Classifiers evaluation: SVM & Logsitic Regression & Decision Tree

Lingcong Shen

redletee@outlook.com

PID: u08297242

Abstract

Model selection is of great importance in classification problems. In this pro-ject, the author evaluated the performance of SVM, Decision Tree, Logistic Regression classifying four datasets with error rate. To ensure the perform-ance is well represented, the datasets are spilled into test data and train data by three test data proportion, and each training process and predicting pro-cess is executed three times.

1. Introduction

Supervised machine learning digs into patterns hidden under the complicated, huge and messy real-world datasets, and provides individuals and companies with a deeper understanding of the real word problem. Mostly, supervised machine learning is applied for the classification problems and regression problems. Here in this essay, the classification will be the only problem to be discussed. Based on supervised machine learning, it is feasible to generate various kinds of classifiers that distinguish data relatively more precisely based on numerous features of observations. And the model selection is an essential part of machine learning because there won't always be sufficient time and calculation capacity when dealing with a real-world problem.

As classifiers are generated by different algorithms, they show different preferences of prediction and perform differently toward the same dataset. Therefore, it's essential to evaluate how classifiers perform while dealing with different kinds of datasets and how classifiers differ from each other while applying for the same dataset.

In this article, I compared the performance of classification of three learning methods (Support Vector Machine (SVM), Logistic Regression, Decision Tree) on four datasets and compared the conclusion with the conclusion of (Caruna and Niculescu-Mizil, 2008). And ACC metric is used for performance evaluation.

2. Method Description:

In this article, I trained these three classifiers on four data sets with three different test data partitions. And for each training process, I used cross-validation to tune the hyper-parameter. To

make sure the error rate correctly reflects the classifiers' performance, I repeated training processes three times for each classifier for each dataset.

2.1 Parameters

2.1.1 SVM (Support Vector Machine)

In this part, SVM classifier is using rbf kernel. I vary the parameters of C and gamma to find the best parameter. I used {0.1,1,10,100} for C and {1e-7,1e-6,1e-5,1e-4} for gamma. I use GridSearchCV() from sklearn to find the hyper-parameter through cross-validation.

2.1.2 Logistic Regression

Concerning Logistic Regression, I vary two kinds of parameters. One is C of logistic regression, the other one is thresholds of prediction. For C, {0.1,1,10,100} is used to tuning hyper-parameter. Threshold parameter is varied by factors of 0.05 from 0 to one.

2.1.3 Decision Tree

In the training process of decision tree, best max depth of the decision tree is achieved through grid search. The parameters of max depth list is {1,2,3,4,5,6,7,8,9,10,11,12,13}.

2.2 Evaluate Performance Metrix

The performance of the classifier is evaluated through its training error rate and its testing error rate. In order to obtain an error rate without accidental results, every training process is repeated three times.

2.3 Cross-Validation

Cross-Validation is of two usages, one is for obtaining hyper-parameter, the other one is for calculation error rate.

Training data, after splitting the original dataset into training one and testing one, are split into training and validation data. Then, the author applied cross-validation to calculate the error rate. The parameter that helps classifiers obtaining the lowest error rate would be hyper-parameter. And error rates of these classifiers' prediction would be the returned error rate.

3. Experiment Datasets

Four datasets were picked and cleaned to examine the performance of classifiers. In this part, a description of dataset, data wrangling process will be delivered. Some of the datasets are clean enough to be used directly, while some datasets contain repeated data, null data as well as information-less features. Besides, a assumed application of classification will be delivered accordingly.

To be noticed, due to the time limit, I took 7000 observations from each dataset for classification instead of using all the data in the datasets.

3.1 Crime in Chicago from 2001 to 2003

3.1.1 Data summary:

The dataset is provided by the Chicago Police Department, containing 6,890,000 observations with 22 features. This dataset reflects reported incidents of crime (with the exception of murders where data exists for each victim) that occurred in the City of Chicago from 2001 to 2003 (Chicago Police Department,2019)

Labels of observations are assigned according to whether the criminals are arrested. Features of the dataset are listed below: ID, Case Number, Date, Block, IUCR, Primary Type, Description, Location Description, Arrest, Domestic, Beat, District, Ward, Community Area, FBI Code, X Coordinate, Y Coordinate, Year, Updated On, Latitude, Longitude, Location.

