Introduction to Machine Learning

Rodrigo Petricioli

November 10, 2021

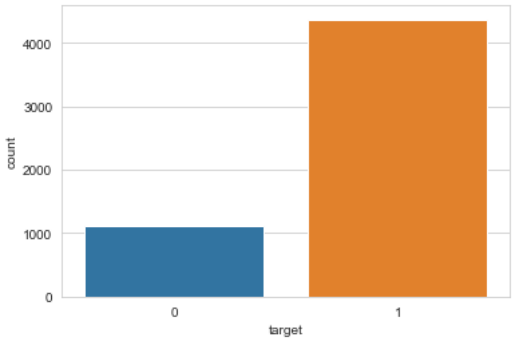
Classification Model Report

ISE 364

Contents

1 Introduction

The objective for this project is to train the most efficient Classification model possible to predict the target value of the given data. The original data contains 5,497 observations of 12 features and a traget value. Feature 0 is a binary feature containing A or B. Features 1 through 11 are numeric variables. Finally the target value is either 1 or 0.
For this specific experiment the actual Features are unkown, and only the Feature values are known.
Classification reports are models that require the use of machine learning algorithms. They use a fraction of the database to "train" the model to later "test" it on the remaining data to predict or classify a selected feature. Use email data to predict wether or not a new email is spam or not.
From the 5,497 observations, 4,374 have a target value of 1 (79.5%), and 1,123 have a target value of 0 (20.5%).



2 Model Selection

Using the original database I experimented with four different classification models to select the best performing one. For this specific experiment I used Logistic Regression, K-Nearest Neighbors, Decision Trees, and Suport Vector Machines.
For each of the models I divided the data randomly into a train set and a test set. The train set is composed of 70% of the observations (3,847 observations) and the test set is composed of the remainder 30% (1,650 observations).
Since the actual features are not known, I ran each model three times, with three different datasets. The first is the original data with the *Feature 0* values changed from A and B, to 1 and 0, respectively. I will refer to this data set as *Raw Data*. Second is the same data but using the *StandarScaler* function in Python to scale the data to be distributed as a Standard Normal

distribution with mean 0 and standard deviation 1. This will be referred to as Scaled Data.
The third data set uses the scaled data for *Features 1-11* and maintaining the *Feature 0* with its values of 0 and 1. I will refer to this data set as *Scaled Adjusted Data*.
These four models are a type of binomial or binary classification model. Binary or binomial classification is the task of classifying the elements of a given set into one of two groups. Typically the groups are encoded as y=0 or y=1. Since our target is binary, this is the appropriate model.
I fitted the models for the Raw Data, Scaled Data, and Scaled Adjusted Data. I used a classification matrix to determine the accuracy, precision, recall, and f1 score of the models.
Precision is defined as the ratio of true positives to the sum of true and false positives.
Recall is defined as the ratio of true positives to the sum of true positives and false negatives.
The F1 is the weighted harmonic mean of precision and recall. The closer the value of the F1 score is to 1.0, the better the expected performance of the model is.
This process is all coded in Python using Jupyter Notebook. The file *ProjectRP* contains the whole process for data analysis and model selection and comparison.

3 Results

3.1 Logistic Regression

The Logistic Regression Model Results are:

Score	RD	SD	SAD
Model Accuracy	80%	81%	81%
Precision (t=0)	63%	66%	64%
Precision (t=1)	81%	82%	82%
Recall (t=0)	13%	17%	17%
Recall (t=1)	98%	98%	97%
f1 Score (t=0)	21%	27%	27%
f1 Score (t=1)	89%	89%	89%

*Where RD=Raw Data, SD=Scaled Data, SAD=Scaled Adjusted Data.

This results show that for the Logistic Regression models, the data set that best fits is the Scaled Data. This makes sense since the features are unknown and the scaled data is more robust for the model given that we don't know how the specific features behave.

3.2 K-Nearest Neighbors

The K-Nearest Neighbors Model Results are:

Score	RD	SD	SAD
K Neighbors	23	17	17
Model Accuracy	80%	79%	79%
Precision (t=0)	62%	0%	0%
Precision (t=1)	80%	79%	79%
Recall (t=0)	4%	0%	0%
Recall (t=1)	99%	100%	100%
f1 Score (t=0)	8%	0%	0%
f1 Score (t=1)	89%	88%	88%

*Where RD=Raw Data, SD=Scaled Data, SAD=Scaled Adjusted Data.

This results show that for the K-Nearest Neighbor models, the data set that best fits is the Raw Data. We can clearly see that the Recall and f1 score for the target = 0 is very low. This could be explained by the difference of target = 1 observations vs target = 0. Since there are significantly more observations that result in target = 1, the model tends to fit new observations to target = 1 better than those for target = 0.

3.3 Support Vector Machines

The Support Vector Machines Model Results are:

Score	RD	SD	SAD
Model Accuracy	79%	81%	80%
Precision (t=0)	0%	62%	61%
Precision (t=1)	79%	82%	81%
Recall (t=0)	0%	16%	15%
Recall (t=1)	100%	97%	97%
f1 Score (t=0)	0%	26%	25%
f1 Score (t=1)	88%	89%	89%

*Where RD=Raw Data, SD=Scaled Data, SAD=Scaled Adjusted Data.

This results show that for the Support Vector Machines models, the data set that best fits is the Scaled Data. Since this model uses distance, the data needs to be scaled so the model can perform correctly. In this case there is no significant difference between the Scaled and the Scaled Adjusted data sets, but the Scaled Data set performs best.

3.4 Decision Tree

The Decision Tree Model Results are:

Score	RD	SD	SAD
Model Accuracy	79%	67%	67%
Precision (t=0)	49%	12%	12%
Precision (t=1)	86%	78%	78%
Recall (t=0)	48%	9%	9%
Recall (t=1)	87%	82%	82%
f1 Score (t=0)	48%	11%	11%
f1 Score (t=1)	87%	80%	80%

*Where RD=Raw Data, SD=Scaled Data, SAD=Scaled Adjusted Data.

This results show that for the Decision Tree models, the data set that best fits is the Raw Data. This makes sense given that no scaling is needed for this model. The original data trains the model better unscaled.

4 Conclusion

In conclusion, after testing several models with raw, scaled and ajdusted data sets, I found that the one that predicts the data best is the Logistic Regression with the Scaled Data set. For the new observations this is the model that I will use to try and best predit those observations.

5 New Observations

This exercise serves the purpose of selecting a model that can best predict any given set of observations with the same features. The final model will be trained using this entire data set to then use the new observations for testing how well it predicts the target outcome. In the next 24 hours I will receive the new observations data set which I will use as the "test" observations and observe and record how the model performs using the same metrics. The "train" data will be the entire current data set.