Recipe To Classify

Case Study: DA 2016 Challenge

Supervisor: Dr. Toktam Khatibi

Student: Rasoul Norouzi

Data Mining Course Project

April, 2018

Problem

IDA 2016 Challenge

Challenge Description Available HERE:
https://people.dsv.su.se/~isak-kar/IDA2016Challenge.txt

Data Insight

Never have complete confidence in metadata

- Determine trainin and test data shape

  By doing this, we get the number of records and the Matrix dimensions

- Determine variables data type

  Some times columns in the dataset do not have the correct type as a result of missing values

Data Insight

Never have complete confidence in metadata

- Examine classes and class imbalance for both training set and test set

  Class imbalance means that there are unequal numbers of cases for the categories of the label. Class imbalance can seriously bias the training of classifier algorithms. In many cases, the imbalance leads to a higher error rate for the minority class

Data Insight

- Visualization
  - For numerical Attributes: Box-Plot

    Sufficient differences in the quartiles for the feature to be useful in separation the label classes

  - For ordinal attributes: Bar-Plot

    The key to interpreting these plots is comparing the proportion of the categories for each of the label values. If these proportions are distinctly different for each label category, the feature is likely to be useful in separating the label.

Data Pre-Processing

- Code label as a binary variable

  Positive class = 1

  Negative class = 0

Data Pre-Processing

- Treat missing values

  understand how missing values are represented (in our case “na”)

  As a rule of thumb, if a variable has more than 60% missing values feel free to remove the variable

  If the number of missing values are low (less than 10%), feel free to remove the missing entries

  If the number of missing values are high, use expectation maximization (EM), SMOTE or nearest neighbors algorithms

Data Pre-Processing

Removing Outliers

Box-Plot and whiskers is a good tool for detecting and removing outliers

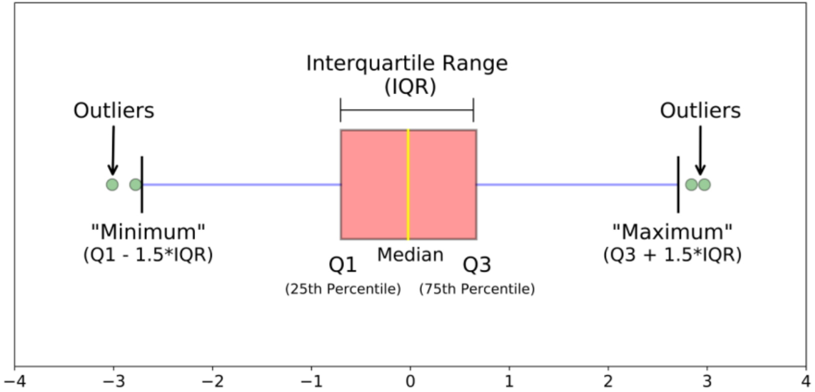


Image from Vishal Agarval

Data Pre-Processing

- Remove Duplicate Rows

  Number of duplicate rows= total number of rows - number of unique rows

Data Pre-Processing

- Feature engineering and transforming variables (ordinal attributes)

  Histograms attributes are a kind of ordinal attributes

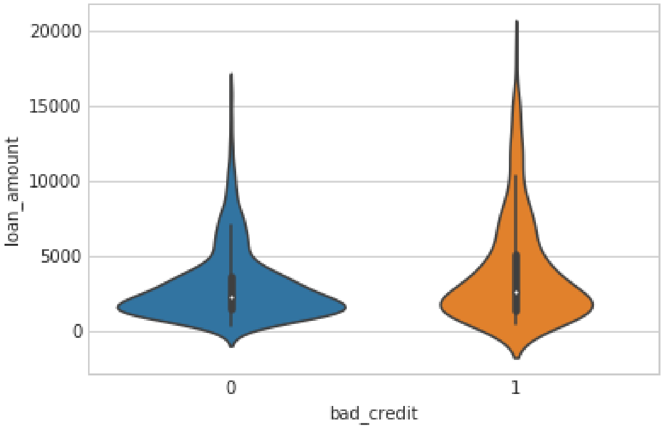
  Arrange the categories in ascending order and assign a number to each category based on its index number (from one)

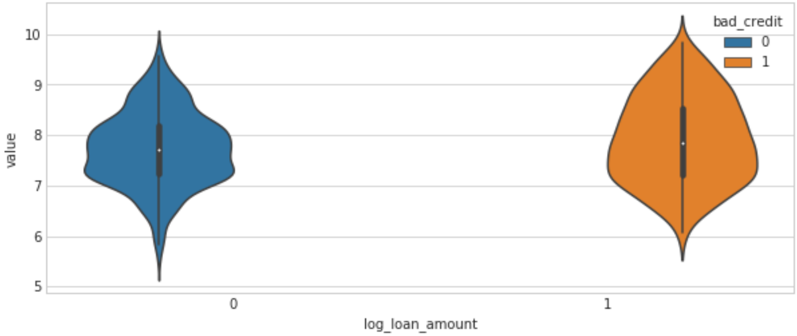
  For example, bin_1 = 1, bin_2 =2 , ...

