Fintech Credit Scoring Techniques for Evaluating P2P Loan Applications – A Python Machine Learning Ensemble Approach

Rekha Ramesh Shenoy\textsuperscript{1}, Sabah Mohammed\textsuperscript{2}, Jinan Fiaidhi\textsuperscript{3}
\textsuperscript{1,2,3}Department of Computer Science, Lakehead University, ON, Canada
\textsuperscript{1}rshenoy@lakeheadu.ca, \textsuperscript{2}mohammed@lakeheadu.ca, \textsuperscript{3}jfiaidhi@lakeheadu.ca

Abstract

Financial Technology (fintech) has been widely recognized as one of the most important innovations in the financial industry and is seen to evolving at a very rapidly. It holds the promise of reshaping the financial industry by creating a diverse financial landscape by providing stability, improving quality and most importantly reducing costs. One such fintech tool is the “Peer to Peer Lending” (also known as “P2P Lending ”), which refers to companies that match lenders and borrowers without the use of the traditional banking systems. They are intermediaries that are usually online investment platforms that offer identity verification, proprietary credit models, loan approval, loan servicing and legal and compliance. This can be an attractive alternative for a borrower as loans can be applied for online, anonymously, and in a timely fashion. It is also beneficial for borrowers that do not have any previous credit history to be shown. Fintech develops a credit scoring model based on the credit risk evaluation. This model establishes itself in the use of online data sources, alternative credit models and variety of machine learning and data analytics techniques to estimate risks involved in the lending process and to minimize the operating costs. In this paper, we propose a stacking ensemble of machine learning classifiers that combines data preprocessing with different learning algorithms. We then compare the results of the bare bone classifiers with our stacking ensemble classifier The ensemble model developed gives a better performance than each of single classifiers that constitute the process of credit scoring.

Keywords: fintech tools, credit scoring, machine learning algorithms, feature reduction, outliers, scikit-learn, regression, clustering, Bayesian, neural networks, forests, ensembles, bagging, boosting, stacking.

1. Introduction:

Managing customer credit is an important issue for each commercial bank; therefore, banks take great care when dealing with customer loans to avoid any improper decisions that can lead to loss of opportunity or financial losses. The manual estimation of customer creditworthiness has become both time- and resource-consuming. Moreover, a manual approach is subjective (dependable on the bank employee who gives this estimation), which is why devising and implementing programming models that provide loan estimations is the only way of eradicating the ‘human factor’ in this problem \cite{9}.

The current computerized credit scoring systems are based on classical statistical theories are widely used. However, these models are less resilient when it comes to large
amounts of data input; as a consequence, some of the assumptions in the classical statistics analysis fail [6]. In all types of business startups and established small businesses, many of these businesses are seeking some additional funding that is too small for an angel investor to get a return for their effort. Banks also think it’s not worth their time. However, the amount necessary may be too much to finance on a credit card, or perhaps the entrepreneur doesn’t want to use that method. That’s where peer-to-peer (P2P) lending is working to fill that lending gap and why we are considering this lending alternative and to evaluate the credit scoring for such lending system. This leading credit scoring evaluation may be a solution for many small businesses that are struggling with just tapping smaller funding amounts. Peer-to-peer lending involves borrowing money from your peers, including other businesspeople and investors who are interested in relatively small financing amounts.

As researched on various news and reports on Fintech, it was observed that these startups are mainly based on business models that target eminent services that are in demand such as Wealth Management, Payments, Lending, Crowd-funding, Capital Markets and Insurance. Therefore, construction of credit scoring models for startup loans requires data mining techniques. This process may use variety of data bins including demographic characteristics, historical data on payments and statistical techniques. While building a machine learning model for credit scoring based on “bin” of characteristics with value and ranges is something that much better that the legacy statistical methods, the bins are meant to maximize the separation between known good cases and known bad cases which largely depend on the dataset selected and machine learning model used in the training stage. However, ensemble methods have been called the most influential development in Data Mining and Machine Learning in the past decade [9]. They combine multiple models into one usually more accurate than the best of its components. Ensembles can provide a critical boost to industrial challenges -- from investment timing to drug discovery, and fraud detection to recommendation systems -- where predictive accuracy is more vital than model interpretability.

A lot of emphasis thus is given for choosing an efficient ensemble algorithm in the fintech organizations. The use of different classifier methods has varied over time wherein single classifiers were used in the initial period, however each of these classifiers showed some deficiencies in generating a good result given different datasets. Hence, the fintech organizations and researchers started experimenting with complex models and introducing newer techniques which resulted in hybrid / ensemble methods. The only difference between hybrid and ensemble methods was that hybrid methods introduced data preprocessing and filtering on the datasets before training the model and later, while ensemble methods focused on the classifier learnings of different base classifiers. Our paper aims to build a hybrid system based on clustering and classification and then feeds this processed data to an ensemble to predict the classification with the best results given any datasets.

The remainder of the paper is organized as follows. In Section 2, the problem definition, related literature survey and the datasets used for the analysis are described. In Section 3, we present the details of methodology and the relevant machine learning algorithms are presented. Section 4 provides a complete report on the implementation. Based on the observations and results of these experiments, Section 5 draws conclusions and future research directions.

