Proposal: Identifying Home Credit Application Default Risk

Joel Klein, Rakesh Jothishankar, Suriyadeepan Narayanasamy, Ashley Thornton

Luddy School of Engineering, Indiana University

INFO 526: Applied Machine Learning

James Shanahan, Ph. D.

April 12, 2021

Abstract

Numerous consumers throughout the United States struggle to receive loan support from banks due to lacking credit history. Home Credit is a service whose goal is to provide loan opportunities for this underserved population. Failing to build and implement an accurate repayment detection method assumes major consequences. Missed financial interest opportunity unfolds when denied consumers can truly repay the loan. Consequently, if a loan is granted to consumers likely to default, Home Credit may not recoup the principal. This paper aims to address this issue by proposing a machine learning approach using Home Credit internal and external loan application and credit payment history data for automatic loan default detection. We introduce a simple and explainable logistic regression algorithm with the loan application data. Additionally, we explore more advanced machine learning and deep learning algorithms such as gradient boosting machines and neural networks to improve default classification. The results will show strong performance comparable to existing algorithms scoring near the top of the open-source *Kaggle* leaderboard.

Data

The data provided by Home Credit will be used throughout the project and is available to download from the *Kaggle* website in CSV format. A brief description of the available data is mentioned below:

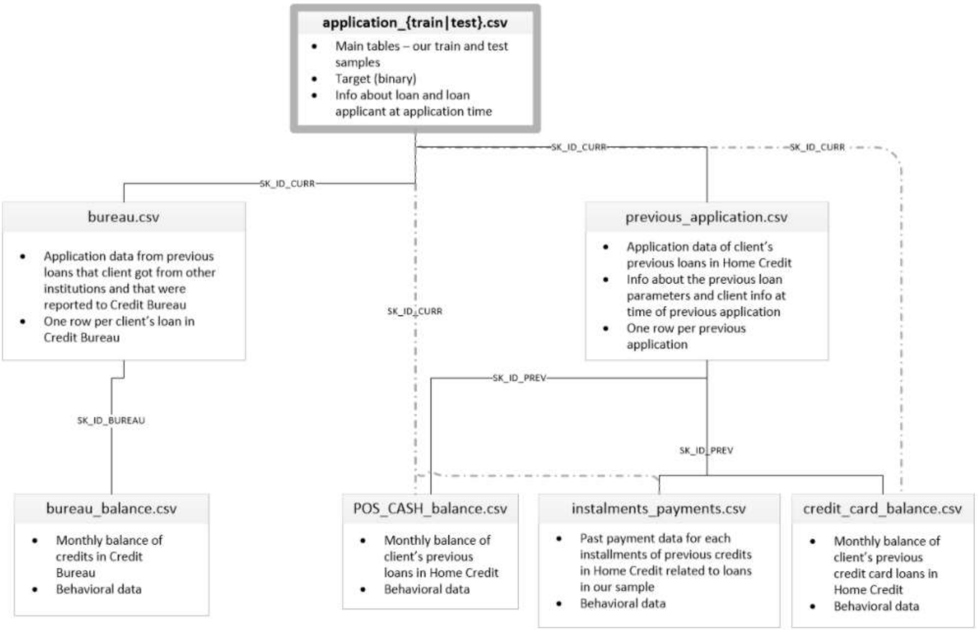

application_{train|test}.csv
Main tables – our train and test samples
Target (binary)
Info about loan and loan applicant at application time
SK_ID_CURR
SK_ID_CURR
SK_ID_CURR
bureau.csv
Application data from previous loans that client got from other institutions and that were reported to Credit Bureau
One row per client's loan in Credit Bureau
SK_ID_CURR
previous_application.csv
Application data of client's previous loans in Home Credit
Info about the previous loan parameters and client info at time of previous application
One row per previous application
SK_ID_BUREAU
SK_ID_PREV
SK_ID_PREV
bureau_balance.csv
Monthly balance of credits in Credit Bureau
Behavioral data
POS_CASH_balance.csv
Monthly balance of client's previous loans in Home Credit
Behavioral data
instalments_payments.csv
Past payment data for each installments of previous credits in Home Credit related to loans in our sample
Behavioral data
credit_card_balance.csv
Monthly balance of client's previous credit card loans in Home Credit
Behavioral data

Figure 1: Home Credit Default Risk Data

*The **Application Data*** file contains information about each loan application at Home Credit. Each loan is identified by the primary key SK_ID_CURR. This file has the most information about the client – gender, income, family status, education, etc. Train and test versions of the file are available. The train version of the file contains the “TARGET” field, which has a value of either “0” or “1”. “0” indicates that a loan has been repaid, whereas “1” indicates that a loan has not been repaid. The test version of the file does not contain the “TARGET” field. 307.5 thousand records with 122 columns are present in the train version of the file, whereas 48.7 thousand records with 121 columns are present in the test version of the file.

The ***Bureau Data*** file contains information about every client's financial information from the various institutions. Prior loan information is present, and it has its own row in this file. 1.7m records with 17 columns are present in this file.

The ***Bureau Balance*** file contains monthly data about previous credits in each bureau. A single credit can be present in multiple rows – one for each month of the credit length. 27.3 million records with 3 columns are present in this file.

The ***Previous Application*** file contains information about client's previous loan in Home Credit. Various loan parameters from the past along with client information at the time of previous application is available. Each prior application has one row in this file. Each row is uniquely identified by a primary key – SK_ID_PREV. 1.7 million records along with 37 columns are present in this file.

The ***POS Cash Balance*** file contains monthly balance information maintained by clients in their previous loan in Home Credit. Each row contains one month of a credit balance, and a single credit can be present in multiple rows. 10 million records along with 8 columns are present in this file.

The ***Installments Payment*** file contains information about installment payments made by the clients in their previous loan in Home Credit. One row for every payment made and one row for every payment missed is present in this file. 13.6 million records along with 8 columns are present in this file.

The ***Credit Card Balance*** file contains information about monthly balances maintained by the clients in their previous credit card loans in Home Credit. 3.8 million records along with 23 columns are present in this file.

Additionally, “HomeCredit_columns_description.csv” file is available, containing the definition for all columns present in the seven data files.

Exploratory Data Analysis (EDA)

The available datasets will be analyzed and investigated using visualization methods. As part of this step, patterns in the given data, relationship between fields, outlier or anomaly detection and data treatment will be done. Visualizations generated as part of this step will help in detecting outliers, which could impact estimations. These outliers could have been due to artificial or natural errors. Once outliers are determined, it will then help in finalizing the needed manipulation on the data to obtain the best results. Data treatment will be done either by deletion or by imputation using mean, median or mode.

EDA – Initial Assessment

Target value distribution has been explored on the training dataset. There is an imbalance between the number of records indicating that the loan was repaid and those indicating that the loan was not repaid.

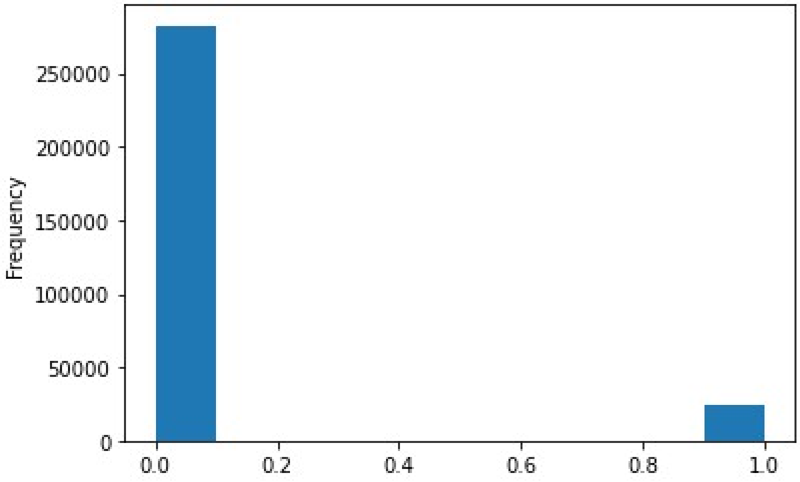
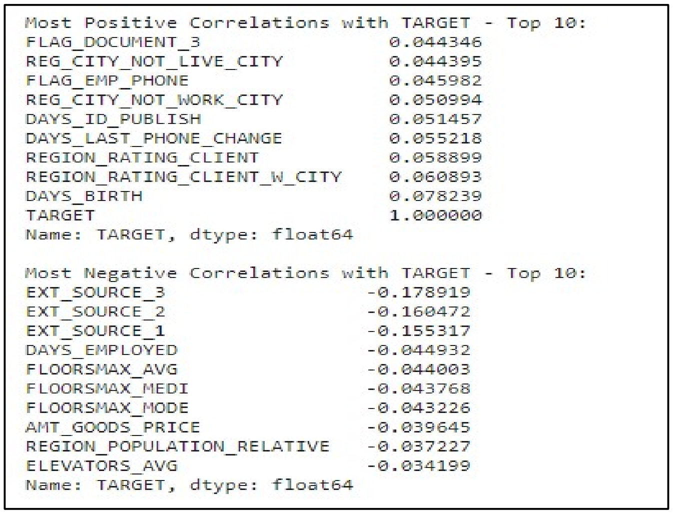


Figure 2: Distribution of target variable

The correlation of the fields against the target column has been explored and provided below:

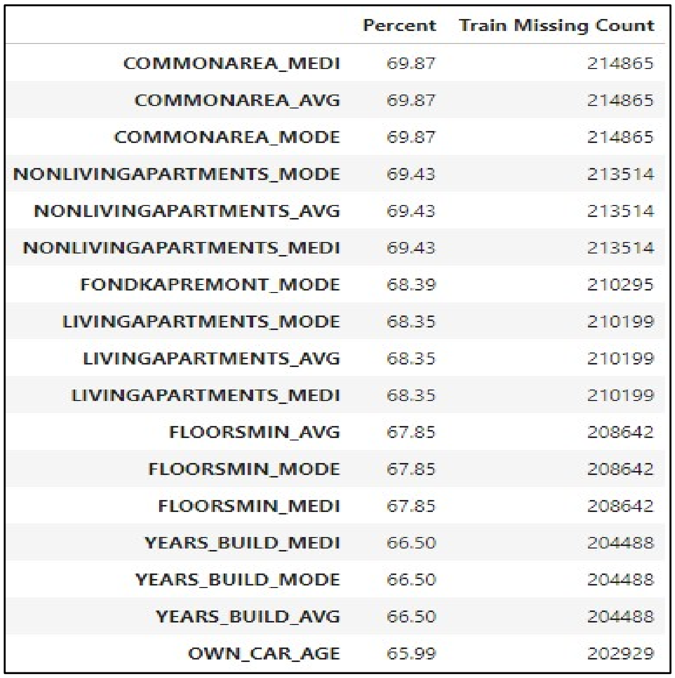


```
Most Positive Correlations with TARGET - Top 10:
FLAG_DOCUMENT_3                0.044346
REG_CITY_NOT_LIVE_CITY         0.044395
FLAG_EMP_PHONE                 0.045982
REG_CITY_NOT_WORK_CITY         0.050994
DAYS_ID_PUBLISH                0.051457
DAYS_LAST_PHONE_CHANGE         0.055218
REGION_RATING_CLIENT           0.058899
REGION_RATING_CLIENT_W_CITY    0.060893
DAYS_BIRTH                     0.078239
TARGET                         1.000000
Name: TARGET, dtype: float64

Most Negative Correlations with TARGET - Top 10:
EXT_SOURCE_3                 -0.178919
EXT_SOURCE_2                 -0.160472
EXT_SOURCE_1                 -0.155317
DAYS_EMPLOYED                -0.044932
FLOORSMAX_AVG                -0.044003
FLOORSMAX_MEDI               -0.043768
FLOORSMAX_MODE               -0.043226
AMT_GOODS_PRICE              -0.039645
REGION_POPULATION_RELATIVE   -0.037227
ELEVATORS_AVG                -0.034199
Name: TARGET, dtype: float64
```

Figure 3: Correlations with target variable

The percentage of missing values has been checked on the training dataset. There are no columns with more than 70% missing data. However, there are 17 columns which have over 60% missing data.



	Percent	Train Missing Count
COMMONAREA_MEDI	69.87	214865
COMMONAREA_AVG	69.87	214865
COMMONAREA_MODE	69.87	214865
NONLIVINGAPARTMENTS_MODE	69.43	213514
NONLIVINGAPARTMENTS_AVG	69.43	213514
NONLIVINGAPARTMENTS_MEDI	69.43	213514
FONDKAPREMONT_MODE	68.39	210295
LIVINGAPARTMENTS_MODE	68.35	210199
LIVINGAPARTMENTS_AVG	68.35	210199
LIVINGAPARTMENTS_MEDI	68.35	210199
FLOORSMIN_AVG	67.85	208642
FLOORSMIN_MODE	67.85	208642
FLOORSMIN_MEDI	67.85	208642
YEARS_BUILD_MEDI	66.50	204488
YEARS_BUILD_MODE	66.50	204488
YEARS_BUILD_AVG	66.50	204488
OWN_CAR_AGE	65.99	202929

Figure 4: Missing data investigation

Methodology

Pipelines

Pipelines are used to concatenate multiple processes from preprocessing, feature engineering, model training, and model evaluation.

Baseline – logistic regression (regularized)

As part of pre-processing, we will create sub-pipelines for processing categorical and numerical features. Some numeric features may require log or square root transformations to normalize the distribution. We will use only existing features available in the provided data. Based on the exploratory data analysis, we will select the numerical or categorical features. We will perform imputation for missing numeric data using the 'Mean' or 'Median' strategy based on skewness in the data. The numerical features will undergo standardization, and the categorical features will undergo one hot encoding transformation, or ordinal encoding based on their data type. We will consolidate the sub pipelines into a data preparation pipeline. We will create the full pipeline by combining the data preparation and logistic regression model steps. If needed, we will do a grid search using the full pipeline to determine the best regularization and penalty parameters for logistic regression. We will use this full pipeline to perform training, tuning, evaluation, and prediction.

Ensemble – random forest, xgboost

The features from the bureau loan and balance, previous application, and payment history files will be merged to the application data after aggregation data operations.

The bureau loan balance data will be merged to the bureau previous loan data (by the bureau loan id) and aggregated balance features will be generated for each previous bureau loan. Aggregations may include min, max, mean, standard deviation, and skewness. This bureau data set will then be merged to the application data (by the application loan id) and aggregated bureau features will be generated for each loan application. Aggregations may include min, max, mean, standard deviation, and skewness.

The previous application balance data sets will be merged to the previous loan application data (by the previous application id) and aggregated balance features will be generated for each previous loan application. Aggregations may include min, max, mean, standard deviation, and skewness. This previous application data set will then be merged to the application data (by the application loan id) and aggregated previous application features will be generated for each loan application. Aggregations may include min, max, mean, standard deviation, and skewness.

We will update the data preparation sub-pipeline from the baseline to use new features. The feature engineering step from the initial pipeline will use the joined data output from the aggregation and join operations. We will update feature engineering sub-pipelines to derive and transform new domain-specific fields to increase modeling performance. For ensemble models, the missing data handing step will be modified as we anticipate features with high amounts of missingness.

We will use grid search with five-fold cross validation to find the best hyperparameters and evaluate performance for the random forest and xgboost models. We will implement feature selection based on feature importance evaluation from initial random forest and xgboost methods to reduce the large feature set and compare results without feature selection. We will modify the pipelines according to the different experiments we conduct.

Neural Network – multilayer perceptron

We will use Keras/TensorFlow to create the multilayer neural network. The cost function will be binary cross-entropy. We will use data from phases 1 and 2 to train the neural network. We will determine the number of hidden layers by exploring the different options and understanding perception. We will perform the grid search with parameters like the number of batches, the number of epochs, different learning rates, etc. We plan to use kerasclassifer as part of sklearn and merge it with the data pipeline we created in phase 2 to perform tuning, training, evaluation, and prediction.

Models

Baseline – logistic regression (regularized)

For the first phase of the project, phase 0, the only model we will fit is logistic regression with the target being a binary field of 1 if the customer had difficulties with repayment and 0 if no issues incurred. For this baseline model, we plan to use all available variables within the application data source using one hot encoding for categorical features. The accuracy score obtained from this model will be used as a target to beat for all subsequent models. We plan to utilize the training dataset from *Kaggle*, randomly splitting into 80% for our training purposes and 20% as our test set. We will use the random state field for reproducibility.

Ensemble – random forest, xgboost

For the next phase of the project, phase 1, we plan to test out additional machine learning algorithms as well as utilize additional data sources. For the data, we will join in the bureau, previous application, and payment history data and apply feature selection as outlined in the Pipeline section of this report. From our new dataset, we will again apply logistic regression, plus random forest, xgboost, and any other algorithms we suspect will perform well. We will run each model multiple times adjusting the settings, such as the learning rate for logistic regression and the tree depth for random forest and xgboost, to find the model with the highest accuracy score.

Neural Network – multilayer perceptron

For the last phase of the project, phase 2, we will implement a neural network on the dataset created in phase 1. For this model, we will vary the settings such as the number of hidden layers until we are content with our accuracy score. At this point, we will decide on our final model based on a number of evaluation metrics, described in the next section.

Evaluation Metrics

Within the *Kaggle* competition, models are evaluated based on the area under the ROC curve between the predicted probability and the observed target. Therefore, this will be our main evaluation metric. In addition, we plan to calculate and review the classification accuracy, F1 score, precision, and recall for each model helping to inform our final model decision.

Project Timeline

CRISP-DM

This project will abide by the Cross Industry Standard Process for Data Mining (CRISP-DM) framework. The CRISP-DM framework consists of six revolving sections: business understanding, data understanding, data preparation, modeling, evaluation, and deployment. Since this modeling exercise is for research purposes only, the deployment step is ignored.

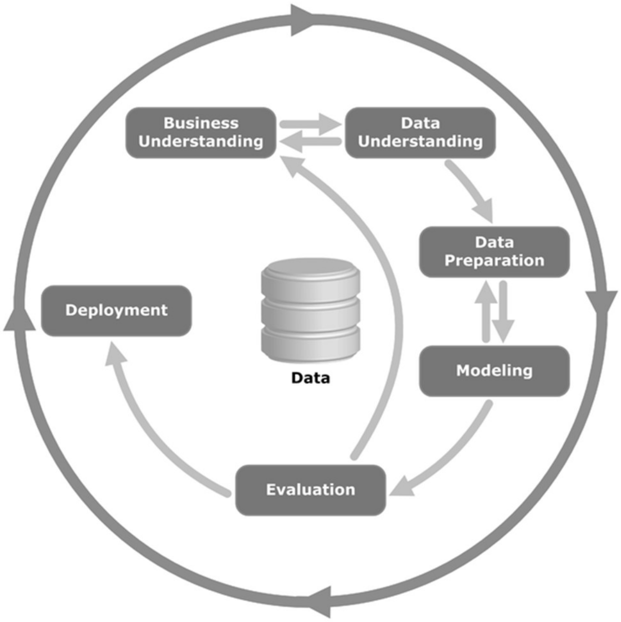


Figure 5: CRISP-DM Framework

Roles & responsibilities

Roles and responsibilities are assigned to each sub-phase of the CRISP-DM process. There are three phases to the project contributing a subset of incremental deliverables towards final submission.

Phase 1:

The deliverable for phase one is a baseline logistic regression default detection model leveraging only the loan application source data.

No.	Deliverable Phase 1: Baseline Model & Report (application data)	Responsible
1.1	Load data	Ashley & Rakesh
1.2	Exploratory data analysis	Ashley & Rakesh
1.3	Test design (models & metrics)	Joel & Suriya
1.4	Pipeline (baseline)	
1.4.1	Feature engineering (baseline)	Joel & Suriya Ashley & Rakesh
1.4.2	Models (baseline)	Joel & Suriya
1.4.3	Hyperparameter tuning (baseline)	Joel & Suriya
1.5	Evaluation & results (baseline)	Joel & Suriya
1.6	Kaggle submission	Joel & Suriya
1.7	Brief report	Ashley & Rakesh Joel & Suriya

Table 1: Phase 1 Steps & Responsible Team Members

Phase 2:

Phases two and three extend phase 1, performing more advanced techniques attempting to increase default detection performance. The phase two objective is to build more advanced tree-based ensemble models with the addition of previous Home Credit application and payment history data as well as third-party bureau application and payment history data.

No.	Deliverable Phase 2: Ensemble Model & Report (application, bureau, previous application, payment history data)	Responsible
2.1	Load data (bureau, previous application data, payment history data)	Joel & Rakesh
2.2	Exploratory data analysis (bureau, previous application data, payment history data)	Joel & Rakesh
2.3	Test design (models & metrics)	Ashley & Suriya
2.4	Pipeline (ensemble)	
2.4.1	Feature engineering (ensemble)	Joel & Rakesh Ashley & Suriya
2.4.2	Models (ensemble)	Ashley & Suriya
2.4.3	Hyperparameter tuning (ensemble)	Ashley & Suriya
2.5	Evaluation & results (ensemble)	Ashley & Suriya
2.6	Kaggle submission	Ashley & Suriya
2.7	Brief report	Joel & Rakesh Ashley & Suriya

Table 2: Phase 2 Steps & Responsible Team Members

Phase 3:

The final phase, phase 3, extends phase 2 learnings to enable a deep learning neural network classification model fitting and evaluation. The end deliverables consist of a final model selection, *Kaggle* submission, final report, and short research presentation.

No.	Deliverable Final Submission: NN Model & Report (application, bureau, previous application, payment history data)	Responsible
3.1	Test design (models & metrics)	Joel & Ashley
3.2	Pipeline (DNN)	
3.2.1	Feature engineering (DNN)	Joel & Ashley
3.2.2	Models (DNN)	Rakesh & Suriya
3.3	Evaluation & results (DNN)	Rakesh & Suriya
3.4	Kaggle submission	Rakesh & Suriya
3.5	Final report	Everyone
3.6	Final presentation	Everyone
3.6.1	Prepare presentation	Everyone
3.6.2	Record presentation	Everyone
3.6.3	Post presentation	Everyone

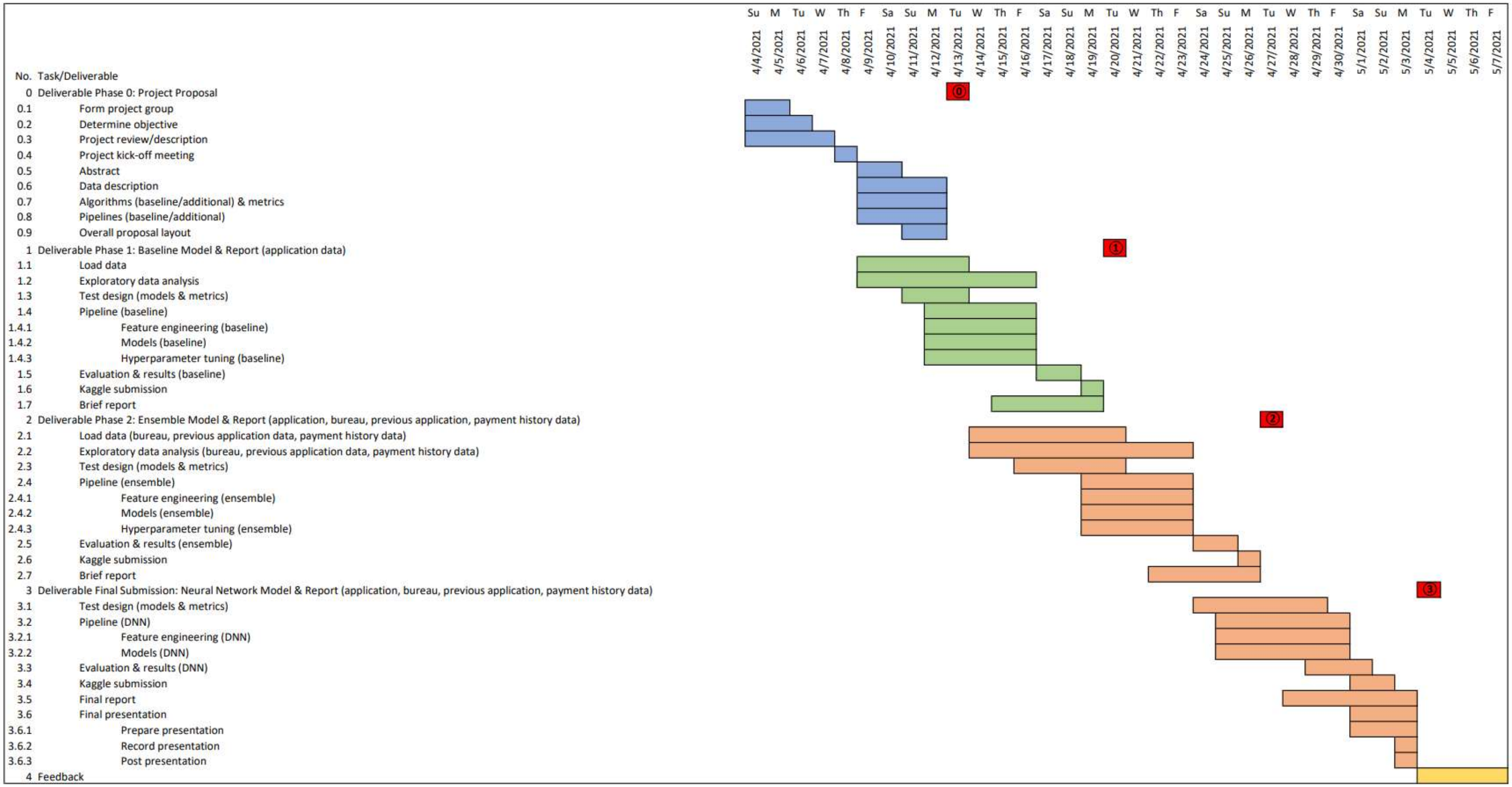
Table 3: Phase 3 Steps & Responsible Team Members

For further deliverable timing detail, view the Gantt outline in figure 3 of the Appendix.

References

Home Credit Group. (2018, May 17). *Home Credit Default Risk*. Kaggle. https://www.kaggle.com/c/home-credit-default-risk/

Appendix



No.	Task/Deliverable
0	Deliverable Phase 0: Project Proposal
0.1	Form project group
0.2	Determine objective
0.3	Project review/description
0.4	Project kick-off meeting
0.5	Abstract
0.6	Data description
0.7	Algorithms (baseline/additional) & metrics
0.8	Pipelines (baseline/additional)
0.9	Overall proposal layout
1	Deliverable Phase 1: Baseline Model & Report (application data)
1.1	Load data
1.2	Exploratory data analysis
1.3	Test design (models & metrics)
1.4	Pipeline (baseline)
1.4.1	Feature engineering (baseline)
1.4.2	Models (baseline)
1.4.3	Hyperparameter tuning (baseline)
1.5	Evaluation & results (baseline)
1.6	Kaggle submission
1.7	Brief report
2	Deliverable Phase 2: Ensemble Model & Report (application, bureau, previous application, payment history data)
2.1	Load data (bureau, previous application data, payment history data)
2.2	Exploratory data analysis (bureau, previous application data, payment history data)
2.3	Test design (models & metrics)
2.4	Pipeline (ensemble)
2.4.1	Feature engineering (ensemble)
2.4.2	Models (ensemble)
2.4.3	Hyperparameter tuning (ensemble)
2.5	Evaluation & results (ensemble)
2.6	Kaggle submission
2.7	Brief report
3	Deliverable Final Submission: Neural Network Model & Report (application, bureau, previous application, payment history data)
3.1	Test design (models & metrics)
3.2	Pipeline (DNN)
3.2.1	Feature engineering (DNN)
3.2.2	Models (DNN)
3.3	Evaluation & results (DNN)
3.4	Kaggle submission
3.5	Final report
3.6	Final presentation
3.6.1	Prepare presentation
3.6.2	Record presentation
3.6.3	Post presentation
4	Feedback

Figure 6: Final Project Gantt Timeline