**1. Data Understanding**

The dataset chosen for this assignment is the Home Loan Approval dataset, which is provided by Rushikesh Konapure on Kaggle, a platform widely used by data scientist and machine learning engineers for the purpose of sharing and obtaining datasets for educational and research purposes. The Home Loan Approval dataset can be obtained via the following link:

https://www.kaggle.com/datasets/rishikeshkonapure/home-loan-approval?resource=downloa

d&select=loan\_sanction\_train.csv

There are 2 separate datasets in the format of CSV: the training set and the testing set, where the training set is used to train the machine to predict the approval of a loan and the testing set is to validate the accuracy of the training result. The training set has an extra variable *Loan\_Status* which is the approval state of a home loan (y variable).

However, in this project, only the training dataset file will be used because the testing dataset doesn’t have the y-variable included, hence the testing dataset can’t be used to validate how well or accurate the model can predict the housing loan outcome. A total of 614 rows of data are being used to train and test the model. The dataset will be split into training set (70%) and testing set (30%).

In the attached project folder, there will only be one csv file, which is renamed from loan\_sanction\_train.csv to loan\_sanction.csv.

The Home Loan Approval datasets contain a total of 13 different variables (12 independent variables and 1 dependent variable) which includes nominal, ordinal, ratio and binary attribute types. The variables can be summarized such as the table below:

|  |  |  |  |
| --- | --- | --- | --- |
| **Variable Name** | **Description** | **Attribute Types** | **Values** |
| Loan\_ID | Unique loan identifier for each application | Nominal | Unique Identifier |
| Gender | Gender of the applicant | Binary | Male, Female |
| Married | Marital status of the applicant | Binary | Yes, No |
| Dependents | The number of dependents or children of the applicant | Nominal | 0, 1, 2, 3+ |
| Education | The educational level of the applicant | Binary | Graduate, Not Graduate |
| Self\_Employed | Whether the applicant is self-employed or not. E.g., freelance or own a business | Binary | Yes, No |
| ApplicantIncome | The monthly income / salary of the applicant | Ratio | Numeric values |
| CoapplicantIncome | The monthly income / salary of the co-applicant | Ratio | Numeric values |
| LoanAmount | The amount of loan applied in thousands | Ratio | Numeric values |
| Loan\_Amount\_Term | The term of the loan in months | Ratio | Numeric values |
| Credit\_History | Whether the applicant’s credit history meets the terms and conditions | Binary | 1, 0 |
| Property\_Area | The property location of the applicant | Nominal | Rural, Semiurban, Urban |
| Loan\_Status  (y-variable) | The approval status of the loan | Binary | Y, N |

Binary attributes are variables that only has 2 possible values, usually represented by 1 and 0. In this case, the variable *Gender, Married* and *Education* can arguably consider as binary attributes as each of them are only represented by 2 values in the dataset provided. However, in real life situation, 3 of the attributes mentioned can have more than 2 values representing them, hence they can be considered as nominal attribute too. E.g., *Married* or more commonly known as marital status can be represented with Widow, Married, Divorce and Single. *Education* can be represented with values such as PhD, Bachelor of Degree, Diploma, etc. In this case, they’re all considered binary.

Potential issues or limitations

There are a few potential issues or limitations with the Home Loan Approval datasets:

1. **Incomplete data / Missing values**

The dataset has some missing values in some columns such as Gender, LoanAmount, Credit\_History, etc. This will affect the overall accuracy of the predictive model. For example, without knowing the exact loan amount, the predictive model may not be able to predict the approval of a loan application accurately as it may be one of the important factors affecting the decision. Hence, data preprocessing is required before training the predictive model.

A picture containing application

Description automatically generated

1. **Outliers**

Chart, box and whisker chart

Description automatically generated

As shown in the figure above, the loan amount attribute contains multiple outliers that may potentially reduce the accuracy of the prediction model. Outliers are values or data that fall outside the range, being overly high or overly low compared to other values in the dataset. They can potentially affect the accuracy of the prediction as they may affect the variance and they may skew the mean value, making an inaccurate representation of the central tendency. Besides that, by having outlier values in the training datasets, the predictive model may not be able to predict other or new data sets accurately as they may not be able to adapt to the new data that may not have outliers.

1. **Dataset may be outdated or inconsistent**

The dataset may not be up to date to the current timestamp. Assuming if the dataset is collected within a long period of time, the first and last data collected may have a significant difference due to the difference of environment. For example, the difference in currency. Assuming the data is collected within a period of 5 years, there may be a possibility that 1 USD of 2018 has more purchasing power than 1 USD in 2023. The applicant income in 2018 may have a significant difference compared to the applicant income in 2023 if the inflation rate or any other factors affected the strength of the currency.

Over time, the way that bank approves housing loan may change too. They may consider certain attributes more important than others. The same value of independent variables may result in different outcome due to this, which can cause inconsistency in the datasets, if the datasets contain data that are collected before and after the change occurs.

1. **Dataset has insufficient information**

The dataset may have insufficient information, or it does not have enough specific information which may affects the accuracy of the predictive model trained. For example, *Credit\_History*. Although some applicants may not meet the terms and condition with their credit history, but it is not specific enough telling why they do not meet the criteria. Even some minor factors may affect the credit history to meet or not meet the requirements. Besides that, the dataset does not include the interest rate of the loan. Although some applicants may have the criteria and capability of getting a home loan approval, the interest rate may affect the decision. If the interest rate is too low or too high, it may affect the risk assessment such as whether the applicant is able to pay back the loan after a certain period of time or whether the loan approval is worth the time based on the interest rate.

