Assessment: Credit Card Default Classification

Iris Manola
manola.irida@gmail.com
25 January 2019

CONTENTS

This presentation includes:

1. An introduction to the subject and the tools used.
2. Having a look at the data
3. Find the most influential features
4. Select the set of Machine Learning (ML) models
5. Apply and test the accuracy of those models
6. Decide on the most appropriate one

Detailed line-by-line commenting on the reasoning of each step of this presentation can be found in the Jupyter notebook file.

1. Subject and tools

- Various data from 30.000 bank holders are used in order to create a machine learning algorithm that can predict with the best possible precision if a new customer is more likely to default on his/her payments or not.

- The language used in this assessment is Python 3.5.2 on Jupyter notebook.

2. THE DATASET

- Loading the "AdviceRobo_Test_data.svc" data in a pandas dataframe for a quick look of the first 5 rows.

	Unnamed: 0	X1	X2	X3	X4	X5	X6	X7	X8	X9	...	X15	X16	X17	X18	X19	X20	X21	X22	X23	Y
0	1	20000	2	2	1	24	2	2	-1	-1	...	0	0	0	0	689	0	0	0	0	1
1	2	120000	2	2	2	26	-1	2	0	0	...	3272	3455	3261	0	1000	1000	1000	0	2000	1
2	3	90000	2	2	2	34	0	0	0	0	...	14331	14948	15549	1518	1500	1000	1000	1000	5000	0
3	4	50000	2	2	1	37	0	0	0	0	...	28314	28959	29547	2000	2019	1200	1100	1069	1000	0
4	5	50000	1	2	1	57	-1	0	-1	0	...	20940	19146	19131	2000	36681	10000	9000	689	679	0

5 rows × 25 columns

- There are 23 independent features and 1 default/not default feature in 30.000 samples. This is in total 750.000 features in the dataset.
- The default feature is found under the variable name 'Y' and includes 2 different types:

  0 : Not Default

  1 : Default

2. The dataset

- X1: Amount of the given credit (NT dollar): it includes both the individual consumer credit and his/her family (supplementary) credit.
- X2: Gender (1 = male; 2 = female).
- X3: Education (1 = graduate school; 2 = university; 3 = high school; 0, 4, 5, 6 = others).
- X4: Marital status (1 = married; 2 = single; 3 = divorce; 0=others).
- X5: Age (year).
- X6 - X11: History of past payment. We tracked the past monthly payment records (from April to September, 2005) as follows: X6 = the repayment status in September, 2005; X7 = the repayment status in August, 2005; . . .;X11 = the repayment status in April, 2005. The measurement scale for the repayment status is:

    -2: No consumption;

    -1: Paid in full;

    0: The use of revolving credit;

    1 = payment delay for one month;

    2 = payment delay for two months;

    . . .;

    8 = payment delay for eight months;

    9 = payment delay for nine months and above.

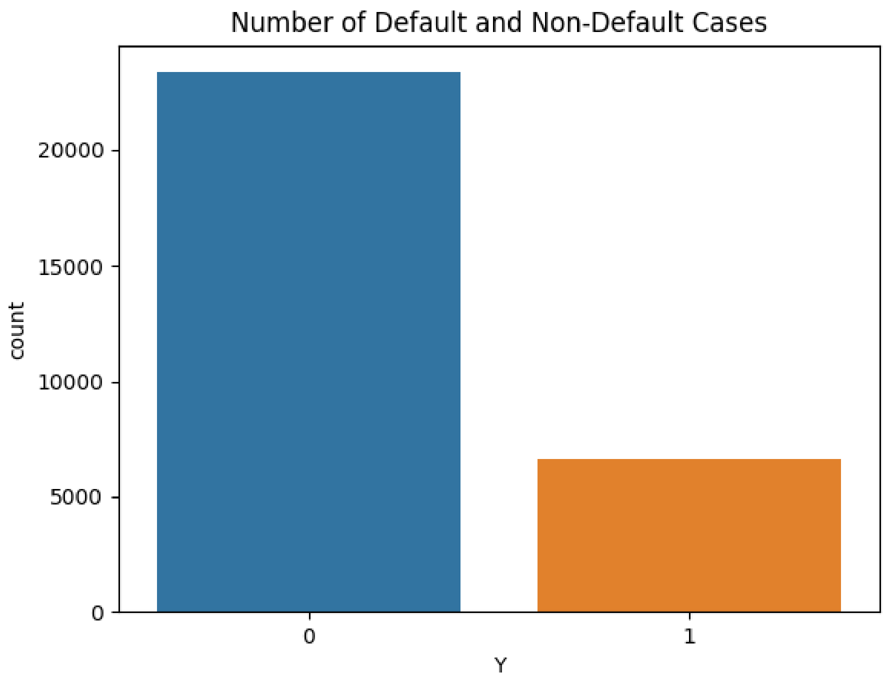
- X12-X17: Amount of bill statement (NT dollar). X12 = amount of bill statement in September, 2005; X13 = amount of bill statement in August, 2005; . . .; X17 = amount of bill statement in April, 2005.
- X18-X23: Amount of previous payment (NT dollar). X18 = amount paid in September, 2005; X19 = amount paid in August, 2005; . . .;X23 = amount paid in April, 2005.
- Y: client's behavior; Y=0 then not default, Y=1 then default

2. The dataset

Are the data clean?

- A df.isnull().sum() counts that there are no features with missing values. So yes, the data are clean.

What is the ration between Default and Non-Default?