3.1.2 Data wrangling:

All the features that contain little useful information on the crime are removed. Features removed are listed: ID, Case Number, IUCR, Updated On, Description, FBI Code; Also, Features that store the same information in a different format are removed. Features removed are listed: Year, Updated on, Longitude, Location.

As for some categorical features like Primary Type, I applied One hot encoding to them to change them into numerical features. After the previous processes were done and the observations with null values are dropped, 3000 observations with 174 features are selected for classification.

3.2 Heart Attack

3.2.1 Data summary:

The dataset is provided by Medicare, containing 12145 rows and 5 columns. (Notice: data is originally from Medicare and it is not citable.) In this dataset, 0 is labeled if the patient has died and 1 is labeled if the patient survives. It has a record of the number of days spent in hospital by patients admitted to hospitals in New York during one year with a primary diagnosis of acute myocardial infarction (heart attack

For this problem, we would like to analyze whether a patient survives. Features are listed below: Los (days spent in hospitals), Age, Sex, DRG, DIED.

3.2.2 Data wrangling:

DRG, representing the identity of the patient, is removed for convenience. And 12000 rows with 4 features are selected after shuffling the dataset and removing all the rows with a null value.

3.3 Adult

3.3.1 Data summary:

The last dataset is Adult dataset from UCI Machine learning repository (Dua & Graff, 2019), containing 4,8842 observations and each observation contains 14 features. Here, I would like to analyze whether this man earns more than 50,000 in a year.

Since categorical features appear frequently in this dataset, one hot encoding is applied to change them into categorical data.

After shuffling the dataset and dropping unnecessary columns and null values, author picked up 7000 observations with 107 features.

3.4 DOTA2 Game result

3.4.1 Data summary:

The dataset is from UCI Machine learning repository (Dua & Graff, 2019), containing information of 102,944 games and 116 features for each game. Labels are assigned to each datapoint based on win-or-lose status of it.

Below is the description of the problem when setting test size at 0.8.

	Features	Train	Valid	Test	%Pos
Dataset1	174	400	200	2400	27.00%
Dataset2	4	1619	810	9716	11.00%
Dataset3	107	933	467	5600	24.39%
Dataset4	115	933	467	5600	53.10%

Figure 1. Description of datasets

4. Evaluation

Concerning the prediction on test dataset, Decision tree outperforms logistic regression and SVM a lot. And SVM is the one that follows up. Logistic regression performs worst among these three classifiers. Decision tree showed dominance in prediction in all datasets except the fourth one. SVM performs better than logistic regression since it has the best prediction for three times. And it often gets the second lowest error rate, greatly outperforming logistic regression.

Classifiers that generate the lowest error rate of a certain dataset with certain test partition are bolded. Classifiers that generate the lowest error of a certain dataset are underlined.

test_rate=0.8	SVM	LOG	DT
D 1	0.2696	0.2696	**0.1494**
D 2	0.1026	0.1089	**0.0893**
D 3	0.1634	0.166	**0.1582**
D 4	0.4314	**0.4225**	0.4725

Figure_5 Testing error with 0.8 test proportion

test_rate=0.5	SVM	LOG	DT
D 1	0.262	0.258	**0.1507**
D 2	0.1006	0.0963	**0.0912**
D 3	0.1446	0.1548	**0.1604**
D 4	0.4197	**0.4142**	0.4733

Figure_5 Testing error with 0.5 test proportion

test_rate=0.2	SVM	LOG	DT
D 1	0.2467	0.26	**0.1422**
D 2	0.1006	0.098	**0.0939**
D 3	0.1393	0.1507	**0.1464**
D 4	**0.4107**	0.4179	0.4624

Figure_5 Testing error with 0.2 test proportion

4.1 Performance of different algorithms.

Considering performance of different classifiers on datasets with all test partitions being considered, Decision tree performs best on classification task of dataset 1, dataset 2 and dataset 3. SVM follows up, having the best prediction on dataset 4. And Logistic regression generates the least satisfying result.

4.1.1 Partition one (test rate = 0.8)

Dataset 1: DT has the best performance, while LOG and SVM generate same error.

Dataset 2: DT performs best, while SVM ranks second.

Dataset 3: DT ranks first. SVM ranks second.

Dataset 4: LOG ranks first, SVM ranks second.

4.1.2 Partition two (test rate = 0.5)

Dataset 1: DT generates the lowest error, while LOG follows up.

