Data Mining Techniques to Identify Credit Risk

Andrew Rhodes

York College of Pennsylvania

E: arhodes4@ycp.edu

Abstract

Using 1000 records of credit data, detailing 29 variables of interest to a bank of a prospective borrower, we look to understand the ability of data mining techniques to predict these credit decisions. The total sample space was partitioned into randomly selected 60% train, 25% validation and 15% test sets. Each of the three models of interest, a logistic regression model, K-nearest neighbor model and neural network model were then specified using the training partition. Once fit, the models were each tested on the validation partition to select the model of greatest accuracy. The logistic regression model is found to have the greatest amount of accuracy and therefore is the model of choice. When the model was tested via the test partition, it was found to have an accuracy of 0.80. In a scenario in which the bank experiences a 5-to-1 loss on bad loans approved vs. good loans approved, none of the three models were profitable. Therefore, we then analyze the probit function of the logistic regression model to understand what propensity of success that should act as the threshold over which the bank should extend credit to maximize profit. This analysis suggests that the bank should approve all loans that return a logistic probit propensity of 0.799 or greater, as this returned the highest net profit on the validation partition assuming a 5-to-1 loss-to-reward ratio. This propensity of success existed in applicants of the 43rd percentile of the validation set.

Chk. Acct. Balance	Owns Residence	Number of Credits	Num. Dependents
Duration of Credit	Retraining	Co-Applicant	Has Phone
Credit History	Job Type	Guarantor	Foreign worker
New Car	Amount	Residence History	% of Income
Used Car	Sav. Acct. Balance	Real Estate Owned	Male & Single
Furniture	Employment History	No Real Estate	* Red cells denote an
Radio/TV	Male & Divorced	Other Installment	inverse relationship
Education	Male & Married	Rents Residence	to credit response

