Loan Prediction

Springboard
Data Science Career Track

Paul Kim

Background

- Continuous growth in personal loan debt.
- Denials of the applications also increases as the number of applications increase.
- Understanding the financial factors those lead to credit score of individual is crucial since it significantly affects the approval rate of loan.

Goals

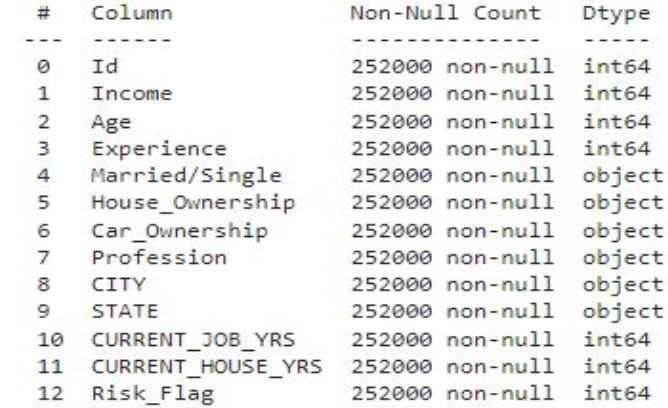
- Develop a machine learning model that predicts clients possibility on loan approval using the ones' financial features.

- Understand the importance of the features on predicting the client's result of the loan application or if a client gets denied, analyze the financial status of the applicant that is causing the rejection.

Data Acquisition

Dataset ‘Loan Prediction’ was retrieved from ‘Kaggle’ in a csv format.

- Id
- Income
- Age
- Experience
- Married / Single
- House_Ownership
- Car_Ownership
- Profession
- City
- State
- Current Job yrs
- Current_House_yrs

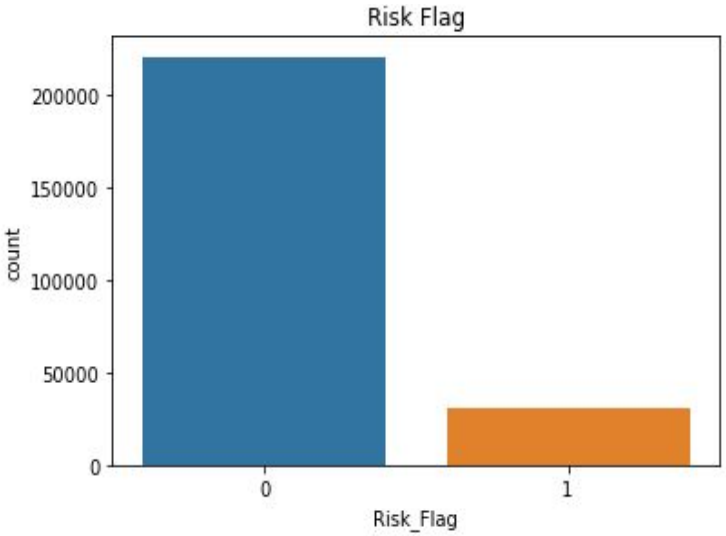


#	Column	Non-Null Count	Dtype
0	Id	252000 non-null	int64
1	Income	252000 non-null	int64
2	Age	252000 non-null	int64
3	Experience	252000 non-null	int64
4	Married/Single	252000 non-null	object
5	House_Ownership	252000 non-null	object
6	Car_Ownership	252000 non-null	object
7	Profession	252000 non-null	object
8	CITY	252000 non-null	object
9	STATE	252000 non-null	object
10	CURRENT_JOB_YRS	252000 non-null	int64
11	CURRENT_HOUSE_YRS	252000 non-null	int64
12	Risk_Flag	252000 non-null	int64

- 13 columns and 252000 rows

Exploratory Data Analysis

Target Variable Proportion

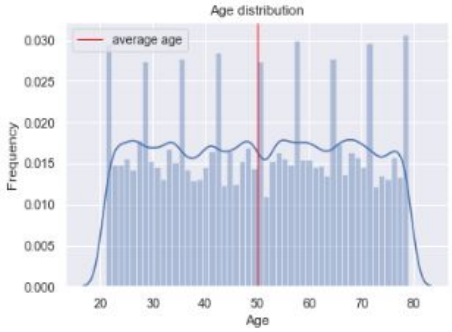


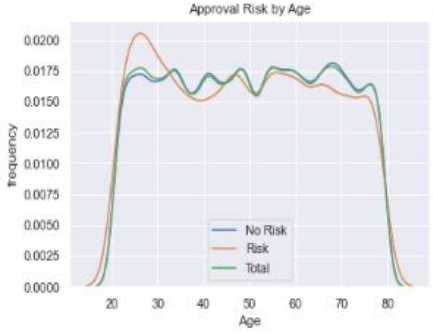
- Target Variable ‘Risk-Flag’ had proportion of the dataset :