1. **Lacking attributes + complexity of loan approval consideration**

Housing loan approval process can be complex and blurry to those who are not working in the industry. We might never know how they are even approved, whether manually by human or automated using some sort of algorithm. There might be a lot more factors that are taken into consideration such as social status, age of property, age of applicant, employment history etc., which are not included in this dataset.

1. **Categorical Variables**

Most of the variables in the Home Loan Approval dataset are categorical variables. For example, *Gender* variables has the value of Male and Female. These values will not be understood by the machine learning model. Hence, encoding process is required for the dataset to turn the categorical variables into numerical variables for the machine to understand and to analyze.

The Home Loan Approval dataset is very useful in various ways as it provides valuable information to the machine learning model on how to analyze and how to predict the home loan approval. With the proper tools, we can use the dataset to do multiple research and analysis such as what are the factors that affects a home loan application hence allowing applicants to pay more attention when applying for one.

However, a predictive model has its own limitation as there may also be situation whereby an application is approved or rejected wrongly, making it to have pros and cons. With the proper steps and consideration taken, we believe these cons can be minimized.

**2. Data Description**

Number of unique values

We can use the .value\_counts() to show out the counts for each unique value in the columns.

A picture containing graphical user interface

Description automatically generated

Mean and standard deviation

We use the .describe() function to show the mean, standard deviation, max and min of the Applicantincome, Coapplicantincome, LoanAmount and Loan\_Amount\_Term attributes.

A picture containing graphical user interface

Description automatically generated

The average applicant income is around $5403.46 monthly and coapplicant income is around $1621.25 monthly. The average loan amount is $146,410 and the average loan amount term is 342 months, which is 28.5 years.

One interesting observation that we can see from the standard deviation values is that they are quite large. For applicant income and coapplicant income, the standard deviation is even larger than the mean value. Hence, we can say that the range and distribution of applicant income and coapplicant income is very large and not close to the mean value. A more spread-out data could potentially be harder to work with since it is more prone to outliers and extreme values. Hence, data preprocessing is important in cleaning up the data before using it to train the models.

Heatmap

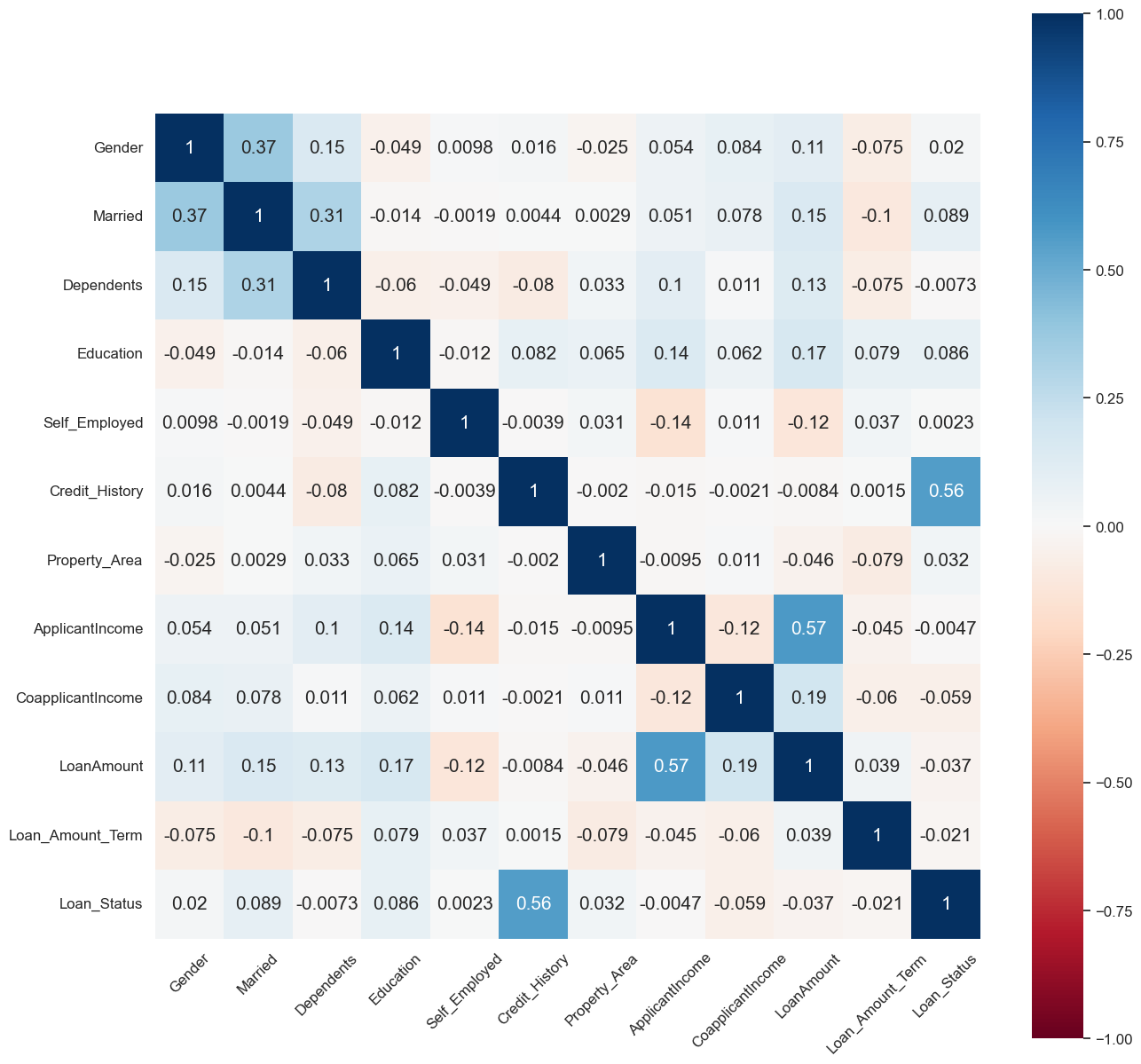
Before building the heatmap, we need to convert all nominal attributes to numeric first, since correlation can only be calculated using numerical values. This is done by .map() function in Python.