2. Research motivations:
The credit scoring model can be said to be a result of a statistical model which evaluates the information of the borrower and calculates an estimate for the probability of the borrower defaulting on his loan. With the advent of newer techniques and algorithms, efficient systems have been built to estimate the likelihood that a borrower may default. The actual task of estimating the risk default has been improved by the credit scoring models as they also include other aspects of credit risk management. The entire risk evaluation is divided into stages which are:

- Pre-application stage: to identify potential applicants.
- Application stage: to identify the acceptable applicants and collect their information
- Performance stage: to identify the possible behavior of the current customers based on customers with similar profiles.

### 2.1 Problem statement:

It is noteworthy to mention that for P2P lending in fintech, the startups are not directly involved in the lending, they do not provide the actual capital that influence the amount of lending. They are only instrumental in matching the borrowers with the lenders who are interested in the borrower’s credit purpose. It is very important to evaluate the objective for which the borrower intends to take the credit. At the same time, the lends also evaluates his intent to benefit from the lending of the credit to the borrower. Since the fintech companies must evaluate the real requirements and meet the right aspects to match the lender efficiently, there might be different objectives / aspects that need to be considered from both the lenders and borrowers point of view such as:

1. What are the factors explaining a loan default in fintech’s P2P lending?
2. How do the P2P lenders associate the borrowers with their risk association?
3. What are the noteworthy attributes/characteristics that are indicative of defaults in P2P lending?
4. How does machine learning prove beneficial in P2P lending?

To answer these questions, P2P lending majorly suffers from the problem of the information asymmetry wherein the borrowers are better informed than lenders of their ability and willingness to pay. This can cause adverse selection where the lenders are unable to distinguish between a good risk / bad risk borrower. The P2P lending platforms evaluate and assign a grade to each loan application which is associated to interest rate based on the credit risk. However, it is observed that the higher the interest, higher is the credit risk observed. While evaluating the credit risk P2P lending considers the loan and borrower characteristics such as Loan characteristics (Loan purpose, loan amount), Borrower characteristics (home ownership, assets, income and employment length) and credit history (expenses, records, ability and patterns in which customer paid previous loans). Using machine learning algorithms in P2P lending is significantly important because it helps apply predictive analysis to enormous amounts of data in real time and produces results that are quick and efficient. They can also help gather information from various online sources to detect rogue investors that are working together across multiple accounts.

Hence, for our project we dive into various machine learning techniques that evaluate different credit scoring datasets available to perform the analysis of comparing different machine learning algorithms. These credit datasets generally use ‘historical’ data which is gathered from various customers to build a scorecard based on their previous or current credit status. Of all the data available for the customer, only features that provide valuable information and impact their credit behavior are examined and analyzed.
2.2 Related work:

In response to the growth of fintech sector, especially in terms of lending and managing huge loan portfolios, various models of credit scoring systems have been implemented and adopted successfully. Various financial institutions have their diverse ways of collecting credit information and hence require different risk analytics systems. The advantage of fintech tools is that they provide for niche and independent analysis systems for the client organizations while trying to build an efficient system that addresses to multiple client demands. Hence, for credit risk scoring systems, it is very necessary that they implement a model that provides efficient solutions irrespective of the dataset that is provided to the model.

Table 1. Scoping Review

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Filtering</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Feature Selection</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
</tr>
<tr>
<td>Dimensionality Reduction</td>
<td>Linear Discriminant Analysis</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Bayesian</td>
<td>Naïve Bayes</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Bayesian Network</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Conditional Decision Trees</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Instance based</td>
<td>k-Nearest Neighbor</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Vector machines</td>
<td>Support Vector Machine (SVM)</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Neural machine</td>
<td>Artificial Neural networks (ANN)</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Regression</td>
<td>Logistic Regression</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Ensembles</td>
<td>Bagging</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Boosting</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Random Forest</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Gradient Boosting Machines (GBM)</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hybrid</td>
<td>√</td>
<td>√</td>
<td>√</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
While building efficient credit scoring models, some focus on the data preprocessing and stress on the importance of handling the data efficiently before training the model. [5] uses the recursive feature elimination approach to evaluate the significance of the features in the classifier by removing each feature step-by-step and evaluating the performance. Based on the feature ranking obtained, only the most significant features are retained. The accuracy of classifier using the selected features gave better accuracy and performance results compared to other methods. [2] observations also indicate that the feature selection method greatly helped reduce the overfitting problem while improving the accuracy. [7] focused on handling imbalanced data while using the traditional classification techniques such as logistic regression (LR), neural networks and decision trees (NN), the researchers also explore the suitability of gradient boosting (GB), least square support vector machines (LS-SVM) and random forests (RF) for loan default prediction. Over the time the researchers moved from classic algorithms and started experimenting more with complex algorithms and systems integrating multiple base classifiers that bought ensembles into effect. [4] authors propose a hybrid model of feature selection and ensemble learning classification algorithm based on valuation approaches viz the SVM classification accuracy, the AUC and parameter settings. [10] introduces the concept of class-wise classification to introduce a new class called ‘borderline’ risk to provide better estimation of potential risky borrower.