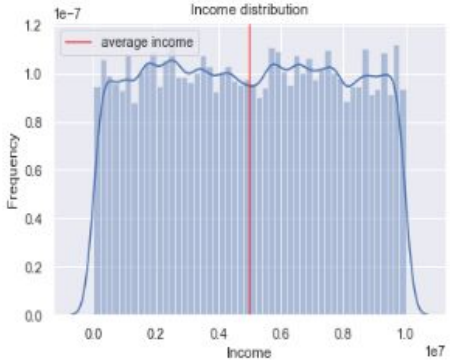
0(No Risk) : 87.7%

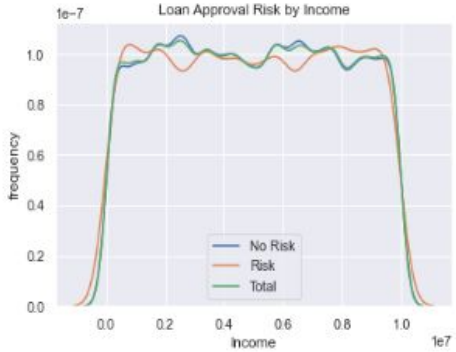
1(Risk present) : 12.3%

Data Distribution





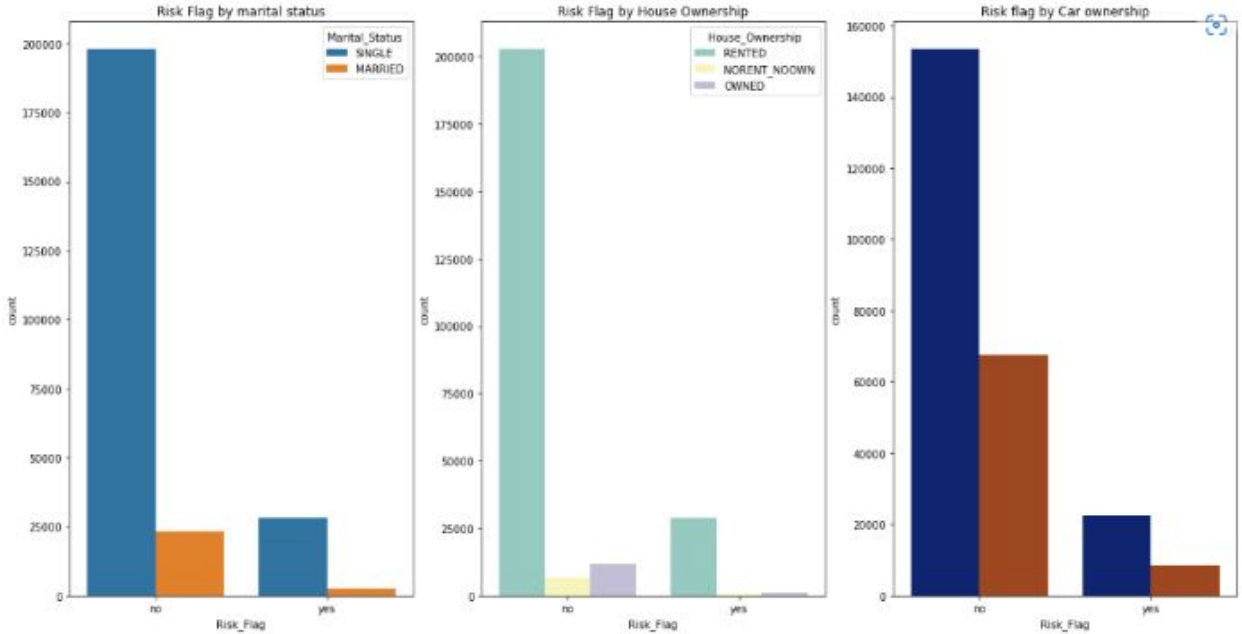




- Age group distribution had an average of 49.95 with minimum of 21 and maximum of 79.
- In terms of target variable, younger age groups had higher risk compared to older age groups.

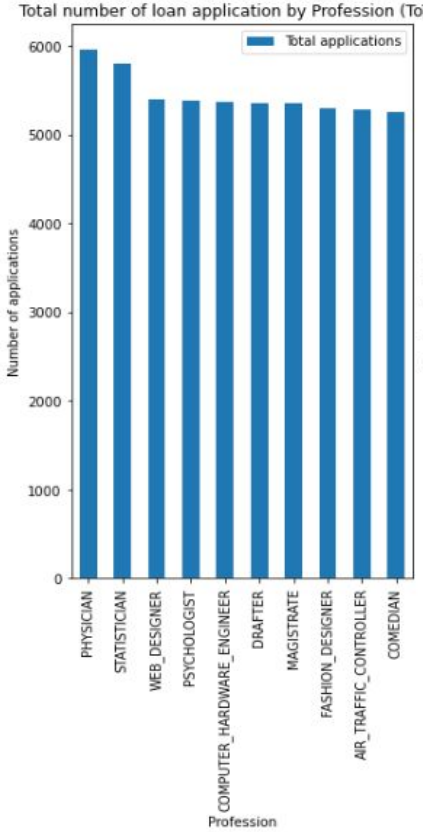
- Distribution of income with mean income of 4997117.
- In terms of target variable, low income and extremely high income group had highest risk.