Data Pre-Processing

- Feature engineering and transforming variables (Numerical attributes)

The distribution of some numerical attributes may be skewed; to mitigate this, use log to transform values.





High Dimensionality

The number of attributes is very high (167)

There are three scenarios:

- Dimensionality Reduction: Personally prefer SVD over the PCA, by dropping the factors with eigenvector lower than one
- Eliminating features with low variance and zero variance (p=80% as threshold)

$$Var(x) = p(1 - p)$$

- Select k best features: RFECV performs pretty well; Since data is highly imbalanced, the AUC metric in joint cross-validation could be the best choice

Class Imbalance

To treat class imbalance there are Four scenario:

- ✔ class_weight : Personally I prefer this one, however is not compatible with some classifiers like Naive Bayes and AdaBoost
- Using SMOTE: The difference in the number of class labels is huge, is not practical
- Using over sampling: The difference in the number of class labels is huge, is not practical
- Using under sampling: for this problem it could be a crazy decision, by doing this at the final stage we will have only 2000 training data

Normalization

Normalize data by using Z-Score

$$Z = \frac{x - \mu}{\sigma}$$

Notice that the scaler is fit only on the training data. Then trained scaler is applied to the test data

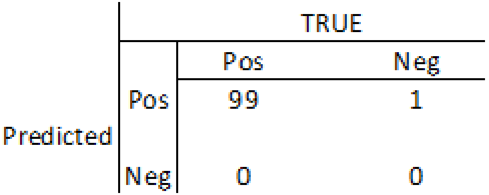
Metric!!!

If the test set is imbalance like the training set, therefore the proposed metric would be extremely misleading.

How?

Suppose we have a test set with 100 examples, 99 of which are labeled "one", and only one of them is labeled "zero".If without any classification algorithm, we label all test data with "one", our algorithm accuracy will be 99% and according to the proposed metric, the cost will be 10 (Cost_1=10)

Cost= Cost_1*No_Instances



		TRUE	
		Pos	Neg
Predicted	Pos	99	1
	Neg	0	0

Implement Classifier

By Nested Cross-validation (grid search) tune the classifier’s hyper-parameters (As data is Imbalance I use AUC metric)

After finding the best Hyper-parameters Values,

go through the training, testing, and comparing various classifiers and be careful about overfitting (using regularization in such situation)

Classifiers

My choices for classification:

- Random-Forest algorithms.
- Logistic Regression
- SVC (a non-linear variation of SVM)
- The number of training data is not bad, so MPL (neural network) could be a good choice
- CatBoost is a powerful algorithm against overfitting

Keep In Mind

Repeat all steps from data preprocessing to the end with different scenarios of attributes reduction, class balancing, and so, then select the best scenario with the minimum cost!

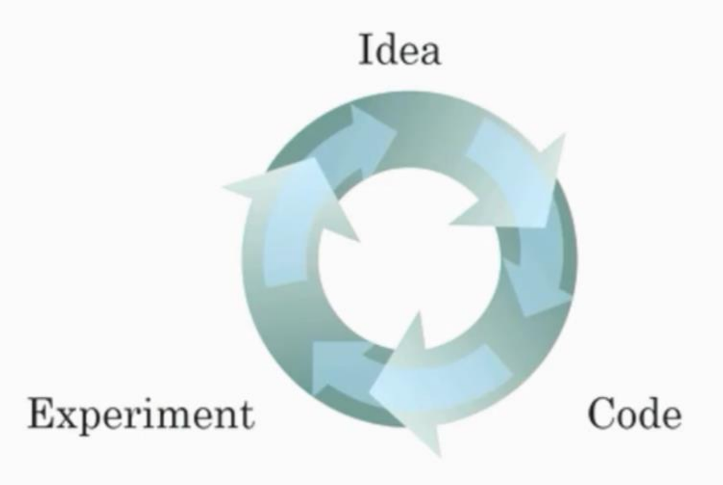


Image by Professor Andre Ng, Deep Learning course, Coursera

Contact

Rasoul Norouzi
rslnorouzi@gmail.com

Website:
https://rasoulnorouzi.github.io/