DSCI 470 Project 2 Loan Prediction Data Set

Ryan Rosiak and Grant Dawson

October 2021

1 Introduction

Our data set was acquired from "kaggle.com": a machine learning and data science community page where people share data sets with one another. We chose a loan prediction data set. Each instance in the data set is a person with data consisting of loan id, gender, married, dependents, education status, employment status, applicant income, co-applicant income, loan amount applied for, loan amount term, credit history check, property area, and loan status. Essentially, this data is to be looked at from the point of view of a bank. A bank cares about return on investment and must look at these specific factors to assess the risk of an individual. This process of assessing risk and determining loan status has been performed for many years. With data science and computers, we can mitigate the timely manner of assessing risk by hand and automate the task by processing large chunks of data at once. Furthermore, we can use machine learning to further assess risk using the computer rather than an actual person.

2 Attribute Analysis

- Loan_ID - Numerical - Quantitative
  - An arbitrary number that uniquely identifies each instance
- Gender - String - Qualitative
  - The gender that the person identifies with
  - Data: Male/Female/Other
  - 80% Male — 18% Female — 2% Other
- Married - String - Qualitative
  - If the person is married or not
  - Data: Yes/No
  - 65% Yes — 35% No
  - 3 Null values
- Dependents - String - Qualitative
  - The number of dependents a sample has
  - Data: Between 1-3(+) years

  - 56% 0 — 17% 1 — 27% 2-3+
- Education - String - Qualitative
  - Whether or not a sample has graduated high school or not
  - Data: Graduated/Not Graduated
  - 88% Graduated — 22% Not Graduated
- Self Employed - String - Qualitative
  - Is the sample point self employed or not
  - Data: Yes/No
  - 13% Yes — 81% No — 5% Not employed
- Applicant Income - Numeric - Quantitative
  - The Monthly income of the applicant
- Co-Applicant Income - Numeric - Quantitative
  - The monthly income of a co-signer of the loan - 0 if no co-sign
- Loan Amount - Numeric - Quantitative
  - Loan amount in thousands
  - Data: between 9 and 700
- Loan Amount Term - Numeric - Quantitative
  - Term of the loan in months
  - Data: Between 12 and 480
- Credit History - Numeric - Qualitative
  - If the samples credit history meets guidelines
  - Data: 0 or 1 / No or Yes
- Property Area - String - Qualitative
  - Type of area the sample lives in
  - Data: Urban/ Semi-Urban/ Rural
- Loan Status - String - Qualitative
  - Whether the sample was approved or not
  - Data: Y/N

3 Research Questions

Here we have provided some preliminary questions to consider before jumping into the major research question. These will be analyzed in the next section.

1. Does more experience mean higher income?
2. Does geographic location mean higher income?
3. Can you make a model to guess loan acceptance? Can prediction models help?
4. Do certain geographic locations have higher rates of loan acceptance?
5. What is the difference in being considered risk/non-risk as it relates to being married/single?
6. Does marital status have an affect your living situation?
7. Do people who graduate have richer connections?
8. Does marital status determine the range of money that you are applying for?
9. Does education status determine a successful credit history check?
10. Does applicant income determine loan amount applied for?
11. Does property area affect loan status?
12. Is there a correlation between loan amount and loan amount term?

4 Graphs/Analysis

To be able to assess the quality of the data set, there must be some preliminary analysis. There are many different attributes and these are closely analyzed along with relationships between them below. Figure 1 shows us the relationship between a sample applicant's income and their co-applicant's income. We want to determine the relationship between the income of the applicant and the income of the co-applicant who is signing on their behalf for the loan. It would be expected that if you make less money you would need aid in paying for a loan. More specifically, a second income to give the bank security in dispersing loan payments. In figure 1 below, we observe that a majority of people that have a cosigner are lower income, with only a few outliers. This is to be expected and reflects a real world depiction of loan data.

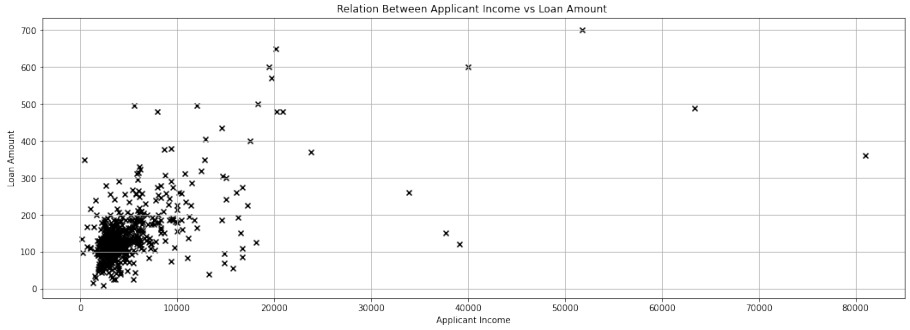

Figure 1: We compare the applicant's income against the Co-Applicant that co-signed on the loan

When looking at applicants, another key relationship to pick up on is how much money the applicant is applying for. A bank is more likely to disperse a loan to a lower income individual if the loan carries less value. A smaller loan amount means less risk that is held over the bank if they were to not get some or all of the money back. Figure 2 displays this relationship perfectly.

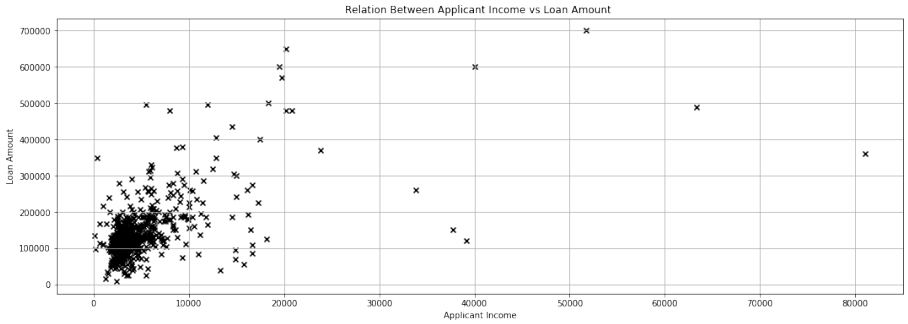

Figure 2: We compare the income of a person to the amount they got in a loan.

The next figure specifically answers question #1 (Questions: 3). It is clear that both graphs are skewed right but there is a much larger tail for graduates. This shows that although the bulk of our data set reflects low income earners, higher income earners are predominantly graduates. Outside of a few non-graduate outliers, the graph clearly shows that an individual tends to have a higher income when graduating from higher education.

