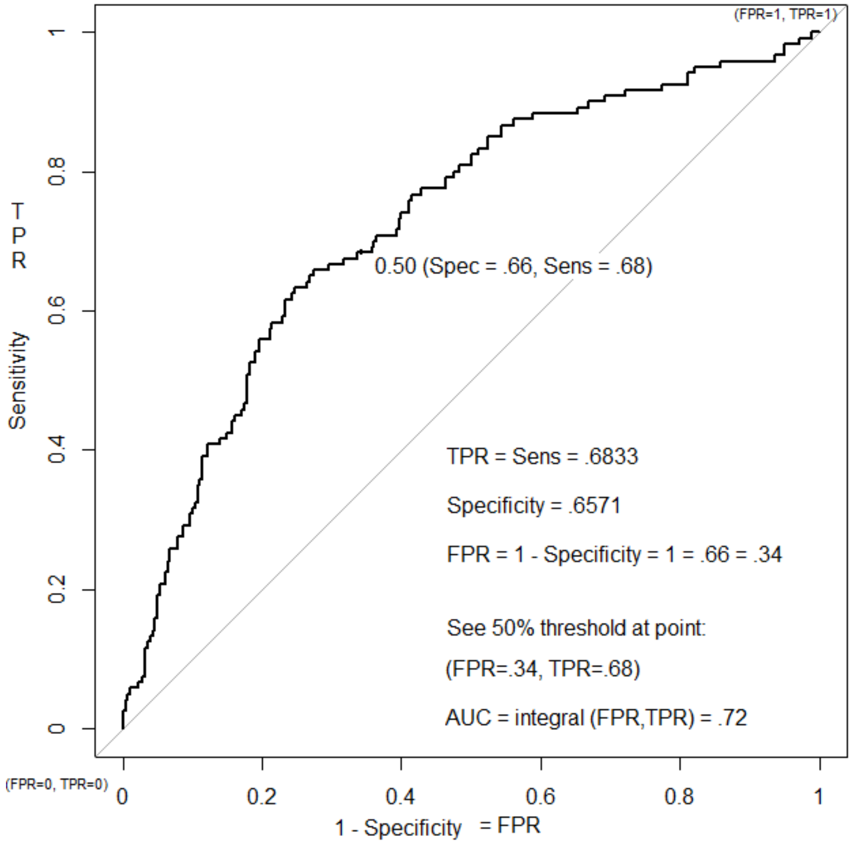


Figure 3.5. ROC curve example for point (FPR,TPR)

before predictive analytics adopted their measurement standards. Medicine and science are the most sensitive to the influence of Type I and II errors while having the greatest need to reduce any harm that may occur based on their diagnostic results. Using their cutting edge research can help provide more discriminating metrics into the field of data science.

3.4. Testing Summary

The test results will be reviewed for AUCROC, Accuracy and Type I and II errors:

AUCROC) TPR (TP/(TP+FN) or TP/(actual yes)) and FPR ((FP/(FP+TN) or FP/(actual no) aka (1-Specificity)) is the integral, area under the curve, of the entire range of thresholds and is measured by both misclassification and accuracy metrics.

Accuracy ((TP+TN)/total) is based on a single threshold initial decision to maximize the classification table and uses no misclassification metrics.

Type I Error - (FP/(FP+TN) or FP/actual no)

Type II Error - (FN/(FN+TP) or FN/(actual yes)

By focusing on the misclassification metrics, FP and FN, it is possible to identify where improvements need to be made. This example demonstrates that what may have appeared to be a harmless high FP result has a false discovery misclassification rate of 54 percent; also see figure 3.6 for examples of misclassification metrics.

Defaulters TP (correctly predicted to default) = 82

FP (incorrectly predicted to default)= 96

Total predicted yes (TP+FP) = 178

false discovery rate = FP/ predicted yes = 96/178 = 54 percent

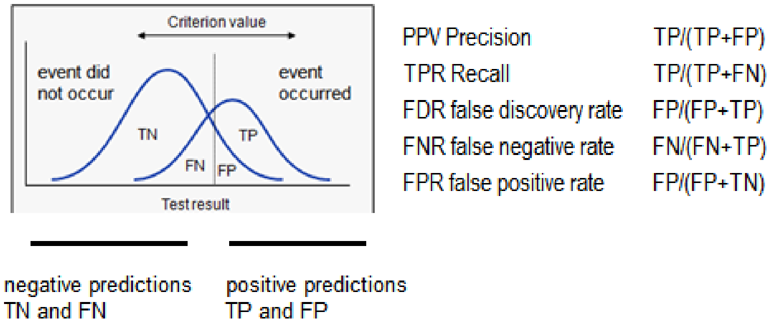

Criterion value
event did not occur
event occurred
TN
TP
FN
FP
Test result
PPV Precision TP/(TP+FP)
TPR Recall TP/(TP+FN)
FDR false discovery rate FP/(FP+TP)
FNR false negative rate FN/(FN+TP)
FPR false positive rate FP/(FP+TN)
negative predictions TN and FN
positive predictions TP and FP

Figure 3.6. Misclassification metrics

3.5. Class Imbalance

The definition of a class imbalance has a large and disproportional difference in the frequency of classifiers which often occurs in fraud risk scoring projects due to the lower incidence of fraud events to the general population of the dataset (Akosa, 2017).

Prior probabilities describe inherent probabilities prior to testing with the assumption that when dividing into two classes a relatively equal number of entries will be distributed into each class as the number of entries grows larger with the random probability of 50/50 for each occurrence. An unequal distribution with one class having a higher proportion of the distribution causes metrics to be misrepresentative (Monard & Batista, 2003).

Imbalanced classes are biased in the direction of the majority class which increases misclassification rates (López el al, 2013). Solutions to address class imbalance include data sampling, algorithm modification and cost sensitive learning. AUCPR, integral of recall and precision, is promoted as the best single classifier metric to measure imbalance. When measuring the minority class (event of default) in the focus of study, there is a fundamental bias toward the majority class which reduces the effectiveness of classification metrics which are then less reliable. Weighting or modifying sampling methods help to correct class imbalance (Akosa, 2017). Majority voting weighting solutions include cost sensitive training or learning what weights misclassification higher for the purpose of minimizing cost. Other options are down sampling or oversampling. Under sampling occurs when the majority class samples are reduced artificially. Over sampling seeks to correct class imbalance across by duplicating the minority class data.

At what point does a class imbalance require a change in metrics? According to Tom Fawcett (Fawcett, 2016), adjustments are needed if the range of data imbalance is less than 10-20 percent which is not the case for either the Australian or German datasets. Since class imbalance is a major factor in fraud cases, the topic has been included to provide a background for future development.

CHAPTER 4

Results

The test process was conducted in a series of trials in order to find the affects of the metrics (Accuracy, AUCROC, Type I and II errors) in ranking models for selection.

Test 1: examines the metrics and how they were applied to 9 models

Test 2: searches for correlations between the 6 base learners for the purpose of finding the most and least correlated pairs to create strong and weaker pairs respectively from which 16 stacked models will be developed.

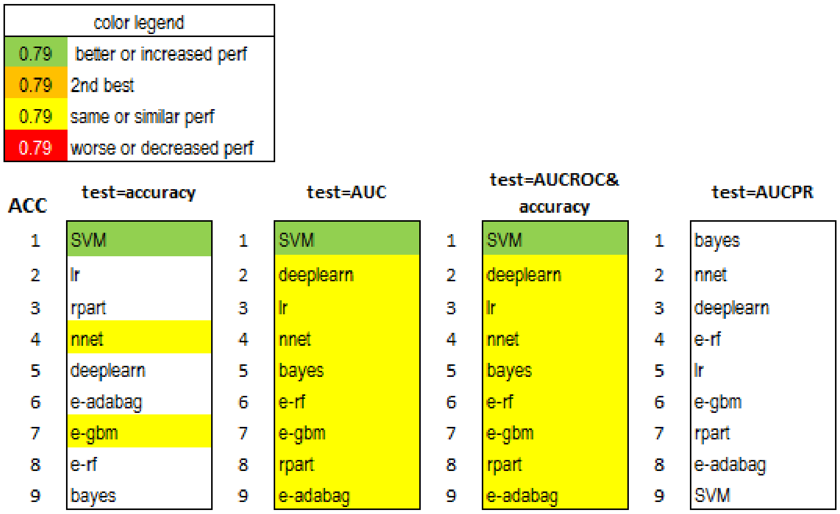
Test 3: displays the magnitude difference in metrics for each model in graphical form.