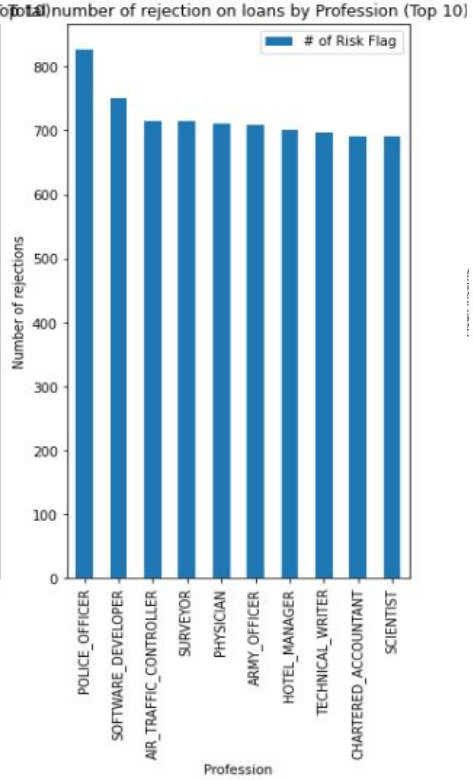
Categorical features distribution

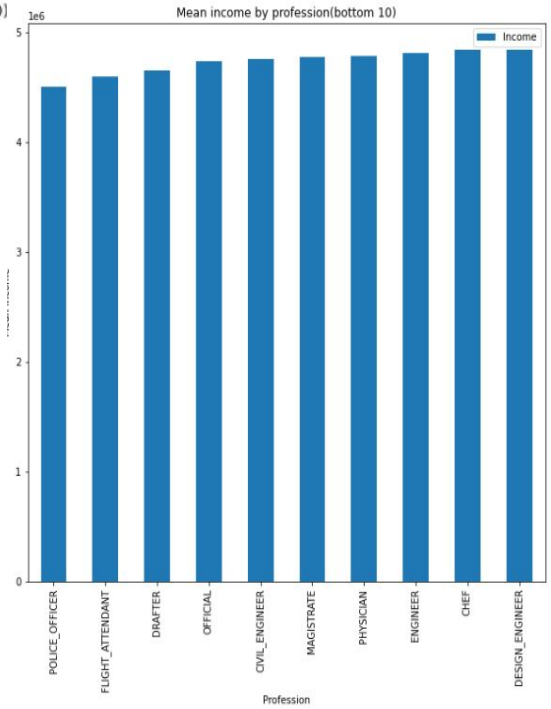


- Marital Status, House ownership and Car ownership distribution.
- Risk Flag distribution within categorical features have similar proportion.