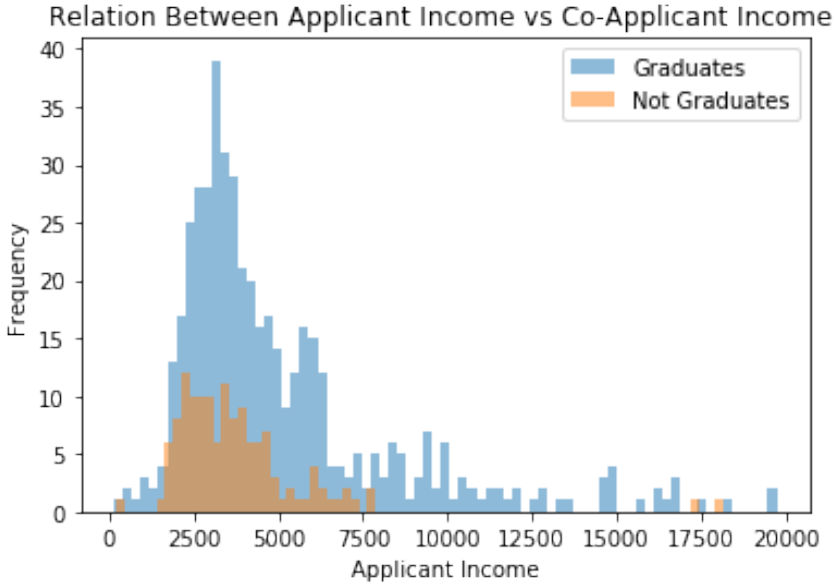

Figure 3: We explore the relationship between Applicant's Incomes and Education

The next graph further answers question #1 (Questions: 3). This alternate graphic shows the spread of income between graduates and non-graduates. Both plots plots are clearly right skewed

but the spread of non-graduates is much smaller and located mainly in the low income bracket in comparison to the much larger spread of graduates. The box and whisker plot gives a better depiction on where the bulk of the data lies. From this, it is even more clear that you tend to have a higher income when you graduate higher education.

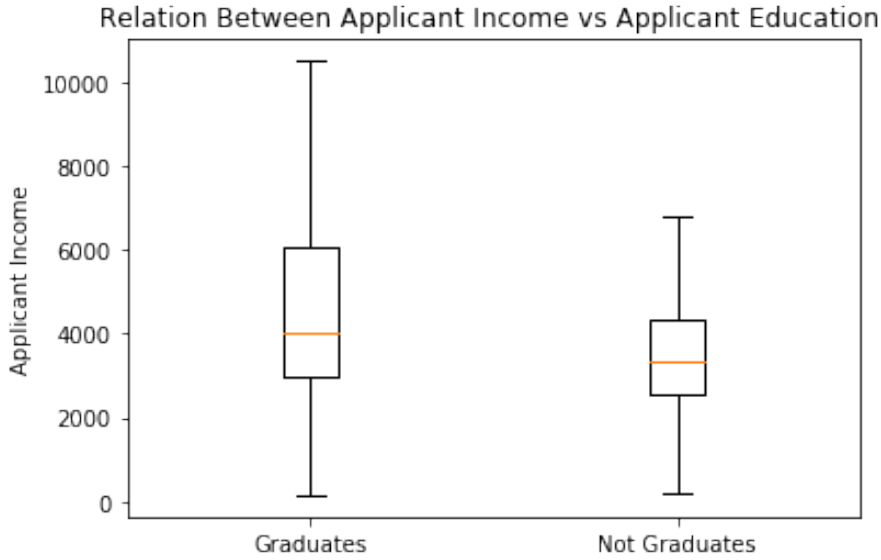

Figure 4: We explore the relationship between Applicant's Incomes and Education

In real life, there are many factors that affect income. Education status is one, but our data set provides many more attributes to draw conclusions about income from. One of these is property area. Figure 5 on the next page aims to answer question #2 (Questions: 3). There are three different property areas that are represented; Urban, Semi-Urban, and Rural. Each property area has a well distributed income that is very similar to one another. This does not give us any clear indication of one property being "better" than another in that one has a clear higher income over another. But, the rural property area does have a slightly more left skew than the others possibly indicating that rural are represents a slightly higher income bracket.

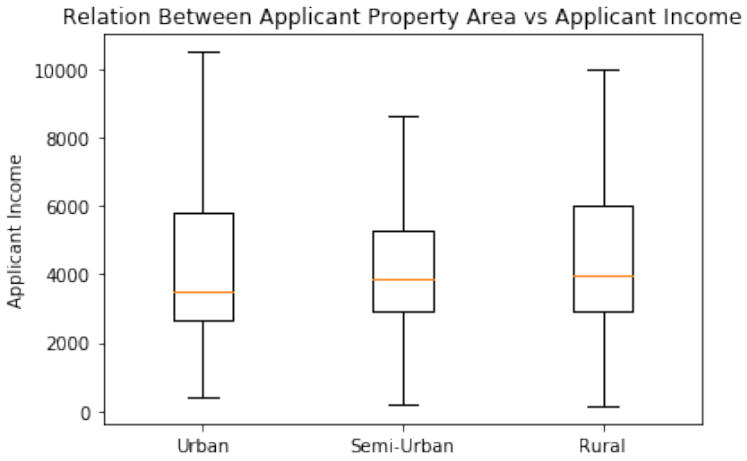

Figure 5: We explore the relationship between Applicant's Incomes and Property Area

Another interesting correlation can be made with the plot below. This plot attempts to answer question #7 (Questions: 3). It is shown that people who have graduated tend to have a more diverse range of co-applicants. Not only is the mean higher for individuals that have graduated but the data is right skewed with a much larger spread. This implicates that people who have graduated have co-applicants with a much higher income. This sort of correlation follows the idea that if you have more money than your family or people that support you generally have more money. This is not true in all cases. But, it is an interesting correlation to analyze.

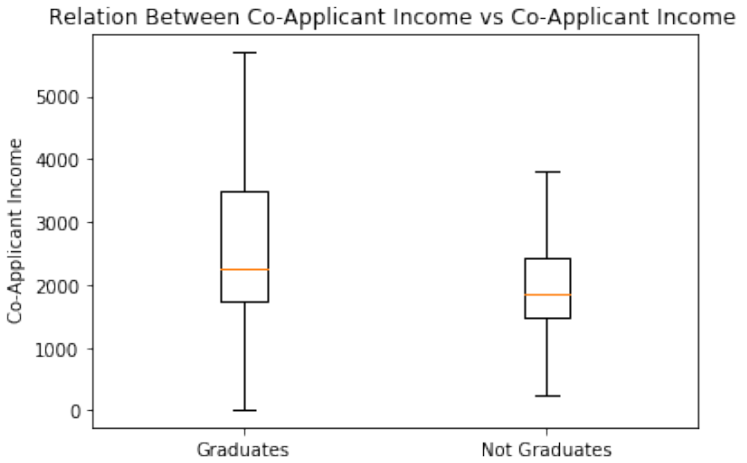

Figure 6: We explore the relationship between Co-Applicant's Incomes and Education

Since the data set contains a wide variety of attributes, it is possible that one plot with a different twist can answer more questions. On the next page, is a graphic that plots applicant income vs loan amount applied for but with a threshold. This is an attempt to separate low income earners from higher income earners. The figure on the next page attempts to answer question #8 (Q: 3).