Although many models have been designed to predict the credit score accurately, there is still not a single ideal or specific classifier among the available models as each model behaves differently with different sets of data. It is important to note that each percentage point can affect the scoring system. Hence, choosing the best model is of the utmost importance as it relates to curbing the huge losses to the financial organization.

Statistical and mathematical formulas have been persistent and used on wide scale to calculate the credit scores, new research studies have proved that the Artificial Intelligence (AI), neural networks, support vector machines and ensemble methods can provide much more quick and accurate analysis as compared to the traditional approaches. Further experiments have shown that the hybrid or ensemble approaches although complex in nature have proven better performance than individual models.

The usual process in any credit scoring model is to use the credit history of the previous borrowers and to compute and predict the likelihood of default risk for new clients. The features or attributes of this historical data are thus used to map and predict the probability of default for a client. The number of features thus form the feature space. In machine learning, it is considered that the more the amount of data, more reliable is the prediction analysis. However, as the dimensionality of the dataset grows, lots of difficulties arise too, as there might be a lot of non-meaningful data in the entire dataset. Large datasets usually tend to have lots of noisy features which can greatly impact the machine learning. Hence, noise should be reduced as much as possible to improve the efficiency and accuracy of the machine learning algorithms.

With the datasets we have used for analysis, our primary observation is that each dataset differs from each other in terms of size, nature of attributes and the information they hold. Hence, it is very important to handle such variances and form an efficient classifier training method. There is also the problem of imbalanced data in large datasets especially in credit risk models, where the number of defaulting customer data is far lesser as compared to the non-defaulters’ data. Another thing to be considered is that for more features in a given dataset, there will be more computation time required which could impact the model accuracy and the prediction results.
Our proposed method focuses on building an ensemble model that focuses on the results of a group of classifiers that are trained on the same dataset and evaluate the best strategies for each dataset. The ensemble built combines the predictions from these different classifiers and gives the final prediction. Through this ensemble, we aim to build a sturdy system that performs the best with different datasets.

2.3 Datasets:

The datasets used in credit score model is based on the historical credit information of the former customers and used to predict the risk factor for a new applicant based on similar behavioral / social attributes. The attributes or features of each loan applicant are mapped to the historical loan accounts and the ideal credit model is built. The only disadvantage of this approach is that these datasets may differ in size, nature and information which usually causes discrepancies in the classifier training process thus missing on capturing the real correlation of the information to the desired scoring. They might contain missing values, redundant values, irrelevant features, erratic data etc. thus affecting the scoring greatly.

2.3.1 German credit dataset:
The German Credit dataset is a publicly available data set and can be downloaded from the UCI Machine Learning Repository2. This dataset contains 1000 entries with 20 categorical attributes that was prepared by Dr. Hofmann. There are 700 samples of credit worthy applicants and 300 samples where credit was not extended. This is based on the 20 attributes that describe credit history, account balances, loan purpose, loan amount, employment status and personal information. These 20 attributes are made of 13 categorical, 3 continuous and 4 binary features and 1 class feature to define good or bad risk.

The cost matrix / status of 1 indicates a good customer whereas status of 2 indicates a bad customer. It is based on the principal that “It is worse to class a customer as good when they are bad (5), than it is to class a customer as bad when they are good (1).”

Based on the correlation heatmap, the attributes of duration, amount, installment rate, residing since, age, number of credits held, and number of dependents showed the correlation with respect to the credit risk status. There were no missing values observed for this dataset.

2.3.2 Australian credit dataset:
The Australian Credit dataset is also a publicly available data set and can be downloaded from the UCI Machine Learning Repository2. This dataset contains 690 entries with 15 numerical attributes that was prepared by Dr. Hofmann. There are 383 samples of credit worthy applicants and 307 samples where credit was not extended. This is based on the 15 attributes that have been changed to meaningless symbols to protect confidentiality of data. This dataset is interesting because there is a good mix of attributes -- continuous, nominal with small numbers of values, and nominal with larger numbers of values2. These 15 attributes are made of 8 categorical, 6 continuous features and 1 class feature to define good or bad risk. The attribute names

---

2 UCI Machine Learning Repository is a collection of databases, domain theories and data generators available for the machine learning community. It is widely used by students, educators and researchers as a primary source of machine learning datasets. URL: https://archive.ics.uci.edu/ml/datasets/Statlog+%28German+Credit+Data%29.
have been hidden to maintain the confidentiality of this dataset, hence they have been named as A1, A2, ... and so on. There are no missing values observed in this dataset.

2.3.3 Give me some credit: This dataset was a part of a data analysis competition in Kaggle3. This dataset consists of 11 features and has 150,000 records. The predictor variable in this dataset is the ‘SeriousDlqin2yrs’ which has a binary value of 0 or 1. The value of 0 indicates that the borrower is a good customer who repays his loan on time. The value of 1 indicates that the borrower is a bad customer or ‘delinquent’ and has defaulted over his loans. The data is based on the customer evaluation for a period of 2 years. These 11 attributes are made of 10 continuous features and 1 class feature to define good or bad risk. Based on the correlation heatmap, the attributes which showed some correlation to the credit status are RevolvingUtilizationOfUnsecuredLines, age, number of times borrower was past due (30-59 days, 60-89 days and 90 days), debt ratio, monthly income, open credit, real loans taken and number of dependants. There were some missing values observed for monthly income and number of dependants attributes.