This heatmap is created to find the correlation between all attributes. This heatmap is created using sb.heatmap() and corr() and the attributes in the heatmap are set in the matrix plot df[[]] and placed infront of corr(). The corelation scale scale bar with the range of -1 to 1 is displayed by using vmin=-1 and vmax=1, center =0.

The color of the heatmap is set to red and blue using cmap = RdBu. Red represents negative correlation while blue represents positive correlation. “Square” is set to True to make the matrix plot to be square. The anot = “True” is also used to enable annotation and the size of the annotation is set to 15 with annot\_kws{“size: 15”}. The set\_xticktables is used to arrange the xtick labels of the heatmap and rotated 45 degree so that we can look at it easily. The horizontal alignment is set to center.

Text

Description automatically generated



From the heatmap above, we have selected to study a few correlations that have the highest correlation score.

1. **Applicant income – loan amount (0.57)**

This indicates a strong positive relationship between the two. If applicant income increases, the loan amount increases too. This makes sense as a person who has higher income tends to buy property that is more expensive too since he can afford it.

1. **Credit history – loan status (0.56)**

This indicates a strong positive relationship between the two. This also shows that credit history is the main consideration for housing loan approval. Using domain knowledge, we know that credit history refers to a person’s track record in paying back the money that he owes. A person with good credit history is known as a credible and trustable person, hence banks are more likely to approve his housing loan.

1. **Married – gender (0.37)**

There is a moderate positive relationship between the two. A person who is married is more likely to be a male, and vice versa. There aren’t many insights to gain from here, it’s just an interesting pattern that is observed.

1. **Married – dependents (0.31)**

There is a moderate positive relationship between the two. A person who is married is more likely to have dependents. This makes sense because in real life, most of those who have dependents or children are those who are already married.

We have also studied some attributes that have low correlation scores.

1. **Loan amount term – credit history (0.0015)**

There is a very weak positive relationship between the two, which means that the loan amount term is not affected by credit history or vice versa. The two can be said to be independent of each other.

1. **Self-employed – married (-0.0019)**

There is a very weak negative relationship between the two, which indicates that the marital status of a person is not determined by whether a person is self-employed or not, or vice versa.

Conclusion

From this data description, we have learnt that there is a large range of values, and potentially some outliers in the applicant income and coapplicant income attributes. The attribute that correlates the strongest or potentially affect the outcome of housing loan approval the most is credit history, or more specifically, whether the credit history meets the guidelines or requirements.

**3. Data Preprocessing**

**Sampling Techniques**

The dataset is obtained from Kaggle. A total of 614 rows of data is obtained. However, on Kaggle, there is no mention of how the data is sampled. Hence, we assumed that simple random sampling is used to gather the data.

**Drop attributes**

The first step of data cleaning is to determine which attributes to use and which to drop. Since there are a lot of attributes, we need to select only the useful attributes. A dataset with a lot of attributes means higher dimensionality and can lead to higher complexity, overfitting, and the curse of dimensionality.

The first method is to use correlation matrix. From there, we can see which attributes relates more strongly to the result, which is the loan status.



From the correlation matrix, we can see that credit history has the strongest correlation with loan status (0.56), hence we need to keep it. Next, we can see that loan amount and applicant income has a strong correlation too (0.57), therefore we can just keep one of them. In this case, we chose to keep the loan amount and drop applicant income as it has lower correlation with loan status.

The correlation of other attributes with loan status is quite similar (all are not too high) and it is harder to determine which attributes to drop. We can’t just drop them all since we believe they contribute to the outcome of loan status collectively. We dropped the attribute that has the lowest correlation among them all, which is self\_employed (-0.002).

Using domain knowledge, we also dropped loan\_id, which we know is useless in determining the outcome. Hence, the remaining attributes are:

Gender, Married, Dependents, Education, CoapplicantIncome, LoanAmount, Loan\_Amount\_Term, Credit\_History, Property\_Area, Loan\_Status

**Fill in missing values**

Using isnull(), we can see that some attributes have some missing values. Instead of deleting those rows of values, we can fill the null values with either mean or mode values.

Text

Description automatically generated with medium confidence

For categorical values (gender, married, dependents, loan\_amount\_term, credit\_history), we replaced the null values with mode values. Loan\_amount\_term is treated as categorical values here because although the values are numeric, it appears in a categorical manner. There are no floating-point values. By using domain knowledge, we know that loan amount term is represented in months and in real life, there is a set options of how long the term can be.

E.g 360, 240, 120 months.

Next, for loan\_amount, we replaced the null values with mean values.

**Identify and remove outliers**

Outliers are data that are significantly different than others and need to be removed to produce more accurate predictions using the data.

Particularly, we will be analyzing loan amount and coapplicant income to determine the outliers and remove them. We will be using histogram and boxplot to understand its distribution and the number of outliers.

Chart, histogram

Description automatically generatedBefore

Chart, histogram

Description automatically generated

Chart, box and whisker chart

Description automatically generatedChart, box and whisker chart

Description automatically generated

The distribution of both loan amount and coapplicant income is heavily skewed and there are a lot of outliers present in both attributes. We shall run the code for removing outliers based on threshold.