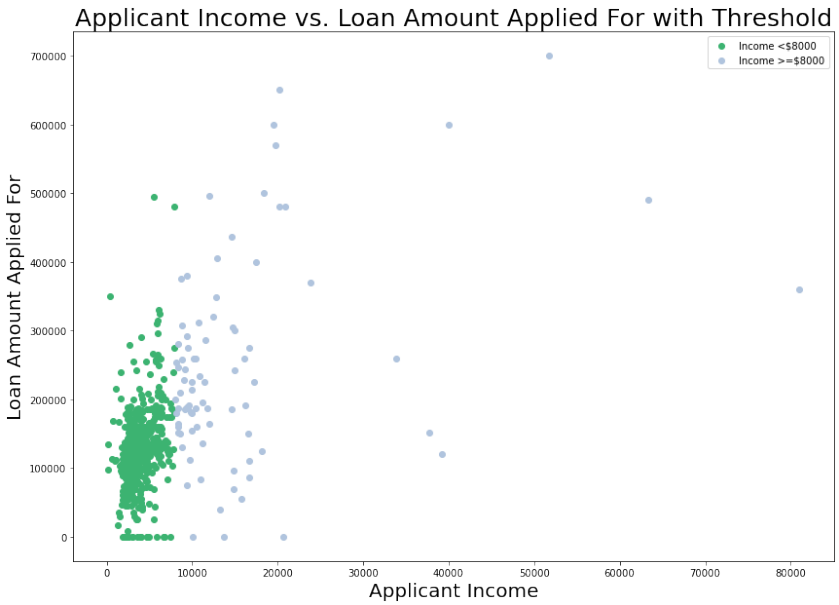

Figure 7: We explore the relationship between Applicant Income and Loan Amount Applied For with an income threshold

From this scatter plot, we can see that the data set consists of a lot of individuals that have a lower income. Specifically, with a threshold applied consisting of individuals who make less than $8000 (green dots) make up a large bulk of the data. This threshold was chosen specifically to determine what individuals are "low income" earners. There is also a weak positive correlation that shows as applicant income increases, so does the amount of the loan that an individual applies for. The low income earners are more than likely applying for smaller loans in the hopes that they are more likely to be approved. These correlations further emphasize figure 1 and 2's conclusions.

So far, a lot of the numerical data has been closely analyzed. However, some non-numeric data analysis can prove to be pivotal in providing a well rounded analysis of the data set. Figure 8 on the next page attempts to answer question #11 (Questions: 3). From this figure, no clear conclusion can be drawn about property status affecting loan status. All property areas are almost equally represented amongst each other. Generally, more than half of the individuals in each property area are getting approved for loans. Although there is some slight variation, the counts are pretty consistent across the board. This lessens the value of the property area attribute for determining loan status due to this lack of variation.

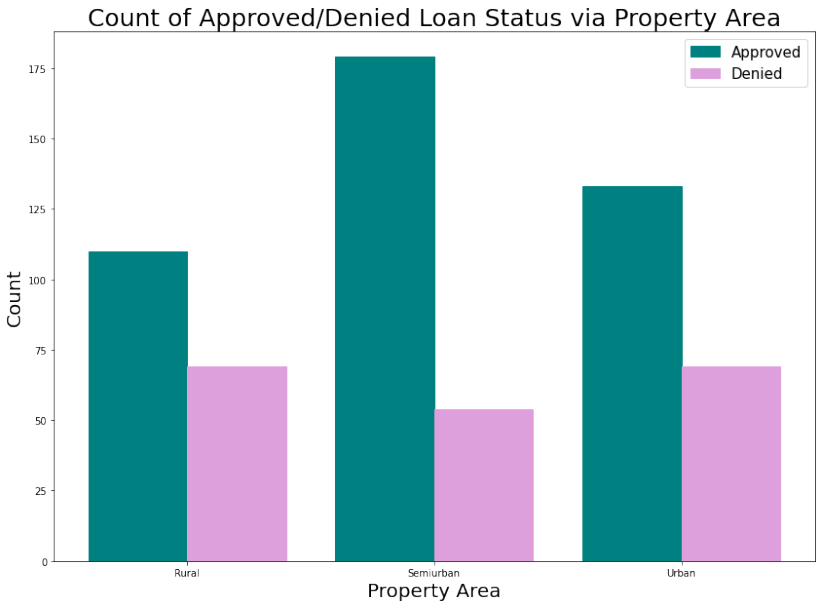

Figure 8: We analyze the count of approved/denied loans based on property area

Another important attribute to look at is education status as it pertains to an individuals credit history check. In reality, most young adults do not have stable credit. This is definitely taken into account by a bank as they know most younger individuals are just starting out. However, this does partially lessen the validity of the credit check. If younger individuals are not held to the same standard as older or non-graduates to graduates, then there might not be any variation as expected. Nevertheless, it would be interesting to see education status affects credit history checks. If an individual passes their credit history check, than that is a huge boon to their application. On the next page, figure 9 attempts to answer question #9 (Questions: 3). As stated above, if an individual passes their credit history check, then they have proof of reliability when determining their loan status. One would naively think that if and individual is a graduate, they are more likely to have a stable job alongside a higher income with more credit eligibility and independence. Therefore, they would be more likely to have a loan approved than someone who is in college because a college student is less likely have a good credit history or one at all. On the other hand, college students who have a small amount of credit may have a greater credit history pass rate because they have less credit that they need to manage due to circumstances of being enrolled in school. The non-graduates could also be individuals who never went to college which could affect the results. While analyzing the graph, it is clear to see that there is only a very slight variation in pass fail credit checks between those who are college graduates and those who are not. The first naive conclusion is favored amongst this variation. However, this does not seem to be enough to show a direct correlation to having better credit history checks for an individual who is a graduate rather than a non-graduate.

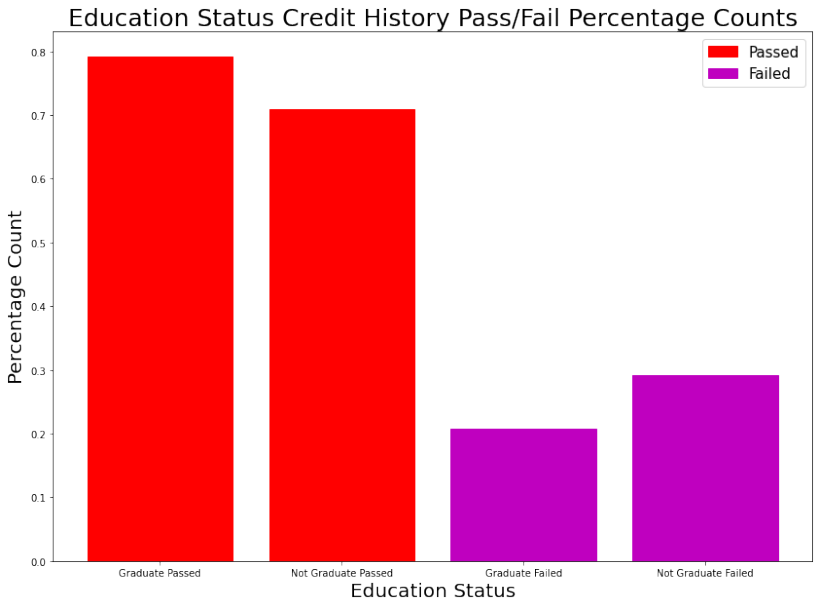

Figure 9: We analyze the count of passed/failed credit history checks based on education status