Distribution by profession

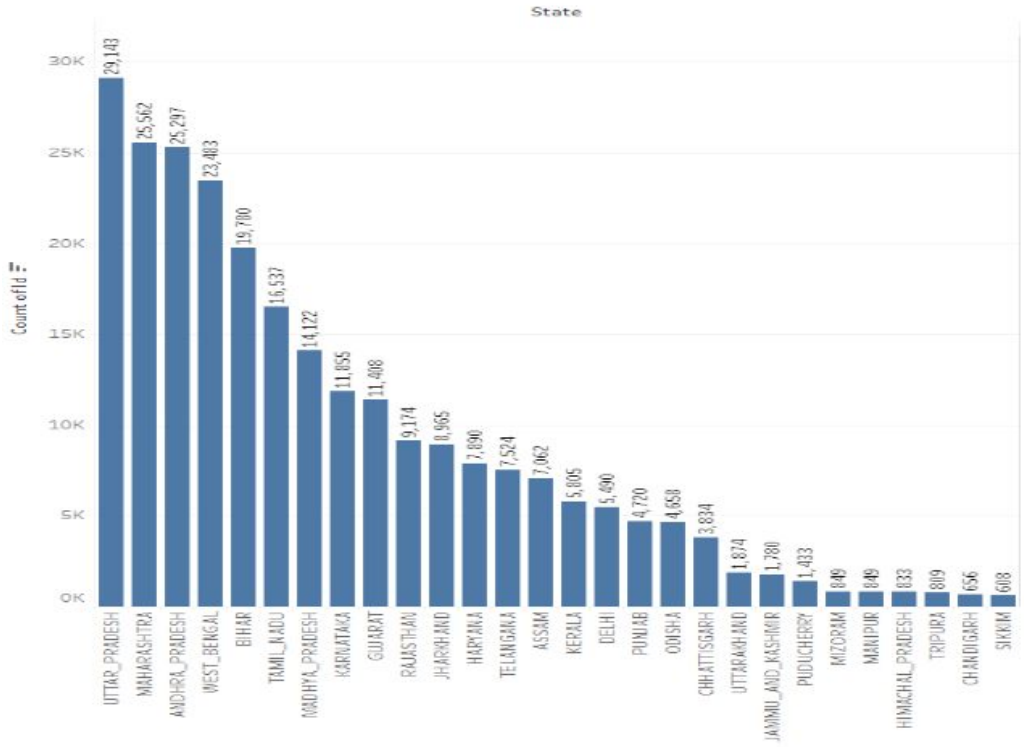






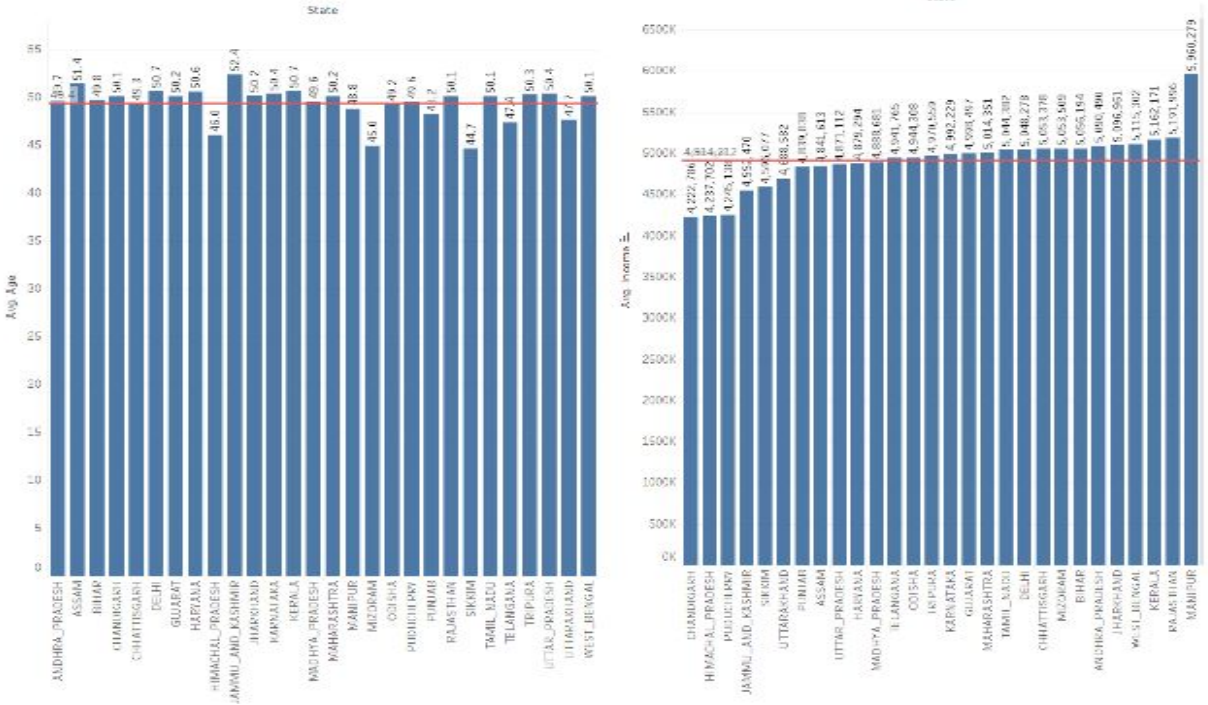
- Total number of applications by profession and 10 most professions with rejections plotted. Second plot and 10 low income have similar list of professions.

City and State Distribution



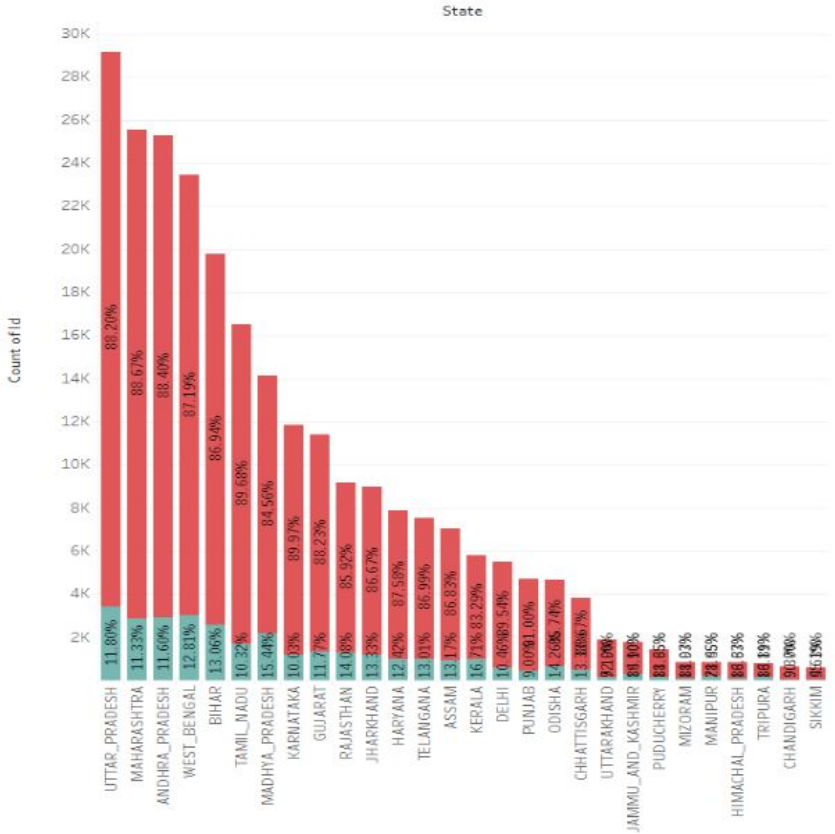
- Left is the population distribution by state.
- The maximum and minimum population by state may be big however, it is due to number of cities in state and each city has similar number of population.