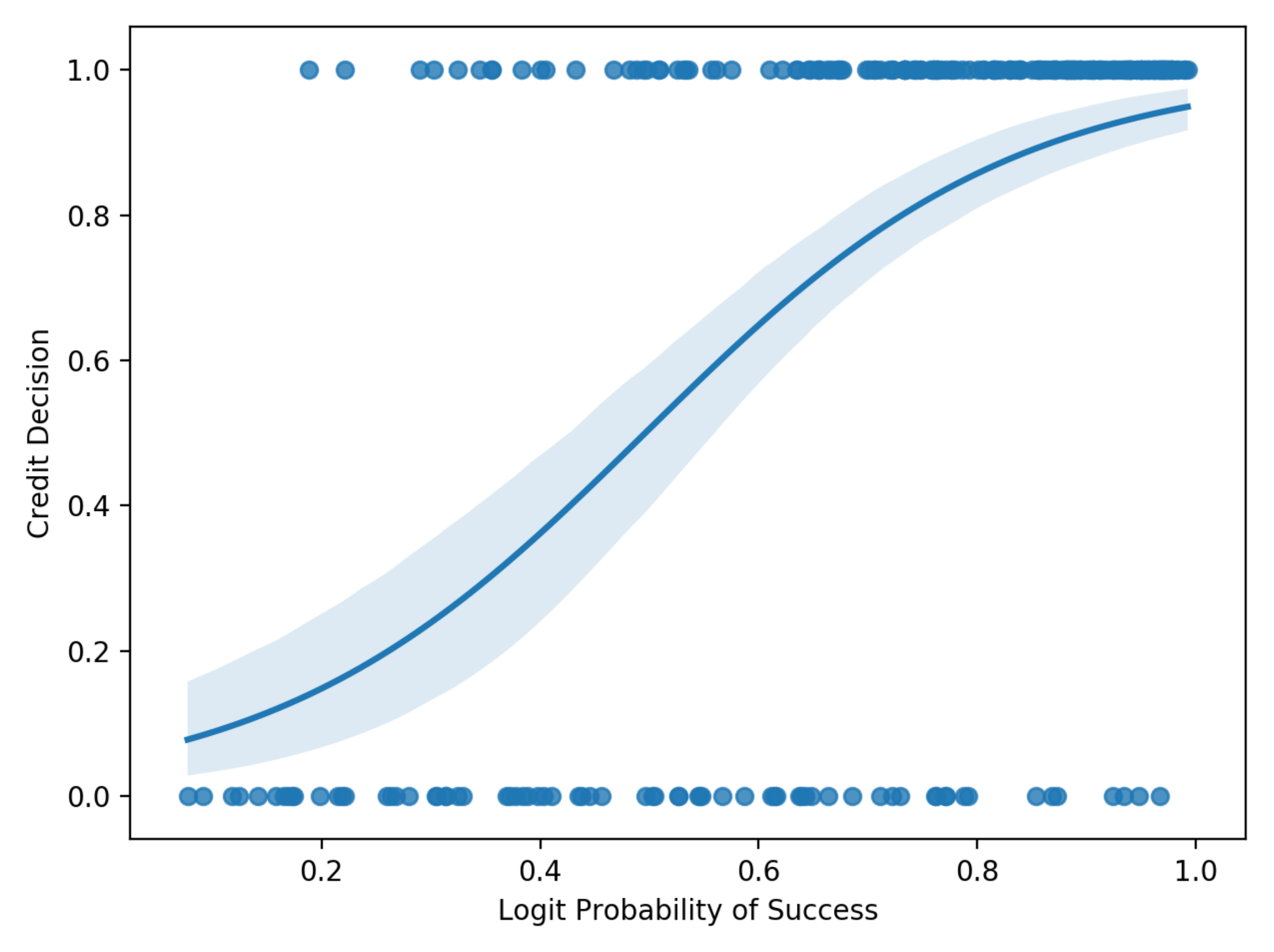
Introduction

A German bank has provided 1000 credit records that include 29 pieces of data that describe the customer's financial position and creditworthiness. These variables (shown in the figure to the left) include variables such as the loan purpose (new/used car, furniture or education), existing account data and if the individual owns a residence. Each of these data acts as the independent variable to explain whether the customer was approved or denied the loan. The data is either binary, categorical or numerical. Although some data points may be categorical, they were treated as numerical variables as they represented a degree of magnitude. For example, the "job type" variable takes the value of 0-4 with 0 indicating the customer being unemployed and 4 indicating the individual is in a management role, which would tend to indicate a greater level of job security. The figure to the left shows some cells being highlighted in red which indicates an inverse relationship to the credit response. For example, the "% of income" variable represents the requested loan amount as a % of the customer's annual income. As this number grows, it tends to explain a decrease in the likelihood of being approved. The three models chosen were a logistic regression, K-nearest neighbor and neural network model. The 1000-piece data set was partitioned into a 60% training set, 25% validation set and 15% testing set. Each model was fit using the training data and then tested against the validation data to choose the model. The selected model was then tested on the test partition.

Logistic Regression Model

Methodology

A logistic regression model allows us to use regression techniques to forecast an outcome when the dependent variable is binary. In our case, the dependent variable is success (credit approved), denoted as 1 and failure (credit denied), denoted as 0. The below figure shows the logistic regression line plotted vs. the actual outcomes of the credit decision. Along the x-axis is the probit propensity of the logistic regression model and along the y-axis is the actual outcome as per the data set. To specify the logistic regression model, we first fit the model to the training data. Once fit, the model is then validated via the validation partition. The model specified utilized a liblinear optimization function to classify the data, which was ideal given the rather small size of the dataset. To regulate the model, an l2 penalty method was chosen in which the penalty is equal to the square of the magnitude of the coefficients.



