University of
St Andrews

CS5014: Machine Learning

Practical 2: Classification of objects using a radar signal and machine learning

Student ID: 180007800

Table of Contents

Introduction

This report describes the derivation of a classification model to identify different objects using radar signals that are reflected on various attributes. This classification is based on a radar signals data set of various objects. There are totally 768 attributes and 80 samples (binary)/ 200 samples (multiclass) in the data set. The samples of radar signals are acquired by placing various objects on top of a radar chip (BGT60 - Infineon Technologies). Firstly, a model for the binary classification of two objects is described. Later on, the model is developed to distinguish between 5 different objects.

To allow the reader to understand all the steps of the model derivation, the report is structured as follows. In the first section, the report describes the measures to ensure high data quality and initial cleaning measurements for achieving a high data quality. In addition, it explains how to split the data sets into training and test set. Then the data is statistically analyzed and described. In the next sections, different classification models are fitted and the results are analyzed and compared with each other. Finally, a conclusion is given to summarize the fitted models and discuss the results.

Data Cleaning

In the Introduction section, it is mentioned that the dataset contains 80 samples(binary)/200 samples(multiclass) with 768 attributes. The attributes are components of the radar signal, so all the attributes are numerical. In addition, there is no deed to handle missing data values because the data set is complete. What's more, the values of all attributes have no too many differences, thus none needs to be removed and there is no necessity to do normalization.

Then both of the two data sets (binary and multiclass) were split up into a training and test sets. It is an important step because the evaluation of a fitted model using training set might be biased by overfitting and that will overestimate performance of the model. Train_test_split function of the scikit-learn package is used to split the data sets, and 20% of the data were selected to be the test sets randomly with 42 random state.

Descriptive statistics

To learn the data in more detail, an important step is to analyse the number of each class. The binary data set has 40 samples for one class and 40 samples for another. Similarly, in the multiclass set there are 50 samples for each class. The values of all attribute are in the same order of magnitude as mentioned in the Data Cleaning section. As for the binary data set, the global minimum is -0.49 with a mean minimum value of -0.099 and the global maximum 0.559 with the mean maximum value of 0.156, and the global mean value is 0.029. As for the multiclass data set, the global minimum is -0.942 with a mean minimum value of -0.362 and the global maximum 0.836 with the mean maximum value of 0.531, and the global mean value is 0.025. The global descriptive statistics for the two data sets can be found in Table 1. Furthermore, more detailed descriptive statistics (binary/multiclass data sets) of each attribute are stored in the CSV files that can be found in the folder Results/DescriptiveStatistics. Although it seems impossible to learn some useful information from the descriptive statistics of the data, it is helpful to understand the data in more detail.

	Binary	Multiclass
Global min	-0.49	-0.942
Mean min	-0.099	-0.362
Global max	0.559	0.836
Mean max	0.156	0.531
Mean	0.029	0.025

Table1: Global descriptive statistics for the binary and the multiclass data set

Logistic regression on the Binary Data Set

I used a Logistic Regression Model to fit both the binary and the multiclass data set. Firstly, different Logistic Regression models for binary data set are analysed.

Firstly, a full Logistic Regression model containing all attributes was fitted with the training data and then was evaluated with the test set. The full Logistic Regression model reached 100% classification rate. And the predictions for the data without the corresponding class ID are stored in Results/Binary/LogRegression.csv.

The 100% classification rate of the full model means the model could be fitted very well for the binary data set. Therefore, several feature selection methods were applied to simplify the model before fitting. The first feature selection method I used was SelectKBest method from the skikit-learn package. The method selects features according to the k highest scores and we can select the number of top features by setting the parameter “k”. In the method, the fault function “f_classif” which only works for classification and the number of top selected features “k = 1” were set[1]. When I applied the SelectKBest method to the binary data set, it returned attribute number 26 that measures the 26th attribute as the best attribute. I fitted a Logistic Regression Model again with the 26th attribute. Classification rate of the model reached 100% again, the high classification rate means the data from the two classes have obvious difference on the selected attribute so that ideally achieve the classification.

Additionally, Principal Component Analysis (PCA) with 10 Components and Recursive Feature Selection (RFE) with 10 Features are also used to select features. For PCA, I set “n_components” as 10 which means selecting 10 components from all the features[4]. For RFE, I defined “n_features_to_select” as 10 which means selecting 10 features from all the features and “step” as 1 that illustrates removes 10 features each iteration[5]. Compared with SelectKBest, the two methods achieve feature selection by reducing dimensionality of features[2][3]. The Logistic Regression Model using the two methods could classify 100% of the samples in the test set. The results of all fitted Logistic Regression Model are shown in the Table 2.

Support Vector Classification on the Binary Data Set

After the Logistic Regression model, several Support Vector Classifier (SVC) models were fitted for the binary dataset. All SVC models applied a linear kernel with the default value of the penalty parameter, C = 1. Similar to the Logistic Regression Model, a full SVC model including all 768 attributes was fitted and the classification rate reached 100%. Similarly, I used SelectKBest, PCA and Recursive Feature

Selection again to simplify the model. The classification rate of all the fitted SVC models reached 100%. The results of fitted SVC models for the binary classification are summarized in the Table 2.