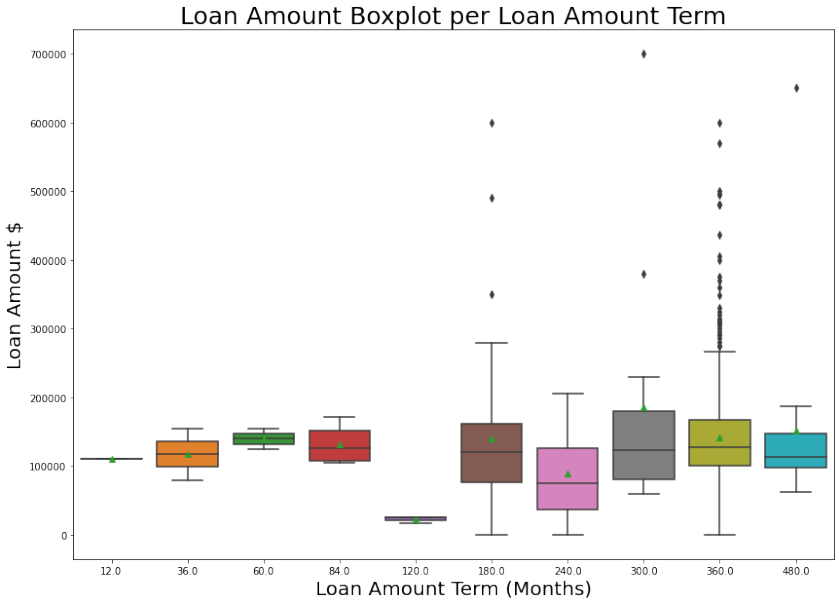

Figure 10: We explore the spread of Loan Amount Data versus Loan Amount Term

Circling back to some more numerical data, looking at the spread of loan amount applied for could further improve some conclusions derived above. Figure 10 above answers question #12 (Questions: 3). These box plots show the spread of loan amount applied for over various loan terms. Almost all

of the loan terms that are represented show roughly their middle 50% of data lies round $1000000. However, as the loan amount term increases, the overall spread of the loan amounts applied for increases. This could be due to individuals wanting various loan amounts at higher loan terms or because lower loan terms are underrepresented in our data set. Regardless, there is a clear increase in the amount of outliers and upper 25% of the data for larger loan terms. This is a reasonable conclusion because if and individual is asking for more money, they are more likely to obtain that over a longer period of time. One key feature pair with this is graduates and non-graduates, depending on where one is in education and income, can determine how much they are applying for.

One last possible correlation to view is the loan amount applied for depending on an individuals marital status. Marital status plays a key role in both how one applies for loans and how much money one applies for. Being married is different stage of life and responsibility. Figure 11 below attempts to answer question #8 (Questions: 3). First and foremost, it is clear to see that the histogram is right skewed. This means that the bulk of our data consists of individuals applying for smaller loan amounts. When looking at the difference between individuals being married or non-married, there is hardly any variation in skew and both statuses are spread almost exactly the same. The only major difference is that married individuals have larger counts than non-married individuals. This can be due to the fact that our data set consists more of non-married individuals. Because of this slight variation, marital status may not play as keen of a role as originally anticipated. The tail of married individuals are clearly applying for higher loans but not by a significant margin.

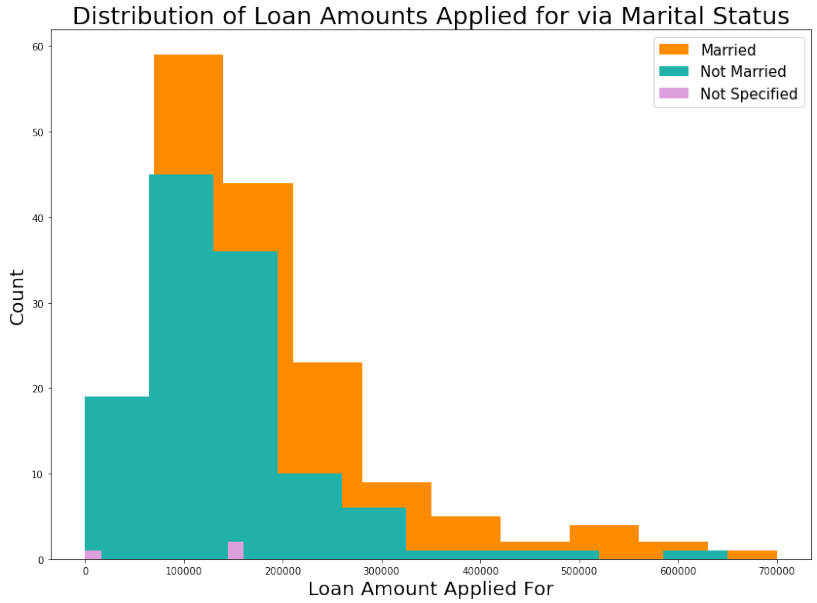

Figure 11: We inspect the relationship between Loan Amounts Applied For and Marital Status

Overall, this loan predication data set proves itself to have valuable information and relationships that can aid our research in determining a loan prediction algorithm. Most relationships that were expected were represented in the data set despite the data set being relatively small in size. This analysis will help determine what attributes will contribute to our algorithms and further push our contributions forward in assessing risk of an individual via various machine learning models.

5 Major Research Question

As stated in the introduction, we want to analyze personal data to assess risk. This can be implemented specifically for a bank or institution. Given our prior analysis, our data set has proven valuable enough to try and run a machine learning model on it. The main question we want answered is; Can I make a machine learning model to predict loan status? This question has a lot of area to cover. But, our data set provides us plenty of information to go off of. In the next two sections, we will use simple logistic regression and k-nearest neighbors classifiers to determine if a machine learning model to predict loan status is possible. Various tools for cross validation and analysis will be used to determine this and will be discussed in the next two sections.

6 Logistic Regression

One of the best yet simple models that is used in classification is logistic regression. Since our problem has only two outcomes, logistic regression is a clear model to use since it works specifically to two outcomes. As explained in "Advantages and disadvantages of logistic regression" [3], a classifier that is tailored to two outcomes is likely to be more accurate than any general purpose classifier. Logistic regression finds the line of best fit on a graph of plotted points using the Sigmoid function for regression. To truly know if this algorithm is worthy to be a classifier for risk, there must be some cross validation. This first set of cross validations can be seen below.

Fold: 3	0.69230769	0.67832168	0.66433566

Fold: 5	0.69767442	0.6744186	0.65116279
	0.63953488	0.65882353	

Fold: 7	0.69354839	0.66129032	0.63934426
0.6557377	0.67213115	0.63934426	0.67213115

The classifier was trained on a standard 70-30 train-test split. The first way the model was evaluated was using simple cross validation using fold sizes of 3, 5, and 7. Cross validation is the act of splitting the training set into n sub-training sets and performing the algorithm on each subset (fold) [2]. Since our training set was rather small, 3 different fold sizes were picked to analyze the quality of the training set. As seen above, the algorithm performed well amongst all tests scoring roughly sub-70% accuracy. Each fold's accuracy scored within 1-5% of each other which is a very good sign that our training set is not being over-fitted.

Cross validation is a good way to get started on determining the validity of the model and the data set it is using. However, there are much more accurate measurements in determining validity [2]. The primary method of model validation is the confusion matrix. Specifically, the precision, recall, and f1-score as it pertains to the confusion matrix. These classification metrics show various rates of good and bad guesses by the model and how they compare to each other. See below the confusion matrix of each of the training folds along with some classification metrics.

