LendingClub Predictions

Overview of the Modelling Steps

Objective

Predict:

- Accepted vs. rejected
- Fully Paid vs. Charged Off
- Grade
- Sub-grade
- Interest Rate

Data Preparation

Select data according to needs of the model:

Accepted and rejected data

- Load cleaned data
- Create sample of 10000 rows from rejected
- Create sample of ‘data_set_size_ratio’ rows from accepted
- Merge samples

Accepted data

- Load cleaned data
- Select paid/charged off
- Drop minority classes in purpose and home ownership
- Pick most recent year (time dependent importance of FICO)
- Create sample of 10000 rows

Cross-Validation, Score Metric Definition, Estimators Setup

Select CV-methods, metrics, and models:

Regression

- CV: RepeatedKFold (splits=5, repeats=10)
- Scoring: MAE
- LinearRegression
- Ridge
- KNeighborsRegressor
- LGBMRegressor
-

Classifiers

- CV: (Repeated)StratifiedKFold (splits=5, repeats=10)
- Scoring: balanced accuracy
- SVC(kernel='linear')
- SVC(kernel=’rbf’)
- LogisticRegression
- Ridge
- KNeighborsClassifier
- DecisionTreeClassifier
- RandomForestClassifier
- LGBMClassifier
- XGBoostClassifier

Features selected and Test - Train - Split

Features in ‘accepted’ data

- Numerical: 'inq_last_12m', 'dti', 'loan_amnt', 'delinq_2yrs', 'pub_rec_bankruptcies', 'fico_range_high', 'annual_inc', 'mo_sin_old_il_acct'
- Categorical: term', 'emp_length', 'home_ownership', 'verification_status', 'purpose'
- Tragets: 'grade', 'sub_grade', 'loan_status_simple', 'int_rate'

Features in ‘accepted’ and ‘rejected data

- Numerical: 'Amount Requested', 'Risk_Score', 'dti'
- Categorical: Employment Length',
- Target: 'App_Status'

LabelEncoder for target classes

Test - Train - Split

- test_size=0.2, random_state=0
- stratify = y for target classes

Preprocessor for Pipeline

Categorical

Ordinal Encoding:

- SimpleImputer(strategy="most_frequent")
- term', 'emp_length'/Employment Length', 'verification_status',

OneHotEncoder:

- SimpleImputer(strategy="most_frequent")
- 'home_ownership', 'verification_status'

Numerical

StandardScaler:

- SimpleImputer(strategy="median")
- 'inq_last_12m', 'dti', 'loan_amnt'/'Amount Requested', 'delinq_2yrs', 'pub_rec_bankruptcies', 'fico_range_high', 'annual_inc', 'mo_sin_old_il_acct'

Feature Selection ‘Sandbox’

SelectKBest

- ‘f_classif’ → selection from f-scores
- Test dependency per feature

Ridge(Classifier)CV

- → selection from ‘Importance’

RFECV

- → selection from ranking
- Test dependency on combination of features

Use this section rather as an orientation than for implementation into pipeline

Scores BEFORE and AFTER Hypertuning

Categorical

Ordinal Encoding:

- SimpleImputer(strategy="most_frequent")
- term', 'emp_length'/Employment Length', 'verification_status',

OneHotEncoder:

- SimpleImputer(strategy="most_frequent")
- 'home_ownership', 'verification_status'

Numerical

StandardScaler:

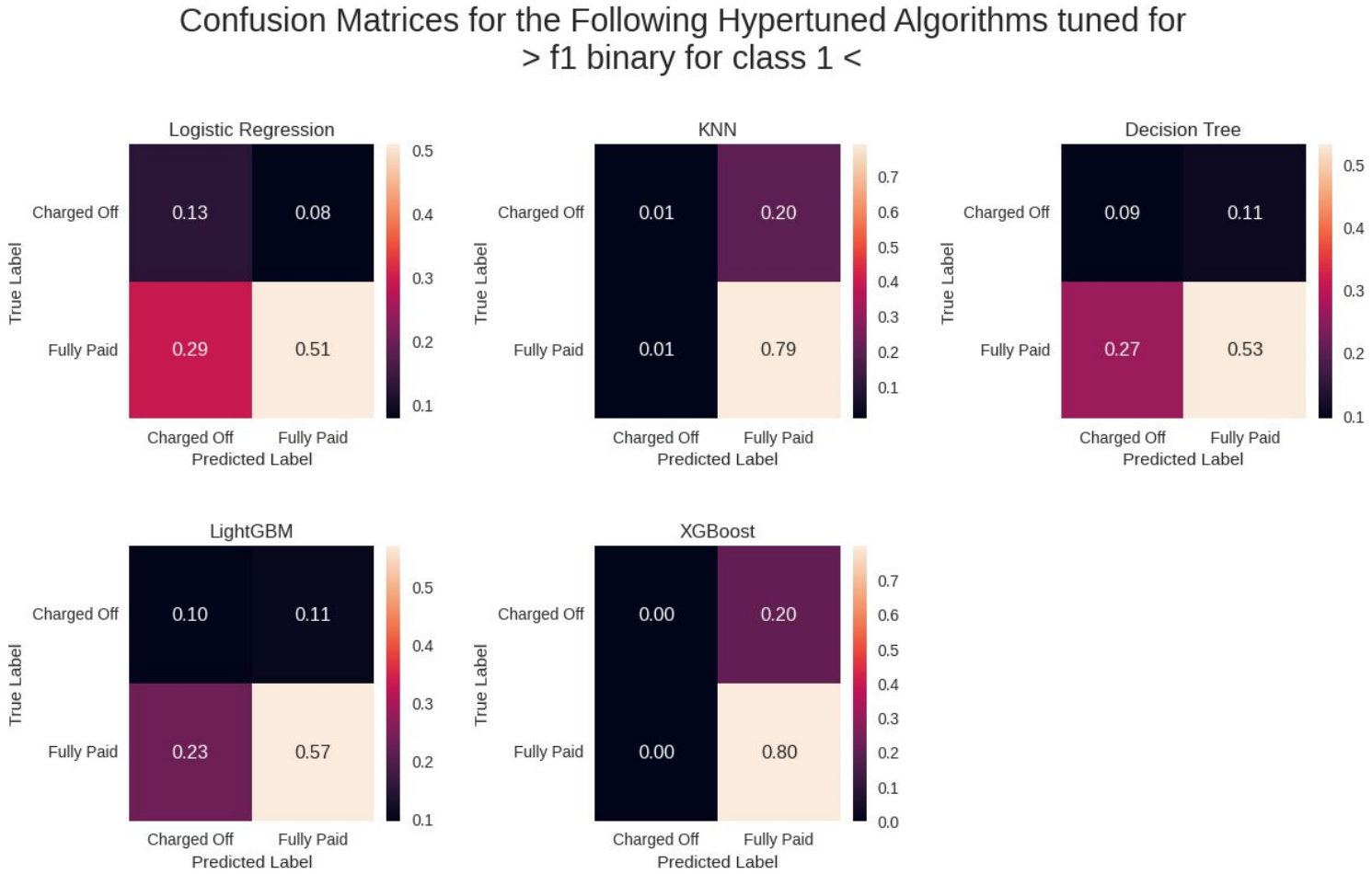
- SimpleImputer(strategy="median")
- 'inq_last_12m', 'dti', 'loan_amnt'/'Amount Requested', 'delinq_2yrs', 'pub_rec_bankruptcies', 'fico_range_high', 'annual_inc', 'mo_sin_old_il_acct'

Experiment:

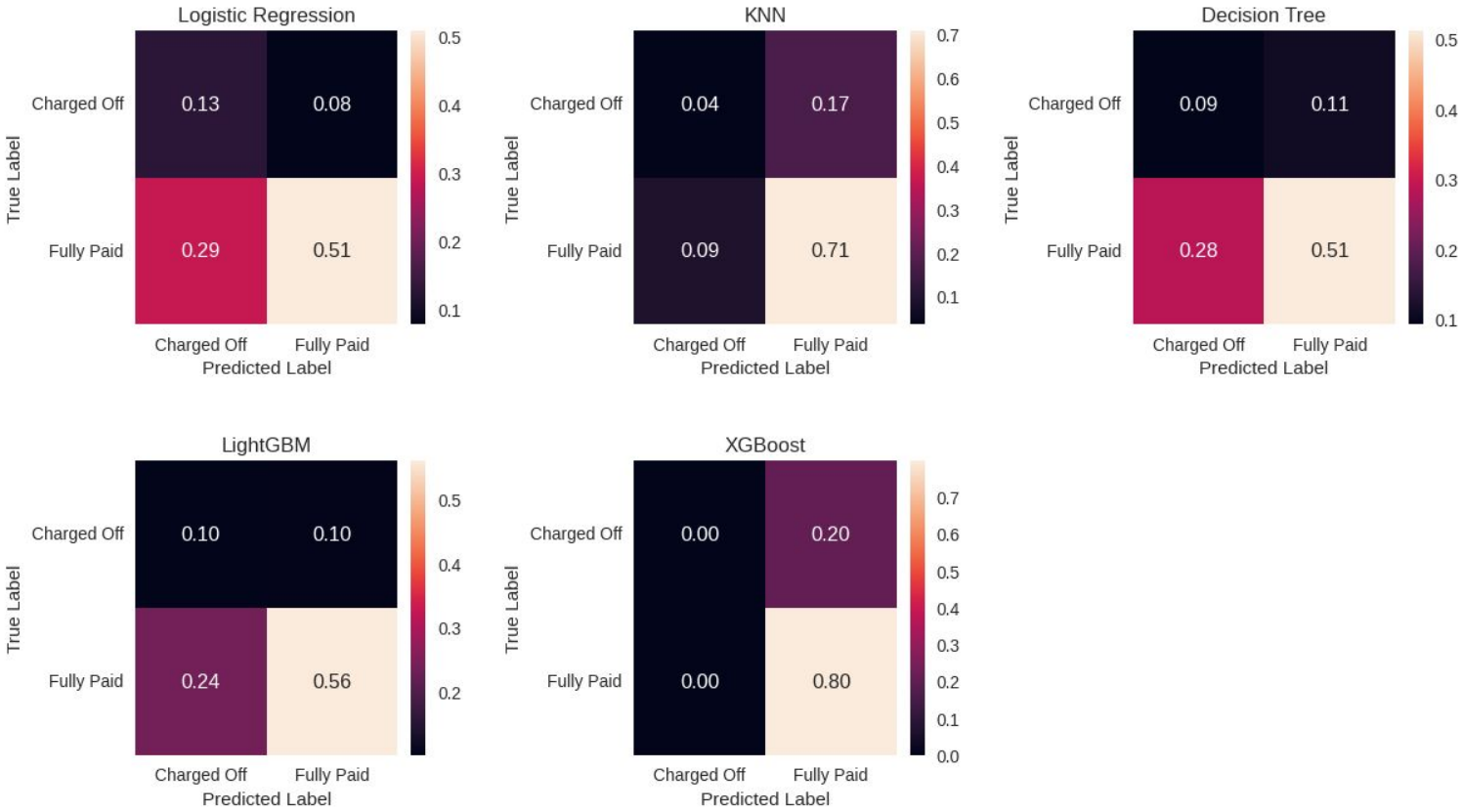
Hypertuning and CV model scores for different score metrics for loan status

Loans status

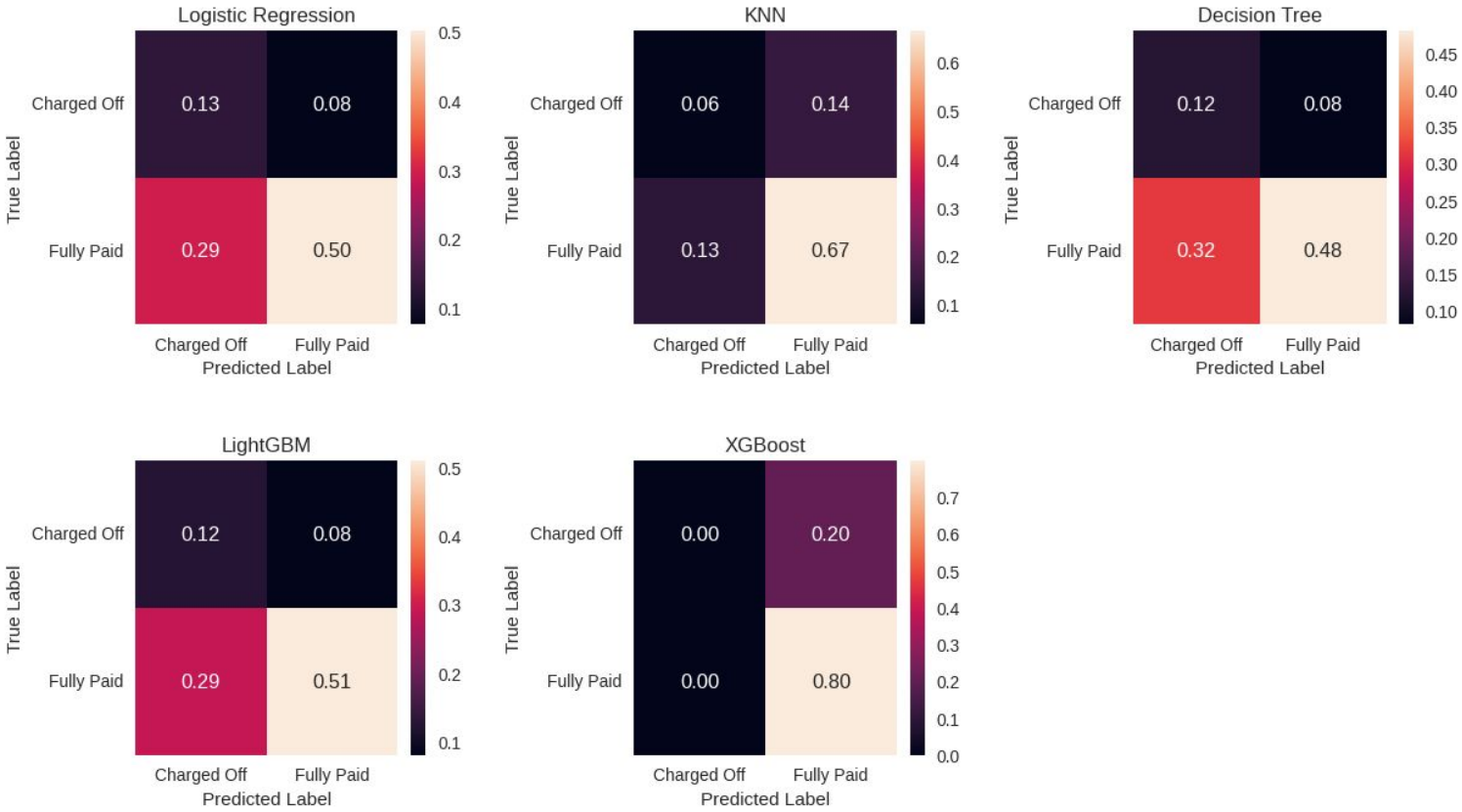
Loans status



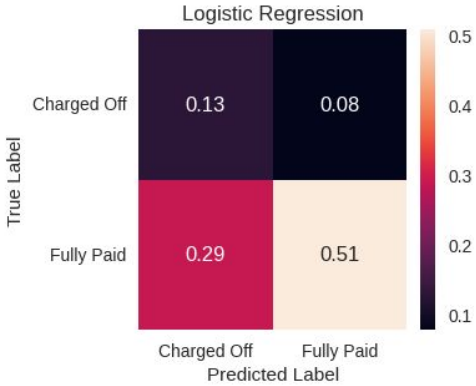
Confusion Matrices for the Following Hypertuned Algorithms tuned for > f1 weighted for class 1 <

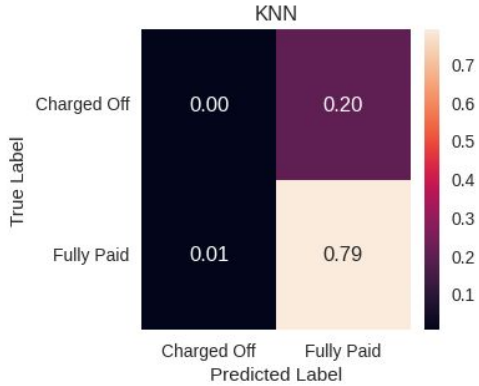


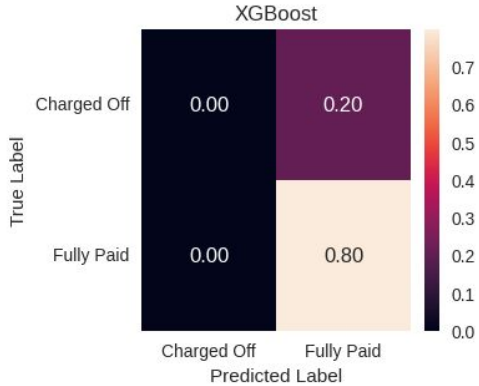
Confusion Matrices for the Following Hypertuned Algorithms tuned for > precision for class 1 <



Confusion Matrices for the Following Hypertuned Algorithms tuned for > recall for class 1 <







Improvements - general

Feature selection/Cleaning

- less strict on mulitcolinearity
- Less strict on dropping but freq-%
- reduce/bundle classes of features
- Missing values in categoricals: Own class instead of imputing
- software, e.g. Boruta
- use logarithmic features

Models and Optimization

- try other model, e.g. CatBoost
- failed attempts: TuneSearchCV (internal bugs), OptunaSearchCV
- Auto-sklearn (grade prediction - not yet finding solutions)

Improvements - model specific

Linear Regression

- clip outliers

Regression/Classifier Trees

- omit scaling of numericals

Loan acceptance/Loan Status

- Language processing of ‘loan_title’/’emp_title’ → purpose
- add states as feature

Grade/Sub-grade

- Classifiers: change class weights manually in estimators
- Linear Regression on ordinal target
- Ordinal Logistic Regression