Test 4: ranks all 25 models by AUCROC, Accuracy, and Type I and II errors for model selection.

The result of both the German and Australian datasets would be examined for how their test results were similar or dissimilar. What are the generalizations that can be made and what are the factors to consider in model selection?

4.1. Result and Summary tables

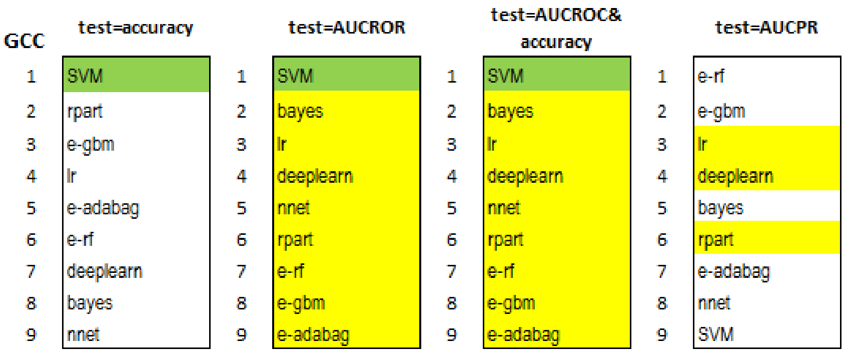
Test 1: Comparing the ranking of models while testing for significant metrics develops a starting hypothesis for metric application.



color legend	
🟩 0.79	better or increased perf
🟧 0.79	2nd best
🟨 0.79	same or similar perf
🟥 0.79	worse or decreased perf

ACC	test=accuracy		test=AUC		test=AUCROC& accuracy		test=AUCPR
1	🟩 SVM	1	🟩 SVM	1	🟩 SVM	1	bayes
2	lr	2	🟨 deeplearn	2	🟨 deeplearn	2	nnet
3	rpart	3	🟨 lr	3	🟨 lr	3	deeplearn
4	🟨 nnet	4	🟨 nnet	4	🟨 nnet	4	e-rf
5	deeplearn	5	🟨 bayes	5	🟨 bayes	5	lr
6	e-adabag	6	🟨 e-rf	6	🟨 e-rf	6	e-gbm
7	🟨 e-gbm	7	🟨 e-gbm	7	🟨 e-gbm	7	rpart
8	e-rf	8	🟨 rpart	8	🟨 rpart	8	e-adabag
9	bayes	9	🟨 e-adabag	9	🟨 e-adabag	9	SVM

Figure 4.1. Australian dataset - initial ranking by metrics



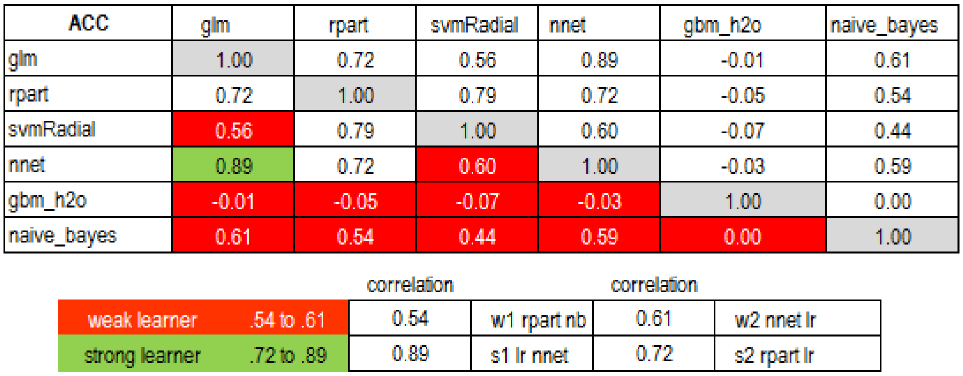
GCC	test=accuracy		test=AUCROR		test=AUCROC& accuracy		test=AUCPR
1	🟩 SVM	1	🟩 SVM	1	🟩 SVM	1	e-rf
2	rpart	2	🟨 bayes	2	🟨 bayes	2	e-gbm
3	e-gbm	3	🟨 lr	3	🟨 lr	3	🟨 lr
4	lr	4	🟨 deeplearn	4	🟨 deeplearn	4	🟨 deeplearn
5	e-adabag	5	🟨 nnet	5	🟨 nnet	5	bayes
6	e-rf	6	🟨 rpart	6	🟨 rpart	6	🟨 rpart
7	deeplearn	7	🟨 e-rf	7	🟨 e-rf	7	e-adabag
8	bayes	8	🟨 e-gbm	8	🟨 e-gbm	8	nnet
9	nnet	9	🟨 e-adabag	9	🟨 e-adabag	9	SVM

Figure 4.2. German dataset - initial ranking by metrics

The results revealed that sorting by AUCROC and Accuracy didn't change model ranking, however Accuracy or AUCPR alone listed the models differently; see figures 4.1

and 4.2. The metrics will be reviewed first AUCROC then Accuracy then appraise the Type I and II errors.

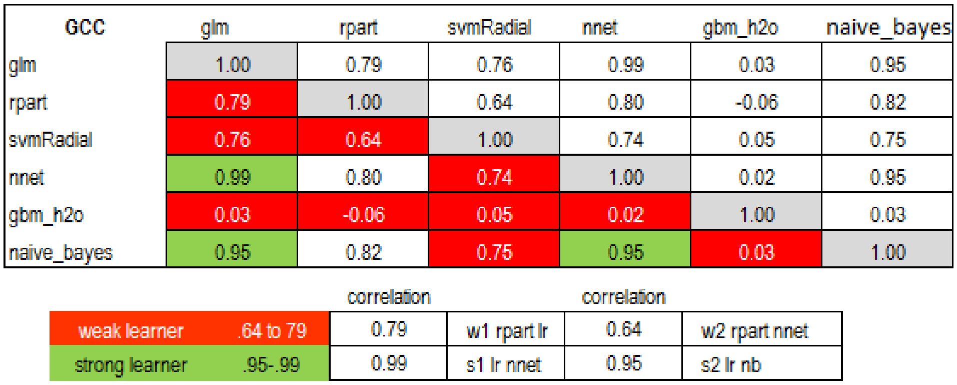
Test 2: The 6 base learner model were compared against each other for correlations; see figures 4.3 and 4.4.



ACC	glm	rpart	svmRadial	nnet	gbm_h2o	naive_bayes
glm	1.00	0.72	0.56	0.89	-0.01	0.61
rpart	0.72	1.00	0.79	0.72	-0.05	0.54
svmRadial	0.56	0.79	1.00	0.60	-0.07	0.44
nnet	0.89	0.72	0.60	1.00	-0.03	0.59
gbm_h2o	-0.01	-0.05	-0.07	-0.03	1.00	0.00
naive_bayes	0.61	0.54	0.44	0.59	0.00	1.00

		correlation		correlation	
weak learner	.54 to .61	0.54	w1 rpart nb	0.61	w2 nnet lr
strong learner	.72 to .89	0.89	s1 lr nnet	0.72	s2 rpart lr

Figure 4.3. ACC Correlations between 6 base learners



GCC	glm	rpart	svmRadial	nnet	gbm_h2o	naive_bayes
glm	1.00	0.79	0.76	0.99	0.03	0.95
rpart	0.79	1.00	0.64	0.80	-0.06	0.82
svmRadial	0.76	0.64	1.00	0.74	0.05	0.75
nnet	0.99	0.80	0.74	1.00	0.02	0.95
gbm_h2o	0.03	-0.06	0.05	0.02	1.00	0.03
naive_bayes	0.95	0.82	0.75	0.95	0.03	1.00

		correlation		correlation	
weak learner	.64 to 79	0.79	w1 rpart lr	0.64	w2 rpart nnet
strong learner	.95-.99	0.99	s1 lr nnet	0.95	s2 lr nb

Figure 4.4. GCC Correlations between 6 base learners

The strong learner1 and weak learner2 for both the Australian and German Credit datasets were the same.

Test 3: Metrics were graphed for all models displaying the magnitude of difference for each metric separately. This barplot effectively shows the changes in metrics in the ranked list of models and specifically for the Type I and II errors; see figures 4.5 and 4.6.