Income and age distribution by State



- Plot represents the average income and average age distribution by the states.
- As the bar plots show the states share similar average income and age.

Target variable distribution by State



- Target variable proportion by state and city was visualized.
- As the graph shows, the proportion of the target variable lies between 10-15 percent in each state.
- The proportion was analyzed to check if separate modeling by state or city is necessary.
- However, we have decided to develop a single model since the city and state have similar characteristics.

Baseline Modeling

Baseline modeling with and without ‘City’ and ‘State’

Logistic Regression Classification Report:

	precision	recall	f1-score	support
0	0.88	1.00	0.93	66301
1	0.00	0.00	0.00	9299
accuracy			0.88	75600
macro avg	0.44	0.50	0.47	75600
weighted avg	0.77	0.88	0.82	75600

RandomForest Classification Report:

	precision	recall	f1-score	support
0	0.94	0.95	0.94	66301
1	0.60	0.55	0.57	9299
accuracy			0.90	75600
macro avg	0.77	0.75	0.76	75600
weighted avg	0.90	0.90	0.90	75600

LGBM Classification Report:

	precision	recall	f1-score	support
0	0.88	1.00	0.93	66301
1	0.66	0.02	0.04	9299
accuracy			0.88	75600
macro avg	0.77	0.51	0.49	75600
weighted avg	0.85	0.88	0.82	75600

Logistic Regression Confusion Matrix:

	precision	recall	f1-score	support
0	0.88	1.00	0.93	66301
1	0.00	0.00	0.00	9299
accuracy			0.88	75600
macro avg	0.44	0.50	0.47	75600
weighted avg	0.77	0.88	0.82	75600

RandomForest Confusion Matrix:

	precision	recall	f1-score	support
0	0.94	0.95	0.94	66301
1	0.60	0.54	0.57	9299
accuracy			0.90	75600
macro avg	0.77	0.75	0.76	75600
weighted avg	0.90	0.90	0.90	75600

LGBM Confusion Matrix:

	precision	recall	f1-score	support
0	0.88	1.00	0.94	66301
1	0.79	0.02	0.04	9299
accuracy			0.88	75600
macro avg	0.84	0.51	0.49	75600
weighted avg	0.87	0.88	0.82	75600

- Baseline modeling with and without ‘City’ and ‘State’ columns created with Logistic Regression, RandomForest and LGBM classifier to check if one performs better or similar.
- As the chart shows, they both have same performance.

Baseline modeling without ‘City’ and ‘State’ after tuning.

Logistic Regression Confusion Matrix:

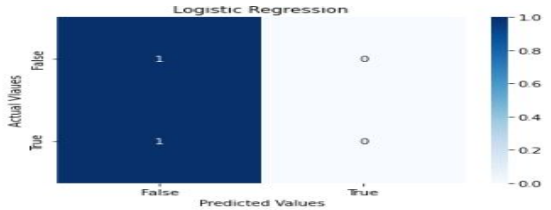
	precision	recall	f1-score	support
0	0.88	1.00	0.93	66301
1	0.00	0.00	0.00	9299
accuracy			0.88	75600
macro avg	0.44	0.50	0.47	75600
weighted avg	0.77	0.88	0.82	75600

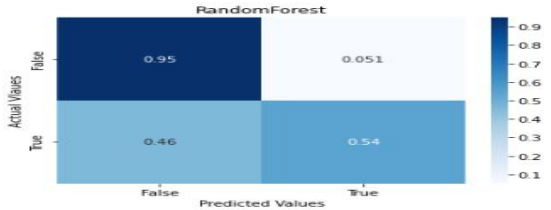
RandomForest Confusion Matrix:

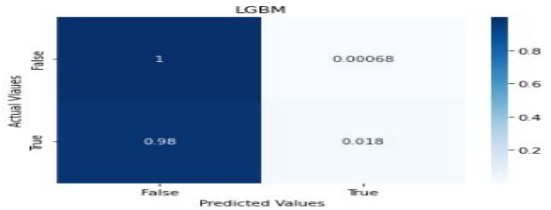
	precision	recall	f1-score	support
0	0.94	0.95	0.94	66301
1	0.60	0.54	0.57	9299
accuracy			0.90	75600
macro avg	0.77	0.75	0.76	75600
weighted avg	0.90	0.90	0.90	75600

LGBM Confusion Matrix:

	precision	recall	f1-score	support
0	0.88	1.00	0.94	66301
1	0.79	0.02	0.04	9299
accuracy			0.88	75600
macro avg	0.84	0.51	0.49	75600
weighted avg	0.87	0.88	0.82	75600







- Hyperparameter tuning done on baseline model without ‘City’ and ‘State’ columns.
- Since removing two columns reduce unnecessary ‘noise’ by the model but still performs the same compared to baseline model with two columns.
- The performance did not improve dramatically but showed slight improvement.