Multi-Layer Perception on the Binary Data Set

Finally, multiple Multi-Layer Perception (MLP) model were fitted for the binary dataset. All MLP models applies 'lbfgs' solver which converges faster and perform better., default value of alpha (0.0001), random state of 1 and 5*5 size of layer[6]. Similar to the Logistic Regression Model and SVC model, a full MLP model

Model	Feature Selection Method	Selected Features (index)	Classification Rate	Specificity	Sensitivity	Precision	Recall
Logistic Regression	None	all	100%	100%	100%	100%	100%
Logistic Regression	SelectKBest	26	100%	100%	100%	100%	100%
Logistic Regression	PCA	10 Components	100%	100%	100%	100%	100%
Logistic Regression	Recursive Feature Elimination	28, 29, 30, 284, 285, 286, 540, 541, 656, 657	100%	100%	100%	100%	100%
SVC	None	all	100%	100%	100%	100%	100%
SVC	SelectKBest	26	100%	100%	100%	100%	100%
SVC	PCA	10 Components	100%	100%	100%	100%	100%
SVC	Recursive Feature Elimination	21, 22, 23, 278, 532, 533, 534, 535, 536, 537	100%	100%	100%	100%	100%
Multi-Layer Perception	None	all	100%	100%	100%	100%	100%
Multi-Layer Perception	SelectKBest	26	100%	100%	100%	100%	100%
Multi-Layer Perception	PCA	10 Components	100%	100%	100%	100%	100%

Table 2: Results of all fitted models for the binary data set

with all 768 attributes was fitted and the classification rate reached 100%. Similarly, to simplify the model, SelectKBest and PCA were applied to reduce the number of features. The classification rate of all the fitted MLP models reached 100%. The results of all fitted models including Logistic Regression, SVC and MLP for the binary classification are shown in the Table 2.

We know that classification rate of all the fitted models reach 100% from the Table 2, so I transferred observe how the models are fitted for the multiclass data sets.

Logistic Regression on the Multiclass Data Set

For the multiclass data set, a full Logistic Regression Model was firstly fitted on the multiclass dataset. According to the features of Logistic Regression, we know it is a linear regression model that only achieves a binary classification, so a one-versus-all scheme is used (the
model for all possible one-versus-all combinations is fitted) for the multiclass data set. When starting the prediction, a sample enters into each model in sequence and then calculates the possibility of each one-versus-all combination, finally the sample is assigned based on the highest possibility. Different from the binary data set, classification rate of the full Logistic
Regression model only reached 95%. It seems more difficult to achieve a perfect classification rate of Logistic Regression Model for the multiclass dataset compared to the binary dataset. Therefore, it is necessary to find models that can achieve a better classification rate.

Support Vector Classifier & Multi-Layer Perception on the Multiclass Data Set

Next to the full Logistic Regression Model, the full Support Vector Classifier was fitted for the multiclass dataset. Similar to the full Logistic Regression Model, it only reached a classification rate of 95% on the test data. Later a full Multi-Layer Perception model was fitted. With the 5*5 size of layer, the model also only reached classification rate of 95%. This is why for second Multi-Layer Perception model the size of layer was increased to 7*7. Not unexpected, the classification rate of the Multi-Layer Perception model increased to 100%. The detailed results were summarized in the Table 3.
Since the full Logistic Regression Model, SVC model and Multi-Layer Perception model only reached classification rate of 95% for the multiclass dataset, there is no need to reduce the number of features. However, in order to compare the different feature selection methods, multiple models with these feature selection methods were fitted. Not unexpected, the fitted Logistic Regression Model, SVC Model as well as Multi-Layer Perception model with SelectKBest method reached a lower classification rate (47.5%, 40% and 82.5% respectively). Then I increased the number of selected features by changing the value of k to 10, the classification of these models still did not appear obvious improvement. However, in comparison to SelectKBest method that selected 10 features, the prediction performance improved obviously when models with PCA (10 components) were fitted for the multiclass dataset (the details can be found in the Table 3). Therefore, PCA outperforms SelectKBest method when simplify classification models.

Model	Feature Selection Method	Selected Features (index)	Classification Rate
Logistic Regression	None	all	95%
Logistic Regression	SelectKBest	50	47.5%
Logistic Regression	SelectKBest	48, 49, 50, 51, 52, 53, 562, 563, 564, 565	60%
Logistic Regression	PCA	10 Components	90%
SVC	None	all	95%
SVC	SelectKBest	50	40%
SVC	SelectKBest	52, 285, 382, 383, 393, 395, 400, 563, 564, 618	55%
SVC	PCA	10 Components	95%
Multi-Layer Perception (layer_size(5,5))	None	all	95%
Multi-Layer Perception (layer_size(7,7))	None	all	100%
Multi-Layer Perception (layer_size(7,7))	SelectKBest	50	82.5%
Multi-Layer Perception (layer_size(7,7))	SelectKBest	52, 285, 382, 383, 393, 395, 400, 563, 564, 618	92.5%
Multi-Layer Perception (layer_size(7,7))	PCA	10 Components	95%

Table 3: Results of all fitted models for the multiclass data set

Summary & Discussion

At the beginning of the report, it analyzed the reasons why there is no need to handle missing data and reasons of splitting data sets into training and test set. Then the binary and multiclass data sets are statistically described which provides more details and it is helpful to understand the data.