	Accept	Reject
Good	31	42
Bad	21	156

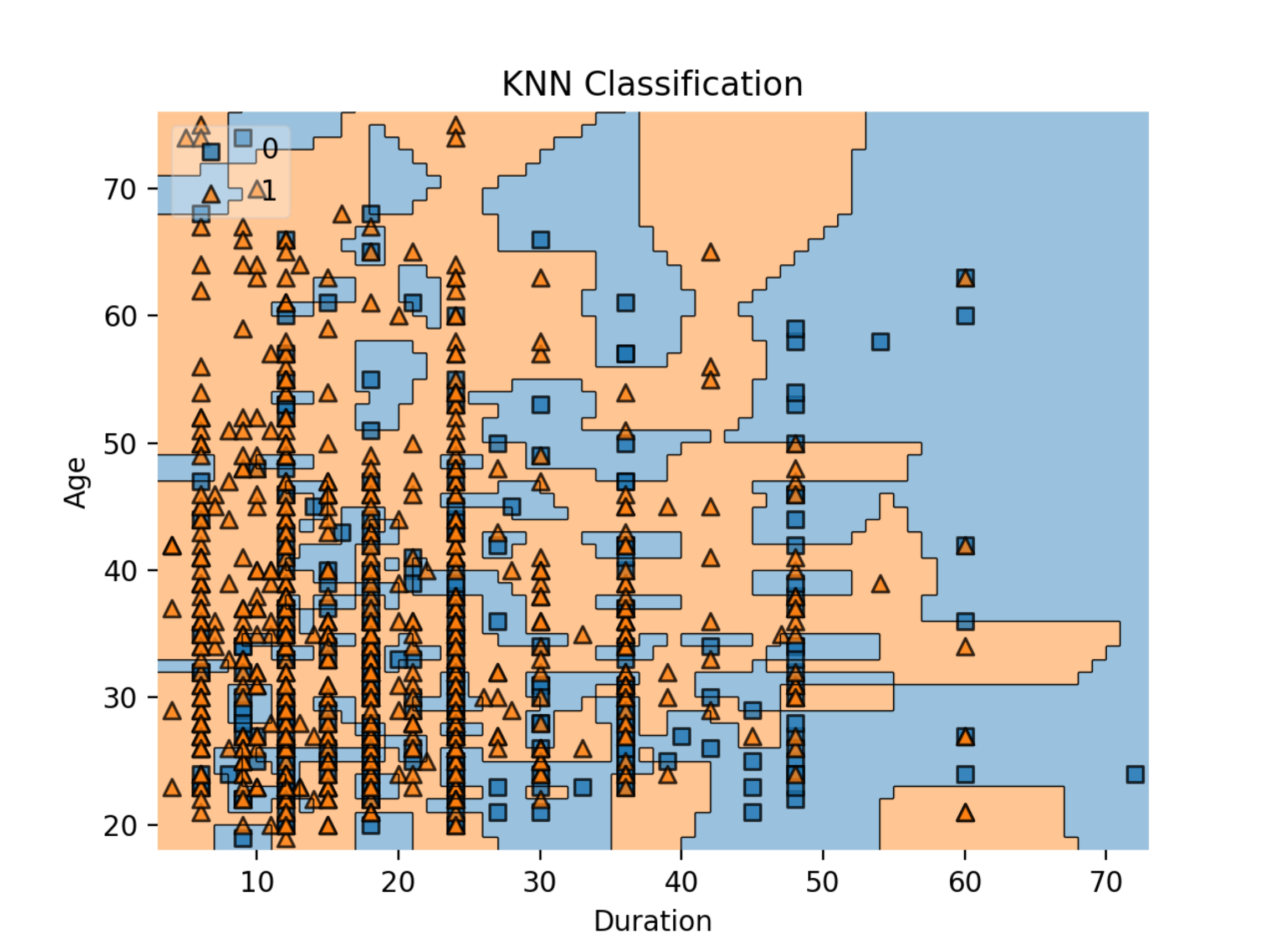
Results

The resulting logistic model, when validated on the validation dataset, had a favorable accuracy score of 0.75 with a confusion matrix as shown above. The confusion matrix indicates that of the 73 customers that were approved for a loan by the bank, 31 were approved by the logistic regression model. Of the 177 applicants that were not approved by the bank, 156 were not approved by the logistic regression model. Although the model has favorable accuracy, a 5-to-1 risk-reward ratio would indicate that this model is not in fact profitable. Given these results, a logistic regression model, like the one specified above, would not be favorable for the bank as it would mean that a net loss would be expected if used on actual customers.