Baseline modeling result

Table of results for comparison

		With columns 'City' and 'State'			Without columns 'City' and 'State'		
		Logistic Regression	RandomForest	LGBM	Logistic Regression	RandomForest	LGBM
Value: No Risk	Precision	0.88	0.94	0.88	0.88	0.94	0.88
	Recall	1	0.95	1	1	0.95	1
	F1-Score	0.93	0.94	0.93	0.93	0.94	0.94
Value: Risk	Precision	0	0.6	0.66	0	0.6	0.79
	Recall	0	0.55	0.02	0	0.54	0.02
	F1-Score	0	0.57	0.04	0	0.57	0.04

- As the table shows above, baseline models with and without ‘City’ and ‘State’ columns did not show significant difference in terms of its performance.
- Therefore, dataset without ‘City’ and ‘State’ columns was used for extended modeling.

Extended Modeling

Dealing with imbalanced classification

	LR	DT	RF	KNN	LGBM	XGB
SMOTE	0.618995	0.871019	0.887646	0.851415	0.760767	0.839974
AD	0.738532	0.87254	0.886799	0.847288	0.813638	0.85295
Random under Sampler	0.52336	0.848161	0.874339	0.829167	0.744841	0.799286
Random over Sampler	0.525013	0.877487	0.893889	0.862831	0.762487	0.821548
TomekLinks	0.876997	0.881905	0.89914	0.887804	0.878161	0.885397

	LR	DT	RF	KNN	LGBM	XGB
SMOTE	0.545722	0.85495	0.843395	0.860277	0.73967	0.799389
AD	0.523868	0.852673	0.845409	0.85515	0.716512	0.776185
Random under Sampler	0.540754	0.85477	0.848105	0.828731	0.740021	0.807257
Random over Sampler	0.542205	0.848837	0.84261	0.855413	0.741852	0.806728
TomekLinks	0.5	0.751781	0.74834	0.728194	0.506997	0.564608

- To deal with imbalanced classification of the data, five different resampling technique used on six classifiers.
- Accuracy score, ROC-AUC and classification report was used for evaluation
- Overall SMOTE oversampler showed the best performance and used to deal with imbalance classification.

Hyperparameter Tuning

Logistic Regression accuracy score: 0.43821916335089794
Logistic Regression best parameters: {'clf__penalty': 'l2', 'clf__C': 1.0}
Logistic Regression roc_auc: 0.5429885578598201
Logistic Regression confusion matrix:

	precision	recall	f1-score	support
0	0.89	0.65	0.75	66301
1	0.15	0.44	0.22	9299
accuracy			0.62	75600
macro avg	0.52	0.54	0.49	75600
weighted avg	0.80	0.62	0.69	75600

RandomForestClassifier accuracy score: 0.596945908162168
RandomForestClassifier best parameters: {'clf__n_estimators': 1000, 'clf__min_samples_split': 2, 'clf__max_features': 'sqrt', 'clf__max_depth': 8, 'clf__criterion': 'entropy'}
RandomForestClassifier roc_auc: 0.6331737881559848
RandomForestClassifier confusion matrix:

	precision	recall	f1-score	support
0	0.92	0.67	0.78	66301
1	0.20	0.60	0.30	9299
accuracy			0.66	75600
macro avg	0.56	0.63	0.54	75600
weighted avg	0.83	0.66	0.72	75600

LGBMClassifier accuracy score: 0.7790084955371546
LGBMClassifier best parameters: {'clf__objective': None, 'clf__num_leaves': 200, 'clf__max_depth': -1, 'clf__learning_rate': 0.1, 'clf__boosting_type': 'gbdt'}
LGBMClassifier roc_auc: 0.8421444794392912
LGBMClassifier confusion matrix:

	precision	recall	f1-score	support
0	0.97	0.91	0.94	66301
1	0.54	0.78	0.63	9299
accuracy			0.89	75600
macro avg	0.75	0.84	0.78	75600
weighted avg	0.91	0.89	0.90	75600

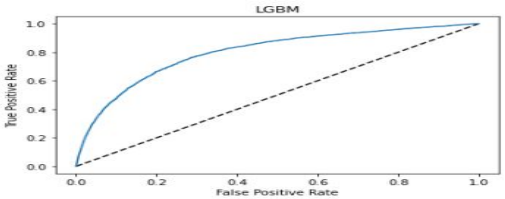
XGBClassifier accuracy score: 0.7772878804172492
XGBClassifier best parameters: {'clf__subsample': 0.6, 'clf__n_estimators': 1000, 'clf__max_depth': 10, 'clf__learning_rate': 0.2}
XGBClassifier roc_auc: 0.842309797435514
XGBClassifier confusion matrix:

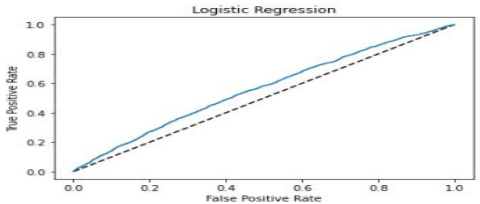
	precision	recall	f1-score	support
0	0.97	0.91	0.94	66301
1	0.54	0.78	0.64	9299
accuracy			0.89	75600
macro avg	0.75	0.84	0.79	75600
weighted avg	0.91	0.89	0.90	75600

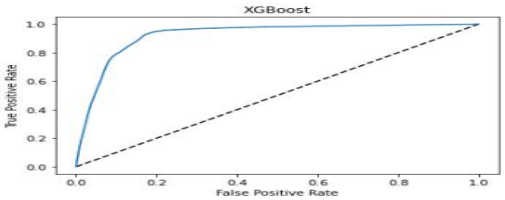
Hyperparameter Tuning continue.

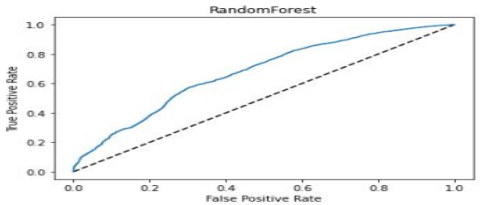
Table of Results

		Logistic Regression	RandomForest	LGBM	XGBoost
Value: No Risk	Precision	0.89	0.92	0.97	0.97
	Recall	0.65	0.67	0.91	0.91
	F1-Score	0.75	0.78	0.94	0.94
Value: Risk	Precision	0.15	0.2	0.54	0.54
	Recall	0.44	0.6	0.78	0.78
	F1-Score	0.22	0.3	0.63	0.64



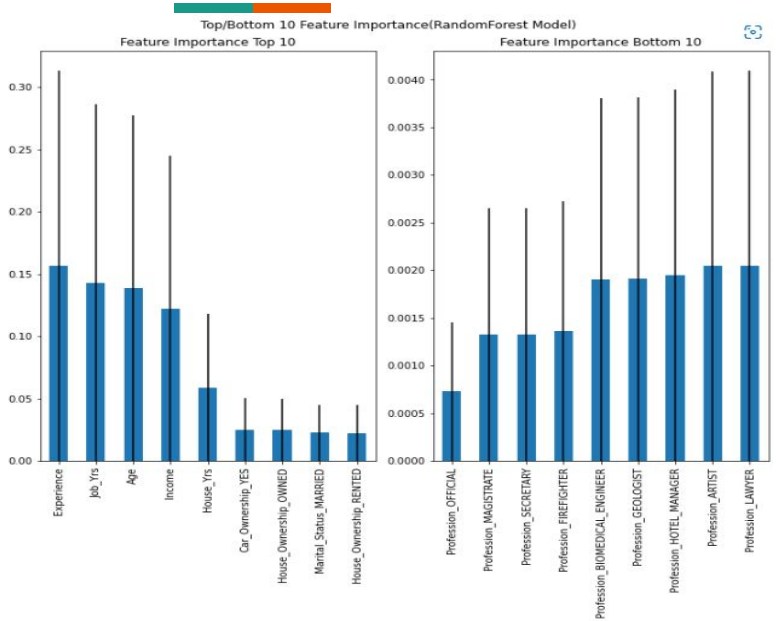


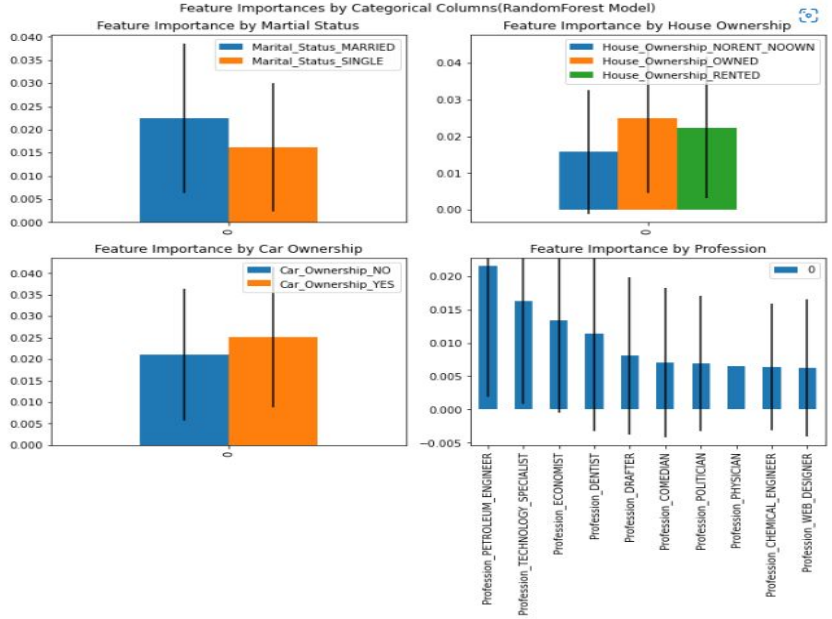




- For hyperparameter tuning, recall score was optimized to minimize the false negative for our project.
- As the results show, XGBoost Classifier performed the best with 84.4 percent of ROC-AUC score.

