Comparison between Visualizations of LIME and SimMachines Workbench – Based on HELOC Dataset by FICO

- Dataset overview:
  Dataset content: Home Equity Line of Credit (HELOC) Dataset
  Size: 10459 records with 24 features
  Data type: all numerical data, no categorical data
  Target: RiskPerformance (Bad-1:5459, Good-0:5000)
  Special values in dataset:
  -9 No Bureau Record or No Investigation
  -8 No Usable/Valid Trades or Inquiries - Usable or valid for Accounts/Trades means inactive, or very old.
  -7 Condition not Met (e.g. No Inquiries, No Delinquencies) - "Condition not met," which implies that the feature/variable searched for a certain event's occurrence in the data, and that event was not found.
- Whole process:
  - Preprocessing:
    a. Remove 588 records with missing values in all features
    b. Feature Engineering: split each column into numerical and categorical columns:
       - For numerical-data columns: treat all special values as Null, keep real numbers values
       - For categorical-data columns: treat all real number as 1(i.e in the same group), keep -7,-8,-9 values
  - Doing Supervised clustering on Python:

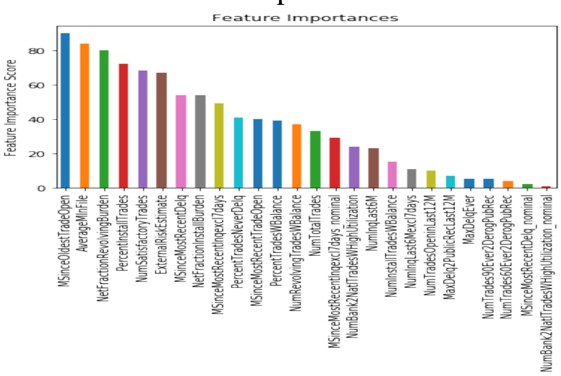
    a. Supervised learning – Xgboost
       - Model &performance

```
Xgboost model:

XGBClassifier(base_score=0.5, booster='gbtree', colsample_bylevel=1,
       colsample_bytree=0.8, gamma=0.3, learning_rate=0.1,
       max_delta_step=0, max_depth=4, min_child_weight=3, missing=None,
       n_estimators=73, n_jobs=1, nthread=4, objective='binary:logistic',
       random_state=0, reg_alpha=0, reg_lambda=1, scale_pos_weight=1,
       seed=27, silent=True, subsample=0.8)
Accuracy: 0.7789
AUC on training dataset:0.862210
AUC on testing dataset:0.803736
```

Not very good but not too bad.

- Feature importance:



Top 10 features (feature importance score>40,descending): MSinceOldestTradeOpen, AverageMInFile, NetFractionInstallBurden, PercentInstallTrades, NumSatisfactoryTrades, ExternalRiskEstimate, MSinceMostRecentDelq, NetFractionInstallBurden, MSinceMostRecentInqexcl7days, PercentTradesNeverDelq

Top Correlation coefficient:

(NumRevolvingTradesWBalance, NumTradesOpeninLast12M)	0.661866
(NumInqLast6M, MaxDelqEver)	0.669120
(NumTradesOpeninLast12M, ExternalRiskEstimate)	0.670877
(NumInqLast6Mexcl7days, MaxDelqEver)	0.673339
(NumSatisfactoryTrades, NumRevolvingTradesWBalance)	0.674215
(NumTrades60Ever2DerogPubRec, PercentTradesNeverDelq)	0.675845
(NumTrades60Ever2DerogPubRec, NumInqLast6M)	0.677402
(MaxDelq2PublicRecLast12M, NumInqLast6M)	0.679375
(NumInqLast6Mexcl7days, NumTrades60Ever2DerogPubRec)	0.681080
(NumInqLast6M, PercentTradesNeverDelq)	0.682192
(NumInqLast6Mexcl7days, MaxDelq2PublicRecLast12M)	0.683039
(ExternalRiskEstimate, NumTrades60Ever2DerogPubRec)	0.684661
(NumInqLast6Mexcl7days, PercentTradesNeverDelq)	0.686419
(NumTradesOpeninLast12M, NumTrades60Ever2DerogPubRec)	0.700333
(NumInqLast6M, NumTrades90Ever2DerogPubRec)	0.703001
(NumTrades90Ever2DerogPubRec, NumInqLast6Mexcl7days)	0.706904
(NumTrades60Ever2DerogPubRec, MaxDelq2PublicRecLast12M)	0.708507
(MaxDelqEver, NumTrades90Ever2DerogPubRec)	0.721690
(MSinceOldestTradeOpen, AverageMInFile)	0.725988
(NumInqLast6M, NumTradesOpeninLast12M)	0.727543
(NumInqLast6Mexcl7days, NumTradesOpeninLast12M)	0.730163
(NumTrades90Ever2DerogPubRec, NumTradesOpeninLast12M)	0.733122
(MaxDelqEver, NumTradesOpeninLast12M)	0.736208
(NumTrades90Ever2DerogPubRec, ExternalRiskEstimate)	0.741375
(NumTradesOpeninLast12M, MaxDelq2PublicRecLast12M)	0.746327
(NumTrades90Ever2DerogPubRec, PercentTradesNeverDelq)	0.751725
(NumTrades90Ever2DerogPubRec, MaxDelq2PublicRecLast12M)	0.762342
(PercentTradesNeverDelq, NumTradesOpeninLast12M)	0.771269
(NumBank2NatlTradesWHighUtilization, NumRevolvingTradesWBalance)	0.791191
(NumTotalTrades, NumSatisfactoryTrades)	0.886282
(ExternalRiskEstimate, MaxDelqEver)	0.890247
(PercentTradesNeverDelq, ExternalRiskEstimate)	0.895690
(MaxDelq2PublicRecLast12M, PercentTradesNeverDelq)	0.907249
(ExternalRiskEstimate, MaxDelq2PublicRecLast12M)	0.908919
(MaxDelq2PublicRecLast12M, MaxDelqEver)	0.924642
(MaxDelqEver, PercentTradesNeverDelq)	0.928059
(NumTrades60Ever2DerogPubRec, NumTrades90Ever2DerogPubRec)	0.975480
(NumInqLast6Mexcl7days, NumInqLast6M)	0.996683

In top 10 features, there are some features with high multicollinearity: 'corr(MSinceOldestTradeOpen, AverageMInFile)= '0.725988', 'corr(ExternalRiskEstimate, PercentTradesNeverDelq)= 0.895690' which make sense: 1. The longer the months since oldest trade open, the longer the average months in file. 2. The higher the credit level(I mean a person have good credit), the higher the percent trades never delinquent. More exploration between variables will be doing later.

b. Supervised learning – HDBSCAN

HDBSCAN source: https://hdbscan.readthedocs.io/en/latest/how_hdbscan_works.html

Introduction

HDBSCAN(Hierarchical Density Based Clustering) is a clustering algorithm developed by Campello, Moulavi, and Sander. It extends DBSCAN by converting it into a hierarchical clustering algorithm, and then using a technique to extract a flat clustering based in the stability of clusters.
I use this algorithm here because it has better performance on computation expense and clustering accuracy.
Source: https://hdbscan.readthedocs.io/en/latest/comparing_clustering_algorithms.html

- Preprocessing:
    - Impute missing values with the nearest neighbor since the algorithm can't do clustering with Null.
    - Only do clustering on target=1(i.e bad) since they are what we really care.

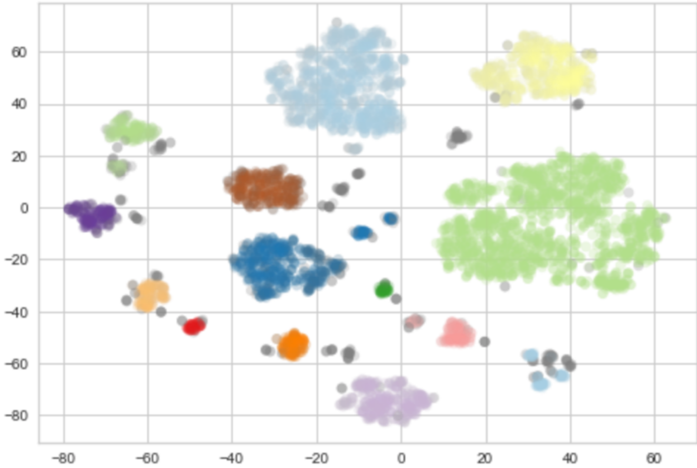
- HDBSCAN result

```
HDBSCAN model:

HDBSCAN(algorithm='best', allow_single_cluster=False, alpha=1.0,
    approx_min_span_tree=True, cluster_selection_method='eom',
    core_dist_n_jobs=4, gen_min_span_tree=False, leaf_size=40,
    match_reference_implementation=False, memory=Memory(location=None),
    metric='euclidean', min_cluster_size=30, min_samples=15, p=None,
    prediction_data=False)

The number of cluster: 15
```