2.3.4 Mock dataset: This dataset is a real-time credit data set that was downloaded from Credit Risk Analytics4 webpage. The original dataset has 887380 entries and 74 attributes. However, there was a lot of missing data for some features. Hence, we took a subset of the original data and built our Mock Dataset. This mock dataset has 884631 samples and 19 attributes. There are 817963 samples of credit worthy applicants and 66668 samples where credit was not extended. This is based on the 19 attributes that describe credit history, account balances, loan purpose, loan amount, loan reimbursement employment status and personal information. These 20 attributes are made of 3 categorical, 15 continuous features and 1 class feature to define good or bad risk. Based on correlation heatmap, attributes like loan amount, term in months for the loan, interest rate, installment amount, employment length, open credit accounts, income, dti, delinquency observed, revolving balance, revolving utility, total payment including total received principal and interest and last payment played a significant correlation to the credit status. Some missing values were observed in this dataset for employment length and revolving utilization.

The datasets can be summarized as shown in [Table 2] as shown below:

<table>
<thead>
<tr>
<th>Dataset</th>
<th>No. of instances</th>
<th>No. of numerical features</th>
<th>No. of ordinal features</th>
<th>No. of nominal features</th>
<th>Class 1: Class 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>German</td>
<td>1000</td>
<td>3</td>
<td>5</td>
<td>13</td>
<td>700:300</td>
</tr>
<tr>
<td>Australian</td>
<td>690</td>
<td>6</td>
<td>0</td>
<td>8</td>
<td>383:307</td>
</tr>
<tr>
<td>Give Me Some Credit</td>
<td>150,000</td>
<td>10</td>
<td>0</td>
<td>0</td>
<td>139974:10026</td>
</tr>
<tr>
<td>Mock Data</td>
<td>884631</td>
<td>15</td>
<td>0</td>
<td>3</td>
<td>817963:66668</td>
</tr>
</tbody>
</table>

3 The Kaggle Public Wiki is a resource for learning statistics, machine learning and other data science concepts, with a strong focus on the practical application of those skills in a competitive environment. The referenced dataset is available at URL: https://www.kaggle.com/c/GiveMeSomeCredit

4 The referenced website is prepared by 3 professors – Prof. Bart Baesens (KU Leuven, Belgium), Prof, Daniel Rösch (Regensburg University, Germany) and Prof. Harald Scheule (Associate Professor at University of Technology, Sydney, Australia for credit risk learning and teaching purpose. The given dataset was obtained by registering through the URL: http://www.creditriskanalytics.net/datasets.html
3. Methodology:

The proposed model is designed in sequential steps such as data filtering, splitting the dataset into training and testing sets, training the model, generating predictions for a set of algorithms on a particular dataset, building an ensemble that takes in input these set of predictions, combines them to generate the final prediction using cross-validation see [Figure. 1].

3.1 Machine learning algorithms used for the hybrid ensemble:

This project aims to compare the performances of different classification techniques with respect to the credit scoring context. The machine learning algorithms that we used in our credit scoring model are as follows:

3.1.1 Logistic Regression: Logistic Regression models are usually used in analyzing models where the outcome variable is either binary or dichotomous. It follows the same principles as a linear regression with the difference observed only in the model and its assumptions. In logistic regression, instead of predicting the value of a variable Y based on the predictor variables like in linear regressions, we calculate the probability of Y being ‘Yes / No’ based on the given known values of the predictor variables. For our project, we shall therefore be focusing on the binary response achieved that determines whether a creditor is a good or bad (i.e. non-defaulter or defaulter). The logistic function is thus written as:

\[ p(x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x)}} \]  

3.1.2 Gradient Boosting Classifier: Gradient Boosting is a type of an ensemble technique used for regression or classification problem. Since the main idea of boosting is to add on new models iteratively to the ensemble, it combines the weak ‘learners’ into a strong learner in an iterative fashion. The principle idea behind these algorithms are to
construct the new base-learners to have maximum correlation with the negative gradient of the loss function of the ensemble\(^5\). The gradient boosting classifier is thus an additive model that allows for the optimization of the arbitrary differentiable loss functions. In each iteration, the regression trees are fit on the negative gradient of the binomial or multinomial loss function\(^6\). The regression trees in this case is usually a decision tree where each leaf is given a score.

The objective function for the gradient boosting classifier can be given as:

\[
J(\theta) = \sum_{i=1}^{n} l(y_i, \hat{y}_i) + \sum_k \Omega(f_k), \quad f_k \in \mathcal{F}
\]

Here, \(y_i\) represents the sum of scores assigned for each leaf in the decision tree, \(f_k\) is the regression tree and is a function that maps attributes to the score, \(\mathcal{F}\) is the space of all the regression trees. Thus, the gradient model predicts \(\theta\) by additive training. It starts from a constant function and keeps on adding new functions \(f_k\) in each iteration. After \(K\) number of rounds, \(\theta\) construction is complete. The value of \(f_k\) is this calculated by minimizing the \(J(\theta)\), During the minimizations, the gradient of loss of functions are uses, thus giving this method the name of ‘gradient boosting’.

