UCF

Lending Analytics Competition

CFE FEDERAL CREDIT UNION

Equal Housing Lender Federally insured by NCUA

Overview

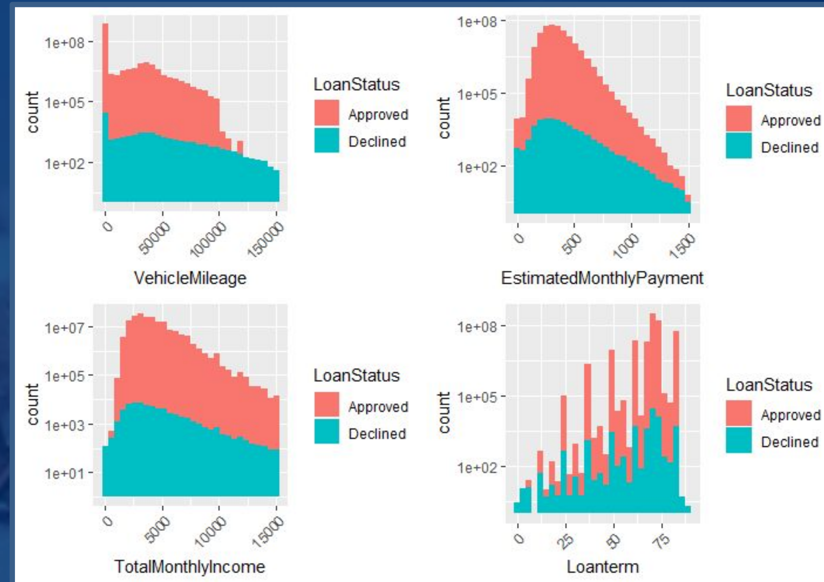
- Introduction
- Exploratory Data Analysis
  - Numerical Feature Validation
  - Categorical Feature Validation
- Feature Engineering
- Model and Classification
- Conclusions and Future Work

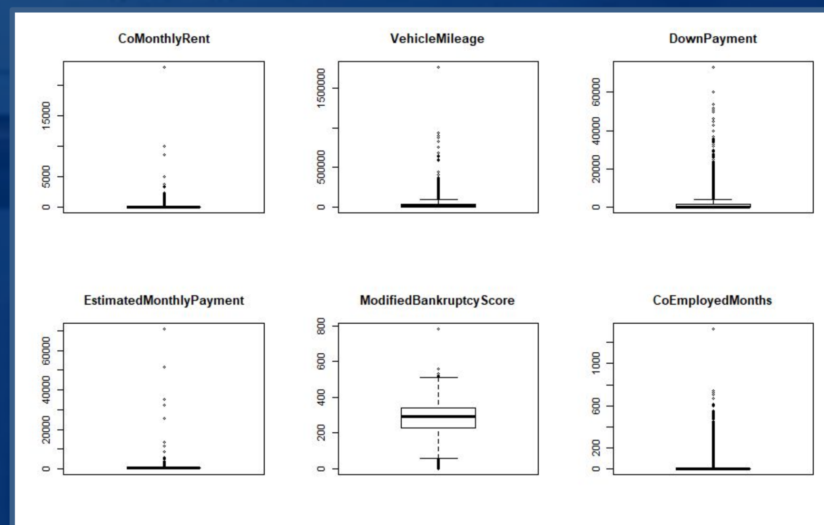
DATASET

- The dataset includes prior CFE vehicle loan data.
- Data includes 37 features.
- Data must be cleaned and analyzed.

Validation

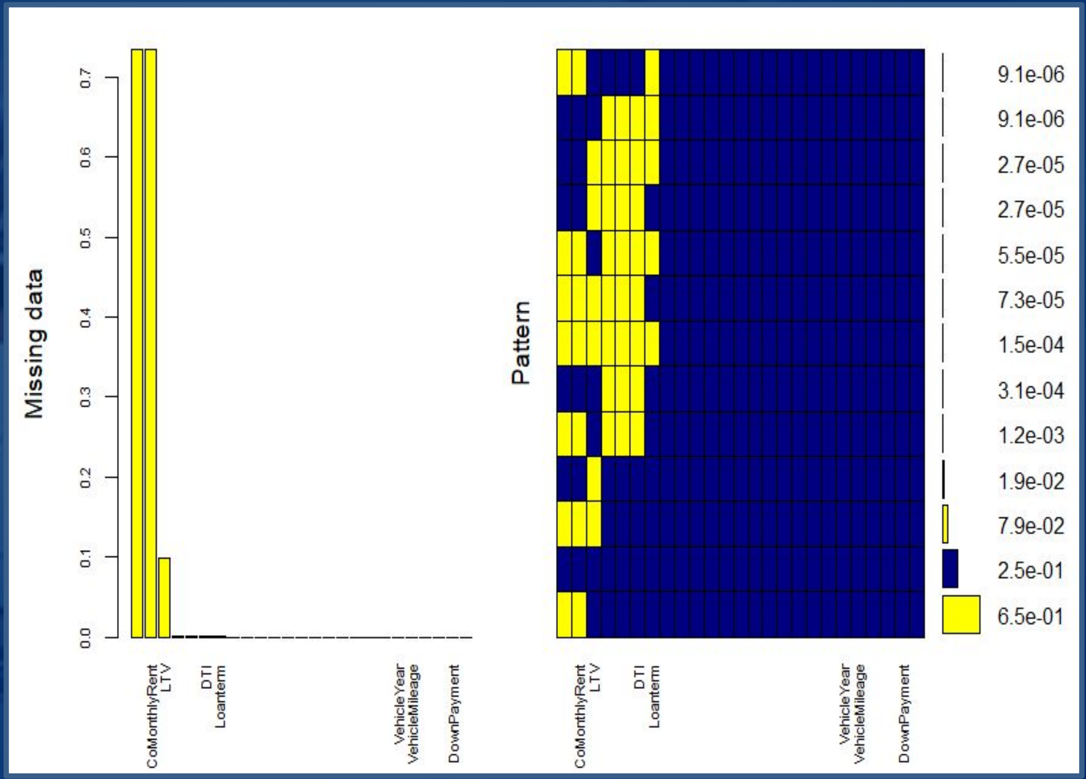
- Outliers:
  - View box plots to determine outliers for each feature
  - Do the values make sense?
- Distributions:
  - Do the distributions make sense contextually?
  - What about how the features compare to LoanStatus?





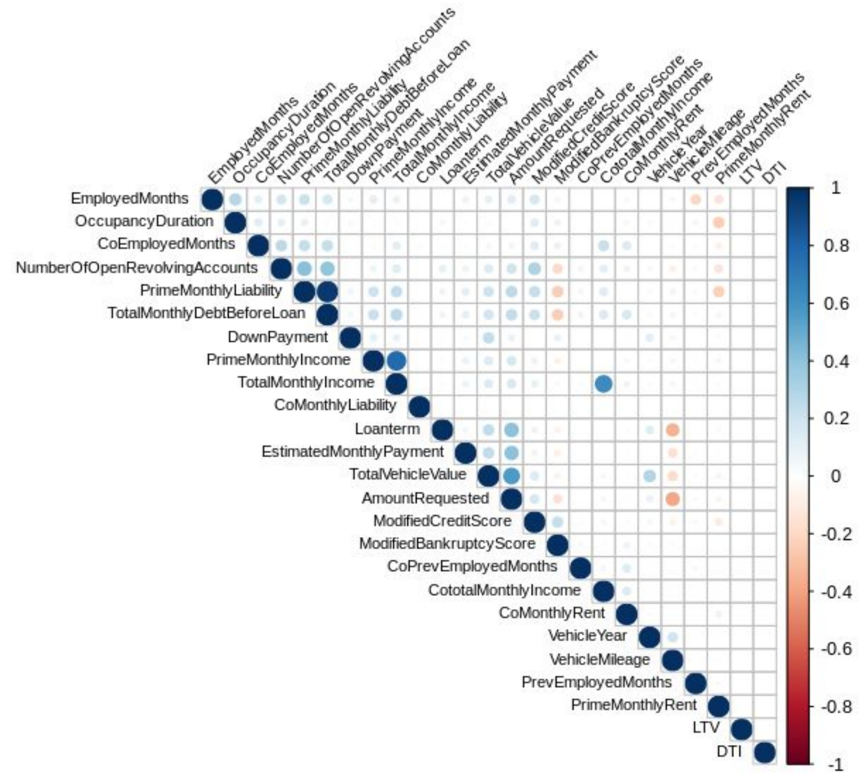
Missing Values

- Oftentimes, entries will have missing values.
- Sometimes, this is appropriate. Other times, it is important we replace the missing value with an estimate.



Correlation

- When two predictors are correlated, this means the value of one influences the value of the other.
- In other words, it can mean that if the value of one predictor increases, the value of the other increases.
- Correlated features need to be removed to increase model performance.



