Bach Quang Minh

Le Ngoc Thach

Nguyen Nhat Quang

Jon Von Neumann Institute – ICT2017

HACKATHON II - 2018

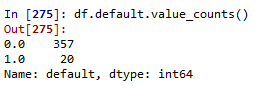
**Problem statement**

Measure the default level of a company based on its financial profile.

**Data exploration**

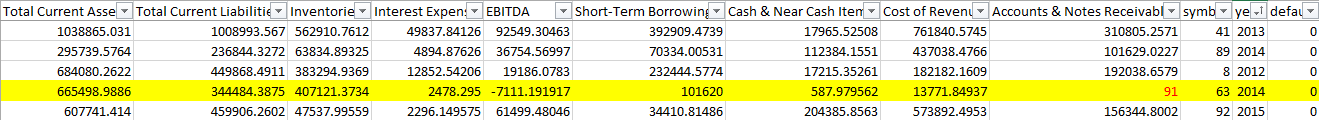
The whole data set contains 12 features and 2 classes: ‘default’-1 and ‘not-default’-0. We identify that this is a binary classification problem.

Here is the distribution of the data class:



As seen, the data suffered from imbalanced-class issue. The class percentage of 1 and 0 are 5.3% and 94.7% respectively.

We also notice that there is one company was having an unusual ‘Account & Notes receivables’ value:



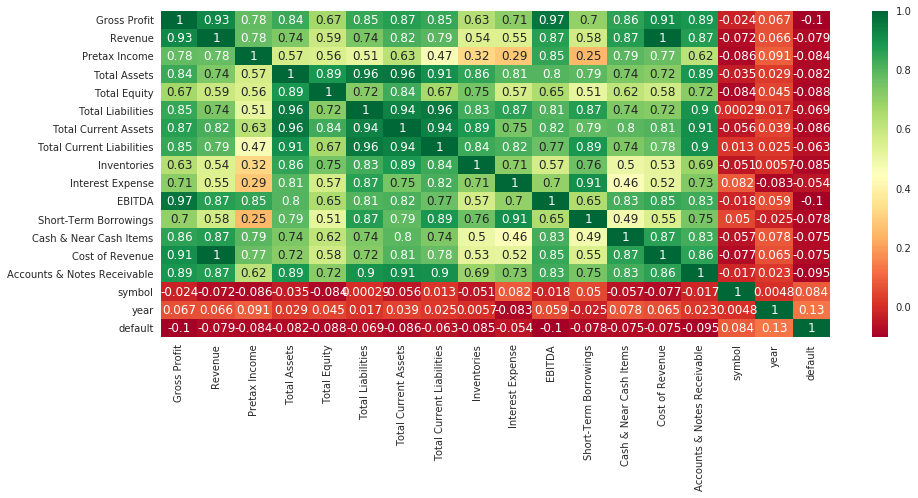
Since, this is the company data, we decided to drop this row before further analysis.

**Proposed method**

Since most columns from the data are real, continuous values. We decided to use Logistic Regression – a classification algorithm which could deal with continuous data.

The measure that we use for evaluation is the True Positive rate.

**Feature selection**



After studying carefully the correlation matrix and combining with our understanding about these finance indicators, we decided to omit the following features:

1. Total equity (because this is just an index calculated from two others indices)
2. EBITDA
3. Cost of Revenue (= revenue – gross profit)
4. Symbol (Just an encoded for each company)
5. Year (How well a company perform in a year is reflected by other index)

We will keep the following features for our first attempt:

'Gross Profit'

'Revenue'

'Pretax Income'

'Total Assets'

'Total Liabilities'

'Total Current Assets'

'Total Current Liabilities','Inventories'

'Interest Expense'

'Short-Term Borrowings'

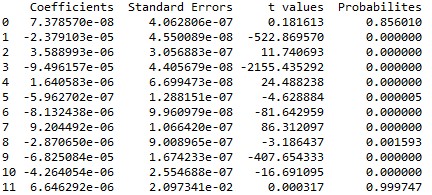
'Cash & Near Cash Items'

We use the train set and split it into train set and validation set with the ratio of 5:1

Here is the result:

|  |  |
| --- | --- |
| Confusion matrix on the train set | Confusion matrix on the test set |

To know if our features are statistically significant, we investigate these statistics:



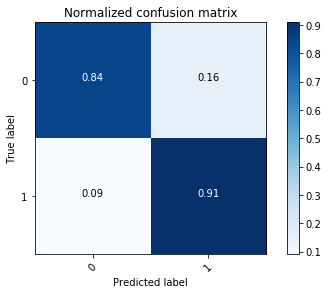
The result shows that some features does not have small p-values. We decided to seek for further feature selections.

First, we use the train-validation ratio as 0.3 and construct **a for loop** for **the number of features** to select the model with **the highest true positive values.**

The loop gives us these features:

['Gross Profit', 'Revenue', 'Pretax Income', 'Total Assets', 'Total Current Assets']

This is the confusion matrix for the validation set



The result obtained is higher and the features are suitable to our understanding.

We decided to use these features to train the model and produce the result for the test set.