3.1.3 Random Forest: A Random Forest classifier is a supervised learning procedure that operates on simple principle of “divide and conquer” wherein sample fractions of data are used to generate a randomized tree predictor on small piece of dataset and then these pieces are aggregated together. Once these decision trees are generated and trained, voting procedure is used to determine the most popular class for each tree and this is selected as the final class determinant for the random forest. Hence, it is also an ensemble method based on bootstrap aggregation or ‘bagging’ method. It uses feature bagging wherein a random subset of features is selected for training of the decision trees.

Random Forest is very simple to use and has proven high accuracy and good prediction results. It also eliminates the concern of overfitting the model as it builds enough trees so that the classifier can divide the data evenly. However, the large number of trees can make the algorithm slower and sometimes ineffective for real-time predictions. They can be said to be fast in train but slow in generating the predictions.

3.1.4 Gaussian Naïve Bayes Classifier: Naïve Bayes classifiers are probabilistic classifiers based on the Bayes theorem of string (naïve) independence assumption between the features\(^7\). The Gaussian Naïve Bayes relies on the assumption that for continuous data, the values associated with each class are distributed according to the Gaussian distribution principle. Thus, it assumes that the all the features may be unrelated to each other.

For Gaussian Naïve Bayes classifier approach, let’s assume that an attribute ‘\(x\)’ contains continuous data. Then the following algorithm segments the data by class and computes the mean \(\mu_y\) and variance \(\sigma^2_x\) for each class as follows:

\(^{5}\) Statement as understood in the article of ‘Gradient Boosting machines, a tutorial’ made available in the research webpage of ‘frontiers in Nuerobiots’ referenced at URL: https://www.frontiersin.org/articles/10.3389/fnbot.2013.00021/full


\(^{7}\) As learned and understood from the brief description of Gaussian Naïve Bayes method on Wikipedia. URL: https://en.wikipedia.org/wiki/Naive_Bayes_classifier
A Naïve Bayes classifier is used to calculate the posterior probability of the class by multiplying the prior probability of the class before seeing any likelihood of the data given its class. Thus, the NB classifier analyses the training set to determine the mapping function which shall decide the final class.

The most important consideration for combining models is to reduce the probability of misclassification based on any single induced model by increasing the system’s area of expertise through different combinations. We use logistic regression as its outputs have a nice probabilistic interpretation, and the algorithm can be regularized to avoid overfitting. Since we are considering different datasets having different attributes behavior, we aim to tackle the variance error and also to implement parallelized model using Random Forest algorithm which is inherently a bagging ensemble technique. We also target to reduce the bias error through the Gradient Boosting classifier which is the boosting ensemble technique. The Gaussian Naïve Bayes relies on the assumption that for continuous data, the values associated with each class are distributed according to the Gaussian distribution principle. Thus, it assumes that all the features may be unrelated to each other, thereby showcasing powerful knowledge representation and reasoning algorithm under conditions of uncertainty. The main intention behind using all the above-mentioned machine algorithms to create a very robust system that performs consistently.

4. Implementation and analytics:

“Ensemble is the art of combining diverse set of learners (individual models) together to improvise on the stability and predictive power of the model.” - Analytics Vidhya8. In every individual machine learning algorithm, there exists a certain limit beyond which it is unable to fit the given data and the accuracy stops. If we try to fit in more data, it leads to ‘data over-fitting problem’. This could be due to various reasons such as differences in population, hypothesis, the given raw data or the modeling technique. Ensembles usually overcome this problem as they use multiple models using different techniques such as mentioned below:

**Bagging:**
Bagging derives its name from Bootstrap Aggregating. It tries to implement similar types of learners on small samples of the data/ training set and then aggregates the model by taking a mean of all the predictions. However, while resampling the data, sometimes some instances get represented multiple times whereas some are left out. Since the individual base classifiers may not be exposed to same of records, voting of their results is carried out. It is used mainly to reduce the variance error defined as the quantification of how the predictions made on the same observations differ from each other. As the training data size increases, there is reduction observed for the variance, thus making the model predictions to be more accurate. We use the Random Forest algorithm which is a bagging technique as a base learner for our implementation.

**Boosting:**
Boosting uses an iterative technique which adjusts the weight of the observation based on previous classification. First, it uses the subset of the original data to produce a series

---

8 https://www.analyticsvidhya.com/blog/2015/08/introduction-ensemble-learning/
of average performing models. Then, it boosts their performance by combining these models together using some cost function (like voting). If the models in the first step are classified incorrectly, then it tries to increase the weight of the observation and vice versa. It is mainly used to reduce the bias error defined as quantification of how much on an average do the predicted values differ from the actual value. Unlike bagging where random subsets are created, boosting creates sampling based on the performance of the previous models. Thus, every new subset has elements that were likely to be misclassified by previous models. We use the Gradient Boosting Classifier algorithm which is a boosting technique as a base learner for our implementation.

**Stacking:**

In stacking, we use a learner to combine results from other individual learners. Instead of using some empirical formula to calculate weights, we introduce a meta-learner which takes in the results from individual learners and uses another approach to estimate the predictions. Hence, an ensemble of individual classifiers is first trained, and the resultant classification output is fed as the input to the meta-learner. It thus reduces either of the bias or variance error depending on which combining meta-learner we have used. Thus, the role of the meta-learner is to discover the best possible way to combine the prediction of the base learners.