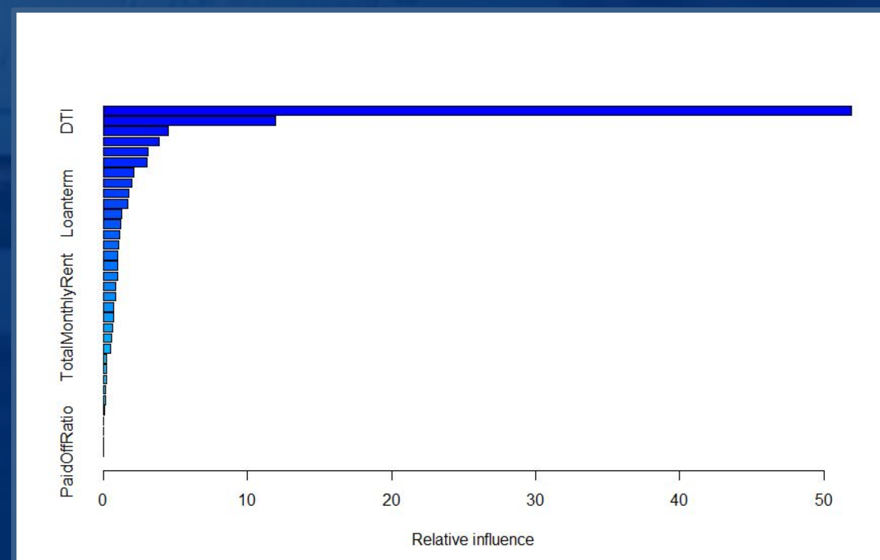
Feature Engineering

- Feature engineering is the process of creating new predictors using acquired contextual understanding of the data and knowledge gained from data analysis.
- Can greatly increase the performance of a model depending on the quality of the features created.

Added Features

- New features created:
  - isPaymentDeficit: Indicates whether or not the applicant can afford the monthly payments.
  - DownToAmountRequested: A ratio of the amount put down to the total value of the loan.

hasEnoughMoney	LoanStatus 0	LoanStatus 1
0	1402	7
1	58986	49459



Feature Selection

- Feature selection is a important process, which helped us to recognize and remove noise or non-relevant features, and therefore improved model accuracy.
- Non-relevant features example: VehicleMake, CoMonthlyRent

Neural Network

- The state-of-the-art Artificial Neural Networks (ANN) is best known for classification tasks.
- Some methods used:
    - Back-propagation
    - Sum of Square Error (SSE) loss function
    - Cross validation
- Things tried:
    - Remove outliers
    - Data normalization
    - Data imputation
- Performance metric:
    - Confusion matrix
    - Accuracy

```
[1] "Confusion matrix for NN:"
> print(nnCM)

pred  0  1
   0 79 28
   1 27 66
> NNaccuracy <- sum(diag(nnCM))/sum(nnCM) #accuracy
> cat('Accuracy for NN: ', NNaccuracy*100, "%")
Accuracy for NN:  72.5 %
```

Models

- We went through several models : Neural Network, Logistic Regression with Lasso, Random Forests, before choosing Boosting. Boosting gives us the best result.

- We chose this model because of its high performance and flexibility, and it can avoid overfitting, which means the model can give good result also on real life data.

Model	Accuracy
Neural Network	72%
Logistic Regression	80%
Random Forests	86%
Boosting	88%

Final Solution with Boosting

- After chose Boosting, we did parameter tuning to achieve higher accuracy.
- Checked false positive and false negative rate. If we set threshold from 50% to 70%, and false positive rate reduce from 7% to 3.5%.

```
predicted.boost    0    1
              0 2706  263
              1  381 2143
> mean(predicted.boost == data.test$LoanStatus)
[1] 0.8827598762
```

Prediction criteria 0.5

```
predicted.boost.1    0    1
                0 2892  570
                1  195 1836
> mean(predicted.boost.1 == data.test$LoanStatus)
[1] 0.8607318405
```

Prediction criteria 0.7

Conclusions

- Our model achieved a cross-validation accuracy of 88%.
- Most important features are ModifiedCreditScore and DTI.

	var	rel.inf
ModifiedCreditScore	ModifiedCreditScore	57.45204634753
DTI	DTI	13.20460846023
EstimatedMonthlyPayment	EstimatedMonthlyPayment	4.10811302500
AmountRequested	AmountRequested	3.89263536231
VehicleMileage	VehicleMileage	3.78650731701
LTV	LTV	2.40609935557
MemberIndicator	MemberIndicator	2.32276472486

Future Work

- Try xgboost R library to reduce computation time
- Work on feature engineering and parameter tuning further
- Use Python (SciKit-learn, Seaborn, Tensorflow)
- Python machine learning libraries are mostly multi-core enabled by default, unlike R libraries.
- Create more complex neural network.

?

QUESTIONS