K-Nearest Neighbor (KNN) Model

Methodology

A K-Nearest Neighbor model utilizes a cluster analysis technique to identify data points in the train partition similar to that of the datapoint of interest in the validation partition. The KNN model allows for specification of the amount of "neighbors" or test data points that the researcher requires to be used to identify the validation data point outcome. To find the best K value to use, or the amount of neighbors required to characterize the validation data, the model was fit using the training partition for a K value of 1 through 100 and then validated on the validation partition. The K value that resulted in the highest accuracy on the validation partition was a K value of 1. The weights applied to each datapoint of the training partition were uniform, instead of the typical inverse distance method, as it resulted in a smaller degree of overfitting when reflecting on the confusion matrix. The figure below illustrates the KNN classification via 2 of the 29 dimensions of the data set: age and duration. Orange represents success (1).



	Accept	Reject
Good	25	56
Bad	59	110

Results

The resulting KNN model, when validated on the validation dataset had a rather unfavorable accuracy score of 0.54 with a confusion matrix as shown above. The confusion matrix indicates that of the 81 customers that were approved for a loan by the bank, 25 were approved by the KNN model. Of the 169 applicants that were not approved by the bank, 110 were not approved by the KNN model. A 5-to-1 risk-reward ratio would indicate that this model is not in fact profitable. Given these results, a KNN model, like the one specified above, would not be favorable for the bank as it would mean that a net loss would be expected if used on actual customers.