After 1st iteration

Chart, histogram

Description automatically generatedChart, box and whisker chart

Description automatically generatedChart, box and whisker chart

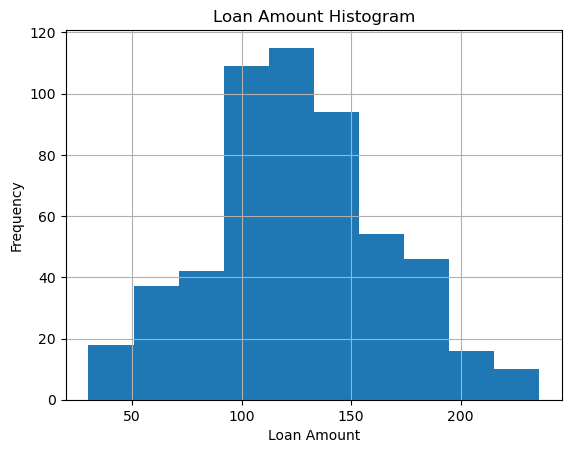
Description automatically generatedChart, histogram

Description automatically generated

There are still some outliers, but it is much better than before the first iteration. Iterative outlier removal, which is the repeated process of recalculating the mean, standard deviation and threshold values to remove outliers, can be carried out. More outliers will be removed for every iteration, however there is a risk of overfitting, so it is best to be careful not to iterate for too many times. In this case, we’ll run the code for the second time for loan amount only since there are still quite some outliers present.

After 2nd Iteration

Chart, box and whisker chart

Description automatically generated

The distribution looks much better now, and the presence of outliers is minimal. Hence, we can proceed to the next step. The number of data rows have been reduced from 614 to 541.

**Encoding Categorical Variables**

We are using the get\_dummies() method in the Pandas library to convert categorical variables into indicator variables using one-hot encoding. The first step is to split the dataframe into X and y. Then, we selected the columns that contains categorical or object values. Those columns are:

Gender, Married, Dependents, Education, Credit\_History, Property\_Area

The categorical columns are then being processed using get\_dummies() method. Taking the example of gender, it is split into gender\_female and gender\_male. They are mutually exclusive. If a person is female, gender\_female will have the value of 1 while gender\_male is 0.

A picture containing text, plaque, screenshot

Description automatically generated

**Normalization**

The last step of data pre-processing is normalization. Normalization is the process of transforming values into a range that is similar or small, to make the prediction and calculation easier and faster. It is also important to ensure that no attribute dominates over others.

The scaling method we used is MinMaxScaler, which is included in the SkLearn library. The columns that require scaling is LoanAmount, CoapplicantIncome and Loan\_Amount\_Term, because their value is very large.

Text

Description automatically generated with medium confidence

Graphical user interface, text

Description automatically generated with medium confidenceAfter normalization, the values fall between the range of 0 and 1, which is much smaller and similar with other columns’ values. The values are also rounded to 4 decimal places.

**4. Visualization**

In this section, we will visualize the data and its attributes using appropriate graphs, with brief explanation, to provide a summary of the data.

Visualization of nominal / binary attributes

1. Chart, bar chart

   Description automatically generatedLoan Status (y-variable)

Graphical user interface, text, application, email

Description automatically generated

We can see that out of a total of 614 loan application, around 69% (422) was approved while the remaining 31% (192) was rejected.

1. Gender

Chart, bar chart

Description automatically generatedGraphical user interface, text

Description automatically generated

We can see that majority of the loan applicants are male, with 81% (489) of them being male and 19% (112) being female.

1. Marital status

Chart, bar chart

Description automatically generatedGraphical user interface, text

Description automatically generated

We can see that among the 611 of the loan applicants 65% (398) was married and 35% (213) was unmarried.

1. Chart

   Description automatically generatedDependents

Chart

Description automatically generated

Majority of the loan applicants has 0 dependents, with only a few that have 3 or more dependents.

1. Education

Chart, bar chart

Description automatically generatedGraphical user interface, application

Description automatically generated

Majority of the loan applicant were graduated (346 more than the number of loan applicants who are not graduated).

1. Self-employed

Chart, bar chart

Description automatically generatedGraphical user interface, text, application

Description automatically generated

Majority of the loan applicants were self-employed (418 more than the number of loan applicants that are not self-employed).

1. Graphical user interface, text, application

   Description automatically generatedCredit History

Chart, bar chart

Description automatically generated

Majority of loan applicants had credit history that met the guidelines or requirements.