Confusion Matrix Fold: 3	
7	133
5	284

Precision Score: 0.6810551558752997
Recall Score: 0.9826989619377162

F1 Score: 0.8045325779036826

Confusion Matrix Fold: 5	
4	136
8	281

Precision Score: 0.6738609112709832
Recall Score: 0.972318339100346
F1 Score: 0.7960339943342777

Confusion Matrix Fold: 7	
2	138
7	282

Precision Score: 0.6714285714285714
Recall Score: 0.9757785467128027
F1 Score: 0.7954866008462623

From here, we look at the confusion matrix and various classification metrics that can be derived from the tables above. As stateed in "Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems" [2], the confusion matrix is a 2x2 matrix, the top left corner being a true negative, the top right corner being a false positive, the bottom left corner being a false negative, and the bottom right corner being a true positive. This matrix gives a great depiction of what the model is consistently guessing and what the model is getting wrong. As seen in all folds, the model is guessing a lot of trues (positives). This means that the model is guessing that many people are deserving of a loan. For almost all of the true predictions, the model was correct. This can be shown in our recall percentage and f1-score being very high. The recall is also known as the true positive rate. Since many true values were guessed to be true and were correct in proportion to the amount of values that were guessed to be negative and were false the recall is almost perfect. The precision is lower because the model was guessing a lot of true and the proportion of true positives to false positives is much smaller. The f1-score favors a high precision and recall. Our f1-score falls right in between our precision and recall. This is to be expected. Overall, the model is predicting that many people should get a loan despite there being a larger amount of individuals who should not be getting a loan based on the training data.

Further analysis of this data can be seen on the next page. An important factor in tuning the model is to look at the threshold parameter as it pertains to the precision and recall. The precision and recall curve on the next page can offer us insight as to where to set the threshold to further improve our model.

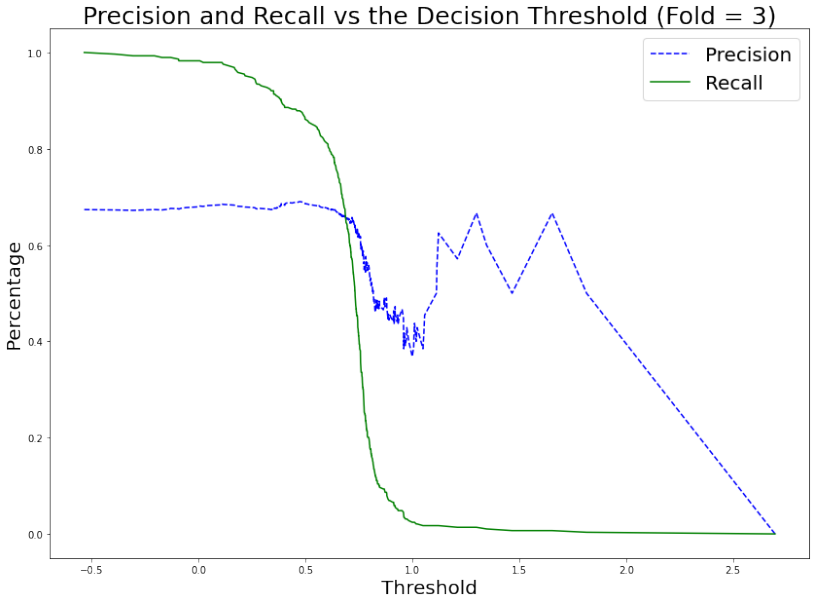

Figure 12: We view the relationship between precision and recall with 3 folds

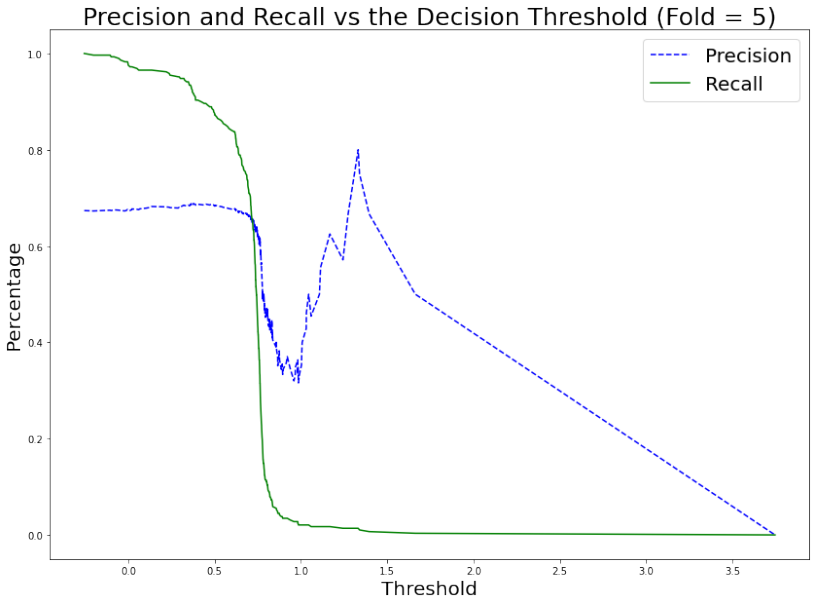

Figure 13: We view the relationship between precision and recall with 5 folds

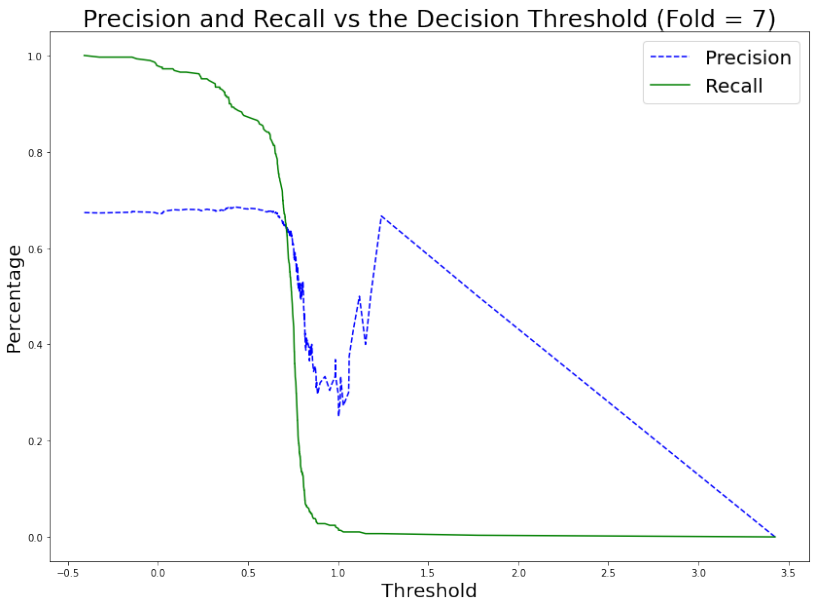

Figure 14: We view the relationship between precision and recall with 7 folds

The 3 plots shown plot the precision vs the recall over various thresholds. All 3 plots should look very similar since cross validation yielded similar accuracy percentages. From the graphs, it is clear that the precision curve is volatile while the recall curve is almost perfect. Since we want the best of both worlds (i.e. both precision and recall to be high) picking a threshold where the two functions meet at their highest point will be an optimal strategy in improving the model.