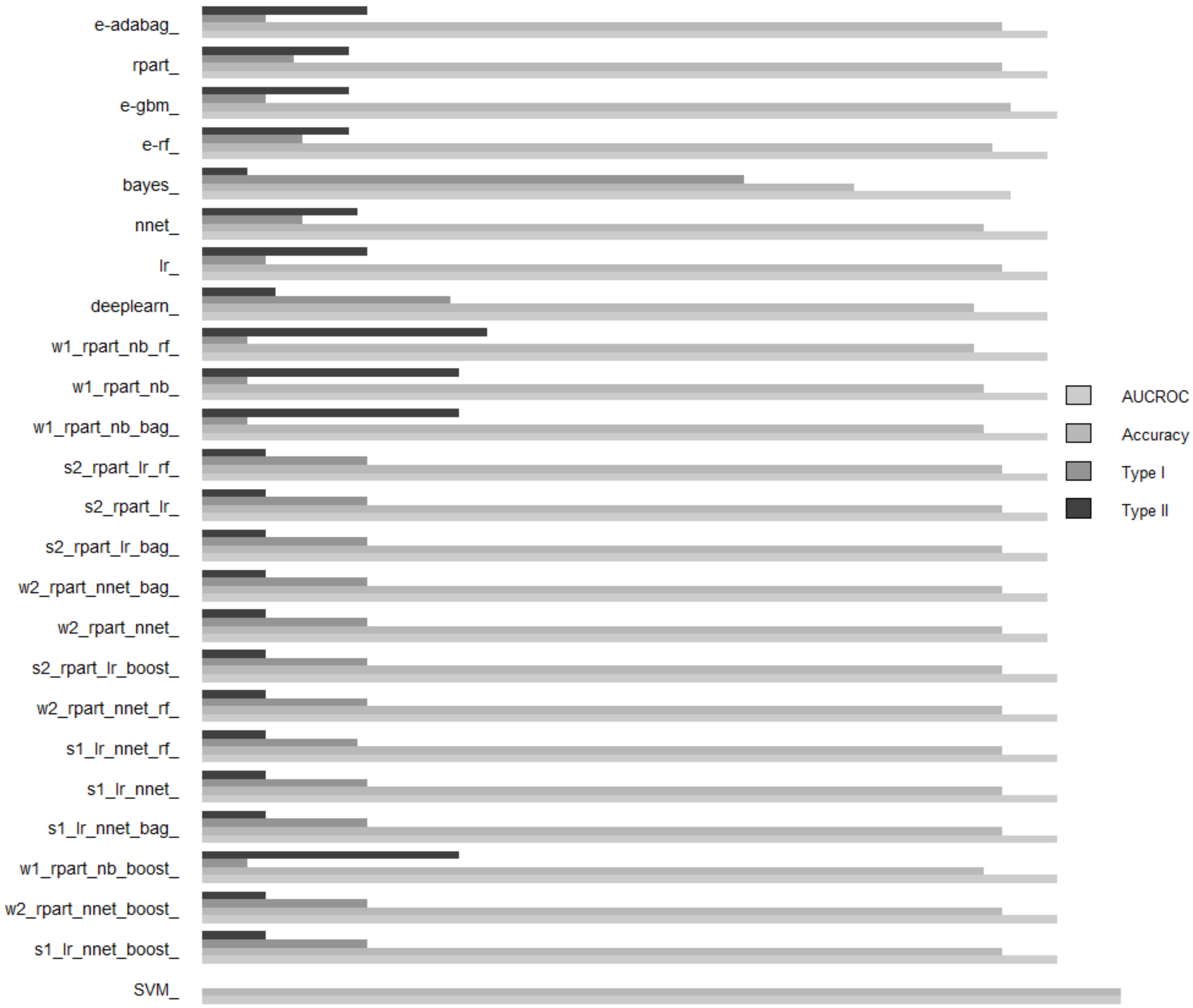


Figure 4.5. ACC magnitude of classification vs misclassification metrics

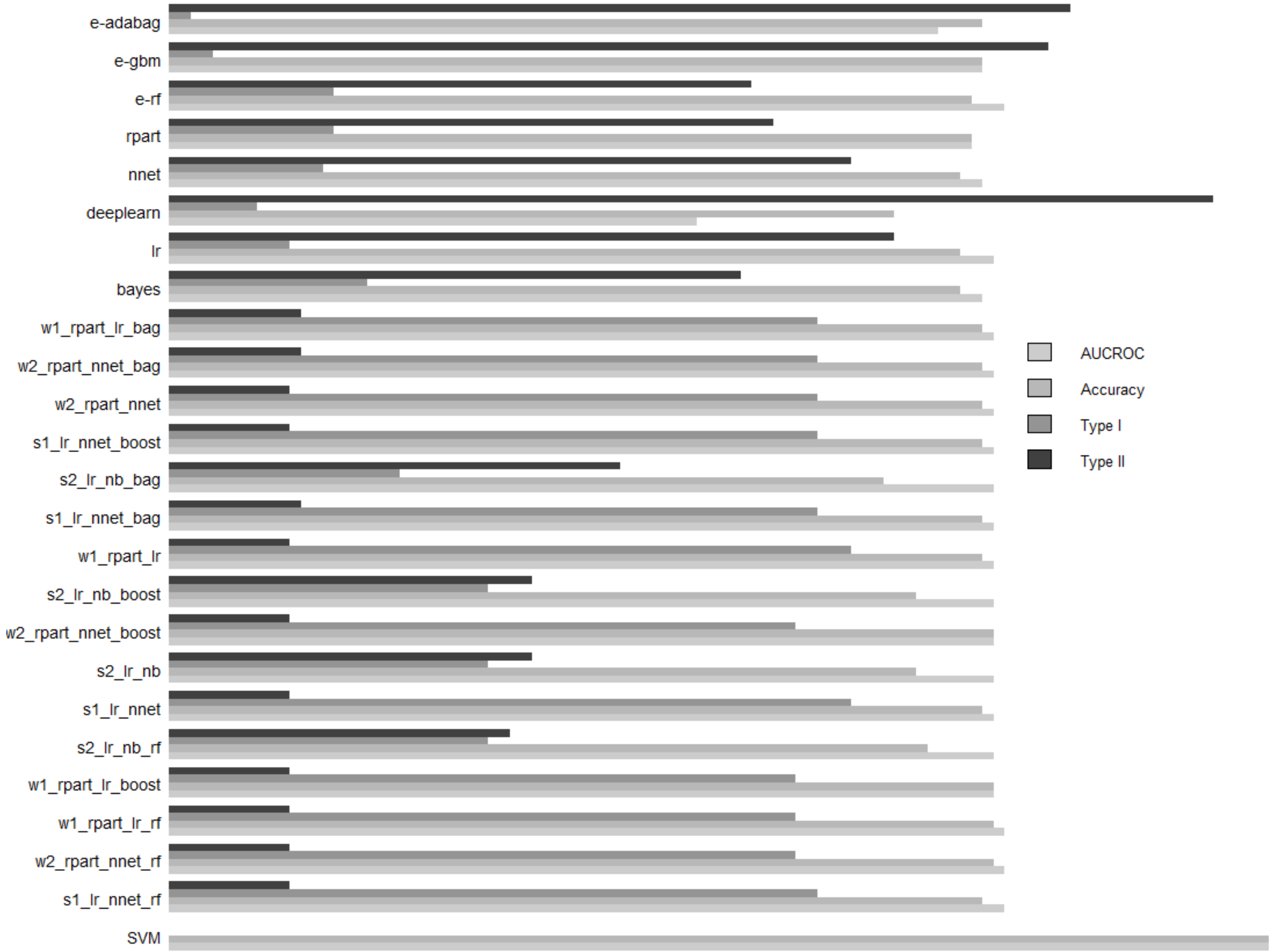
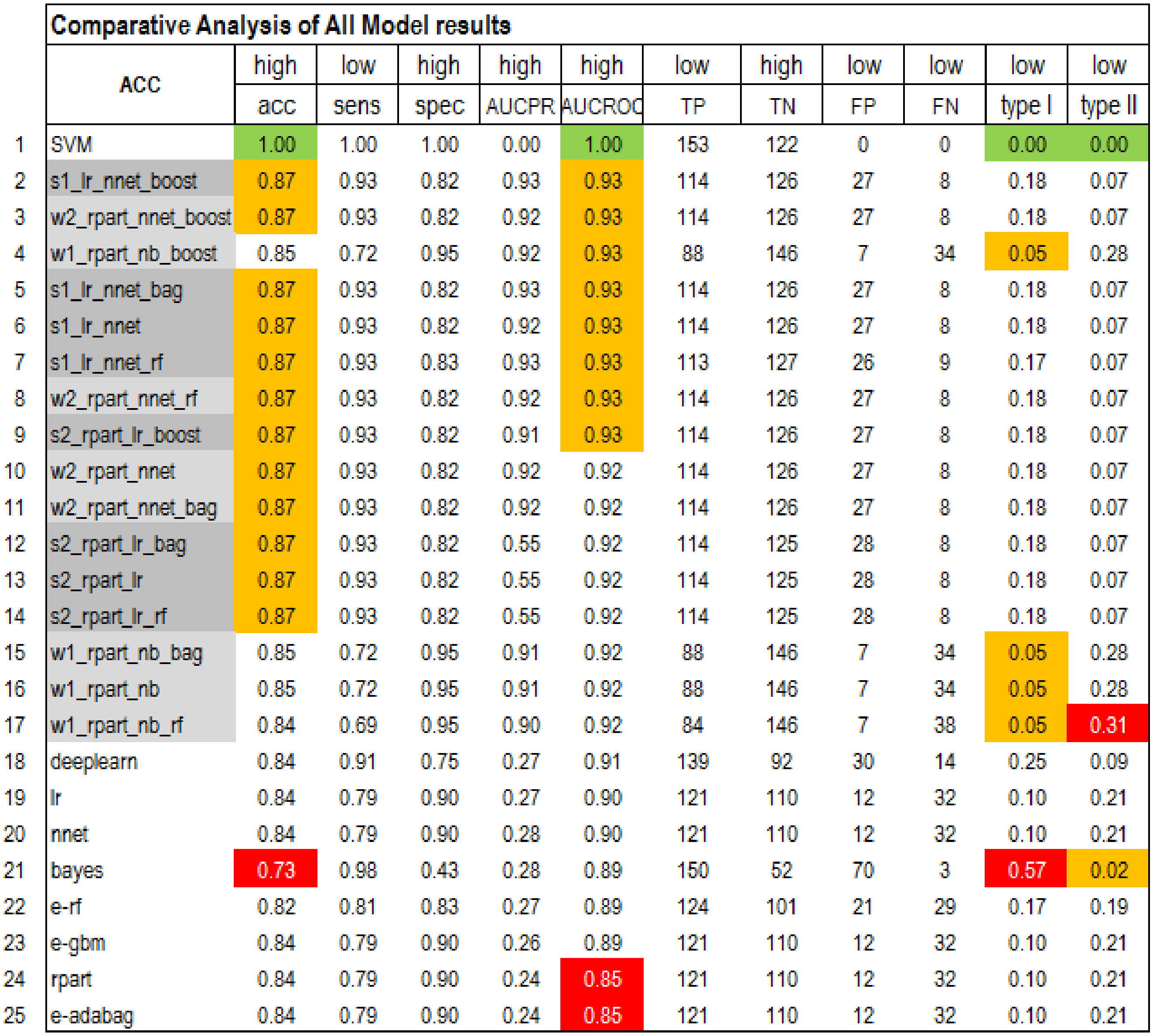