Visualization of clusters using t-SNE



There is no proper benchmarks for HDBSCAN to evaluate its performance, it performances bad on internal indices like silhouette score, however, I keep using it since it intuitively does good clustering based on visualization, also, the further performance of cluster exemplars on LIME can prove that it did a good job.

- Keep cluster exemplars for further use

(Cluster exemplar source: https://hdbscan.readthedocs.io/en/latest/api.html#id33
A list of exemplar points for clusters. Since HDBSCAN supports arbitrary shapes for clusters we cannot provide a single cluster exemplar per cluster. Instead a list is returned with each element of the list being a numpy array of exemplar points for a cluster – these points are the "most representative" points of the cluster.)

c. LIME:

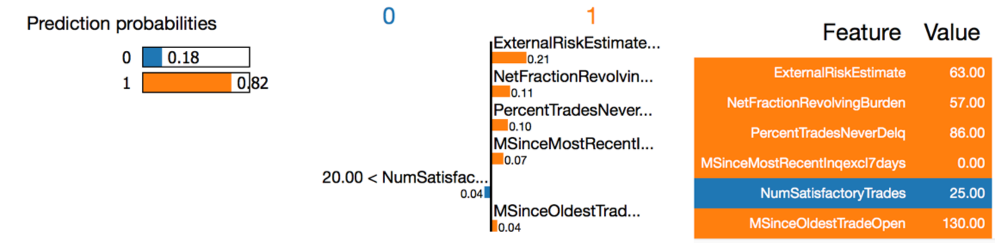
Source: https://github.com/marcotcr/lime
LIME is about explaining what machine learning classifiers (or models) are doing.
We use our Xgboost model here and let it explain our cluster exemplars. Since it can only do explanation on records in test set (i.e records can't be used for model building), I used the intersecting records of test set and cluster exemplars, which are qualified for LIME to explain the result of the supervised clustering we did before.

Since the intersecting records can't cover the whole clusters, we can only get explanation on part of clusters.

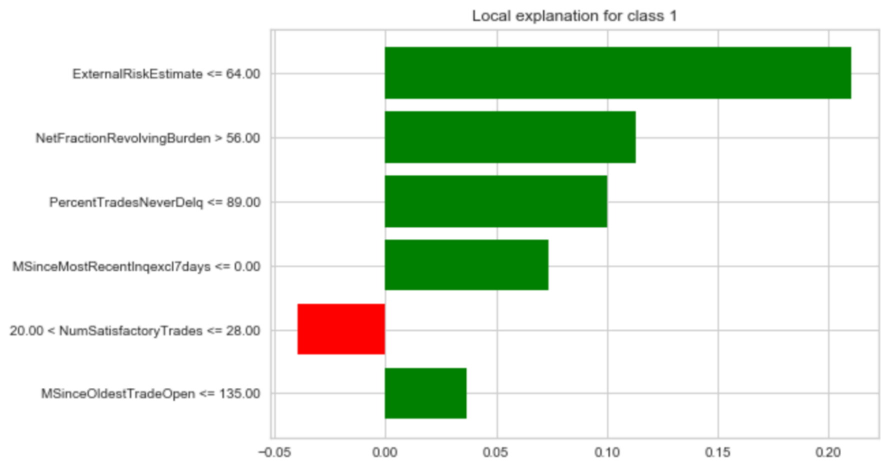
LIME Result:
Generally speaking, LIME results are good, the explanation for each feature fit their monotonicity constraint.

Cluster 1:

cluster 1 - index: 1122



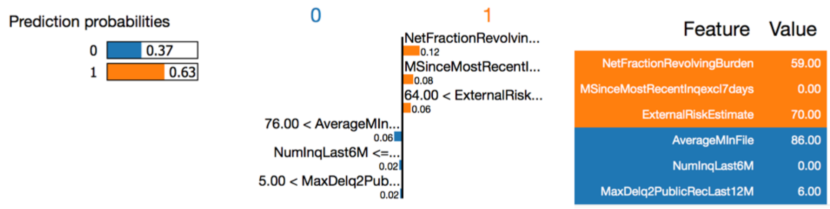
cluster 1 - index: 1122



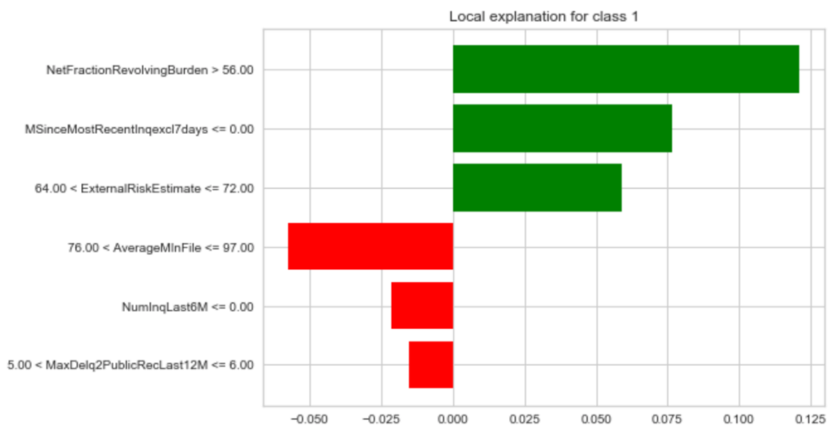
Cluster 2:

Exemplars 1:

cluster 2 - index: 485



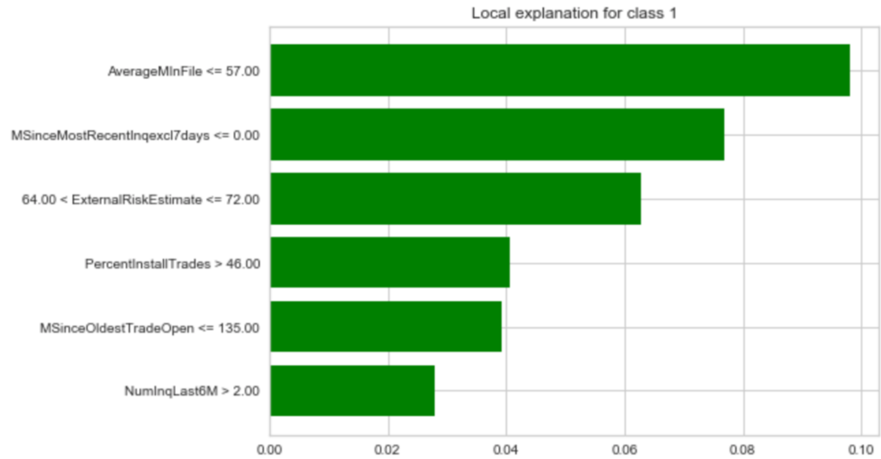
cluster 2 - index: 485



Exemplars 2:

cluster 2 - index: 2643

cluster 2 - index: 2643

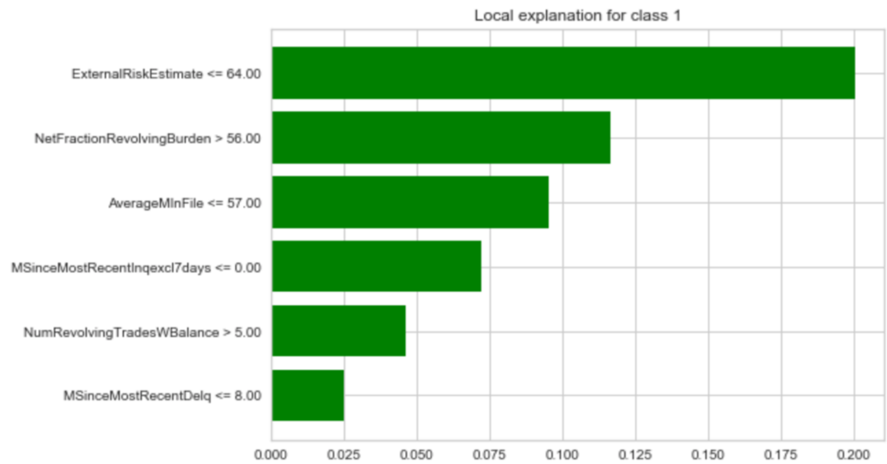


Cluster 6:

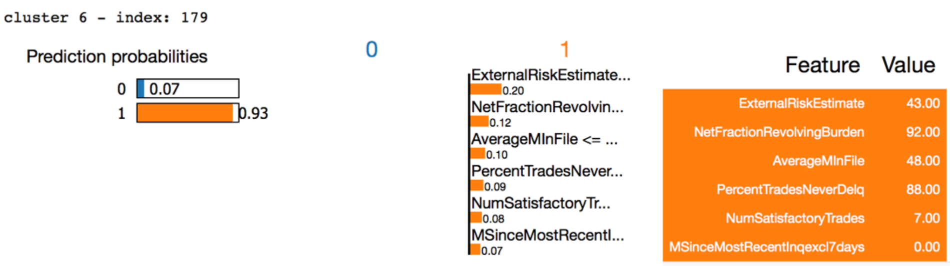
Exemplars 1:

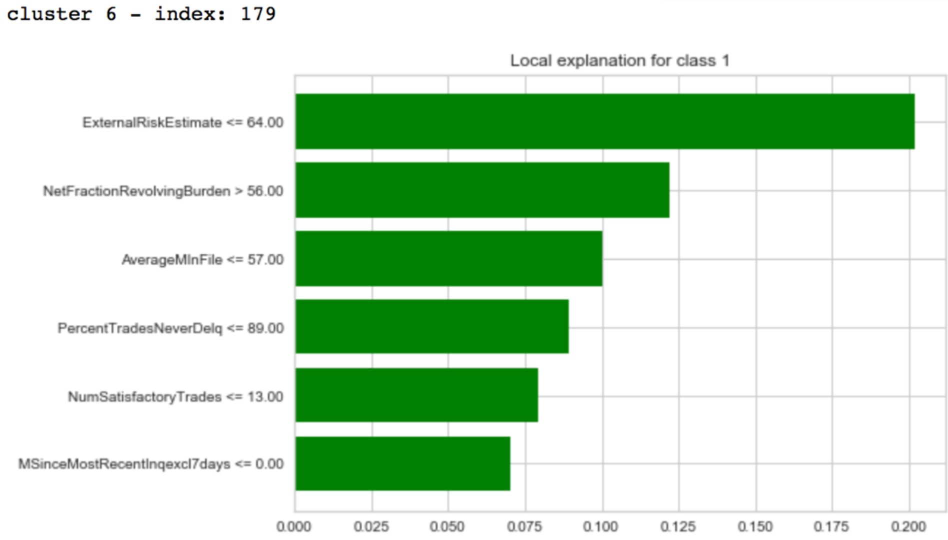
cluster 6 - index: 274

cluster 6 - index: 274

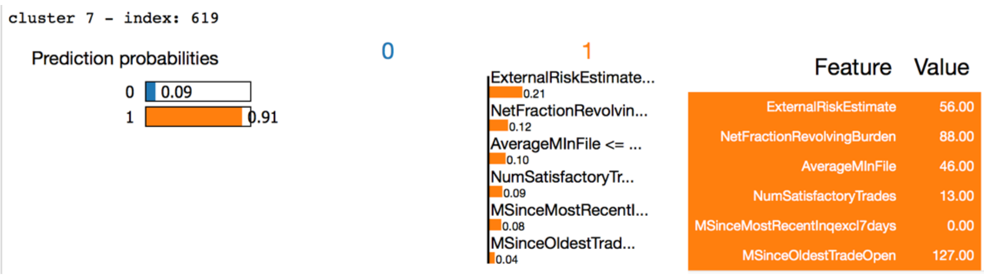


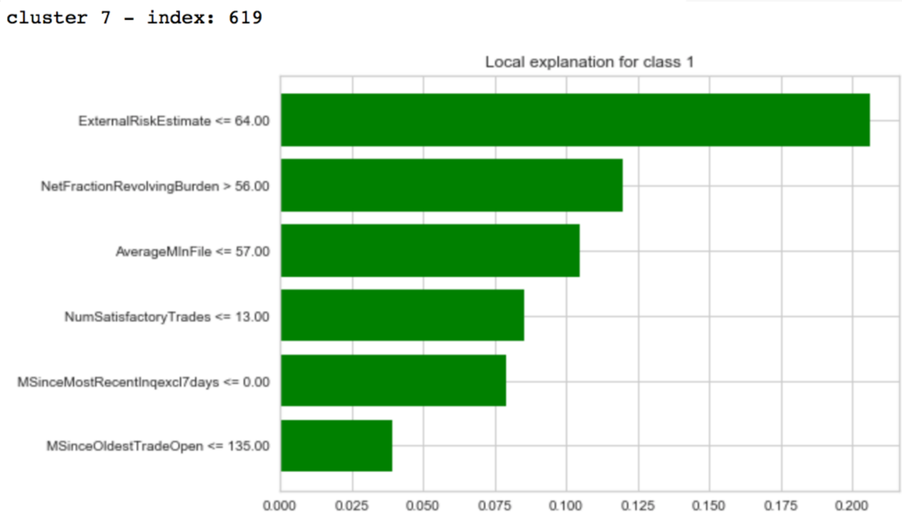
Exemplars 2:





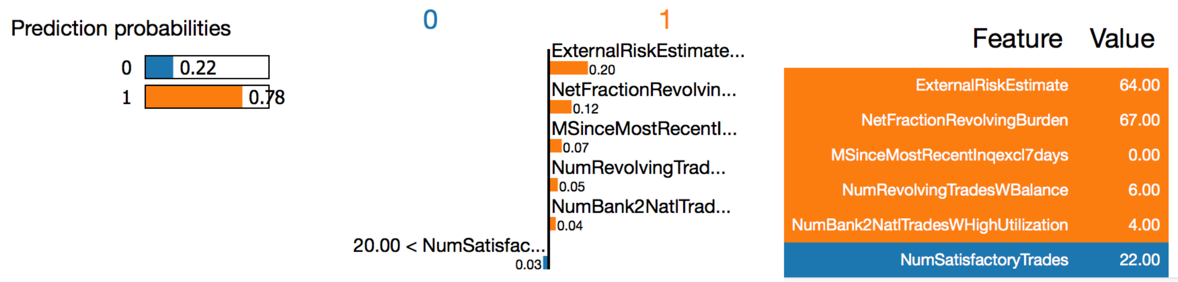
Cluster 7 :



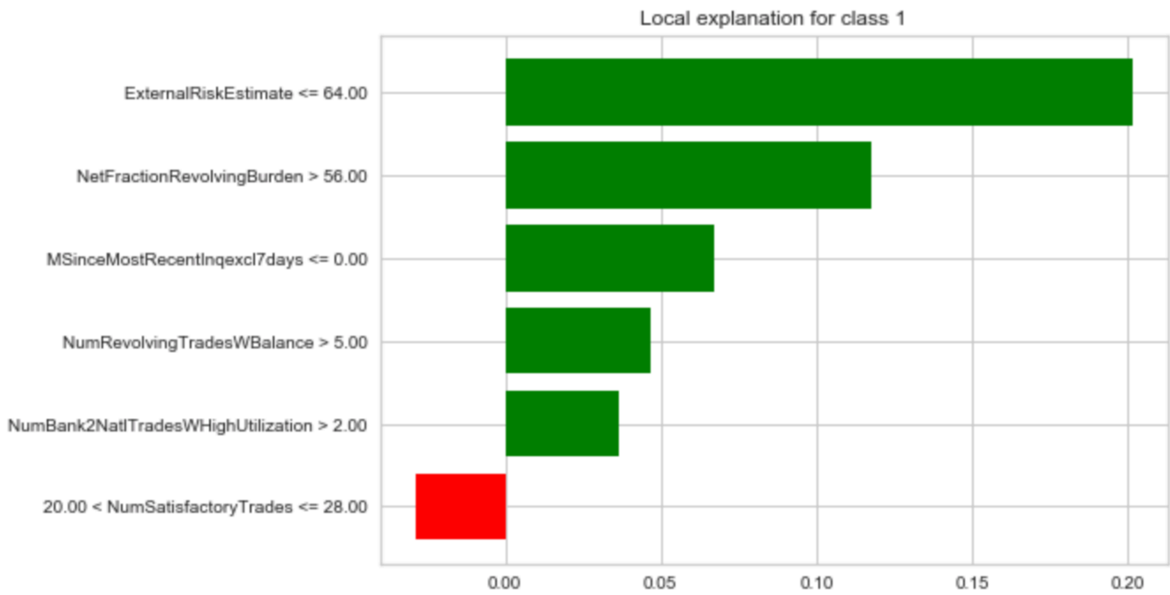


Cluster 9:

cluster 9 - index: 284

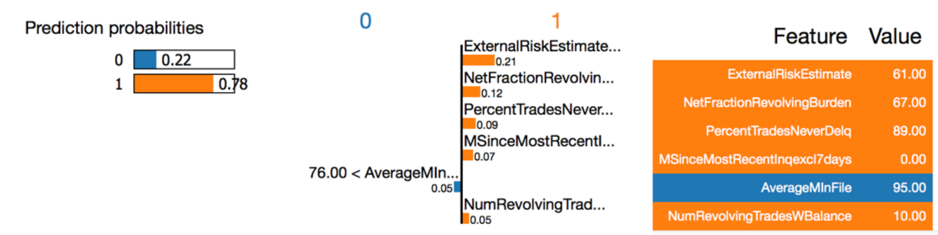


cluster 9 - index: 284

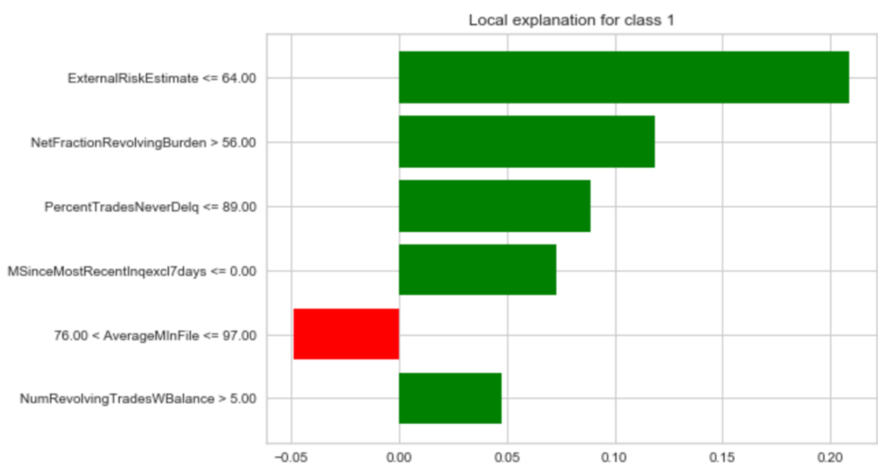


Cluster 10:

cluster 10 - index: 224

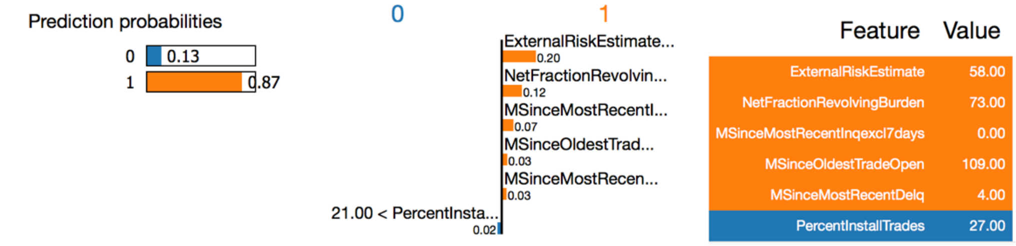


cluster 10 - index: 224

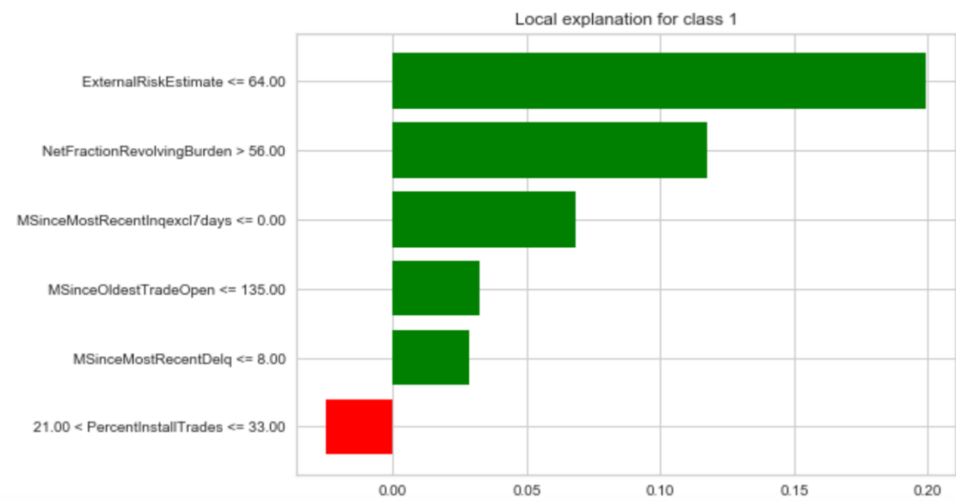


Cluster 13:

```
cluster 13 - index: 1828
```



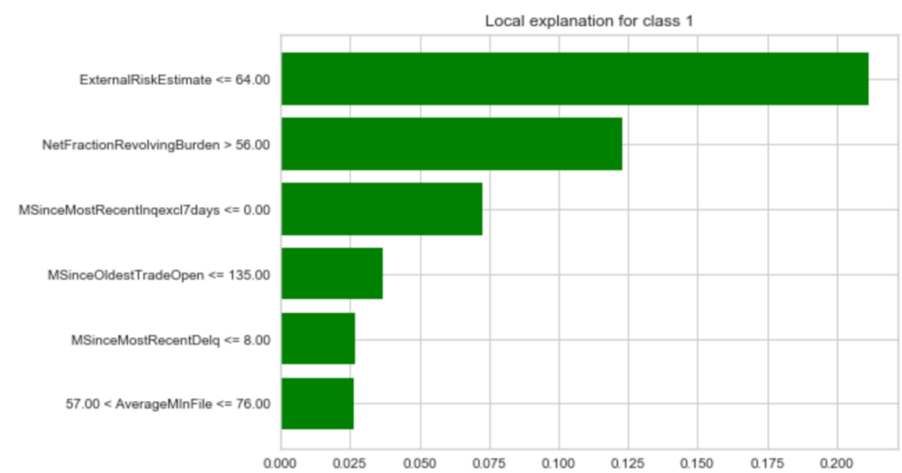

```
cluster 13 - index: 1828
```



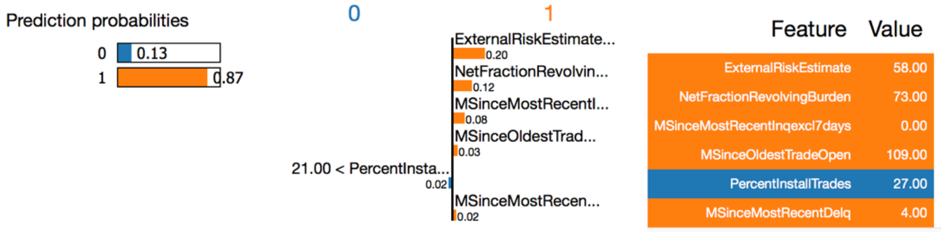
One problem about LIME is that explanation may change a little for each run:

```
cluster 13 - index: 1828
```


```
cluster 13 - index: 1828
```



cluster 13 – index: 1828



cluster 13 – index: 1828