Another method of validating the model is to look at the receiver operating characteristic curve (ROC curve) along with the ROC area under the curve (ROC AUC) [2]. The ROC curve is a way to measure the recall (true positive rate) vs the false positive rate. Ideally, the recall curve should bow up and to the left as far away from the black line as possible. See the next page for the ROC curve and corresponding ROC AUC scores.

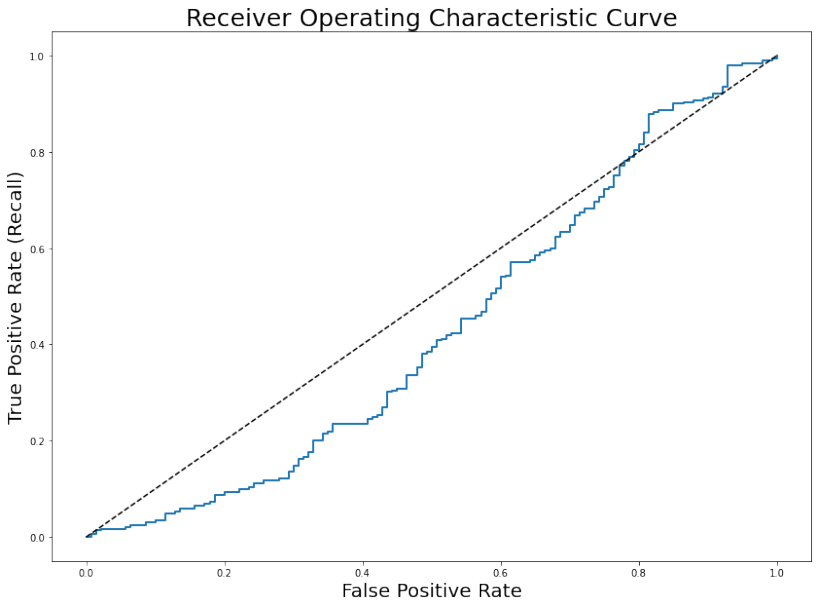

Figure 15: We view the receiver operating characteristic curve showing the relationship between the true positive rate and false positive rate with 3 folds

ROC AUC Score: 0.43084527928818594

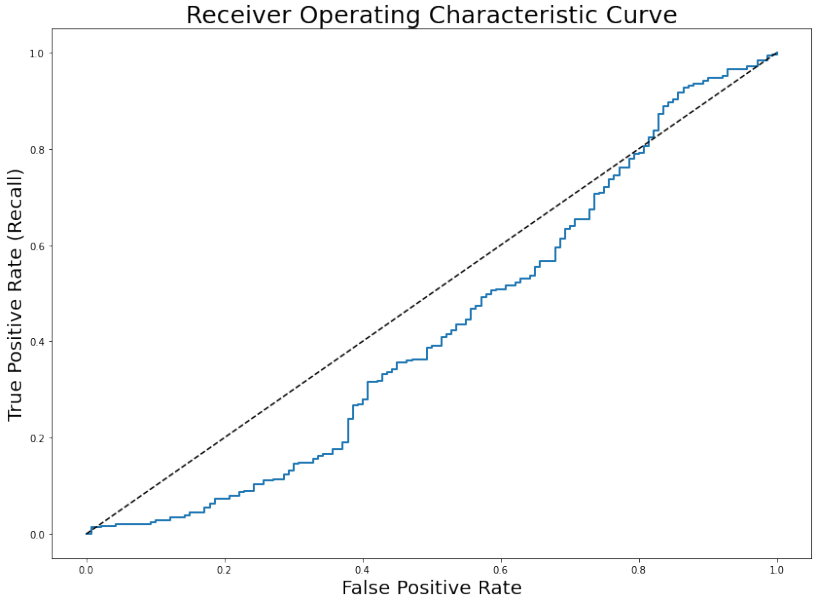

Figure 16: We view the receiver operating characteristic curve showing the relationship between the true positive rate and false positive rate with 5 folds

ROC AUC Score: 0.42587740978744437

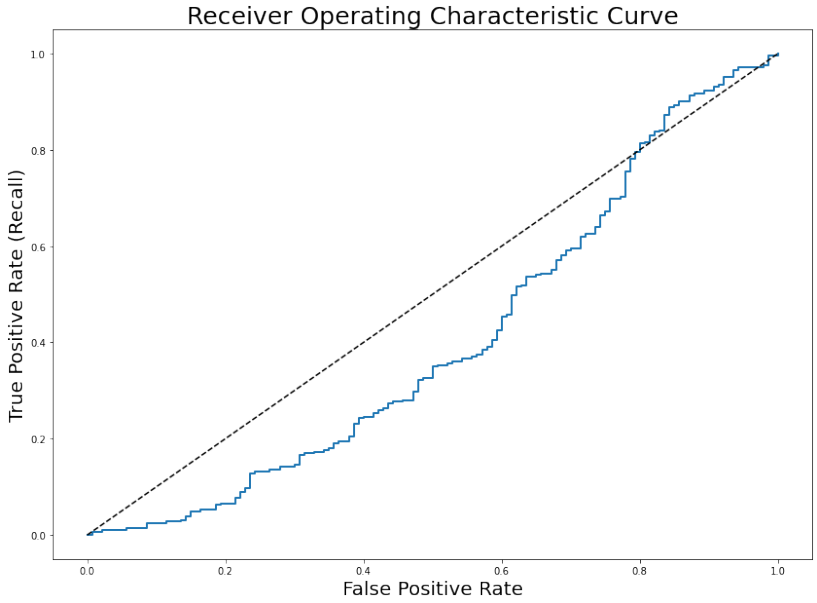

Figure 17: We view the receiver operating characteristic curve showing the relationship between the true positive rate and false positive rate with 7 folds

ROC AUC Score: 0.40548690064261006

These ROC curves are a cause for concern. This is because our ROC curve is favoring a more false positive rate on average. As stated previously, ideally, we want our true positive rate to dominate as the training set increases which would mean the number of true positives is increasing. Also, we want the ROC AUC score to be as close to 1 as possible while favoring the true positive rate side. Although the ROC scores are not great, it is clear that logistic regression still has potential in giving a decent guess at whether an individual is deserving of a loan or not.

To finish off this round of model validation, it is always appropriate to look at the raw accuracy score of the model when predicting on the test set. This score can be seen below.

Accuracy Score: 0.668997668997669

This accuracy score was run on the entire training set and matches that of the cross validation analyzed earlier. This is to be expected but does not provide much more to go off of.

After this major preliminary analysis, it would only make sense to pick a threshold that maximizes the precision and recall from from our precision and recall graph. This can be done but with ill affect. This is because logistic regression chooses the the class that has the highest probability when returning a prediction. In our case of 2 classes, the probability of each class is .5 meaning the "threshold" is .5. This is known because of how the Sigmoid function is analyzed in logistic regression. When the probability is above .5, the classifier returns 1 or yes. When the probability is less than or equal to .5, the classifier returns 0 or no.

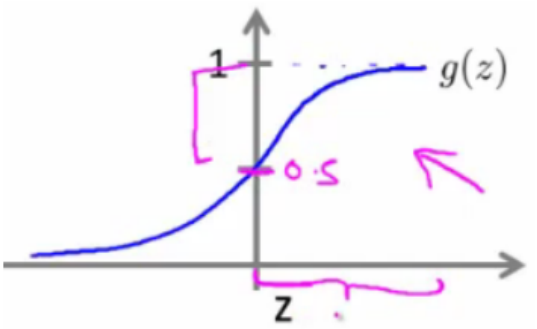

Figure 18: (Geron [2])

Introducing a special threshold would only change the proportion of false positives to false negatives due to the characterization of the regression not changing but the probability at which we select a yes or no is changing. As a result, picking a threshold does not pose any benefit in terms of improving the current logistic regression model. However, there are a couple more parameters that can be tweaked in order to get an overall better accuracy percentage from the model.

Another way that the model could be tweaked is adding more attributes. However, since there are no more attributes in the data set that contain ratio data, there is no way to draw a "numerical" meaning to those attributes, rendering them unusable.

The last way that the model can be tweaked is to change the test/train split. Below, a graph to show the accuracy change over change in test/train split can be analyzed.

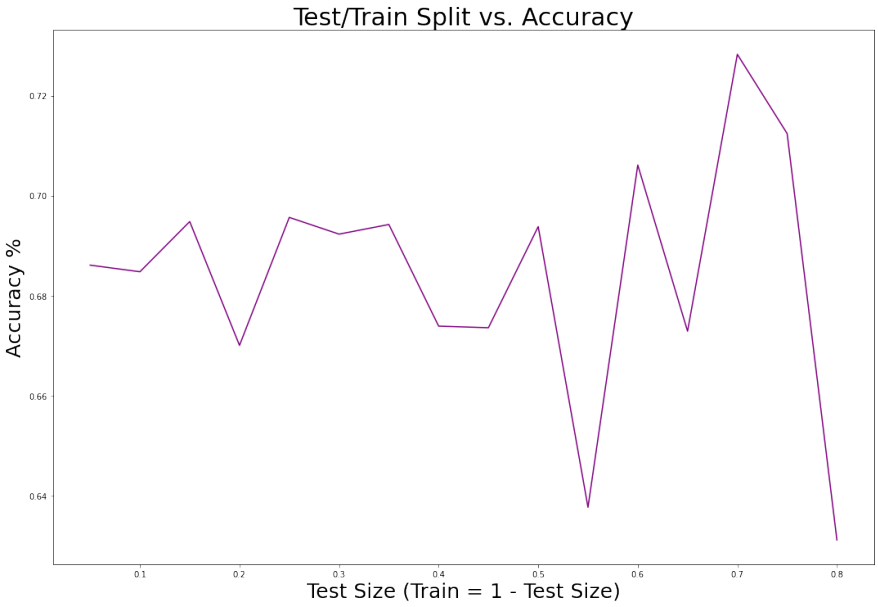

Figure 19: We analyze above the accuracy % change over time as we decrease the training set size and increase the test size

It is clear to see from the graph above that there is no real variation in accuracy percentage as

the training set decreases in size and the amount of test data increases. Little variation is good because there is less of a chance that the model is being over fitted. However, there is not much to go off of because it is not very clear where the best split lies.

Throughout this analysis of logistic regression classifying loan status, it is clear to see that the model performs relatively well based on the the training set given and the attributes provided. The model is very much skewed in the direction of passing individuals who apply for the loan. This can be a good thing because one might want business to continue and individuals to obtain loans. Assessing risk in loan approval must have a limit so that even individuals with unfavorable qualities still have a chance to receive a smaller loan. This model performs relatively well in guessing who deserves a loan but has a rather high rate of false positives. The general consensus on the quality of the model depends on what is more valued. Would a loan prediction model be better that it accepts most but only declines individuals who are clearly not cut out for a loan? Or, that it declines most individuals and only seeks out individuals that provide near perfect credit histories and ideal non-risk characteristics? From our analysis, a model that accepts most and declines the worst is much more intuitive for business. Therefore, this model could definitely be used to predict loan approval. The drawback is that the model does not leave much room for improvement [3]. While on average the model is performing at about 70% accuracy, it would be beneficial to have more parameters to tweak in hopes of increasing the precision while keeping the near perfect recall where it is currently.

7 K Nearest Neighbors

Another great simplistic model that is used in classification is K-Nearest Neighbors. This algorithm works by plotting each instance as a n-dimensional point on a graph. Each dimension is an attribute. Lastly, the algorithm uses one parameter k which is the number of "closest" points to check and vote for what the current point should be classified as. Since we are using this in binary classification, the only options for each point of classification are yes and no. This makes classification of each point very explicit. However, the training data given can highly affect the outcome. Such as, if there is a disproportionately high amount of classifications of yes and a low amount of classifications of no or vice versa. This is one of the drawbacks of this algorithm. Regardless, this algorithm can still provide accurate predictions. On the next page, see some of the outcomes of our tests.

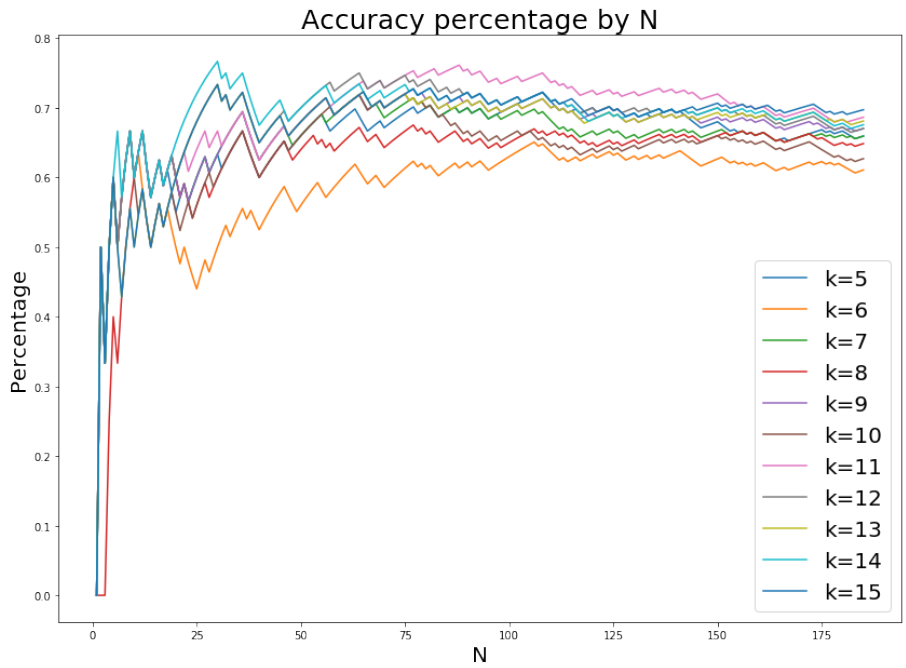

Figure 20: We analyze above the accuracy % change over time as we increase in the number of points tested on a specific KNN trial

K and Final Accuracy	
K=5	0.6590
K=6	0.6108
K=7	0.6595
K=8	0.6486
K=9	0.6108
K=10	0.6270
K=11	0.6865
K=12	0.6703
K=13	0.6811
K=14	0.6757
K=15	0.6973

In the table above there are ten K-Nearest Neighbors trials ranging from k 5 to 15. Each trial was given the same data in the same order. In turn, we can analyze which does the best, worst, and average. It is clear to see that k equaling 15 has the best result for accuracy with the next best being 14 then 9 and so on. The range of accuracy between all the k values tested was less than 10%.