The above three methods can thus be compared as shown in the below table:

<table>
<thead>
<tr>
<th>Table 2. Comparison between Ensemble Techniques</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Comparison</strong></td>
</tr>
<tr>
<td><strong>Method</strong></td>
</tr>
<tr>
<td><strong>Subset creation</strong></td>
</tr>
<tr>
<td><strong>Function to combine into single model</strong></td>
</tr>
<tr>
<td><strong>Suitable for</strong></td>
</tr>
<tr>
<td><strong>Goals</strong></td>
</tr>
<tr>
<td><strong>Example</strong></td>
</tr>
</tbody>
</table>

Ensembles are considered to be one of the most popular methods used in machine learning. They combine predictions from different models and generate a final prediction. Any number of base models can be combined to form the ensemble, the more the better! Ensembles not only improve prediction, but they also help reduce errors in the prediction. They do so by averaging out the irregularities, thus smoothening the decision boundaries.

Datasets usually comprise of varied attributes and features which may or may not be correlated to each other but are very important factors which determine the credibility of the expected predictions. However, as suggested earlier, they may sometimes include redundant or irrelevant features which make it difficult to train the models, thereby reducing the accuracy and performance of the given model. Hence, it is very important to determine the nature of these attributes and process them accordingly.
We have used the Python packages of pandas, numpy, scikit-learn and matplotlib designed specially to help with data analysis and visualization. These packages also have helped in getting the data ready for building our model.

4.1 Data filtering and pre-processing

Before we implement our training model, it is necessary that we recognize and remove any data or information that stands out to make our model uniform. “Observation which deviates so much from other observations as to arouse suspicion it was generated by a different mechanism” — Hawkins(1980). This was defined to explain outliers which are extreme values that tend to deviate from other observations in a given dataset. These outliers are majorly caused by data entry errors, incorrect measurement errors, experimental errors, data processing errors, sampling errors etc.

The first step in our credit modelling was to perform data filtering – this resulted in reduction of the original data set size to retain a meaningful data set while not affecting the data integrity. It helps to smoothen the decision boundaries thereby helping achieve the targeted prediction with much better accuracy and performance. Only important and most relevant attributes are retained and modelled for training and testing, improving not only the accuracy but also reducing the computational costs. The first step we perform is to identify the missing or NaN values in the given datasets. Since the datasets are mainly read in the ‘csv’ file format, the missing symbols are replaced with empty strings “” which is interpreted to NaN in the python packages. The German and Australian datasets did not have any null values, however the datasets GiveMeSomeCredit and Mock Data showed 29731 and 7165 number of total null values respectively. Listwise Deletion for null vales was performed.

```python
In [ ]: # Cleaning the data to remove all attributes with missing data (NaN/null)
dataset_df = pd_df.replace('',np.nan)
dataset_df = dataset_df.dropna(axis="rows", how="any")
```

**Figure 2. Code snippet for removing NaN values**

4.2 Outlier detection and removal:

The quality of the samples plays a very critical role in the modelling of the implemented classifier as misclassified patterns generated generally throw a lot of errors in the model thus affecting its accuracy. Using scatterplot from the matplotlib package we have identified isolated or inconsistent values based on clustering approach wherein each feature was classified with respect to the credit status. The assumption made in this case, is that these isolated values are those values that tend be far away from the continuous clusters.

For the outlier removal process, we used the normal distribution and standard deviation approach specially for numerical attributes to identify the starting and ending range values of the continuous values. In this approach, we tried to remove the outlier points by removing any points that were out of the range of (Mean ± 2*SD). The numpy ‘mean’ and ‘std’ functions were used to obtain the mean and standard deviation for each attribute. Once we get the range of the final list, we removed the rows of values that had values for the attributes outside the given range.
The heat map feature is used to show the correlation between the independent features with respect to the credit status.

4.3 Balancing the dataset:

Since the number of samples where the credit was extended was significantly larger than the samples where the credit was rejected, there is a huge possibility of the classifier system being biased and tending towards credit worthiness and extension. This could make the model highly unstable showcasing inaccurate predictions. Hence, we attempted to balance the dataset by randomly choosing a sample of records with class 0 which was equal to the number of samples belonging to the class 1 in the given datasets. This would help the classifier models to learn about each class equally and thus make for a better prediction model see [Figure 2]
4.4 Implementation of the Ensemble:

The ideal ensemble is the one that consists of highly accurate predictors which at the same time disagree as much as possible. Hybrid ensembles deal with the combination of base learners trained using different algorithms and their predictions are given as input to another ensemble learner that generalizes the resulting prediction based on the input probabilities. Ensemble learning method is a commonly used approach by researchers where multiple base classifier outputs are pooled to provide the decision. In our implementation, we focus on the Stacking technique, where the base classifier results are processed and used as an input to a meta classifier that generates the final prediction. It is recommended to use as many different models as possible. Hence, for our hybrid system, we have used the base learner algorithms as logistic regression, random forest, gradient boosting classifier and Gaussian Naïve Bayes classification methods. Using these models, we create a prediction matrix that corresponds to the predictions generated by each model. We observe that for each dataset the base model with highest accuracy is different. Logistic Regression model performs best for German and Mock datasets, Gradient Boosting Classifier works best for GiveMeSomeCredit dataset and Random Forest performs best for Australian dataset. The training and test set data is divided into 70-30% ratio and the performances of the base learners are verified on the basis of the ROC-AUC score.