1. Property Area

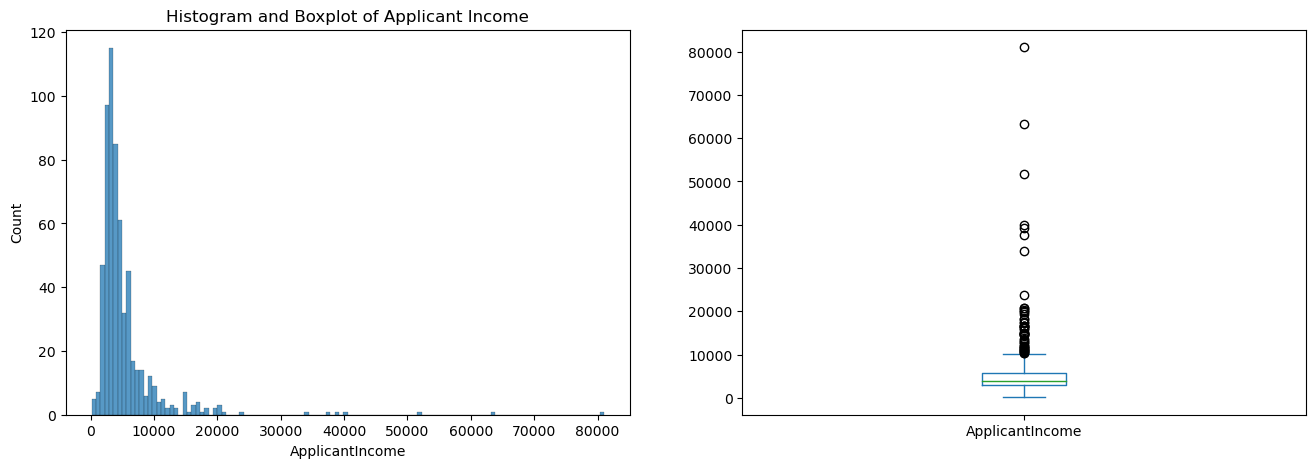
Chart, bar chart

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generated

Semi-urban area has the highest number of loan applicants while rural areas have the lowest number of applicants.

Visualization of numeric attributes

1. Applicant income

Both the histogram and boxplot showed that the range is large and the distribution is heavily skewed to the left. There are quite a number of outliers that exceeded the upper range too.

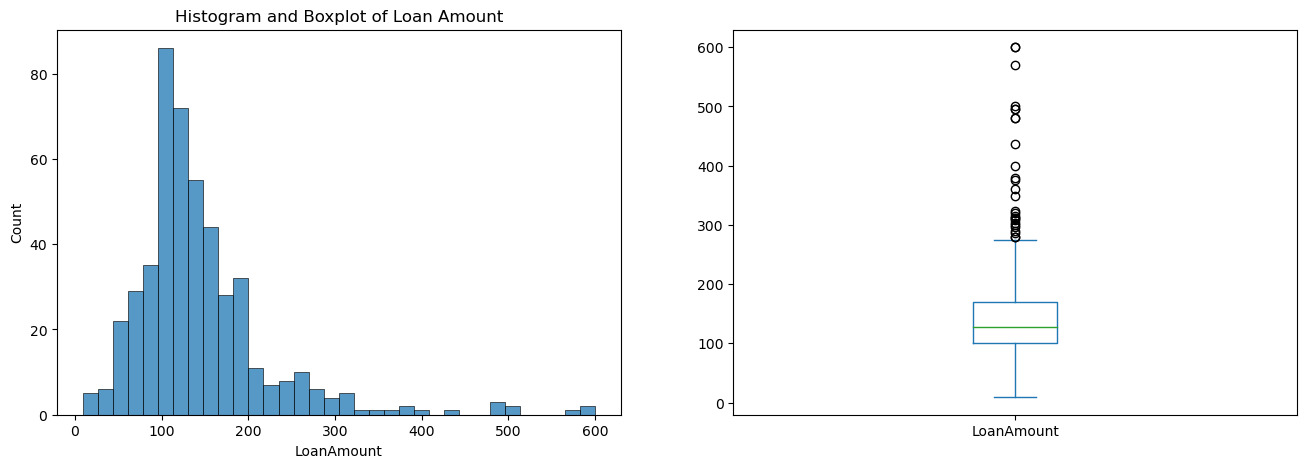
1. Coapplicant income

Chart

Description automatically generated

Coapplicant income has a similar situation as applicant income, where the distribution is skewed to the left and there are quite a number of outliers that exceeded the upper range.

1. Loan amount



Once again, loan amount has a similar situation as applicant income, where the distribution is skewed to the left and there are quite a number of outliers that exceeded the upper range.

1. Chart

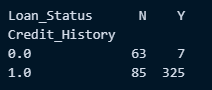
   Description automatically generatedLoan amount term

The distribution is skewed to the right. The mode value, which is 360 months is much higher than the other loan amount term values. However, loan amount term acts like a nominal attribute here, since in the application process, there is a set option of what the applicant can choose from. Although it contains numeric data, there are no continuous values.

Visualization of combination of attributes

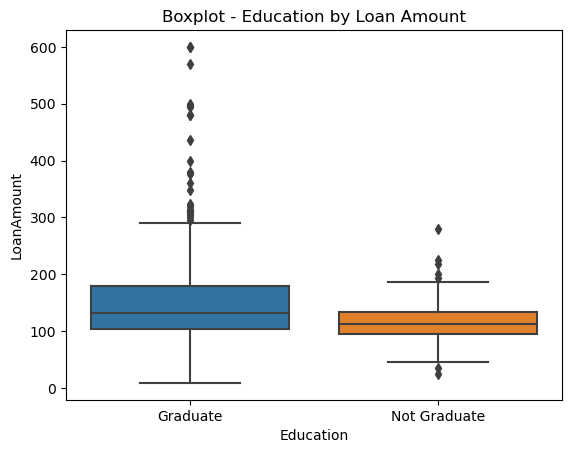
We have selected several sets of attributes to visualize and study, ones that have helped us to gain insight into the dataset.