After looking at the initial performance of K-Nearest Neighbors (KNN), we can start discussing the selection process of the parameter k. In KNN, k refers to the classification of a point and how many other points are taken into consideration to classify the newest addition. The underlying functionality of KNN places the new point to be classified into an n-dimensional vector space where each dimension is an attribute used in training. The algorithm then looks at the k closest points and the majority classification of what is presented will be the classification for the new point. However,

this idea of looking at k other points can get quite costly because as one increases the dimension size there are more points to take into consideration for distance. Also, as one increases k there are more points to take into consideration which can affect the outcome.

The next question is where to go from here. The only parameter to look at in KNN is k. Despite this case, there is more that can be analyzed as it pertains to picking k.

If k is too small, this would lead to over-fitting and if k is too large, this will lead to under-fitting. That means that there must be a sweet spot in choosing a k value. In KNN, one wants to have low variance and low bias as described by Tzu-Chi Lin in "Day 3 — K-Nearest Neighbors and bias-variance trade-off", [4]. Tzu-Chi discusses the idea of finding the perfect k value among other things.

Looking back at the table, there is very little difference in accuracy between each k chosen test. As a result, we can say a chosen k value between 9-15 would be adequate. However, there is no way to determine a definite k sweet spot with this given variance. Therefore, we must look further into other metrics to get closer to finding the right k value.

Since this is a classification algorithm, one can look at the accuracy and the confusion matrix to get a better understanding of the performance of the model.

It is important to understand a confusion matrix and it's corresponding classification metrics before looking at the actual data. A high precision means that the ratio of true positives to false positives is large. Precision is solely based on the positive guess ratio which may not contain everything one wants to know about their model. A high recall means that the ratio of true positives to false negatives is large. Recall, aka the true positive rate, has a larger impact on how well a model is performing. In a perfect world, one would want to see their recall and precision to be 1 or as close to 1 as possible. The f1-score is relative to how high precision and recall are to each other. This means that the f1-score is closer to 1 when both recall and precision are closer to 1. This information is reinforced by Harikrishnan [1] and Aurélien [2].

Confusion Matrix Meaning:	
Correctly Deny Loan	Incorrectly Accept Loan
Incorrectly Deny Loan	Correctly Accept Loan

See below the confusion matrices for various chosen k within our "sweet spot" range and some outside.

Confusion Matrix: k=15	
1	55
3	126
Scores:	
Precision Score	0.696
Recall Score	0.977
F1 score	0.813

Confusion Matrix: k=9	
5	51
16	113
Scores	
Precision Score	0.689
Recall Score	0.876
F1 score	0.771

What can be obtained from these tables is that even though each k had similar accuracy, k=9 was guessing false negatives at a much higher rate than k=12.

In conclusion, to determine what k to choose in the end is dependent on what classification metric we are more interested in. This is a much similar situation as our logistic regression model. If we want a higher true positive rate, than we will look for a higher recall. If we want a higher true positive to false positive ratio then we will look for a higher precision. As a result, we need to determine if we want to accept more individuals on the off chance that some with more risk make it through or deny most individuals and only accept individuals with pristine records. As stated in the prior section, from a business point of view, it is better to accept more than accept only a few. Therefore, k-nearest neighbors can also be used as an acceptable model. However, this model also falls into the same trap as logistic regression in that it is very simple and there is not much room for improvement after a sweet spot k has been chosen.

8 Conclusion

Our analysis of simple machine learning models attempting to assess risk for loan prediction would not be complete without analyzing the two side by side. Below, we look at the classification metrics of the best run from both models.

Confusion Matrix Fold: 7	
7	133
5	284
Scores	
Precision Score	0.681
Recall Score	0.983
F1 score	0.805

Table 1: Logistical Regression

Confusion Matrix: K=15	
1	55
3	126
Scores:	
Precision Score	0.696
Recall Score	0.977
F1 score	0.813

Table 2: K Nearest Neighbors

From the tables above, we can see that each model is running into similar issues. Both algorithms have a very high recall. This is good because our true positive rate is very high meaning the proportion of true positives to false negatives is large. However, the precision of both models is lacking in comparison. Despite this, the precision is still relatively high. We are looking at a consistent average of 70% precision and some models can not even feasibly achieve this. But, we want a model that can be as close to perfect as possible. Thankfully, recall is what we want to be higher because a high true positive rate means a higher rate of acceptance over all. We want the higher rate of acceptance from a business perspective because one would rather accept more individuals with some risk rather than only accept individuals that have pristine records. Banks make money off the loans that they disperse when they are receiving money back from the client. Outside of this conclusion, there is not much more that we can go off of because the models are so similar. Based on the classification metrics, it would seem that either model is consistent and accurate enough to be used as a loan prediction model. Sadly, neither algorithm leaves much room for improvement. This also can be a cause for concern because we are currently only using 5 dimensions of attributes. We do not know how more attributes will affect our results. In a more realistic setting,

there are a lot more attributes to consider when assessing risk. In conclusion, these models both can be used as a method to assess risk for loan prediction. However, in a more realistic setting, because there is no substantial analysis on how much higher dimensions affect the model and there is not much room for improvement, we can not conclude that these models would be good enough for loan prediction in a real bank.

9 Annotations

"Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems" is a scholarly text book used for understanding various machine learning concepts and and methodologies [2]. The text book was used for learning ways to cross validate and analyze various classification metrics of classification models. These tools were used primarily in section 6 and 7 of the paper. Some of the tools provided from the book include the precision vs. recall curve and the confusion matrix. These tools were used in determining whether our models could determine loan prediction or not.

"Advantages and disadvantages of logistic regression" is a scholarly online article that consists of reasons to use logistic regression, how logistic regression works, and how to improve logistic regression based on specific data sets [3]. Specifically, the article was used to analyze the validity in running a logistic regression model on our data set and how to improve the model based on the results received after each test run. This article was predominantly used in section 6 of the paper and was a large influence in using logistic regression as one of our chosen models.

"K-nearest neighbors and bias-variance trade off" is a scholarly online article that consists of ways to validate K-nearest neighbors classifier, choose which attributes to include in the model training, and what parameters to tweak to improve the model [4]. This article was predominantly used in section 7 of the paper. The specific tools that were used in this paper are choosing various K's for predictions, finding the sweet spot for K (most accurate), and choosing the correct combination of attributes to include in the training set.

"Confusion Matrix, Accuracy, Precision, Recall, F1 Score" is a scholarly online article that consists of examples and analysis of what a confusion matrix is and what the different scores represent [1]. In this article, we are given the direct equations to derive the different scores and examples of what the confusion matrix represents. This article was mainly used as another source to understand confusion matrices and other classification metrics. Additionally, the source provided enough information for us to do a fair analysis in our conclusion. This source was primarily used in section 6, 7, and 8 of the paper.

References

[1] Harikrishnan N B. Confusion matrix, accuracy, precision, recall, f1 score, Dec 2019.

[2] Aurélien Géron. *Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: Concepts, tools, and techniques to build intelligent systems.* O'Reilly Media, 2019.

[3] Khushnuma Grover. Advantages and disadvantages of logistic regression, Jun 2020.

[4] Tzu-Chi Lin. Day 3 - k-nearest neighbors and bias–variance tradeoff, Dec 2018.