Table 3. Validation results for base classifiers.

<table>
<thead>
<tr>
<th></th>
<th>Logistic Regression</th>
<th>Gradient Boosting Classifier</th>
<th>Random Forest</th>
<th>Naïve-Bayes</th>
</tr>
</thead>
<tbody>
<tr>
<td>German</td>
<td>0.865</td>
<td>0.825</td>
<td>0.991</td>
<td>0.812</td>
</tr>
<tr>
<td>Australian</td>
<td>0.913</td>
<td>0.886</td>
<td>1</td>
<td>0.89</td>
</tr>
<tr>
<td>GiveMeSomeCredit</td>
<td>0.85</td>
<td>0.852</td>
<td>0.877</td>
<td>0.858</td>
</tr>
<tr>
<td>Mock Data</td>
<td>0.852</td>
<td>0.847</td>
<td>0.827</td>
<td>0.818</td>
</tr>
</tbody>
</table>

We then define a meta learner that will generate the final prediction. For our research we have re-used the Gaussian Naïve Bayes as a meta learner. Bayesian networks are considered to easily model the complex relationships among the different variables especially if they are discrete in nature. They do not have any requirements on the distribution of the underlying variables as the relationship between variables are explicitly represented by acyclic graphs.

We further split the entire training set into training and prediction set equally for the base learners. Therefore, we have one training set (Xtrain_base, ytrain_base) and another is prediction (Xpred_base ypred_base) which will generate the prediction matrix to be fed as input to the meta learner. Once the base learners are trained and generate the prediction matrix, we feed this prediction matrix to the meta learner (Gaussian Naïve Bayes model) and further train this meta learner. The meta learner thus generates the final prediction see [Figure 3].

---

Until now, we have trained the base learner and the meta learner on only 50% of the dataset, hence a lot of crucial information may be lost. To overcome this loss, we use the K-fold cross-validation method. In this method, the base learners are trained again wherein a copy of the base learner is fitted on K-1 folds thus predicting the left data. For the number of folds specified, the entire process is iterated. It is recommended to keep more number of folds to ensure the entire data is uncaptured. Thus, for each of the 10-fold cross validation the given data set is first partitioned into 10 equal sized sets, then each set is in turn used as the test set while the classifier trains on the other nine sets. This entire process of fitting an ensemble with cross-validation is termed as ‘stacking’. This process helps both the base learners as well as meta learners to train on the complete datasets. It is observed that stacking results in a sizeable improvement in the performance and generates the best score. To measure how well our models perform, we use the ROC-AUC score, which trades off having high precision and high recall see [Figure 4].

4.5 Results and discussion:

In a ROC (Receiver Operating Characteristic) curve, the true positive rate (also termed as Sensitivity) is plotted in a function of the false positive rates (also termed as Specificity). As shown in the figures below, the accuracy of the model is measured by the area under the ROC curve referred to as AUC. In a ROC curve, we plot the ‘True Positives’ on Y-Axis and the ‘False-Positives’ on the X-axis. The ‘True Positives’ are the correctly predicted positive values which means that the value of the actual class and value of predicted class is both yes. ‘False Positives’ are values wherein the actual class is yes, but the predicted class is no. As per definition, an area of 1 represents a perfect test whereas an area of 0.5 represents a worthless test. The more an ROC curve is lifted up and away from the diagonal the better the model is.10 In other words, the greater the

---

10 https://ashokharanl.wordpress.com/2014/03/14/a-very-simple-explanation-for-auc-or-area-under-the-roc-curve/
AUC, the more accurate is our test model. In our analysis of the comparison of the base learners as compared to the ensembles, we were able to achieve the AUC for the ensembles to be closer to the top left-hand borders as possible indicating better the accuracy of the model. See [figures 5-8].

We have plotted the ROC-AUC score of the base learners and the hybrid ensemble as:

![Figure 5. German Data ROC](image)

![Figure 6. Australian Data ROC](image)

![Figure 7. Give Me Some Data ROC](image)

![Figure 8. Mock Data ROC](image)

Now that we have calculated the ROC-AUC, we verify the accuracy of the final hybrid ensemble model by plotting the confusion matrix (also known as the error matrix). It is a specific table layout that allows the visualization of the performance of the hybrid model. In the confusion matrix, each column represents the instances in the predicted class while each row represents the instances in the actual class. The confusion matrix (CM) for each of the datasets using the hybrid ensemble is shown as follows see [Figures 9-12]:

![Confusion Matrix for German Data](image)

![Confusion Matrix for Australian Data](image)

![Confusion Matrix for Give Me Some Data](image)