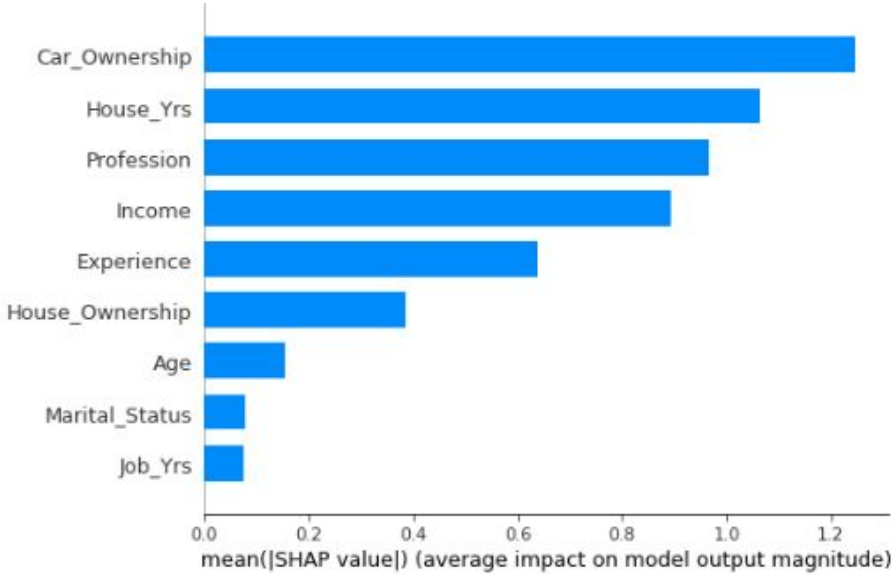
Feature Impact (RandomForest Classifier)





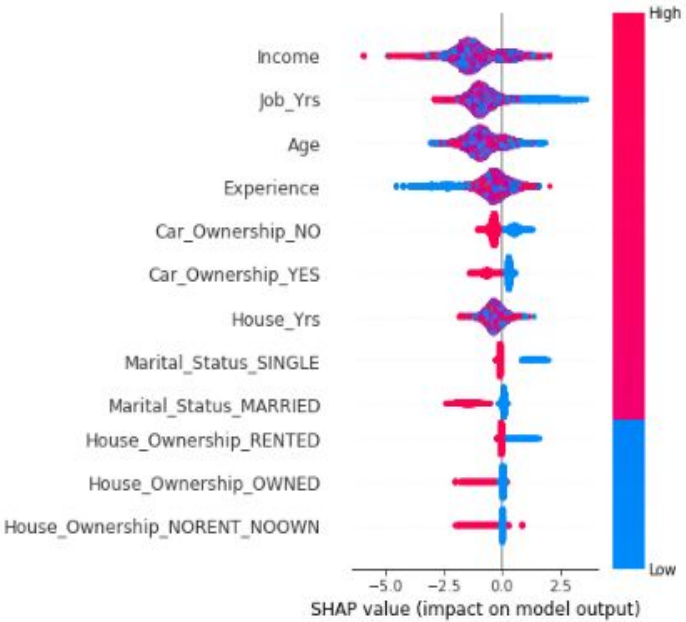
- Feature importance in Random Forest Classifier analyzed by top and bottom ten and categorical features.
- Experience, Job years, Age and Income are considered the most important features on predicting the target variable in the model.

Feature Impact (XGBoost Classifier)



- SHAP values of the features are calculated for XGBoost Classifier and encoded columns' values are added for analysis.
- As the graph shows, Car ownership, house years, profession and income are considered the most important features.

Feature Impact (XGBoost Classifier) continue.



- Beeswarm summary of SHAP values with encoded columns.

More Risk:	Less Risk:
High income	Low Income
High number of Job years	Low number of Job years
Older age	Younger age
Lower years experience	Marital status single
Marital Status married	House rented
House owned or not owned/rented	

Future Work

- Merge another dataset with more features that explains samples financial status

- Develop models for each of ‘State’ therefore clients have different prediction depends on the state they are living in

Recommendations

- Provide detailed counseling to clients where the applications are rejected.
- Provide final check for clients before the actual application.
- Lenders can use it for marketing purpose by identifying the individuals who needs loan services.

Appendix

Talty, A. (2021, November 14). *How Do Personal Loans Work?* Forbes Advisor. https://www.forbes.com/advisor/personal-loans/how-do-personal-loans-work/

Suknanan, J. (2022, April 14). *6 personal loan lenders that'll get you funded in as little as 1 business day*. CNBC. https://www.cnbc.com/select/6-personal-loans-thatll-get-you-funded-in-as-little-as-1-business-day/

Thank you

Email: paulmkim97@gmail.com

Linkedin:https://www.linkedin.com/in/paul-kim-15301514a/

Github:https://github.com/paulkimDSN/Capstone-3-

Tableau:https://public.tableau.com/app/profile/paul5347/viz/edac3/StateandCityVisualization