Figure 4.6. GCC magnitude of classification vs misclassification metrics

The ACC models were more effective based on the reduced number of Type I and II errors.

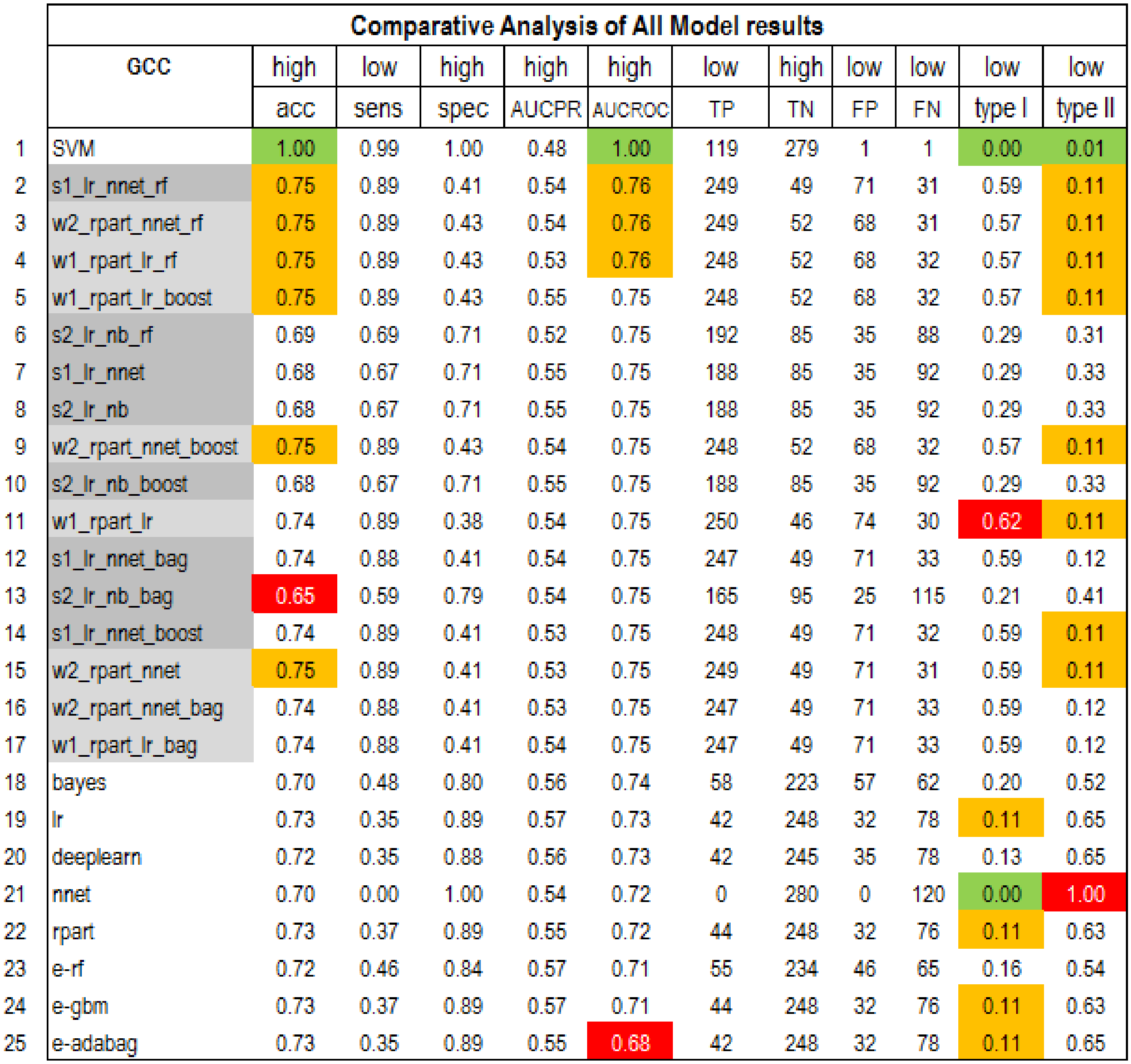
Test 4: The final test ordered all 25 models by AUCROC, Accuracy, and Type I and II errors for model selection; see figures 4.7 and 4.8. The Support Vector Machine had the highest AUCROC and Accuracy metrics with the lowest Type I and II errors. Each dataset had different results for the rest of the 24 of the 25 models except that the 16 stacked models were in the top 17 models for both data sources.