For the binary data set, Logistic Regression, Support Vector Classifiers (linear kernel) as well as Multi-Layer Perception were fitted. These models with all 768 attributes reached a classification rate of 100%. Therefore, some feature selection methods, like Univariate Feature Selection, Principal

Components Analysis or Recursive Feature Elimination were used to simplify the models. Surprisingly, all models with a feature subset reached a classification rate of 100%.

Because the classification rate of all fitted models for the binary data sets had reached 100%, there is no need to further improve the performance and simplify the models. Therefore, I moved on to the multiclass data set. Firstly, a Logistic Regression Model and Support Vector Classifier as well as Multi-Layer Perception (5*5 size of layer) containing all 768 attributes were fitted for the multiclass data set. The multiclass data set can be expected to be more complicated than the binary data sets, so it is not surprising that the full Logistic Regression, Support Vector Classifier model and Multi-Layer Perception model with 5*5 size of layer only classify 95% of the samples in the test. This is why increase the size of layer from 5*5 to 7*7 for Multi-Layer Perception to optimize the classification model. Not unexpectedly, Multi-Layer Perception with 7*7 size of layer finally reached a classification rate of 100%. Based on the performance differences, Multi-Layer Perception could be confirmed to be more suitable for the multiclass data sets than Logistic Regression and Support Vector Classifier.

Although there is no need to simplify the model for the full Logistic Regression Model and the full Support Vector Classifier, I applied the different feature selection methods to these models and observed the performance differences of models from different feature selection methods. According to the results, it can be concluded that Principal Components Analysis outperformed Univariate Feature Selection when they both selected 10 optimized features. However, the conclusion still needs to be confirmed further by applying these methods to different data sets.

All in all, for the binary data sets, it is extremely hard to compare different classification models because all the models reached 100% classification rate. However, it presented differences between different classification models for the multiclass data sets, and Multi-Layer Perception seemed to be more suitable for the multiclass data sets than other models because it can be optimized by changing the size of layer. But the open question remaining is why SVC model with Principal Components Analysis performs better on the multiclass data set than the Logistic Regression Model with the same feature selection method, while the full Logistic Regression Model reached the same classification rate as the full SVC model. Finally, further study could be focused on a more granular classification with more objects.

Reference

[1] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.SelectKBest.html

[2] Ringnér M. What is principal component analysis?. Nature biotechnology. 2008 Mar;26(3):303.

[3] Lin X, Yang F, Zhou L, Yin P, Kong H, Xing W, Lu X, Jia L, Wang Q, Xu G. A support vector machine-recursive feature elimination feature selection method based on artificial contrast variables and mutual information. Journal of chromatography B. 2012 Dec 1;910:149-55.

[4] https://scikit-learn.org/stable/modules/generated/slearn.decomposition.PCA.html

[5] https://scikit-learn.org/stable/modules/generated/sklearn.feature_selection.RFE.html

[6] https://scikit-learn.org/stable/modules/generated/sklearn.neural_network.MLPClassifier.html

Appendix

Class ID	0	1	2	3	4
0	5	0	0	0	0
1	0	11	0	0	1
2	0	1	5	0	0
3	0	0	0	10	0
4	0	0	0	0	7

Table 4: Confusion Matrix for Logistic Regression with all attributes on the multiclass data set

Class ID	0	1	2	3	4
0	5	0	0	0	0
1	0	12	0	0	0
2	0	2	4	0	0
3	0	0	0	10	0
4	0	0	0	0	7

Table 5: Confusion Matrix for SVC with all attributes on the multiclass data set

Class ID	0	1	2	3	4
0	5	0	0	0	0
1	0	12	0	0	0
2	0	0	6	0	0
3	0	0	0	10	0
4	0	0	0	0	7

*Table 6: Confusion Matrix for Multi-Layer Perception (7*7 size of layer) with all attributes on the multiclass data set*

Sample Index	Class Prediction
1	1
2	1
3	1
4	1
5	1
6	1
7	1
8	1
9	1
10	1
11	0
12	0
13	0
14	0
15	0
16	0
17	0
18	0
19	0
20	0

Table 7: Class prediction for the binary task

Sample Index	Class Prediction
1	2
2	2
3	2
4	2
5	2
6	2
7	2
8	2
9	2
10	2
11	0
12	0
13	0
14	0
15	0
16	0
17	0
18	0
19	0
20	0
21	3
22	3
23	3
24	3
25	3

26	3
27	3
28	3
29	3
30	3
31	1
32	1
33	1
34	1
35	1
36	1
37	1
38	1
39	1
40	1
41	4
42	4
43	4
44	4
45	4
46	4
47	4
48	4
49	4
50	4

Table 8: Class predictions for the multiclass task