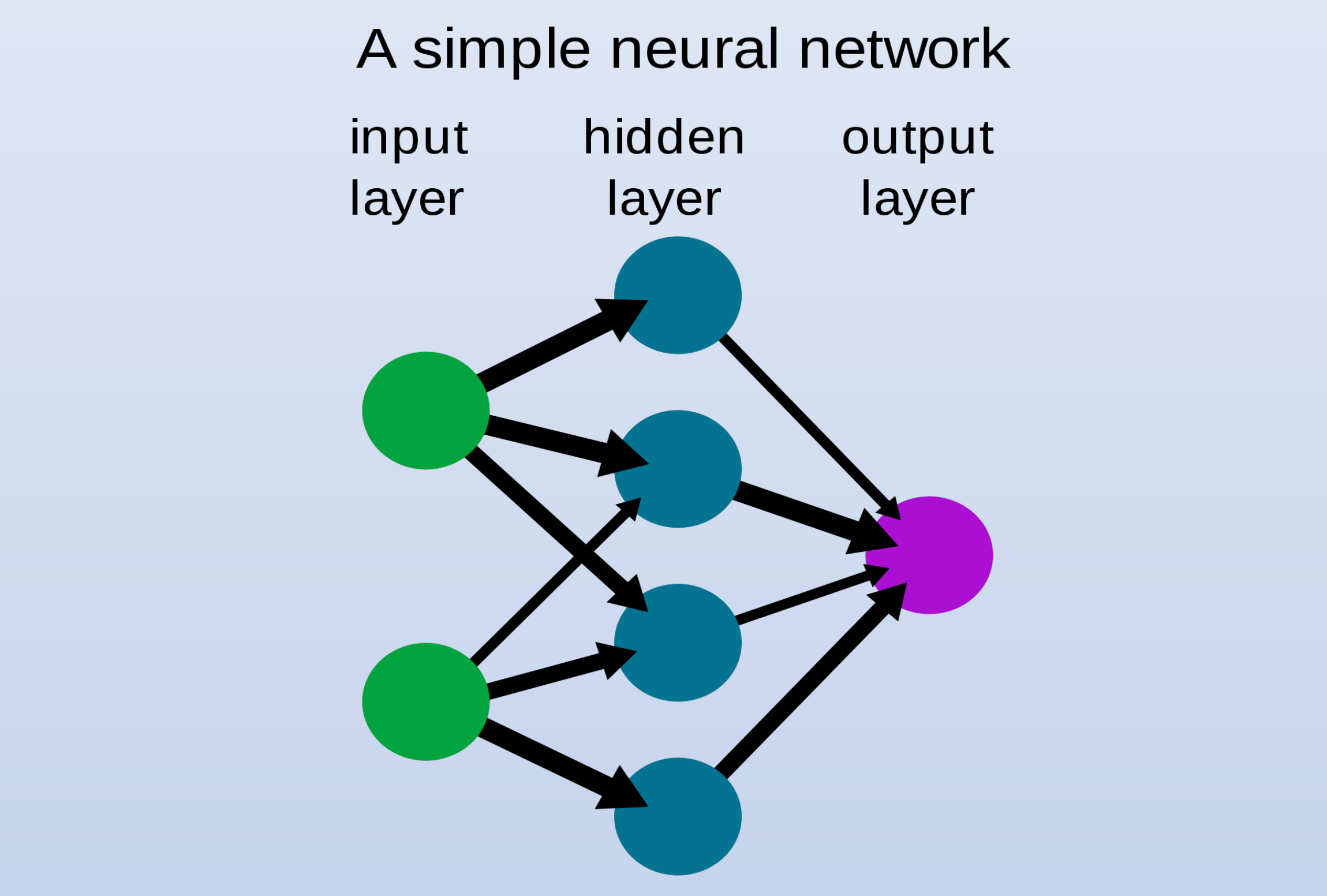
Neural Network Model

Methodology

A neural network model is used to assist in the classification process of the credit decision. It uses a rectified linear unit optimization function for a hidden layer of neurons to produce the output decision of either a 1 (success) or a 0 (failure). The function of the rectified linear unit optimization function is as follows:

$$f(x) = \max(0, x)$$

The training partition was used to optimize the model through testing the amount of hidden layers that would result in the highest amount of accuracy on the validation data. The model that produced the highest accuracy on the validation partition was the model that utilized a hidden layer of 1. A L-BFGS solver function was utilized in the neural network model of choice. The LBFGS method is part of the quasi-newton family of solvers. The graphic below illustrates the neural network structure.



	Accept	Reject
Good	4	68
Bad	14	164

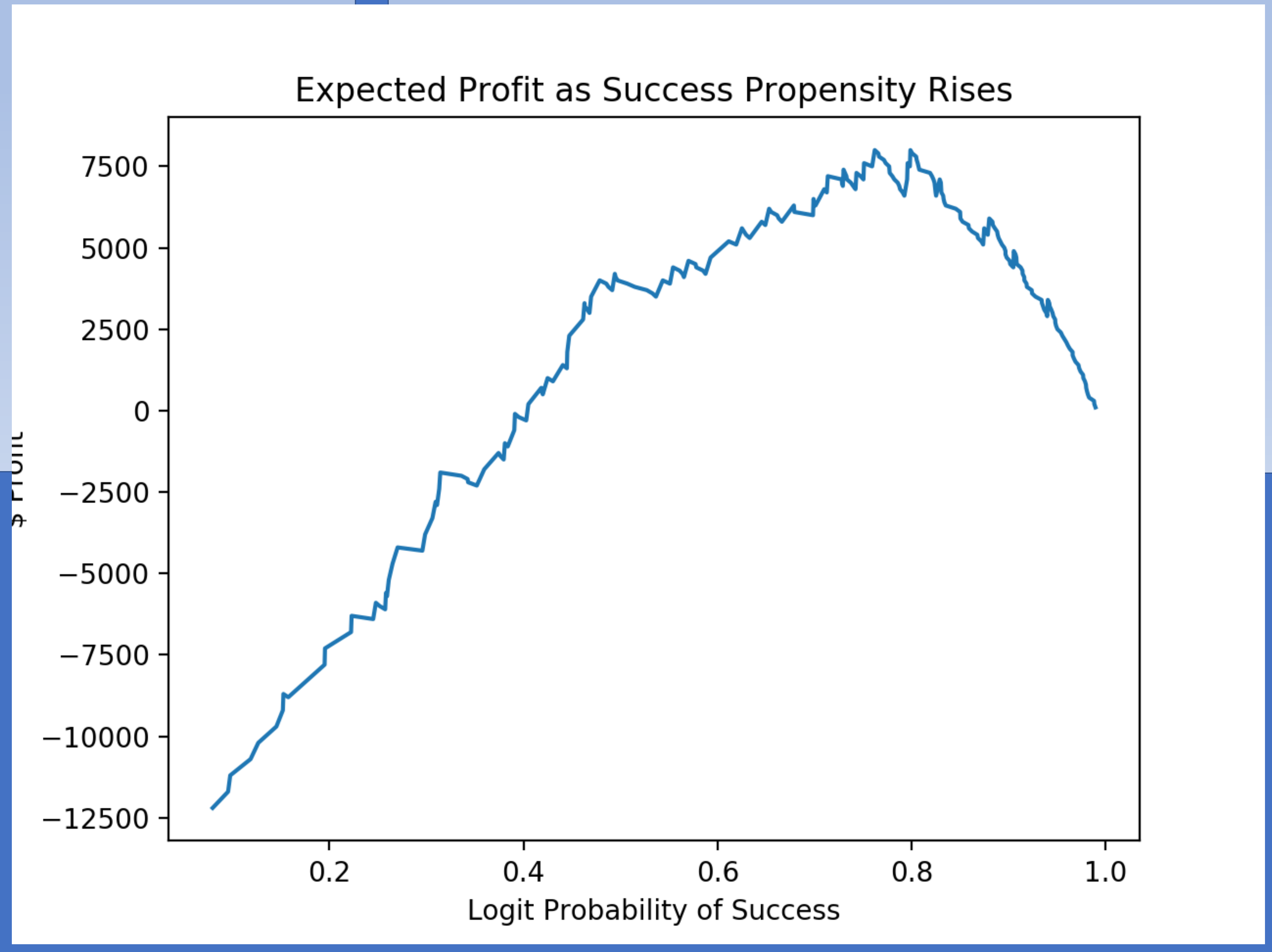
Results

The resulting neural network model, when validated on the validation partition had an accuracy score of 0.67 with a confusion matrix as shown above. The confusion matrix indicates that of the 72 customers that were good per the bank, 4 were approved by the neural network model. Of the 178 applicants that were bad, 164 were not approved by the neural network model. A 5-to-1 risk-reward ratio would indicate that this model is not in fact profitable. It would seem that the neural network model seems to result in a model that is too stringent in classifying credit customers. Given these results, a neural network model, like the one specified above, would not be favorable for the bank as it would mean that a net loss would be expected if used on actual customers.

Conclusion

The models discussed above indicate a lack of profitability for the bank, given a 5-to-1 risk-to-reward ratio that the bank likely experiences. The accuracy of the logit and neural network model are favorable, with accuracy scores of 0.75 and 0.67, respectively. However, the confusion matrix of each of these models would suggest that it is perhaps too poor of a predictor for those data points that are of good credit standing. For almost all the models, it failed to score the majority of the validation partition that was originally approved by the bank as being a success. This resulted in a large amount of opportunity cost for the bank and could denote that the models are too stringent in their credit standards. The vast majority of bad credits of the validation partition were rejected by each model, which would further confirm the rather large degree of stringency in credit selection of each model. The model selected to be tested via the test partition was the logit model. The accuracy score of the logit model on the training partition was 0.80, higher than the same model's accuracy on the validation partition. The confusion matrix of the test partition is shown below, in which 30 of the 51 good credits were accepted by the logit model, and 90 of the 99 bad credits were rejected. A logit model can be further specified by understanding which probit propensity results in the maximum profit—again, considering a 5-to-1 risk-to-reward ratio. The figure to the right shows this function graphed. As the propensity of success rises, the net profit rises until we reach a probit of 0.799. After this point, profit begins to fall as we accept less loans that have a high propensity to be profitable. This would suggest that, if the validation partition is representative of the population, the bank should employ a logit model in which all borrowers that are assigned a probit of 0.799 or higher are granted approval. This customer base was in the 43rd percentile of the validation partition.

	Accept	Reject
Good	30	21
Bad	9	90


Expected Profit as Success Propensity Rises
$ Profit
Logit Probability of Success