3. The most influential features

- The most influential/informative features among the 23 features are those that can explain most of the variance in the data.
- The SelectKBest command of scikit-learn library can easily give the list of the top 8 most informative features and their F-scores (the ration between the explained and unexplained variance):

Most influential features	
Feat_names	F_Scores
X6	3537.7
X7	2239.2
X8	1757.5
X9	1476.8
X10	1304.6
X11	1085.4
X1	724.07
X18	160.4

4. Select the set of models

- We should now select binary supervised classification ML models (algorithms) that would allow us to make trustworthy predictions on the default class if the details of another customer are given.

- The multiclass supervised classification ML techniques that fit in our case and will be tested here are the following:
  1. Logistic Regression
  2. k Nearest Neighbour
  3. Random Forest
  4. Support Vector Machine

5. Apply and test the models

- The data will be divided in the X independent variables (the 23 features) and the Y dependent variable (the 2 default classes), which is the one that we would like to be able to predict given the set of X variables.

- The data will be separated in two, random, non-overlapping sets, which are the training set and the test set.

- The training set will define with the help of the ML algorithms the relations between the data and learn from them. The test set will perform an unbiased evaluation of the fit of each model on the training data. In this case we divide the data in a training set that contains the 80% of the data and the test set that contains the rest of 20%. A sensitivity test (repeating the process for slightly different test sets of 20, 22 and 25% showed that the outcome does not change considerably in the current analysis).

- Because the 23 features are of different magnitudes, in order to be compared they should be transformed to the same level of magnitude. Here we used the StandardScaler module for this.

5. Apply and test the models

There are many tests to assess the accuracy of a ML model. A highly explanatory one is the **Confusion Matrix** algorithm that predicts how many times each model makes a right or a wrong prediction by giving details for the precision and the recall of the model as:

Precision = True Positive / (True Positive + False Positive)

Recall = True Positive / (True Positive + False Negative)

The F1-Score a is the harmonic average of precision and recall, so it also takes into account the false positives and false negatives but can give the overall estimation of the precision of each tested model.

Classification Report – Logistic Regression			
	Precision	Recall	f1- score
Non-Default	0.82	0.98	0.89
Default	0.73	0.23	0.35
weighted average	0.87	0.82	0.78

Classification Report – Random Forest			
	Precision	Recall	f1- score
Non-Default	0.84	0.95	0.89
Default	0.63	0.33	0.44
weighted average	0.79	0.81	0.79

Classification Report – kNN			
	Precision	Recall	f1- score
Non-Default	0.83	0.95	0.89
Default	0.62	0.29	0.38
weighted average	0.78	0.81	0.78

Classification Report – SVC			
	Precision	Recall	f1- score
Non-Default	0.84	0.96	0.9
Default	0.7	0.34	0.46
weighted average	0.81	0.83	0.8

5. Apply and test the models

From the information below we can conclude that, although the differences are small, the overall best performing model is the **SVC** model!

The general accuracy of the model is 80%, that can be assumed high enough for a trustworthy model to predict the type of default if the relevant features are given.

Classification Report – Logistic Regression			
	Precision	Recall	f1- score
Non-Default	0.82	0.98	0.89
Default	0.73	0.23	0.35
weighted average	0.87	0.82	0.78

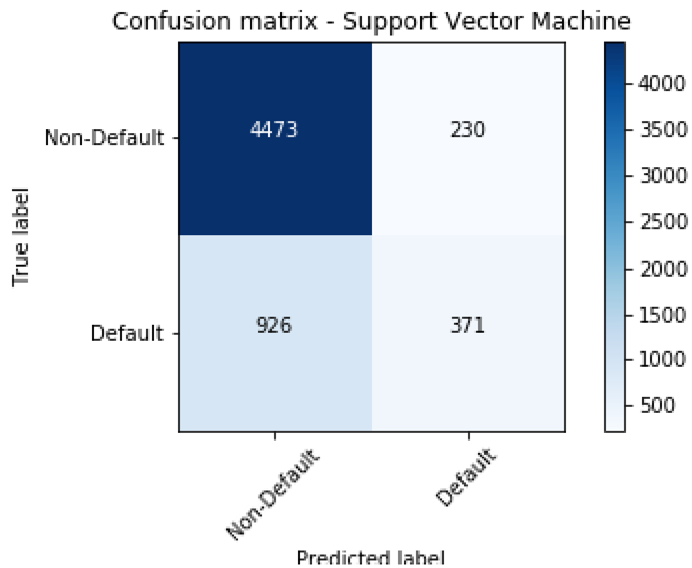
Classification Report – Random Forest			
	Precision	Recall	f1- score
Non-Default	0.84	0.95	0.89
Default	0.63	0.33	0.44
weighted average	0.79	0.81	0.79

Classification Report – kNN			
	Precision	Recall	f1- score
Non-Default	0.83	0.95	0.89
Default	0.62	0.29	0.38
weighted average	0.78	0.81	0.78

Classification Report – SVC			
	Precision	Recall	f1- score
Non-Default	0.84	0.96	0.9
Default	0.7	0.34	0.46
weighted average	0.81	0.83	0.8

6. THE MOST PRECISE MODEL

- Let's plot and example of the Confusion Matrix of the SVC



So, for example, if we make 6000 predictions (the test set) using the SVC model we will predict correctly 4473 times that the customer will not default a payment and 371 times correctly that the customer will default.

But we will also assume 926 times that the customer will default, while he/she won't, and 230 times the customer will be assumed trustworthy while he/she will default.

(keep in mind that with a different random selection of the train and test data these numbers will change slightly, but they give a good estimation)

Thank you for your time!

I look forward to your evaluation.