1. Loan Status and Credit History

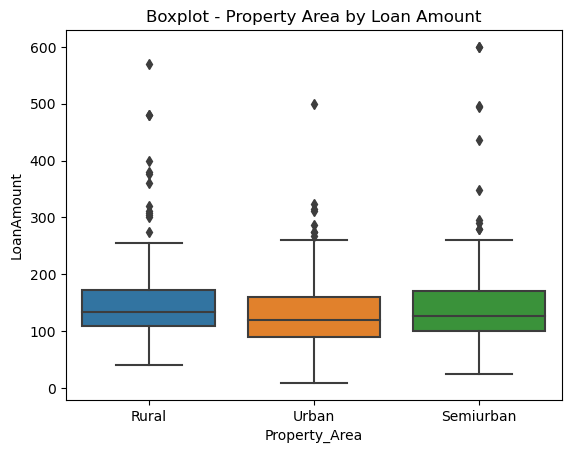
Chart, bar chart

Description automatically generated

A stacked bar chart is used to visualize the proportion of approved and rejected loan application based on credit history alone. For those who have a credit history that meet the guidelines or requirements, 97.89% of them are approved. For those whose credit history doesn’t meet the guideline, 90% of them are rejected. From here, we can deduce that credit history could be one of the important attributes in the determination of housing loan approval.

1. Education and Loan Amount

For those who are graduate, their loan amount covers a larger range, lower min value and higher max value. There are also a lot of outliers, who applied for loan amount that is much higher than the normal range. For non-graduate, there are also some outliers, but not as much as graduates. Overall, we can say that graduate apply for a higher loan amount than non-graduates.

1. Property area and loan amount

One interesting observation that we can see here is the even range of loan amount across all 3 property areas. We would assume that those who apply for housing loan for property in urban area will apply for a higher loan amount, but that wasn’t the case here. Even more interestingly, housing loan for rural areas has higher min value than urban or semiurban areas. Semiurban housing loans also have more upper outliers, compared to rural and urban properties.

However, housing loan amount doesn’t always directly relate to housing price. Some who have higher income might choose to pay more in down payment and apply for housing loan for the remaining amount.

(For a full list of visualization, please refer to Data Visualization.ipynb)

**5. Research Questions**

Here are the research questions that our team has decided to go with for this data mining project:

1. Main objective - Given gender, marital status, number of dependents, education status, property area, presence of credit history, coapplicant income, loan amount and loan amount term, how accurately can we predict the outcome of loan status?
2. How does k-fold cross validation affect the performance and prediction accuracy of models?
3. Relating to social injustice issues, do gender, marital status and education play a part in affecting the outcome of loan status?
4. Which attributes relate strongly to the outcome of loan status?

**Question 1**

*Main objective - Given gender, marital status, number of dependents, education status, property area, presence of credit history, coapplicant income, loan amount and loan amount term, how accurately can we predict the outcome of loan status?*

Our main objective of this data mining project is to determine the outcome of housing loan approval using the following attributes:

* Gender
* Marital Status
* Number of dependents
* Education status
* Property area
* Credit history
* Coapplicant income
* Loan amount
* Loan amount term

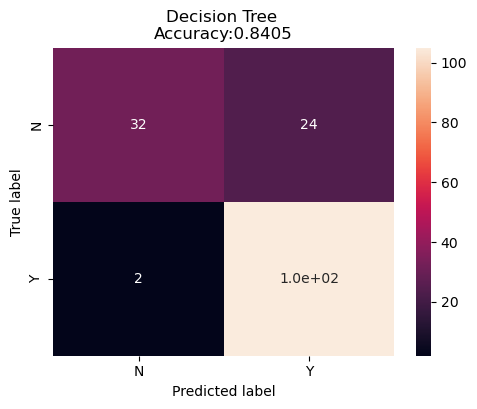
We have used 3 prediction models to predict the outcome (loan status) using the above attributes. The models are decision tree, Naïve Bayes classification and neural network. Different models work better in different cases. Therefore, we have implemented 3 models to determine which prediction model is better at predicting the results based on this dataset.

Decision Tree

Decision tree works by having a root node, then splitting it into two nodes, which can be either terminal node or decision node. Decision nodes will split until it reaches a terminal node, or the maximum depth is reached. Each decision node will split based on the condition of one of the attributes. The attribute is selected based on attribute selection measure.

Diagram

Description automatically generated with medium confidenceIn our case, we have selected to use entropy to calculate the impurity, with maximum depth of 3. We have selected these measures since they produce the best results, after trial and error with different values.

****

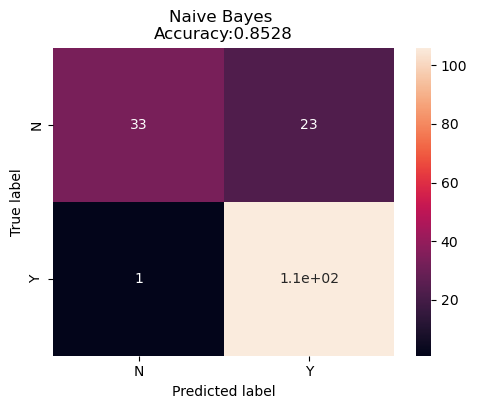
Based on the confusion matrix, we can see that the accuracy is quite high at 0.84. The confusion matrix also shows that there is a high percentage of true positive. The weakness of this model is at predicting negative cases, where the ratio of true negative and false negative is quite similar (true negative: 32 instances, false negative: 24 instances). If the model can predict the negative values accurately, we foresee this model having extremely high accuracy.

Naïve Bayes Classification

Naïve Bayes is another simple classification algorithm that we have decided to use. Naïve Bayes classifier works based on the Bayes Theorem which calculates the probability of an event based on a condition. Its advantage is that it is easy to implement, and it produces good results most of the time, which is true in this case.

A screenshot of a computer

Description automatically generated with medium confidence



The model accuracy is 0.853 which is slightly higher than the decision tree model. However, it has the same weakness as the decision tree model which is low accuracy at predicting negative cases. It can be seen that the ratio of true negative and false negative is quite similar in this case too. The recall value for case = 0 is also low, at just 0.59. Recall value is the percentage of actual positives that are predicted as positive.