Comparative Analysis of All Model results

	ACC	high	low	high	high	high	low	high	low	low	low	low
		acc	sens	spec	AUCPR	AUCROC	TP	TN	FP	FN	type I	type II
1	SVM	1.00	1.00	1.00	0.00	1.00	153	122	0	0	0.00	0.00
2	s1_lr_nnet_boost	0.87	0.93	0.82	0.93	0.93	114	126	27	8	0.18	0.07
3	w2_rpart_nnet_boost	0.87	0.93	0.82	0.92	0.93	114	126	27	8	0.18	0.07
4	w1_rpart_nb_boost	0.85	0.72	0.95	0.92	0.93	88	146	7	34	0.05	0.28
5	s1_lr_nnet_bag	0.87	0.93	0.82	0.93	0.93	114	126	27	8	0.18	0.07
6	s1_lr_nnet	0.87	0.93	0.82	0.92	0.93	114	126	27	8	0.18	0.07
7	s1_lr_nnet_rf	0.87	0.93	0.83	0.93	0.93	113	127	26	9	0.17	0.07
8	w2_rpart_nnet_rf	0.87	0.93	0.82	0.92	0.93	114	126	27	8	0.18	0.07
9	s2_rpart_lr_boost	0.87	0.93	0.82	0.91	0.93	114	126	27	8	0.18	0.07
10	w2_rpart_nnet	0.87	0.93	0.82	0.92	0.92	114	126	27	8	0.18	0.07
11	w2_rpart_nnet_bag	0.87	0.93	0.82	0.92	0.92	114	126	27	8	0.18	0.07
12	s2_rpart_lr_bag	0.87	0.93	0.82	0.55	0.92	114	125	28	8	0.18	0.07
13	s2_rpart_lr	0.87	0.93	0.82	0.55	0.92	114	125	28	8	0.18	0.07
14	s2_rpart_lr_rf	0.87	0.93	0.82	0.55	0.92	114	125	28	8	0.18	0.07
15	w1_rpart_nb_bag	0.85	0.72	0.95	0.91	0.92	88	146	7	34	0.05	0.28
16	w1_rpart_nb	0.85	0.72	0.95	0.91	0.92	88	146	7	34	0.05	0.28
17	w1_rpart_nb_rf	0.84	0.69	0.95	0.90	0.92	84	146	7	38	0.05	0.31
18	deeplearn	0.84	0.91	0.75	0.27	0.91	139	92	30	14	0.25	0.09
19	lr	0.84	0.79	0.90	0.27	0.90	121	110	12	32	0.10	0.21
20	nnet	0.84	0.79	0.90	0.28	0.90	121	110	12	32	0.10	0.21
21	bayes	0.73	0.98	0.43	0.28	0.89	150	52	70	3	0.57	0.02
22	e-rf	0.82	0.81	0.83	0.27	0.89	124	101	21	29	0.17	0.19
23	e-gbm	0.84	0.79	0.90	0.26	0.89	121	110	12	32	0.10	0.21
24	rpart	0.84	0.79	0.90	0.24	0.85	121	110	12	32	0.10	0.21
25	e-adabag	0.84	0.79	0.90	0.24	0.85	121	110	12	32	0.10	0.21

Figure 4.7. ACC model performance comparison



Comparative Analysis of All Model results

	GCC	high	low	high	high	high	low	high	low	low	low	low
		acc	sens	spec	AUCPR	AUCROC	TP	TN	FP	FN	type I	type II
1	SVM	1.00	0.99	1.00	0.48	1.00	119	279	1	1	0.00	0.01
2	s1_lr_nnet_rf	0.75	0.89	0.41	0.54	0.76	249	49	71	31	0.59	0.11
3	w2_rpart_nnet_rf	0.75	0.89	0.43	0.54	0.76	249	52	68	31	0.57	0.11
4	w1_rpart_lr_rf	0.75	0.89	0.43	0.53	0.76	248	52	68	32	0.57	0.11
5	w1_rpart_lr_boost	0.75	0.89	0.43	0.55	0.75	248	52	68	32	0.57	0.11
6	s2_lr_nb_rf	0.69	0.69	0.71	0.52	0.75	192	85	35	88	0.29	0.31
7	s1_lr_nnet	0.68	0.67	0.71	0.55	0.75	188	85	35	92	0.29	0.33
8	s2_lr_nb	0.68	0.67	0.71	0.55	0.75	188	85	35	92	0.29	0.33
9	w2_rpart_nnet_boost	0.75	0.89	0.43	0.54	0.75	248	52	68	32	0.57	0.11
10	s2_lr_nb_boost	0.68	0.67	0.71	0.55	0.75	188	85	35	92	0.29	0.33
11	w1_rpart_lr	0.74	0.89	0.38	0.54	0.75	250	46	74	30	0.62	0.11
12	s1_lr_nnet_bag	0.74	0.88	0.41	0.54	0.75	247	49	71	33	0.59	0.12
13	s2_lr_nb_bag	0.65	0.59	0.79	0.54	0.75	165	95	25	115	0.21	0.41
14	s1_lr_nnet_boost	0.74	0.89	0.41	0.53	0.75	248	49	71	32	0.59	0.11
15	w2_rpart_nnet	0.75	0.89	0.41	0.53	0.75	249	49	71	31	0.59	0.11
16	w2_rpart_nnet_bag	0.74	0.88	0.41	0.53	0.75	247	49	71	33	0.59	0.12
17	w1_rpart_lr_bag	0.74	0.88	0.41	0.54	0.75	247	49	71	33	0.59	0.12
18	bayes	0.70	0.48	0.80	0.56	0.74	58	223	57	62	0.20	0.52
19	lr	0.73	0.35	0.89	0.57	0.73	42	248	32	78	0.11	0.65
20	deeplearn	0.72	0.35	0.88	0.56	0.73	42	245	35	78	0.13	0.65
21	nnet	0.70	0.00	1.00	0.54	0.72	0	280	0	120	0.00	1.00
22	rpart	0.73	0.37	0.89	0.55	0.72	44	248	32	76	0.11	0.63
23	e-rf	0.72	0.46	0.84	0.57	0.71	55	234	46	65	0.16	0.54
24	e-gbm	0.73	0.37	0.89	0.57	0.71	44	248	32	76	0.11	0.63
25	e-adabag	0.73	0.35	0.89	0.55	0.68	42	248	32	78	0.11	0.65

Figure 4.8. GCC model performance comparison

CHAPTER 5

Model Selection

The datasets performed quite differently except for the SVM being selected the top model in both. The question is why the performance was so inconsistent? Classification performance is based on how well the algorithms generalize the data structure. The two datasets differed in class balance, number of entries, and data structure which is reflected in the metrics. The Australian dataset which had 690 entries with a relatively equal class balance rated better consistently with reduced Type I and II errors over the German dataset of 1000 entries and a 30/70 percent class balance. The model performance is also determined by the choice of variables selected and values included in the dataset which are direct causal relationships to the response binary risk scoring classifier.

Logistic Regression was 19th of 25 models for both datasets. In the German dataset, randomForest stacked models were in 3 of the top 4 of best models with the Australian stacked models showing boosting in 3 of the top 4 of best models. The strong and weak learners were inconsistent in performance for both models other than being in 16 of the top 17 models for both the German and Australian datasets.

Once AUCROC ranking was sorted, any additional sorting for the Type I and II errors was not necessary for both datasets. A review of Precision and AUCPR did not prove to be helpful.

5.1. Next Steps

The initial study was based on creating a new fraud risk scoring application, however building a new dataset proved to be a daunting task because it would have to reflect a significant amount of application experience to build and calibrate the source data in a field that does not yet exist. As a result, the Australian and German credit risk scoring datasets were substituted. Critical issues to be researched in each dataset are class imbalance, optimal cutoff points, reducing misclassification rates, improving validation and resampling methods, using different languages and platforms that meet client requirements, managing parallel and/or distributed processing needed for the computationally intensive ensemble models with cloud environment options, and including the addition of other measurements such as Kappa statistics, Gini Index, and false discovery and detection error rates.

In a separate parallel study, the work completed on the business aspect of the new risk scoring application proved to be very useful since a creative way was found in extending the range of the binary risk classifier while adding significantly greater utility to the application itself. Even though there is less imbalance with a creative solution, risk scoring classifiers are by nature imbalanced and require building a toolkit and methodology enabling rebalancing.

Other ensemble methods may be useful in working with class imbalance (Galara et al., 2013) as a viable option to current methods. The EUSBoost algorithm, AdaBoost.M2, is a later version of Adaboost.M1 (RUSBoost) used in this study. The algorithm uses random undersampling for preprocessing prior to using EUSBoost.

CHAPTER 6

Conclusion

The main goal of this study was to optimize classification performance for risk scoring applications by thoroughly reviewing the elements of the metrics used to measure improvements, experimenting with ensemble stacking methods to reduce misclassification, and developing a test strategy to monitor where the metrics were changing in order to be able to select the best model.

The Support Vector Machine outperformed all other models, however each dataset ranked the rest of the 24 models very differently. Since the model generalizes the data structure, how the specific dataset characteristics can account for their performance is the question. Is it the class imbalance, the number of entries, the data structure itself, the variables selected, data preprocessing or model tuning?

The performance of the stacked models was similar for each data source ranking 16 of the stacked models in the top 17 best models but very inconsistent for specific model ranking.

A note about Logistic Regression since it is very likely the most common algorithm in the business market place. It did not fare well in the base model comparison as 19th out of 25 ranked models for both data sources. Logistic Regression coefficients are used in the risk scorecard development. The model's ranking will have a strong effect on the confidence level of the final scorecard.

There is another dimension to model selection which brings into the equation the volume of data and model selection order. With a greater volume of data, algorithms have an increased opportunity to find the real pattern in the data structure (Ng, 2013 and 2016). According to Ng, each new paper on algorithm improvement that succeeds the previous paper's performance may actually prove to be based on the increased volume of data used in model testing. This hypothesis does examine the essence of what is really being tested during research which is conducted in a development environment with engineered test datasets with sizes that are smaller than a normal deployment environment. The awareness that development and deployment dataset sizes and complexity actually are a factor in model performance increases the need to follow up research findings when recommendations are actually deployed. The feedback from deployment could provide invaluable information regarding the model selection process. Each production dataset is unique which does not provide information for generalization. The feedback would be very useful for the broader understanding of algorithm comparative analysis, the factors to be reviewed and considered, as well as their effects on the process.

The model results in this study showed that the performance ranking of model algorithms for finding the generalized pattern of the dataset should be based on empirical methods closely examining the changes in key metrics rather than on research assumptions for model selection.

The key implication in performance optimization is the creation of more efficient predictions that can then be prescribed back into an organization as insights. In fields such as fraud where imbalanced classes are inherently an issue to be addressed, finding effective tools to balance classes is currently a considerable modeling challenge. Other areas for future work include finding optimal threshold cutoffs and using more discriminating metrics (such as Kappa statistics, Gini Index, Precision and AUCPR, for example) that are required to maximize performance when the class balance is more skewed.

Bibliography

[1] Abdou, Hussein A. (2009), An evaluation of alternative scoring models in private banking, The Journal of Risk Finance; London 10.1 (2009): 38-53

[2] Abdou, Hussein A., Pointon, John (2011) CREDIT SCORING, STATISTICAL TECHNIQUES AND EVALUATION CRITERIA: A REVIEW OF THE LITERATURE, Article first published online: 22 JUN 2011 DOI: 10.1002/isaf.325

[3] Akosa, J. (2017). Predictive Accuracy: A Misleading Performance Measure for Highly Imbalanced Data, Oklahoma State University, Retrieved from http://support.sas.com/resources/papers/proceedings17/0942-2017.pdf

[4] Anderson, T. W. (2003) An Introduction to Multivariate Statistical Analysis, New York: Wiley Interscience 3rd Edition

[5] Avery, R. B., Bostic, R. W., Calem, P. S., Canner, G. B. (2000), Credit Scoring: Statistical Issues and Evidence from Credit Bureau Files, Real Estate Economics 2000 V28 3:pp. 523-547

[6] Baesans, B., van Gostel, T., Viaene, S., Stepanova, M., Suykens, J., Vanthienen, J. (2003). Benchmarking state of the art algorithms for credit scoring. Journal of the Operational Research Society, 54(5), pp. 627-635

[7] Boroujeni, M. S. (2016). Introduction to Machine Learning Methods and CARET package in R, Nov. 01 2016, Ecole Polytechnique Federal De Lausanne

[8] Boyes, W. J., Hoffman, D. L., Low, S. A. (1989) An Econometric Analysis of the Bank Credit Scoring Problem, Journal of Econometrics 40 (1989) 3-14, North-Holland,

[9] Brownlee, J. (2014), Classification Accuracy is Not Enough: More Performance Measures You Can Use, Brownlee, Jason : Author, Retrieved from http://machinelearningmastery.com/classification-accuracy-is-not-enough-more-performance-measures-you-can-use/

[10] Brownlee, J. (2015), Jump-Start Machine Learning in R Apply Machine Learning with R Now, Brownlee, Jason : Author, Retrieved from http://MachineLearningMastery.com

[11] Caruanna, R., Niculescu-Mizil, A., Crew, G., Ksikes, A. (2004) Ensemble Selection of Models, Department of Computer Science, Cornell University, Proceedings of the twenty-first international conference on Machine learning 1-58113-838-5, Retrieved from http://www.cs.cornell.edu/ caruana/ctp/ct.papers/caruana.icml04.icdm06long.pdf

[12] Cetinkaya-Rundell, Mine (2017) The best of Bayesian Statistics, A Coursera class from Duke University, Retrieved from https://www.coursera.org/learn/bayesian

[13] Cohen, J. (1960) A coefficient of agreement for nominal scales, Educational and Psychological Measurement 20, 37-46.

[14] Cornell University Lecture Notes (2003) Performance Measures for Machine Learning, Cornell University Computer Science class CS 578

[15] Crook, J. N. (1996) Credit scoring: An overview. Working paper series No. 96/13, British Association, Festival of Science. University of Birmingham, The University of Edinburgh.

[16] Crook, J. N., Edelman, D. B., Thomas, L. C. (2007) Recent developments in consumer credit risk assessment, European Journal of Operational Research 183 (2007) 1447-1465

[17] Davis, J., Goadrich, M. (2006), The relationship between precision recall and ROC curves, in: Proceedings of the 23th International Conference on Machine Learning (ICML2006), ACM, 2006, pp. 233-240

[18] Daumé III, H. (2012), A Course in Machine Learning, Daumé III, Hal: Author, Retrieved from http://ciml.info

[19] Descoins, A. (2013), Why accuracy alone is a bad measure for classification tasks, and what we can do about it, published by Tryo Labs website, Retrieved from https://tryolabs.com/blog/2013/03/25/why-accuracy-alone-bad-measure-classification-tasks-and-what-we-can-do-about-it/

[20] Demir, N. (2017) Ensemble Methods: Elegant Techniques to Produce Improved Machine Learning Results, Demir, Necati: Author, Published by KDNuggets website, Retrieved from https://www.toptal.com/machine-learning/ensemble-methods-machine-learning

[21] Dietterich, T.G. (2000) An Experimental Comparison of Three Methods for Constructing Ensembles of Decision Trees: Bagging, Boosting, and Randomization , 40: 139. doi:10.1023/A:1007607513941, Department of Computer Science, Oregon State University, Corvallis, OR 97331, USA, August 16, 1999

[22] Durand, David, (1941). Risk elements in consumer instalment Credit, National Bureau of Economic Research, Inc.

[23] Fawcett, T. (2005) An introduction to ROC analysis, Institute for the Study of Learning and Expertise, 2164 Staunton Court, Palo Alto, CA 94306, USA Available online 19 December 2005, http://people.inf.elte.hu/kiss/13dwhdm/roc.pdf

[24] Fawcet, T. (2016) Learning from Imbalanced Classes, Fawcett, Tom : Author, Retrieved from Silicon Valley Data Science website https://svds.com/learning-imbalanced-classes/

[25] Galara, M., FernÃąndez, A., Barrenecheaa, E. , Herrerac, F. (2013) EUSBoost: Enhancing ensembles for highly imbalanced data-sets by evolutionary undersampling, Science Direct Pattern Recognition Volume 46, Issue 12, December 2013, Pages 3460-3471

[26] Hall, Patrick, Dean, Jared, Kabul, Ilknur Kaynar, Silva, Jorge (2014) An Overview of Machine Learning with SASÂő Enterprise MinerâĎć, SAS Institute Inc. Paper SAS313-2014, Retrieved from https://support.sas.com/resources/papers/proceedings14/SAS313-2014.pdf

[27] Hand, J., Henley, W. E. (1997), Statistical Classification Methods in Consumer Credit Scoring: a Review, J. R. Statist. Soc., A (1997) 160, Part 3, pp. 523-541

[28] H2O.ai Team (2016) H2O website, URL http://h2o.ai

[29] Huval, B., Coates, A., Ng, A. (2013) Deep learning for class generic object detection, eprint arXiv:1312.6885 December 2013; Cornell University Library, Retrieved from https://arxiv.org/abs/1312.6885v1

[30] James, G , Witten, D., Hastie, T., Tibshirani, R. (2013), An Introduction to Statistical Learning with Applications in R, Springer Publications

[31] Jeatrakul, P., Wong, K.W. and Fung, C.C. (2010) Data cleaning for classification using misclassification analysis, Journal of Advanced Computational Intelligence and Intelligent Informatics, 14 (3). pp. 297-302.

[32] Jha, G. (2007) Artificial Neural Networks and Its Applications, Advances in Data Analytic Techniques, Indian Agricultural Statistics Research Institute (I.C.A.R.) Library Avenue, New Delhi-110012, 2007

[33] Keilwagen, J., Grosse, I., & Grau, J. (2014). Area under Precision-Recall Curves for Weighted and Unweighted Data. PLoS ONE, 9(3), e92209.

[34] Koh, H., Tan, W., Goh, C. (2006) A two-step method to construct credit scoring models with data mining techniques, Int. J. Bus. Inform., 1 (1) (2006), pp. 96-118

[35] Korting, T. S. (2006) C4.5 algorithm and Multivariate Decision Trees, Video file, Retrieved from https://www.youtube.com/watch?v=8-vHunc4k8s

[36] Kraus, A. (2014) Recent Methods from Statistics and Machine Learning for Credit Scoring, Dissertation an der FakultÂĺat f́ur Mathematik, Informatik und Statistik der Ludwig-Maximilians-Universitĺat Ḿunchen, March 2014

[37] Kuhn, M., Kjell J. (2013), Applied predictive modeling, Springer Publishing; Retrieved from http://www.springer.com/us/book/9781461468486

[38] Kuhn, M. (2014) Caret Package Webinar, by the Orange County R User Group, Feb 27, 2014, Retrieved from https://www.youtube.com/watch?v=7Jbb2ItbTC4

[39] Kuhn, M. (2015). Building Predictive Models in R Using the caret Package. Journal of Statistical Software, Vol. 28, No. 5., pp. 1-26.

[40] Lavrenko, V. (2014), Decision Trees (1-6), Full lecture: http://bit.ly/D-Tree , Video file, Retrieved from https://www.youtube.com/watch?v=eKD5gxPPeY0

[41] LeDell, E. (2015) Intro to Practical Ensemble Learning, Group in Biostatistics University of California, Berkeley April 27, 2015

[42] Lee, T. H., & Jung, S. (1999). FORECASTING CREDITWORTHINESS: LOGISTIC VS. ARTIFICIAL NEURAL NET. Journal Of Business Forecasting Methods & Systems, 18(4), 28.

[43] Lo, H.-Y., Chang, C.-M., Chiang, T.-H., Hsiao, C.-Y., Huang, A., Kuo, T.-T., Lai, W.-C., Yang, M.-H., Yeh, J.-J., Yen, C.-C., Lin, S.-D., (2008), Learning to improve area-under-FROC for imbalanced medical data classification using an ensemble method, SIGKDD Explorations 10 (2) (2008) 43-46. doi: 10.1145/1540276.1540290).

[44] Lopes, C, Bajracharya, S., Ossher, J., Baldi, P. (2010). UCI Source Code Data Sets [http://www.ics.uci.edu/ lopes/datasets]. Irvine, CA: University of California, Bren School of Information and Computer Sciences, Retrieved from https://archive.ics.uci.edu/ml/datasets.html

[45] López,V., Fernandez, A. G., Palade, V., Herrera, F. (2013), An Insight into Classification with Imbalanced Data: Empirical Results and Current Trends on Using Data Intrinsic Characteristics. Information Sciences 250 (2013) 113-141 doi: 10.1016/j.ins.2013.07.007

[46] Louzada, F., Ara, A., Fernandes, G. (2016), Classification methods applied to credit scoring: A systematic review and overall comparison, Department of Applied Mathematics and Statistics, University of Sao Paulo, Sao Carlos, Brazil b Department of Statistics, Federal University of Sao Carlos, Sao Carlos, Brazil P and D Inovation in Analytics, Serasa-Experian, Sao Paulo, Brazil, February 2016

[47] Luo, C., Wu, Desheng, Wu., Dexiang (2016), A deep learning approach for credit scoring using default swaps, Journal of Engineering Applications of Artificial Intelligence, Retrieved from http://www.sciencedirect.com/science/article/pii/S0952197616302299

[48] Madyatmadia, E. D., Aryuni, M. (2005), COMPARATIVE STUDY OF DATA MINING MODEL FOR CREDIT CARD APPLICATION SCORING IN BANK, Journal of Theoretical and Applied Information Technology 20th January 2014. Vol. 59 No.2 copyright 2005 - 2014 JATIT & LLS

[49] McHugh, M. L. (2012). Interrater reliability: the kappa statistic, Biochemia Medica, 22(3), 276-282.

[50] MedCalc Software (2017) Medcalc ROC curve analysis in MedCalc, MedCalc Statistical software, copyright 1993-2017 MedCalc Software bvba,https://www.medcalc.org/manual/roc-curves.php

[51] Monard, M.C., Batista, G (2002), Learning with Skewed Class Distributions, Retrieved from http://conteudo.icmc.usp.br/pessoas/mcmonard/public/laptec2002.pdf

[52] Ng, A. (2013), Self-Taught Learning and Unsupervised Feature Learning, Retrieved from https://www.youtube.com/watch?v=n1ViNeWhC24

[53] Ng, A. (2016) Nuts and Bolts of Applying Deep Learning, Deep Learning School on September 24/25, 2016, Video file, Retrieved from https://www.youtube.com/watch?v=F1ka6a13S9I

[54] Quinlan, J. R. (1986) Induction of Decision Trees, Machine Learning 1, 1 (Mar. 1986), 81-106

[55] RAY, S. (2015a), 7 Regression types you should know! , August 14, 2015, Retrieved from https://www.analyticsvidhya.com/blog/2015/08/comprehensive-guide-regression/

[56] RAY, S. (2015b) 5 Easy questions on Ensemble Modeling everyone should know, September 30, 2015, Retrieved from https://www.analyticsvidhya.com/blog/2015/09/questions-ensemble-modeling/

[57] Saito T, Rehmsmeier M. (2015) The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets. PLoS ONE 10(3): e0118432. https://doi.org/10.1371/journal.pone.0118432

[58] Santini, M. (2015) Lecture 4 Decision Trees (2): Entropy, Information Gain, Gain Ratio, Video file,

[59] Shadram, A. (2016) Measuring Performance of Classifiers, Retrieved from URL http://shahramabyari.com/2016/02/22/measuring-performance-of-classifiers/, Author Shadram, Abyari February 22, 2016

[60] Sing, T., Sander, O., Beerenwinkel, N., Lengauer, T (2005) ROCR: visualizing classifier performance in R. Bioinformatics 21(20):3940-3941 (2005).

[61] Singh, R. (2011) Macro Designing and Comparative Evaluation of Various Predictive Modeling Techniques of Credit Card Data, Masters Thesis in the Computer Science and Engineering Program Thapar University June 2011

[62] Shahram, A. (2016), Measuring Performance of Classifiers, Shahram, Abyari: Author, Retrieved from URL http://shahramabyari.com/2016/02/22/measuring-performance-of-classifiers/

[63] Stergiou, C., Siganos, D. (2017) NEURAL NETWORKS, Retrieved from https://www.doc.ic.ac.uk/~nd/surprise^96/journal/vol4/cs11/report.html#Introduction to neural networks

[64] Thomas, L.C. (2000) A survey of credit and behavioral scoring: forecasting risk of lending to consumers, International Journal of Forecasting 16 (2000) 149-172

[65] Thomas, L. C., Edelman, D., Crook, J. (2002), Credit Scoring and its applications, Monographs on mathematical modeling and computation. SIAM.

