Motivation
Following the Financial Crisis of 2008, credit institutions faced a number of loan defaults. This jeopardized the profitability of banks due to a number of nonperforming loans. Following these developments, there is a greater need for models not only to detect the risks at higher accuracy, but also deliver the detection at an early stage.

Problem Definition
The goal is to use a borrower’s financial history to predict whether an individual is likely to default on a loan or not. The input is given as financial history and the output is either 0 to mean ‘paid off’ or 1 to mean ‘defaulted’.

Features
The original dataset had 132 features and 819,501 observations. We removed irrelevant, scarce and protected features and retained 24 best features. E.g. mortgage account, annual income, emp_length etc.

Feature Encoding
We encoded the categorical features using feature hashing and one hot encoding. For feature hashing, we hashed the zip codes in a smaller set of finite integer values and fed these values to our model. We used the one hot encoding scheme to transform each attribute into m binary features where the label corresponding to the attribute is encoded as 1 and the rest are zeros.

Models/Approaches
Training and Test Size
Training set: ~655690
Test set: ~163901

Logistic Regression
For logistic regression we settled for a linear solver and balanced class weights.

Random Forest
The random forest model used 100 estimators since that’s what helped us achieve the best accuracy.

Light GBM
This implementation makes use of binary log loss and very low learning rate of 0.001

Multi Layer Perceptron
For MLP we used ReLu activation and adam optimization.

XGB Boost
For this implementation we decided to use early stopping so as not to overfit the data.

Results/Analysis

<table>
<thead>
<tr>
<th>Model</th>
<th>Logistic Regression</th>
<th>MLP</th>
<th>Random Forest</th>
<th>Xgb Boost</th>
<th>Light GBM</th>
</tr>
</thead>
<tbody>
<tr>
<td>Precision</td>
<td>0.70</td>
<td>0.78</td>
<td>0.78</td>
<td>0.78</td>
<td>0.78</td>
</tr>
<tr>
<td>Recall</td>
<td>0.67</td>
<td>0.82</td>
<td>0.82</td>
<td>0.82</td>
<td>0.82</td>
</tr>
<tr>
<td>Accuracy</td>
<td>0.67</td>
<td>0.81</td>
<td>0.81</td>
<td>0.82</td>
<td>0.82</td>
</tr>
</tbody>
</table>

Challenges
- There were a lot of empty or sparse columns and rows which needed intensive prepossessing and feature engineering.
- There were a lot of anonymous features, e.g., zip code, which limited the full potential of feature encoding and evaluation.
- There is a huge disparity in the number of defaulted and number of paid off loans in our dataset.

Future
- We plan to under-sampling negative examples so that we end up with an equal number of positive and negative examples.
- Since under-sampling may result in fewer training examples overall, another approach that can be taken is collecting more data for the positive examples so that we have an almost equal representation of positive and negative classes in our dataset.

References
- Goyal, “Credit Risk Prediction Using Artificial Neural Network Algorithm,”
- Personal Loans Borrow up to $40,000 and get a low, fixed rate