![Confusion Matrix for Mock Data](image)
Once, we have the visual representation of the confusion matrix, we proceed to calculate the accuracy, precision and recall values for the predictions. Accuracy is the ratio of correctly predicted observation to the total observations. Although accuracy is one of the greatest measures and expected to have the highest value, consideration also needs to be given the symmetry of the datasets. Precision is the ratio of correctly predicted positive observations to the total predicted positive observations. High precision relates to low false positive rate. Recall is the ratio of correctly predicted positive observations to all the observations in the actual class. Accuracy works best if false positives and false negatives have similar cost. If the cost of false positives and false negatives vary, it’s better to look at both Precision and Recall. The cumulative results of the hybrid model for these parameters are shown in the following table:

**Table 4. Performance Parameters for the Hybrid model**

<table>
<thead>
<tr>
<th>Parameters/Model</th>
<th>German</th>
<th>Australian</th>
<th>GiveMeSomeCredit</th>
<th>Mock Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accuracy</td>
<td>0.78</td>
<td>0.87</td>
<td>0.77</td>
<td>0.75</td>
</tr>
<tr>
<td>Recall</td>
<td>0.77</td>
<td>0.97</td>
<td>0.7</td>
<td>0.82</td>
</tr>
<tr>
<td>Precision</td>
<td>0.8</td>
<td>0.82</td>
<td>0.83</td>
<td>0.71</td>
</tr>
</tbody>
</table>
Following table represents the results obtained on different datasets using the base learners and the hybrid ensemble:

### Table 5. Comparison of Results for Base Learners and Hybrid Ensemble.

<table>
<thead>
<tr>
<th>Dataset/Method</th>
<th>Logistic Regression</th>
<th>Gradient Boosting Classifier</th>
<th>Random Forest</th>
<th>Gaussian Naïve Bayes</th>
<th>Hybrid Ensemble</th>
<th>K-fold validation</th>
</tr>
</thead>
<tbody>
<tr>
<td>German</td>
<td>0.825</td>
<td>0.812</td>
<td>0.765</td>
<td>0.768</td>
<td>0.826</td>
<td>0.847</td>
</tr>
<tr>
<td>Australian</td>
<td>0.886</td>
<td>0.89</td>
<td>0.894</td>
<td>0.873</td>
<td>0.898</td>
<td>0.908</td>
</tr>
<tr>
<td>Give Me Some Credit</td>
<td>0.852</td>
<td>0.858</td>
<td>0.835</td>
<td>0.847</td>
<td>0.855</td>
<td>0.861</td>
</tr>
<tr>
<td>Mock Data</td>
<td>0.847</td>
<td>0.818</td>
<td>0.759</td>
<td>0.714</td>
<td>0.848</td>
<td>0.851</td>
</tr>
</tbody>
</table>

As observed, Logistic Regression performs best for German and Mock Datasets, Gradient Boosting Classifier works best for GiveMeSomeCredit credit dataset and Random Forest works best for Australian dataset. The hybrid system seems to outperform the base learners for all the datasets except GiveMeSomeCredit. However, an important point to note here is that the Hybrid system at this stage is trained only on partial dataset, hence we might be losing a great deal of important samples that affect the performance. To overcome this problem, we had implemented the K-fold validation and if we look at the K-fold results, we observe that the hybrid system when trained over the complete dataset outperforms the base learners despite the different datasets. Using both the bagging (Random forest) and boosting (Gradient Boosting classifier), we make our system robust enough to reduce the variance as well as the bias error. This is our desired outcome as we are looking for an ideal classifier system that provides the best results despite the variance of datasets unlike the individual base learner models performance which varies with every dataset.

### 5. Conclusions and future work

Our study proposed the complete procedure for designing a hybrid ensemble for efficient credit scoring analysis. Considering the real-world datasets are made up of inconsistent and uncorrelated data, it is necessary to build an efficient credit scoring system that performs the best irrespective of the nature of datasets that are provided to them. This was the idea of our study, and we were able to train and build an efficient system that gave outperforms the individual best base classifier performance. Data filtering approach help us remove the inaccurate / unrelated samples from our datasets helped the classifiers to distinguish between the classes efficiently and thus define the decision boundaries to be specific. We surveyed a lot of researches that described better ways of preprocessing the data, using different approaches to build the ensemble methods. However, most of them pointed to the concern wherein although ensembles outperform the base learners, it is of crucial importance when deciding with which base learners would build an efficient ensemble. The hybrid ensemble we propose investigated these concerns and was able to achieve the goal of using efficient diverse learning algorithms and being able to provide the best optimum results for each of the applied datasets.
If observed from the ensemble point of view, the Bagging (Random Forest) and Boosting (Gradient Boosting Classifier) perform the best for different sets of data which are Australian and Give Me Some Credit datasets respectively but when applied individually. However, since our stacking ensemble combines these methods, we take advantage of both bagging which reduces variance and boosting which reduces bias error and proves itself to be a ‘champion model’. Hence, our hybrid ensemble of preprocessed data with stacking proves to be a much more robust and accurate system.

We have built our hybrid ensemble using the stacking approach with cross validation which allows both the base and the meta learners to train on the full dataset. Our study into stacking process brought forth some concerns such as computational complexity. The more the base learners, the more efficient the ensemble, however, this could also slow down the analysis significantly. Parallel processing is a suitable solution for this issue but again in parallel processing; we must assign each process its own memory allocation. For bigger the dataset and higher number of base learners, this could mean a great deal of memory consumption and time complexity. This shortcoming can be addressed in the future work.

References:


