Predicting Mortgage Approvals

Lewis Spencer (April 2019)

1. Executive Summary

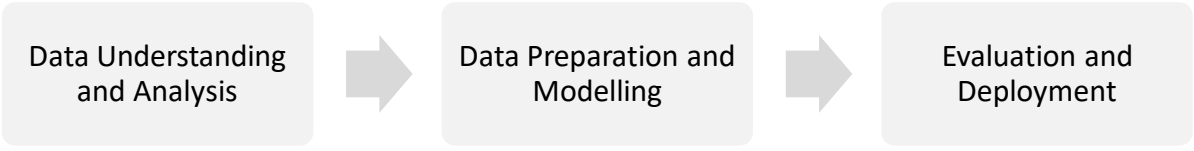
This document outlines an analysis and prediction of mortgage approvals using Government data. The analysis is based upon a dataset of 500,000 observations and 21 features which was adapted from the Federal Financial Institutions Examination's Council (FFIEC). After exploring the data, collecting summary statistics and producing visualisations to explore the relationship between the target variable (accepted mortgages) and other features in the dataset, a predictive model was developed to classify new data. The model was trained and tuned to provide optimal performance – a re-iterative process which involved feature engineering and hyper parameter optimisation following the CRISP-DM methodology (cross-industry standard practice in data mining). The model was validated by accuracy of classification. The result was a prediction accuracy of 71.06% in validation and 71.14% in deployment against new data.

There were 14 significant features found in this analysis, of which the most significant were:

- **Property Type:** indicates whether the loan or application was for a one-to-four-family dwelling (other than manufactured housing), manufactured housing, or multifamily dwelling; available values are 1-- One to four-family (other than manufactured housing), 2-- Manufactured housing, 3—Multifamily.
- **Loan Purpose:** Indicates whether the purpose of the loan or application was for home purchase, home improvement, or refinancing; available values are: 1 -- Home purchase, 2 -- Home improvement, 3 – Refinancing.
- **Applicant Race and Ethnicity:** Race of the applicant; available values are: 1 -- American Indian or Alaska Native, 2 – Asian, 3 -- Black or African American, 4 -- Native Hawaiian or Other Pacific Islander, 5 – White, 6 -- Information not provided by applicant in mail, Internet, or telephone application, 7 -- Not applicable, 8 -- No co-applicant.
- **Lender:** A categorical with no ordering indicating which of the lenders was the authority in approving or denying this loan
- **Loan Type:** Indicates whether the loan granted, applied for, or purchased was conventional, government-guaranteed, or government-insured; available values are: 1 -- Conventional (any loan other than FHA, VA, FSA, or RHS loans), 2 -- FHA-insured (Federal Housing Administration), 3 -- VA-guaranteed (Veterans Administration), 4 -- FSA/RHS (Farm Service Agency or Rural Housing Service)
- **Co-applicant:** (bool) - Indicates whether there is a co-applicant (often a spouse) or not

Methodology

The project followed a re-iterative process of feature selection which was to begin with all the features and then return to data analysis to remove less significant features. An outline of the project's methodology can be seen below:



2. Exploratory Data Analysis

Individual Feature Statistics

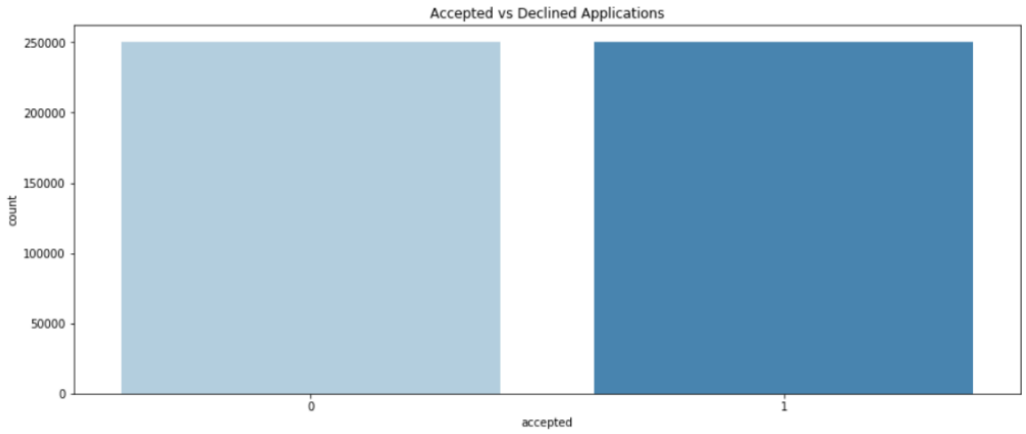
	row_id	loan_type	property_type	loan_purpose	occupancy	loan_amount	preapproval	msa_md	state_code	county_code	applicant_ethnici
count	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.00000
mean	249999.500000	1.366276	1.047650	2.066810	1.109590	221.753158	2.764722	181.606972	23.726924	144.542062	2.03622
std	144337.711634	0.690555	0.231404	0.948371	0.326092	590.641648	0.543061	138.464169	15.982768	100.243612	0.51135
min	0.000000	1.000000	1.000000	1.000000	1.000000	1.000000	1.000000	-1.000000	-1.000000	-1.000000	1.00000
25%	124999.750000	1.000000	1.000000	1.000000	1.000000	93.000000	3.000000	25.000000	6.000000	57.000000	2.00000
50%	249999.500000	1.000000	1.000000	2.000000	1.000000	162.000000	3.000000	192.000000	26.000000	131.000000	2.00000
75%	374999.250000	2.000000	1.000000	3.000000	1.000000	266.000000	3.000000	314.000000	37.000000	246.000000	2.00000
max	499999.000000	4.000000	3.000000	3.000000	3.000000	100878.000000	3.000000	408.000000	52.000000	324.000000	4.00000

population	minority_population_pct	ffiecmedian_family_income	tract_to_msa_md_income_pct	number_of_owner-occupied_units	number_of_1_to_4_family_units	lender	accepted
500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000	500000.000000
5396.982356	31.225669	69158.876302	92.200385	1423.172866	1880.147458	3720.121344	0.500228
2667.723303	25.798784	14478.232811	13.990187	721.027517	893.717989	1838.313175	0.500000
14.000000	0.534000	17858.000000	3.981000	4.000000	1.000000	0.000000	0.000000
3805.000000	11.191000	60071.000000	89.145000	963.000000	1323.000000	2442.000000	0.000000
4975.000000	22.901000	67526.000000	100.000000	1327.000000	1753.000000	3731.000000	1.000000
6379.000000	44.486000	74714.250000	100.000000	1754.000000	2275.000000	5436.000000	1.000000
37097.000000	100.000000	125248.000000	100.000000	8771.000000	13623.000000	6508.000000	1.000000

Further information on individual features can be found at: https://datasciencecapstone.org/competitions/14/mortgage-approvals-from-government-data/page/44/

Univariate and Bivariate Analysis

A small number of columns were missing values, these were imputed with the median values of those columns. The **accepted** feature was the target for prediction in this analysis, interestingly the amount of mortgages accepted vs rejected were equal, therefore to train the model the data was balanced.



The initial hypothesis at this point was that the size of the loan for the home, and the income of the applicant would play a significant role in whether an applicant would be successful. The following plots explore distribution (or univariate frequency) of selected features, and their relationship to the target variable of 'accepted'.

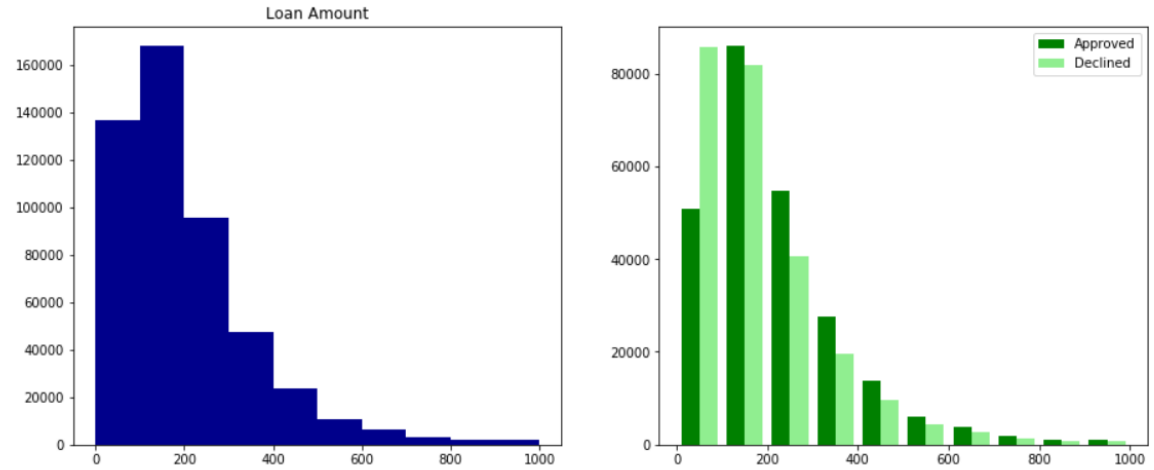


Figure: Shows histogram of loan amount and a comparison of loan amount with approved vs declined. The higher the loan amount seems to contribute to the rate of acceptance with lower loan amounts having a much higher rate of being declined – this may be due to other factors surrounding applicants within that range.

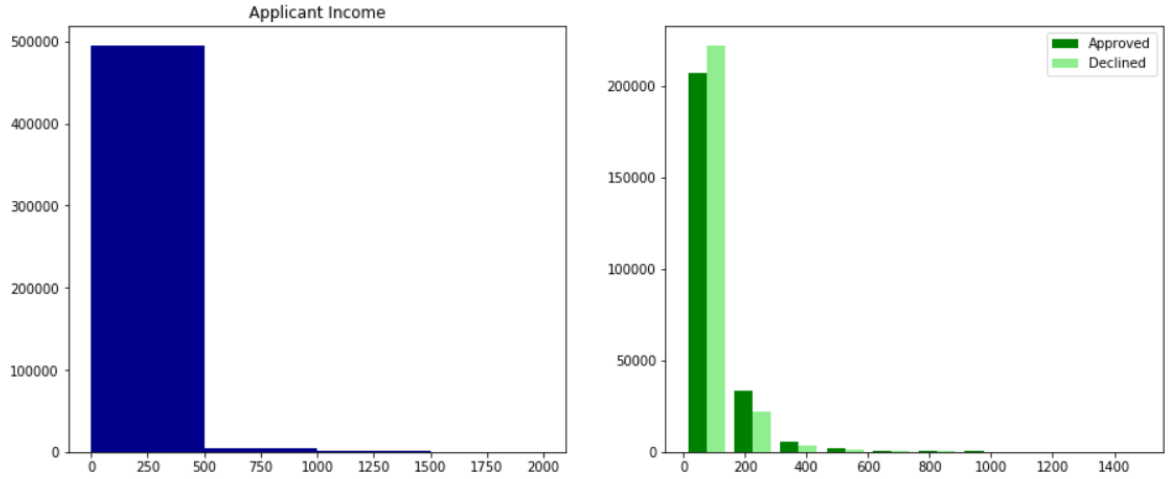


Figure: Histogram of applicant income and a comparison of applicant income with approved vs declined. The majority of applicant's had income between 0 – 500, however in the upper ranges the probability of being accepted increased.

A key insight was the relationship between loan purposes and accepted, alongside the relationship between accepted and race, ethnicity and property class. This highlighted a need to consider ethical implications as race and class are protected characteristics – thus analysis based upon property and race must be in an objective non-discriminatory manner. This relates back to the concepts of proxy discrimination and fairness across groups. As this project will aim to predict in a controlled environment, these features were included and moving forward could uncover unfair bias in the original training dataset. This could be the focus of a further experiment into explaining the cause of protected characteristics uncovering predictive elements in the dataset.

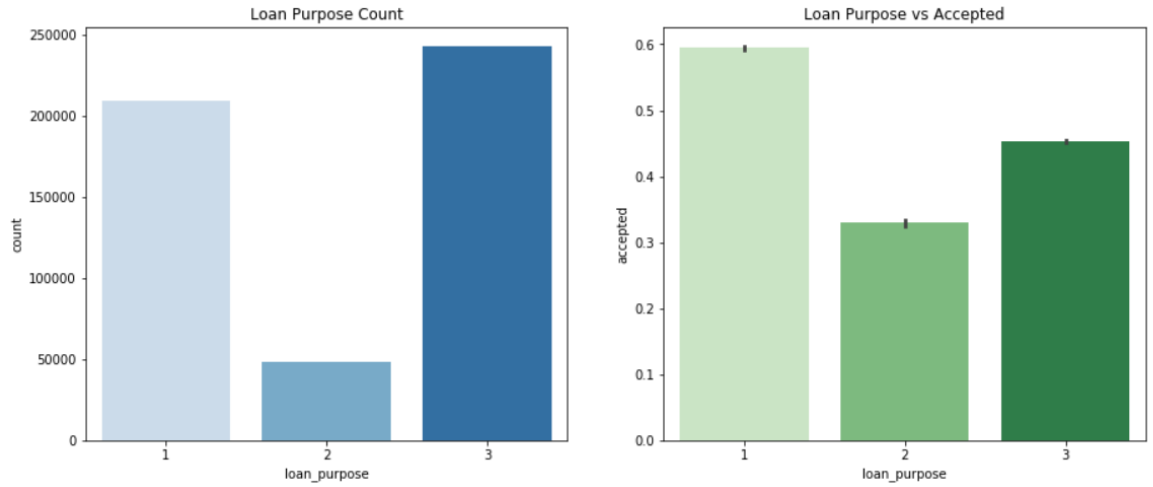


Figure: Although purpose category 1 (Home purchase) contained similar levels to category 3 (Refinancing) the probability of being accepted stood at 60% versus 45% respectively. This is a significant indicator.

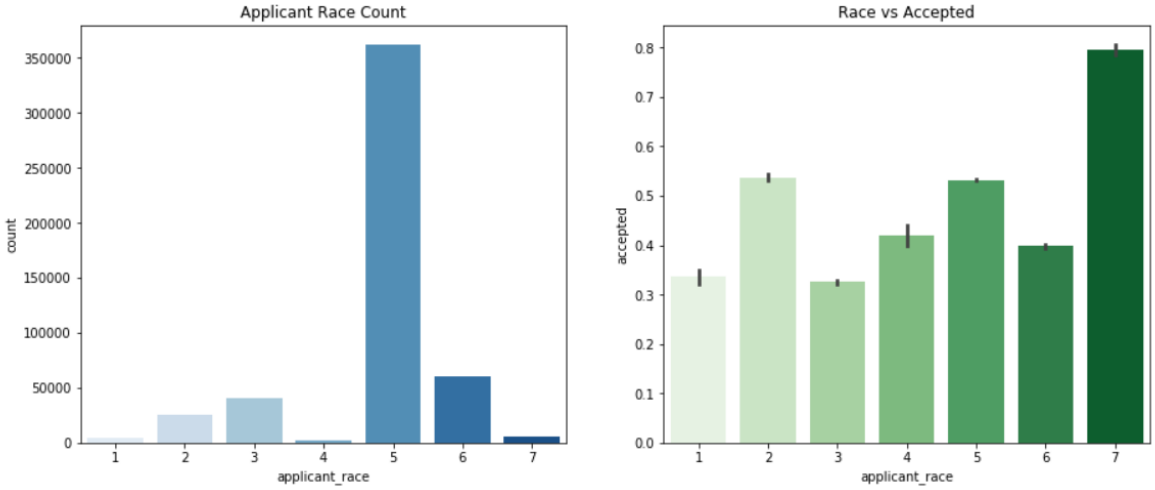


Figure: Category 5 contained 'White' and was the most populous category, however this did not translate to a high probability of acceptance. Category 7 contained 'Not Applicable' which yields an 80% probability of acceptance. Other categories range from 31% to 50% ('Black, African American, American Indian, Alaska Native and Asian). The causation of these probability values is not clear

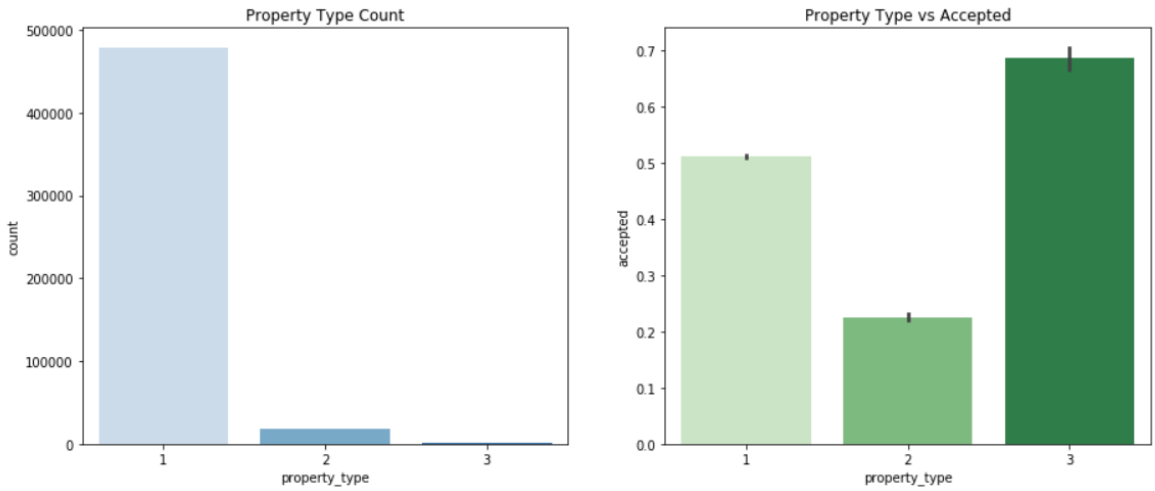


Figure: The least voluminous category 3 (multi-family) had a 60% probability of acceptance versus the most populated category 1 (one to four family) which only had a 50% probability. This is useful for decision trees.

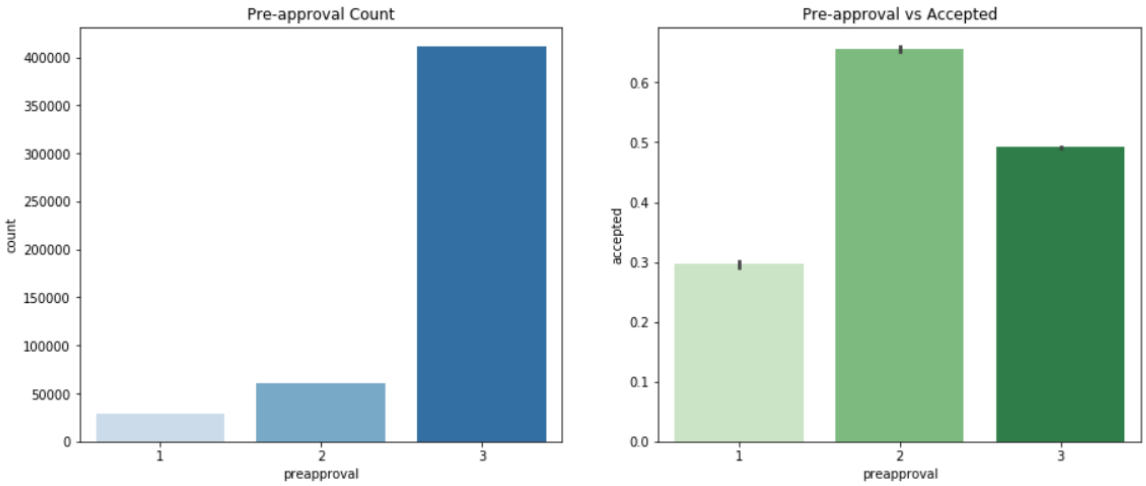


Figure: Category 2 (Pre-approval was not requested) showed a significant probability of acceptance (67%)

Correlation and Apparent Relationships

After exploration, it became clear that enough relationships existing between features and the target to attempt prediction modelling. From the above plots, the strongest relationships proved to be between property type and loan purpose with accepted – which will become more evident later. The numeric features which came from census information proved to be less useful – the median family income was the only feature to be put through feature engineering from this set.

3. Data Preparation

Prior to modelling, a number of data pre-processing steps were carried out to boost the accuracy of the model. The first of which has already been discussed, the missing values were imputed with the median of the columns in which they resided.

Feature Engineering

As part of an iterative data process, features were engineered that proved more useful to the model than some original features. These were:

1. Loan to income ratio: which was the loan amount divided by the applicant's income
2. Location code: which was the state code multiplied by the county code
3. Applicant race and ethnicity: which was applicant race multiplied by applicant ethnicity
4. Family income: transformed to a categorical variable (low, medium and high income)

Encoding

All categorical variables were then encoded using Label Encoder. These variables were co-applicant (true or false) and family income (low, medium and high).

Train Test Split

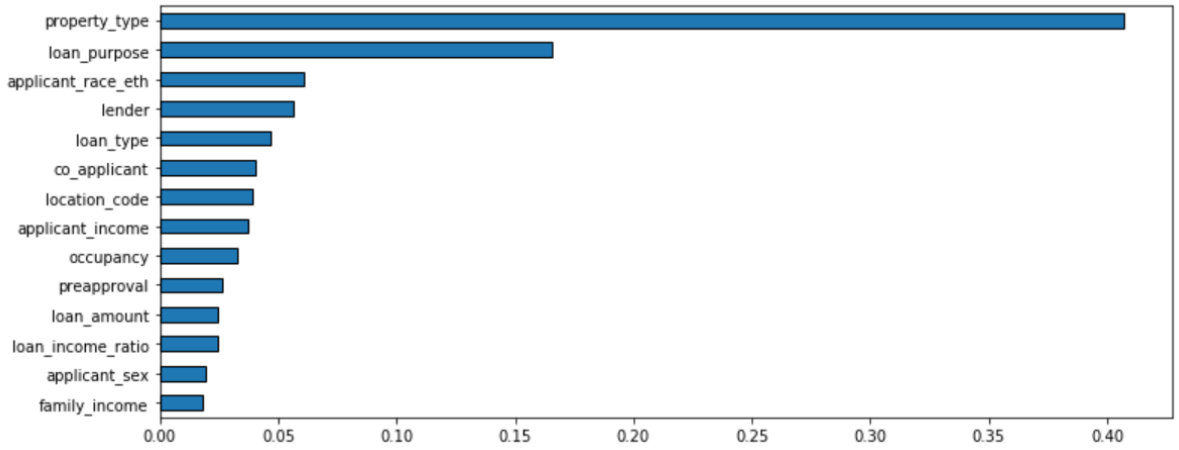
The target **accepted** was removed from the dataset, and 14 key features identified. The data was then split into a train (70%) and test set (30%). This allowed 400,000 observations for training and 100,000 for testing to act as unseen data.

Scaling

The data was scaled using Standard Scaler which standardises the features by removing the mean and scaling to unit variance. The scaler was fit to the train data to prevent information leakage.

4. Modelling

The model used for prediction was the XGB (Extreme Gradient Boosting) Classifier. Research was carried out to discover the optimal tuning of the hyper parameters. XGBoost is an implementation of gradient boosted decision trees designed for speed and performance. This was trained using 70% of the data and tested on the remaining 30%. Following this the training set produced a result of 76.63% accuracy and 71.04% for the test set. As the XGB Classifier used a tree-based method for boosting, a calculation of feature importance was performed:



The final selection of tuned hyper-parameters can be seen below (this, along with data exploration took the longest part of the project). This was created by using a grid search cross validation step to select from a range of parameters the most optimal. The most significant was the max depth and minimum child weight which was essential in combatting the imbalance in the dataset. This was due to there being categorical data which was heavily dominated by a single class – thus created an imbalance which affected the decision making of the algorithm in favour of the dominant class. The tuning of hyper parameters alongside feature engineering saw the biggest increase in accuracy of the model.

```
# build and fit xgb model (approx run time: 10 mins)
xgb = xgboost.XGBClassifier(objective = 'binary:hinge', eval_metric='error', min_child_weight = 5,
                            max_depth = 11, min_samples_split = 2, min_samples_leaf=50, n_estimators = 600,
                            random_state = 50, n_jobs=-1, gamma=0, scale_pos_weight=1, nthred=4,
                            learning_rate=0.1, reg_alpha=0.005, subsample=0.8)
```

5. Evaluation

Confusion Matrix

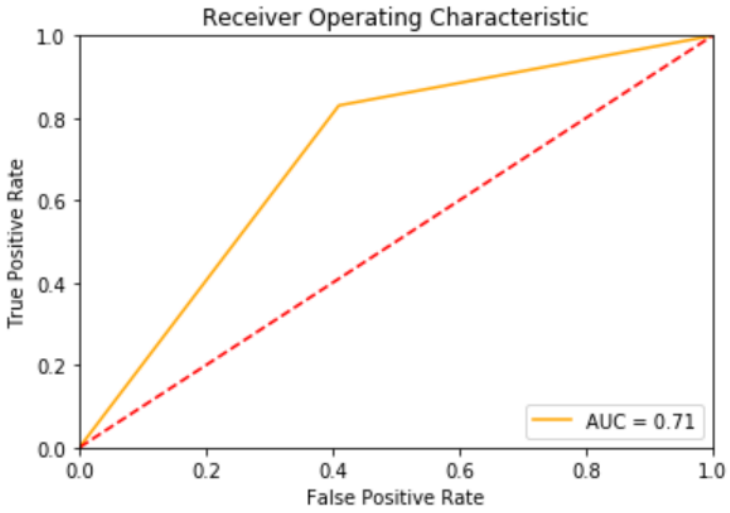
The results of the confusion matrix show inaccuracies in classification, here it is shown that the model produced 8475 false positives and 20481 false negatives. This could certainly be an area to improve on for the future of this data pipeline, and to explore whether false positives or false negatives are more costly or beneficial to data subjects. For example, a false positive could be an applicant that cannot afford the size of the loan applied for – thus creating a debt problem if this solution was used in a live deployment scenario.

	Positive	Negative
Actual Positive	29488	20481
Actual Negative	8475	41556

Accuracy: 0.71004

Received Operator Characteristic (ROC)

The ROC curve for the model is shown here, with the orange line indicating the model's performance versus the central diagonal line which would be equivalent to a random guess. The receiver operating characteristic or ROC is a curve that displays the relationship between the true positive rate on the vertical axis and false positive rate on the horizontal axis. The ROC curve shows the trade-off between true positive rate and false positive rate.



This translates into the following standard performance metrics for classification:

	Positive	Negative
Number of Cases	49969	50031
Precision	0.78	0.67
Recall	0.59	0.83
F1 Score	0.67	0.74

6. Deployment

To deploy the model, the same data preparation and feature engineering was carried out upon the test values dataset to ensure consistency whilst predicting.

Missing data was filled by imputing the median and the data was scaled using Standard Scaler. The output was then saved to a CSV file which could be used to complete the project.

7. Conclusion

The required accuracy for this model was 71% and this model satisfies that requirement. This analysis has shown that whether an applicant will be approved for a mortgage loan can be predicted with a high confidence given these features. In particular, as per the feature importance earlier,

property type and loan purpose have a significant effect on the approval chances – and the model's predictive power. Other features such as income, applicant sex, race, ethnicity, loan type, lender, location, pre-approval, loan amount and loan to income ratio also help to classify approval.

The main challenge in exploring and modelling throughout this project was storing and manipulating the data due to its size. The computation required to apply complex algorithms was higher than usual – therefore an option that taken was to utilise a cloud service platform which can host Jupyter Notebooks. This removed the issue of storing and processing a relatively large dataset on a local machine. Further improvements to the accuracy of the prediction would most likely be marginal, however improvements might be made by attempting a stacking methodology of different machine learning classifiers or using a model built for handling larger volumes of data at increased speeds such as the Light GBM (Gradient Boosting Model).