Both models have low accuracy at predicting negative values. It can be a coincidence that both models have the same weakness, but it is more likely that the problem occurs within the dataset itself. One of the insights that we have gained throughout this project is the impact of outliers in affecting our prediction result (explained in a later part). Therefore, we can deduce that there is a high chance that there are quite a number of outliers for the negative result (loan status = no) in the dataset.

Neural Network

Neural Network is another machine learning algorithm that is more advanced than decision tree and Naïve Bayes classifier. The way it works is similar to how human brain works which consists of neurons. The overall structure is one input layer where the number of neurons usually depends on how many attributes there are, a number of hidden layers with a number of neurons and an output layer where the number of neurons is dependent on the number of possible outcomes or classes.

Its strength is its ability to backpropagates to reduce the mean squared error to get more accurate results. Every time backpropagation occurs, each neuron’s weights and bias are adjusted, and data is feed into it again. The process repeats for a number of times which is set by the user. However, its disadvantage is that sometimes it is hard to determine how many hidden layers and neurons to use and it’s difficult to understand what is going on in the hidden layers.

Text

Description automatically generatedBelow is the code of the neural network structure:

The model consists of 3 layers, one input layer, one hidden layer and one output layer. The input layer has 20 neurons, with 18 input features, and is using the activation function of rectified linear unit (relu). Usually, the number of input neurons corresponds to the number of input features, but in our case, we have more neurons than input features (20 > 18). Each neurons will receive a weighted sum of all the input features, then pass it through the activation to produce an output.

The hidden layer consists of 16 neurons, each using relu activation function and the output layer only has 1 neuron with sigmoid activation function because the output is binary.

A screenshot of a computer

Description automatically generated with medium confidence

We compiled the model using binary cross entropy loss function and adam optimizer. There will be a total of 100 iterations for backpropagation, where the dataset will be broken down and feed into the model with a batch size of 10.

Here is the result produced by the neural network:

Text

Description automatically generated

Conclusion

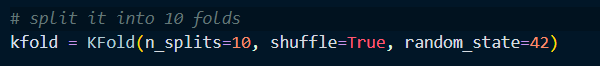
|  |  |
| --- | --- |
| **Model** | **Prediction Accuracy** |
| Decision Tree | 0.8405 |
| Naïve Bayes | 0.8528 |
| Neural Network | 0.8221 |

Overall, we can say that all models can predict the outcome of loan status with high accuracy using the given attributes. Therefore, we can conclude that we can predict the outcome of loan status with high accuracy using data that has the attributes of gender, marital status, number of dependents, education status, property area, credit history, coapplicant income, loan amount and loan amount term. In this project, Naïve Bayes classifier seems to be the best model to use as it yields the highest accuracy.

**Question 2**

*How does k-fold cross validation affect the performance and prediction accuracy of models?*

K-fold cross validation is one of the methods to validate the prediction accuracy. The way it works is by dividing the dataset into equal partitions. The number of partitions = k. For the ith iteration, ith partition is used as test data while others are used as training data. It repeats for k number of iterations. K-fold is helpful in evaluating the model on how well it can predict the outcome based on untrained data and prevent issues like overfitting.



For this project, we have chosen to use k = 10, which is pretty standard. The dataset will be shuffled before being split. Random sampling is used to select the data to include in each fold. Random state is set to 42 to get the same split every time it is run to get consistent result.

Decision Tree



Naïve Bayes Classification

Neural Network

|  |  |  |
| --- | --- | --- |
|  | **Normal Train Test Split** | **K-Fold** |
| Decision tree | 0.8405 | 0.8168 |
| Naïve Bayes classification | 0.8528 | 0.8153 |
| Neural Network | 0.8221 | 0.8333 |

From the table overview, we can see that K-fold’s precision accuracy is all lower than the normal train test split. However, it provides a more accurate representation of the models’ accuracy and performance by using multiple training and test sets. After using K-Fold, we can see that the prediction accuracy is very consistent across 3 models. All of them are around 0.81.

Conclusion

K-fold cross validation does not help in improving prediction accuracy. Prediction accuracy increases in some cases, but that does not mean k-fold has helped to increase the prediction accuracy. In the end, it is a validation method to assess the performance of a model more accurately, not to boost its accuracy.

**Question 3**

*Relating to social injustice issues, do gender, marital status and education play a part in affecting the outcome of loan status?*

To relate the dataset with real life issues, we want to study whether social injustice issues such as discrimination and inequality in gender, marital status and education background affect the outcome of loan status.

A simple approach that we use is to compare the results of 2 different sets of models. One is using the original dataset, and the other with dataset that has the gender, marital status and education background attributes removed from the dataset.

The hypothesis is that if the results produced are similar, then those attributes don’t heavily affect the outcome of loan status. Another way of saying it is that without those attributes, we still can predict the outcome of loan status accurately, hence those attributes are not so important.

Results

The table below showed the results produced:

|  |  |  |
| --- | --- | --- |
| **Model** | **Accuracy** | |
|  | Original Data | Modified Data |
| Decision Tree | 0.8405 | 0.8344 |
| Naïve Bayes | 0.8528 | 0.8466 |
| Neural Network | 0.8221 | 0.8466 |

From the table, we can see that the accuracies of both sets of models are similar. Therefore, we can say that those attributes do not have significant contribution to the outcome of loan status.

Results (K-Fold)

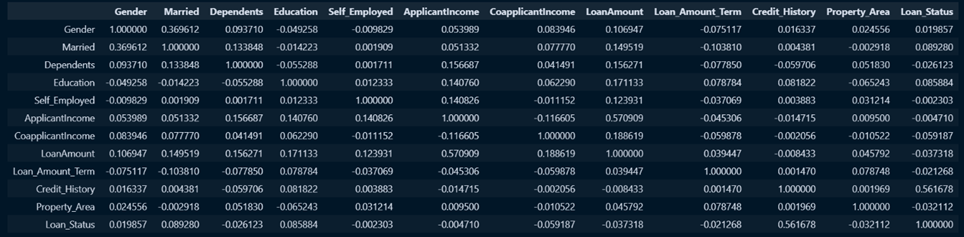
We have also used k-fold cross validation to train and test the 3 models using the modified data set, to get a more accurate presentation of the models’ prediction accuracy.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | | | |
|  | Original Data | Modified Data | Original Data (K-fold) | Modified Data  (K-fold) |
| Decision Tree | 0.8405 | 0.8344 | 0.8168 | 0.8150 |
| Naïve Bayes | 0.8528 | 0.8466 | 0.8153 | 0.8111 |
| Neural Network | 0.8221 | 0.8466 | 0.8333 | 0.8333 |

From the updated table, we can see similar results, the accuracy of the 3 models is similar to when they’re using the original data.

Correlation Matrix

This can be proven once again by the correlation matrix. Their correlation with loan status is quite low:



Gender – 0.019857

Married – 0.089280

Education – 0.085884

Conclusion

Therefore, we can conclude that gender, marital status and education background are not important attributes to consider when determining the outcome of a loan status.

**Question 4**

*Which attributes relate strongly to the outcome of loan status?*

We used Random Forest to check for attributes that relates the strongest to the outcome of loan status. This process is also called feature selection. Random Forest is based on decision trees, and it consist of a few hundred decision trees. It determines the importance of an attribute based on purity of the node produced. The purer the node produced, the higher the importance of the attribute.

Text

Description automatically generatedHere’s the code snippet:

A selection model is created using RandomForestClassifier and the number of decision trees is set to 100. X\_train and y\_train are then fit to the selection model. Sel.get\_support() returns a boolean array that indicates where the features are selected by the selection model. The features that are selected are then added to selected\_feat.

The selected features are:

['CoapplicantIncome', 'LoanAmount', 'Loan\_Amount\_Term', 'Credit\_History\_0.0', 'Credit\_History\_1.0’]

Rebuild the models

We then rebuilt all 3 models using a new dataset that consists of only the selected features. Do note that all the parameters of the models remain the same.

Here are the results:

|  |  |  |
| --- | --- | --- |
|  | **Original Dataset** | **Dataset with selected features** |
| Decision Tree | 0.8405 | 0.8282 |
| Naïve Bayes | 0.8528 | 0.8466 |
| Neural Network | 0.8221 | 0.8466 |

We can see that the accuracies of these models are similar when using the new dataset. Interestingly, neural network has even higher prediction accuracy when using the new dataset. This can be due to the reduced noise caused by other non-important attributes.

Conclusion

Therefore, we can conclude that the attributes that contributes the most to the outcome of a loan status is coapplicant income, loan amount, loan amount term and credit history.

**6. Discussions + Conclusions**

**Insights**

Here are some of the insights that we have gained after completing this data mining project:

Impact of outliers in prediction accuracy

We have built the 3 decision models before realizing we forgot to remove the outliers in the dataset during the data pre-processing phase. This mistake has helped us to discover the impact of the outliers in affecting the prediction accuracy, which can be seen in the diagrams below.

*Before*

Chart, treemap chart

Description automatically generatedChart, treemap chart

Description automatically generated

*After*

**Chart, treemap chart

Description automatically generated**Chart, treemap chart

Description automatically generated

Before removing outliers, we can see that the accuracy never exceeds 0.8, even though we have tried to use different max\_depth values for our decision tree and also used both gini and entropy to calculate the impurities. After removing outliers, the accuracy increases to around 0.84, which is considered quite good and accurate.

After this, we realized just how much outliers can skew the data distribution and alter the prediction model into predicting with lower accuracy.

Feature selection based on importance

It is until the last research question that we realize this: although there are a lot of attributes in this dataset that can be used to determine the outcome of housing loan approval, only a few really determine it. We can actually exclude the other attributes and still build a prediction model that can achieve high accuracy. It might be even faster or more efficient to train and run the model, since there are less attributes.

**Further improvements**

Neural networks

The neural network that we have deployed in this project is just a simple model with one hidden layer only. There are a lot more that goes into building a neural network like deciding how many hidden layers to use, how many neurons for input layer and hidden layers, what activation function to use, how many epochs to use, etc.

For our case, the way we decide how many layers and neurons to use is by trial and error using random values. We believe that there are much more effective ways to do that. However, it is satisfactory for this project as we have achieved quite high accuracy using one hidden layer with the number of neurons that we have selected.

In the future, we should try to truly understand how neural network works and also learn how to choose the number of layers and neurons, etc to build a better model.

Other approach for question 3

*Relating to social injustice issues, do gender, marital status and education play a part in affecting the outcome of loan status?*

For this question, it is more advanced where we really need a solid evidence or proof for it since it is a more serious topic. In our case, we have used a simple approach by dropping the attributes and see whether there are any changes.

However, we are not sure whether that’s the best approach to do it. To improve, more time is needed to research into this topic and found out what’s the best approach and method to build a model around it, to study and find out the true relationships between the 3 attributes and the outcome. More models and tests should be implemented to get solid proofs before coming to a conclusion.