Dataset 2: DT performs best, while LOG ranks second.

Dataset 3: DT has the best performance, while SVM generates the second lowest error.

Dataset 4: LOG ranks first, SVM ranks second.

4.1.3 Partition three (test rate = 0.2)

Dataset 1: DT outperforms LOG and SVM, and SVM outperforms second.

Dataset 2: DT ranks first, and LOG ranks second.

Dataset 3: DT has the best performance, while SVM generates the second lowest error.

Dataset 4: SVM ranks first, while DT has the highest error.

4.2 Same classifiers across different datasets and partitions

The decreasing of error rate appears most frequently, especially for SVM and DT. However, error rates generated by LOG vary randomly across different partitions.

4.2.1 SVM

Dataset 1: Error rate's decreasing is seen as test data proportion goes down. With test data proportion of 20%, SVM classification error rate is 0.2467.

Dataset 2: Error rate's decreasing is seen as test data proportion goes down. However, it is not quite obvious since error rates of classification with 0.2 and 0.5 test data proportion are same. With test data proportion of 20%, SVM classification error rate is 0.1006

Dataset 3: Error rate's decreasing is seen as test data proportion goes down. With test data proportion of 20%, SVM classification error rate is 0.1393.

Dataset 4: Error rate's decreasing is seen as test data proportion goes down. With test data proportion of 20%, SVM classification error rate is 0.4107

4.2.2 Logistic regression

Dataset 1: Error rates randomly vary with different test data proportion. The error rate on dataset with 0.5 test data proportion is the lowest, 0.258.

Dataset 2: Error rates randomly vary with different test data proportion. The error rate on dataset with 0.5 test data proportion is the lowest, 0.0963.

Dataset 3: Error rate's decreasing is seen as test data proportion goes down. With test data proportion of 20%, error rate is 0.1507.

Dataset 4: Error rates vary randomly with different test data proportion. The error rate on dataset with 0.5 test data proportion is the lowest, 0.4142.

4.2.3 Decision Tree

Dataset 1: Error rates vary with different test data proportion. However, some kind of decreasing appears as the error rate on dataset with 0.2 test data proportion is still the lowest, 0.1422.

Dataset 2: The error rate increases as test proportion goes down. The error rate of 0.8 test data proportion is the lowest, 0.0893.

Dataset 3: Error rates vary with different test data proportion. The error rate on dataset with 0.2 test data proportion is still the lowest, 0.1464. Therefore, we could conclude that the decreasing of error rates happens.

Dataset 4: Error rate's decreasing is seen as test data proportion goes down. With test data proportion of 20%, error rate is 0.4624.

5. Conclusion

Overall, Decision Tree performs the best with the lowest error rate. And SVM follows up with the second lowest error rate. Logistic regression, except being used towards the fourth dataset, generates the highest error rate. The decreasing of error rate appears most frequently, especially for SVM and DT. However, error rates generated by LOG vary randomly across different partitions.

The evaluation results differ from the conclusion of Caruna and Niculescu-Mizil. (Caruna and Niculescu-Mizil, 2008) Despite, which shows that SVM > DT > LR. Here in this experiment, Decision Tree outperforms SVM. Meanwhile, SVM still performs better than logistic regression. The cause of SVM's bad performance may be that the datasets been tested in the experiment have relatively smaller sizes, which are around 5 times smaller than those used in the Caruna and Niculescu-Mizil 's research. (Rich, Nikos, & Ainur, 2008)

6. Bouns Point

Four Datasets are chosen from different fields, including public safety, health care, economics and video games. And they are comparably messy and contain lots of information-less attributes. Therefore, it took me much time to identify useful attributes and remove null rows which I think it deserves some extra points.

Works Cited

Chicago, P. (2019, June 14). *Crimes - 2001 to present.* Retrieved from CHICAGO DATA PORTAL: https://data.cityofchicago.org/Public-Safety/Crimes-2001-to-present/ijzp-q8t2

Dua, D., & Graff, C. (2019). Retrieved from UCI Machine Learning Repository: http://archive.ics.uci.edu/ml

Rich, C., Nikos, K., & Ainur, Y. (2008). An empirical evaluation of supervised learning in high dimensions. *ICML '08 Proceedings of the 25th international conference on Machine learning*, Pages 96-103.