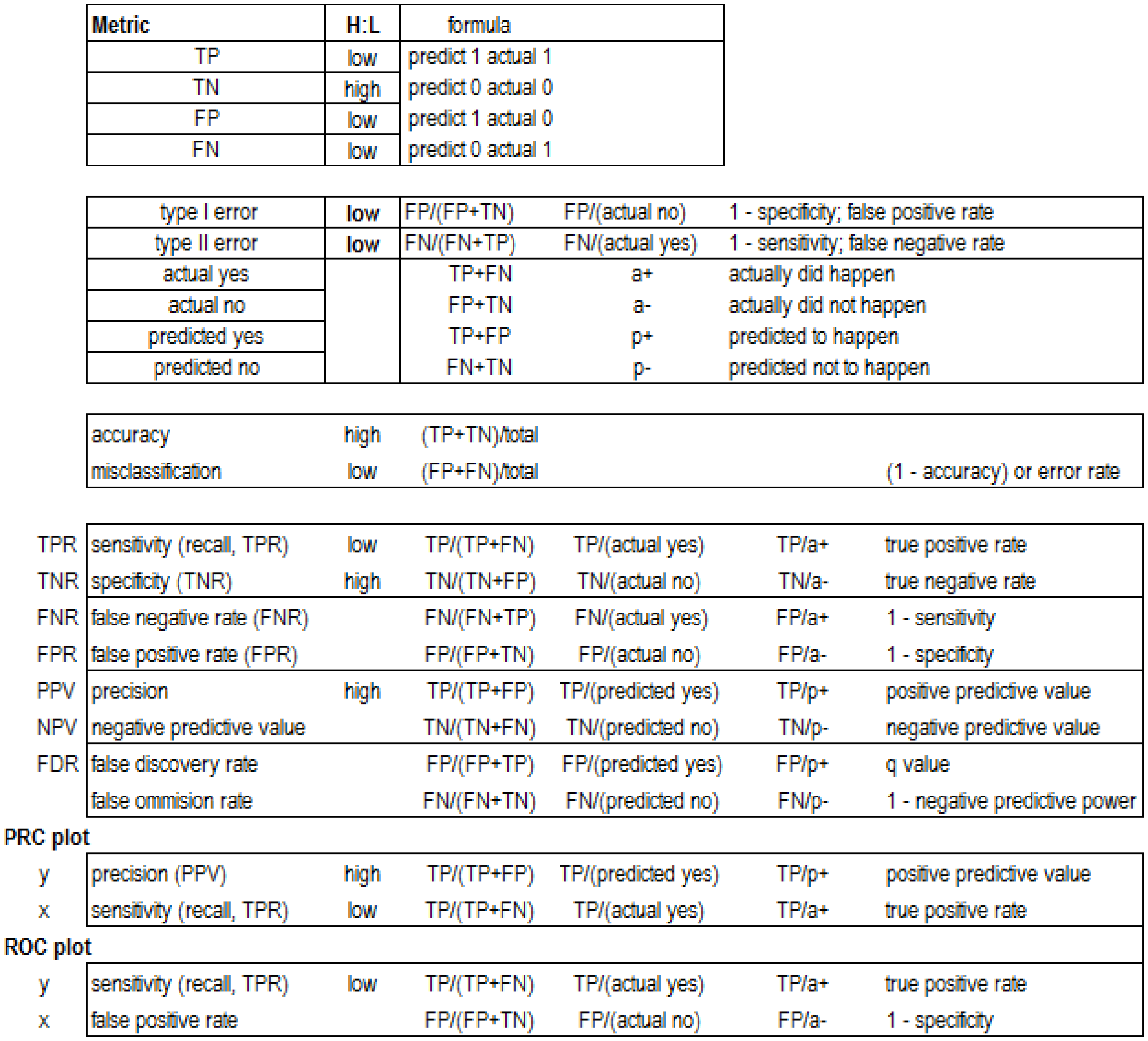
[66] Wang, G., Hao, J., Ma, J., Jiang, H. (2011), A comparative assessment of ensemble learning for credit scoring, Elsevier Journal Expert Systems with Applications 38 (2011) 223-230

[67] Wu, Y., Lee, W. (2014), Alternative Performance Measures for Prediction Models, PLOS published March 7, 2014; Published: March 7, 2014, Retrieved from https://doi.org/10.1371/journal.pone.0091249

[68] Zhu, H., Beling, P.,& Overstreet, G. (2001) A Study in the Combination of Two Consumer Credit Scores. The Journal of the Operational Research Society, 52(9), 974-980. Retrieved from http://www.jstor.org/stable/822776

APPENDIX A

Classification metric summary table



Metric	H:L	formula
TP	low	predict 1 actual 1
TN	high	predict 0 actual 0
FP	low	predict 1 actual 0
FN	low	predict 0 actual 1

type I error	low	FP/(FP+TN)	FP/(actual no)	1 - specificity; false positive rate
type II error	low	FN/(FN+TP)	FN/(actual yes)	1 - sensitivity; false negative rate
actual yes		TP+FN	a+	actually did happen
actual no		FP+TN	a-	actually did not happen
predicted yes		TP+FP	p+	predicted to happen
predicted no		FN+TN	p-	predicted not to happen

accuracy	high	(TP+TN)/total	
misclassification	low	(FP+FN)/total	(1 - accuracy) or error rate

TPR	sensitivity (recall, TPR)	low	TP/(TP+FN)	TP/(actual yes)	TP/a+	true positive rate
TNR	specificity (TNR)	high	TN/(TN+FP)	TN/(actual no)	TN/a-	true negative rate
FNR	false negative rate (FNR)		FN/(FN+TP)	FN/(actual yes)	FP/a+	1 - sensitivity
FPR	false positive rate (FPR)		FP/(FP+TN)	FP/(actual no)	FP/a-	1 - specificity
PPV	precision	high	TP/(TP+FP)	TP/(predicted yes)	TP/p+	positive predictive value
NPV	negative predictive value		TN/(TN+FN)	TN/(predicted no)	TN/p-	negative predictive value
FDR	false discovery rate		FP/(FP+TP)	FP/(predicted yes)	FP/p+	q value
	false ommision rate		FN/(FN+TN)	FN/(predicted no)	FN/p-	1 - negative predictive power
PRC plot						
y	precision (PPV)	high	TP/(TP+FP)	TP/(predicted yes)	TP/p+	positive predictive value
x	sensitivity (recall, TPR)	low	TP/(TP+FN)	TP/(actual yes)	TP/a+	true positive rate
ROC plot						
y	sensitivity (recall, TPR)	low	TP/(TP+FN)	TP/(actual yes)	TP/a+	true positive rate
x	false positive rate		FP/(FP+TN)	FP/(actual no)	FP/a-	1 - specificity

Figure A.1. Summary of classification metrics

APPENDIX B

Australian dataset variables

A1: 0,1 CATEGORICAL (formerly: a,b)

A2: continuous.

A3: continuous.

A4: 1,2,3 CATEGORICAL (formerly: p,g,gg)

A5: 1, 2,3,4,5, 6,7,8,9,10,11,12,13,14 CATEGORICAL (formerly: ff,d,i,k,j,aa,m,c,w, e, q, r,cc, x)

A6: 1, 2,3, 4,5,6,7,8,9 CATEGORICAL (formerly: ff,dd,j,bb,v,n,o,h,z)

A7: continuous.

A8: 1, 0 CATEGORICAL (formerly: t, f)

A9: 1, 0 CATEGORICAL (formerly: t, f)

A10: continuous.

A11: 1, 0 CATEGORICAL (formerly t, f)

A12: 1, 2, 3 CATEGORICAL (formerly: s, g, p)

A13: continuous.

A14: continuous.

A15: 1,2 class attribute (formerly: +,-)

APPENDIX C

German dataset variables

Attribute 1: (qualitative) Status of existing checking account

A11 : ... < 0 DM

A12 : 0 <= ... < 200 DM

A13 : ... >= 200 DM / salary assignments for at least 1 year

A14 : no checking account

Attribute 2: (numerical) Duration in month

Attribute 3: (qualitative) Credit history

A30 : no credits taken/ all credits paid back duly

A31 : all credits at this bank paid back duly

A32 : existing credits paid back duly till now

A33 : delay in paying off in the past

A34 : critical account/ other credits existing (not at this bank)

Attribute 4: (qualitative) Purpose

A40 : car (new)

A41 : car (used)

A42 : furniture/equipment

A43 : radio/television

A44 : domestic appliances

A45 : repairs

A46 : education

A47 : (vacation - does not exist?)

A48 : retraining

A49 : business

A410 : others

Attribute 5: (numerical) Credit amount

Attribute 6: (qualitative) Savings account/bonds

A61 : ... < 100 DM

A62 : 100 <= ... < 500 DM

A63 : 500 <= ... < 1000 DM

A64 : .. >= 1000 DM

A65 : unknown/ no savings account

Attribute 7: (qualitative) Present employment since

A71 : unemployed

A72 : ... < 1 year

A73 : 1 <= ... < 4 years

A74 : 4 <= ... < 7 years

A75 : .. >= 7 years

Attribute 8: (numerical) Installment rate in percentage of disposable income

Attribute 9: (qualitative) Personal status and sex

A91 : male : divorced/separated

A92 : female : divorced/separated/married

A93 : male : single

A94 : male : married/widowed

A95 : female : single

Attribute 10: (qualitative) Other debtors / guarantors

A101 : none

A102 : co-applicant

A103 : guarantor

Attribute 11: (numerical) Present residence since

Attribute 12: (qualitative) Property

A121 : real estate

A122 : if not A121 : building society savings agreement/ life insurance

A123 : if not A121/A122 : car or other, not in attribute 6

A124 : unknown / no property

Attribute 13: (numerical) Age in years

Attribute 14: (qualitative) Other installment plans

A141 : bank

A142 : stores

A143 : none

Attribute 15: (qualitative) Housing

A151 : rent

A152 : own

A153 : for free

Attribute 16: (numerical) Number of existing credits at this bank

Attribute 17: (qualitative) Job

A171 : unemployed/ unskilled - non-resident

A172 : unskilled - resident

A173 : skilled employee / official

A174 : management/ self-employed/highly qualified employee/ officer

Attribute 18: (numerical)

Number of people being liable to provide maintenance for

Attribute 19: (qualitative) Telephone

A191 : none

A192 : yes, registered under the customers name

Attribute 20: (qualitative) foreign worker

A201 : yes

A202 : no