Credit Risk Modelling

Pushkar Sathe and Johann Thiel

Introduction

- Home Credit Group (HCG) is an international lender that focuses on individuals with little or no credit history.
- Lending to individuals with little or no credit history presents HCG with a unique challenge: determining which borrowers are at risk of default.
- Scorecards and other measures can be used to try to assess this risk, but over time they become unreliable due to a variety of factors.
- HCG is hosting a Kaggle competition to predict an individual's default risk.

Stakeholders and Problem

- Stakeholders
  - HCG – avoid lending to high-risk individuals
  - Individual borrowers – obtain loans despite little to no credit history
- Modeling Risk
  - Given a collection of internal and external documents, predict which individuals are likely to default on a loan.
- Stability
  - Once a model is constructed, assess the model's stability over time.
  - This portion favors a model that performs adequately over a longer timeline than one that has a decline in quality even if it has a high level of short-term performance.

Data Set and Challenges

- Large data set, ~26GB
- 32 files from various sources
- ~5.7 billion entries
- ~41% of all entries are NaN
- 465 different features
- Most data is categorical, with a mix of strings, dates and masked values (for e.g. to remove PII).
- Must take less than 12 hours to execute.

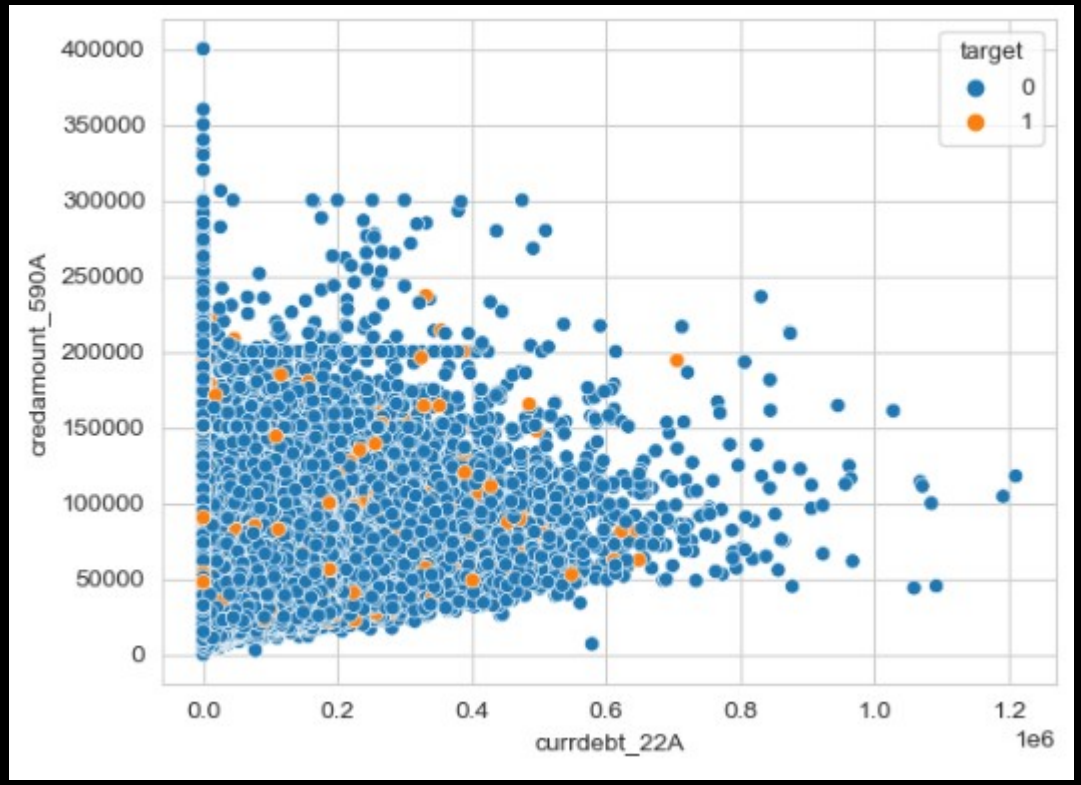
Data Cleanup

- Numerical Only
  - Discarded a significant number of features in favor of numerical ones
  - Discarded NaN entries (significant loss of data)
  - Aggregated using the mean
  - Reduced original dataset to ~100MB

- Categorical Only
  - Converted all numerical features into categorical ones using quartiles
  - Converted NaN entries into their own category
  - Aggregated using the mode
  - Reduced original dataset to a single ~1.8GB file

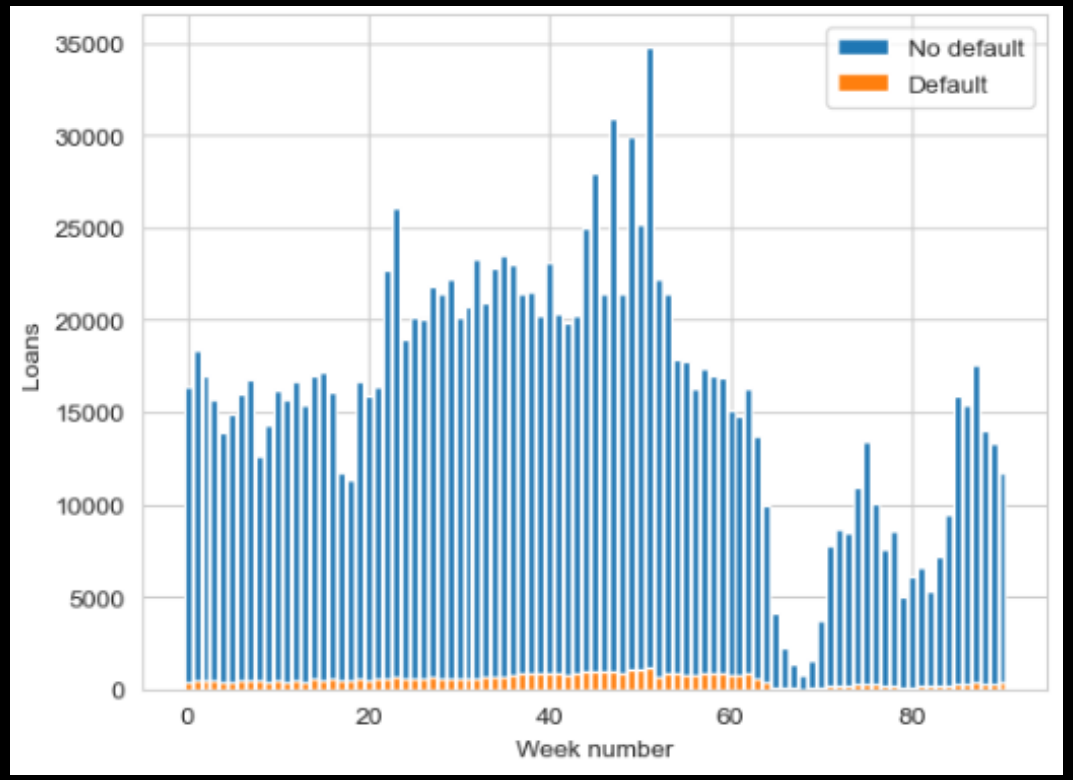
Exploratory Data Analysis

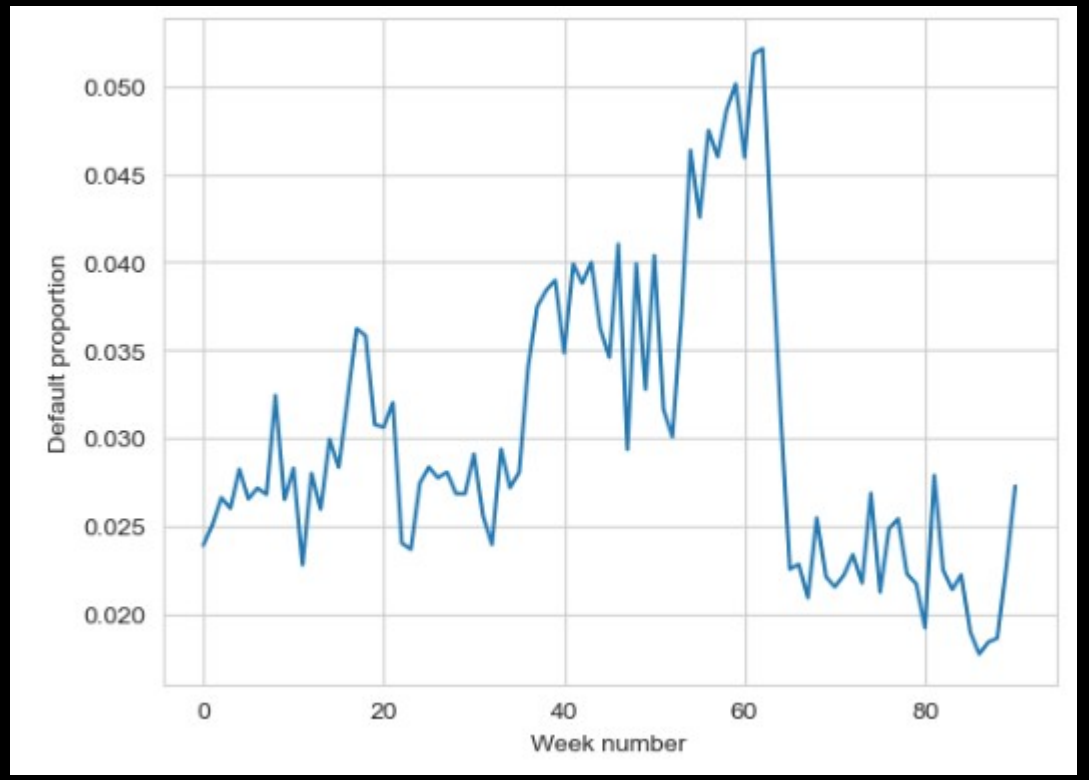
- General Observations
  - Much of the data is masked, making it difficult in some cases to understand a feature
  - Client histories vary vastly
- Numerical Only Dataset
  - Reduces the amount of data available
  - No obvious way separate the target labels
- Categorical Only Dataset
  - Reduces some of the available tools of exploration
  - Adds some value to NaN entries



Exploratory Data Analysis

- Defaults are rare.
- Time itself is a factor but time dependence trends are unclear.





Models

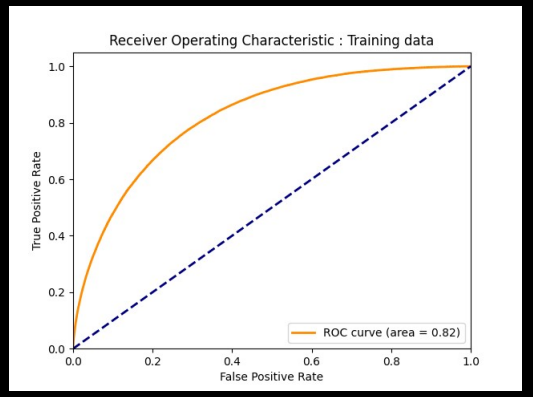
- Logistic Regression
  - Would routinely fail to converge unless the number of features was severely limited (5 were selected at random)
- Decision Tree Classifier
  - Depth 3 trees were used to try to find important features (e.g. -'assignmentdate_4527235D' and 'currdebt_22A')
- AdaBoost using decision tree classifiers
  - Depth 3 trees, 200 estimators
  - Yielded nontrivial results and incorporated a lot of data
- Neural Network (FCN)
  - Trained on binned numerical and one hot encoded categorical data set for over 100 epochs with 2 hidden layers

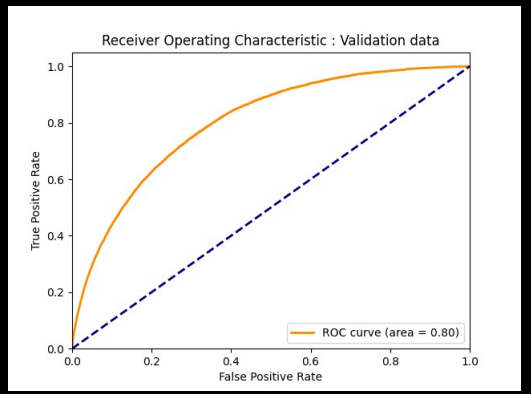
Neural Network

- Only numerical data from master file was considered for the network.
- DataLoader for loading data in batches. Adam optimizer.
- Train-Validation split : 80-20.
- Data overfits to train after

case_id	date_decision	MONTH	WEEK_NUM	target	actualdpd_943P	...	validfrom_1069D
0	1/3/2019	201901	0	0	-1	...	-1
1	1/3/2019	201901	0	0	-1	...	-1
2	1/4/2019	201901	0	0	3	...	-1
3	1/3/2019	201901	0	0	3	...	-1
4	1/4/2019	201901	0	1	3	...	-1
5	1/2/2019	201901	0	0	3	...	-1

DataFrame ingestion





Model Assessment

- An easy method for obtaining high accuracy: always take a 0% chance of default.

- Models were assessed using the receiver operating characteristics (ROC) area under the curve (AUC).

- Model risk results (ROC-AUC)
  - Logistic regression: 0.56251
  - AdaBoost: 0.78133
  - Neural network: 0.82

Model Assessment

- Stability score formula

$$gini_w = 2AUC - 1$$

$$stability = mean(gini_w) + 88.0min(0,a) - 0.5std(r)$$

  - Where $gini_w$ is the gini score per week a and r are the slope and residuals of linear regression on $gini_w$.

- Stability result
  - AdaBoost: 0.53137
  - Neural network: 0.5698

Concluding Remarks

- Much of the work focused on the data set and many of the difficulties in converting to a useful form. This step is critical.
- Additional testing using different parameters in each model could yield better results.
- Additional work to improve the stability score is needed.