Machine Learning: Fourth Home Work

Support Vector Machines

Edoardo Ghini

December 17, 2016

Dipartimento di Ingegneria dell'Università di Roma La Sapienza

Contents

Part I

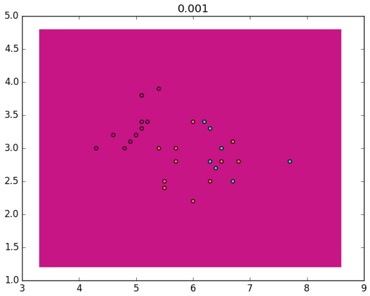
Introduction

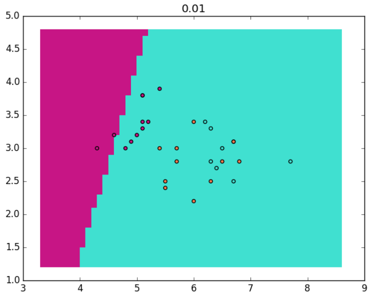
1 Scope

This time the aim of the assignment has been to comprehend the concept behind the classification through Support Vector Machines

2 Objectives

For the purpose of understanding SVM algorithm and theory, it will be executed a classification task with, at first, a linear SVM, and then with a non-linear one in order to proof that non-linearity brings better scores because of more flexible classification boundaries. At the end I will have to validate and test the performances of the classifier with a K-fold approach.





Part II

Development

3 Data Manipulations

At the beginning, after that the first two vectors of a dataset had been loaded, I obtained a separation in train, validation and test data.

4 Linear SVM

4.1 Theory pills

As shown in these figures, I have started with a lineal model that, being a SVM classifier, maximises the distance between the first points of two different classes and the decision linear function. This maximum problem tries to find the greatest area in the data space that can be considered neutral between two data point concentrations. There is also a coefficient (C) that allows a different weighting of the penalty for missclassification.

4.2 Validation and Test phases

The aforementioned figures have been generated with different values of C. In fig(5), it can be seen that for this dataset the best performances on the validation sets where obtained with a C parameters equal to 100.

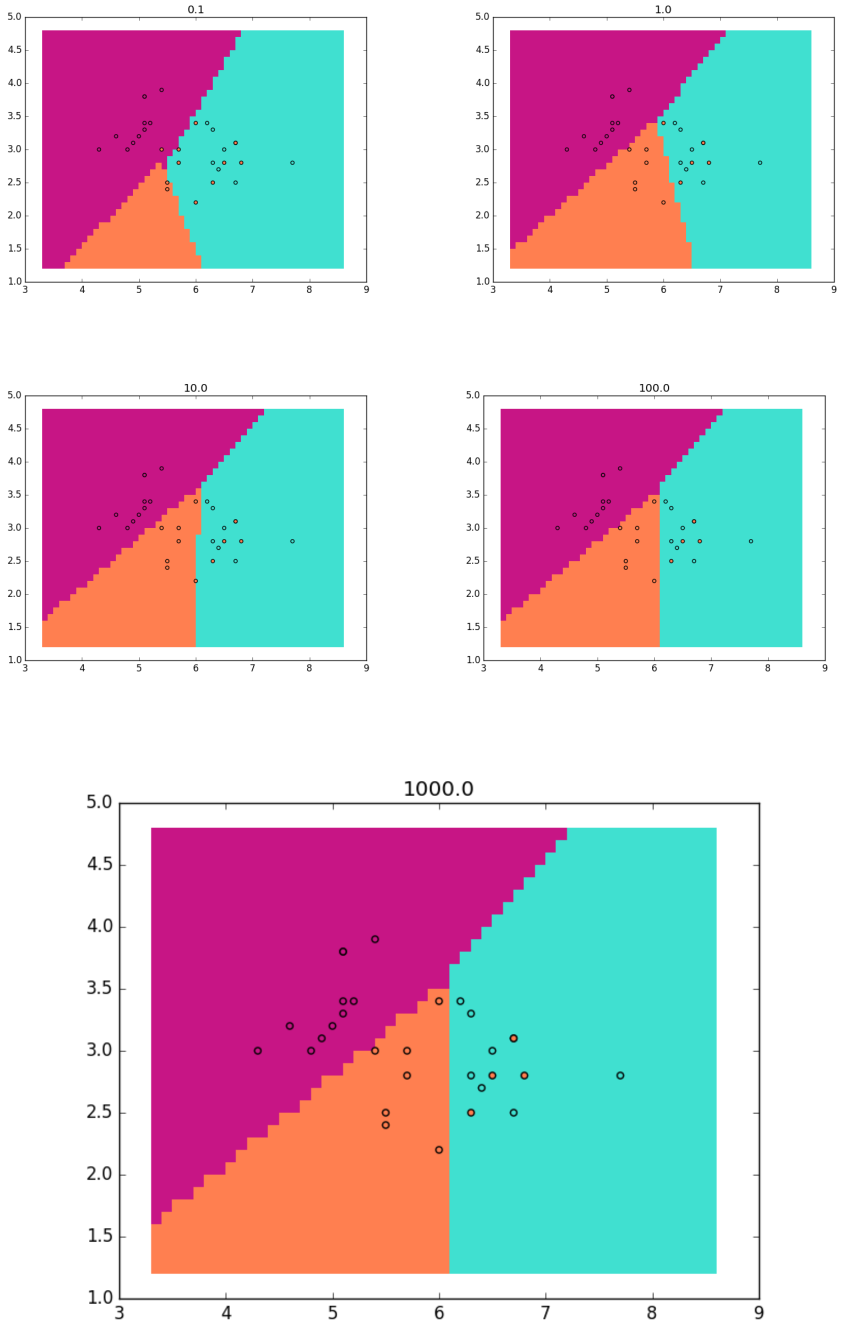


Figure 4

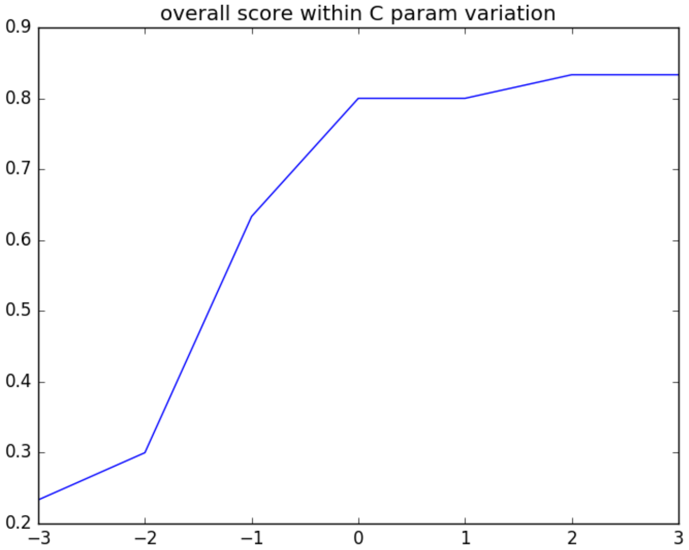


Figure 5

Correspondingly, in fig(6), we can see how this model performs on test data.

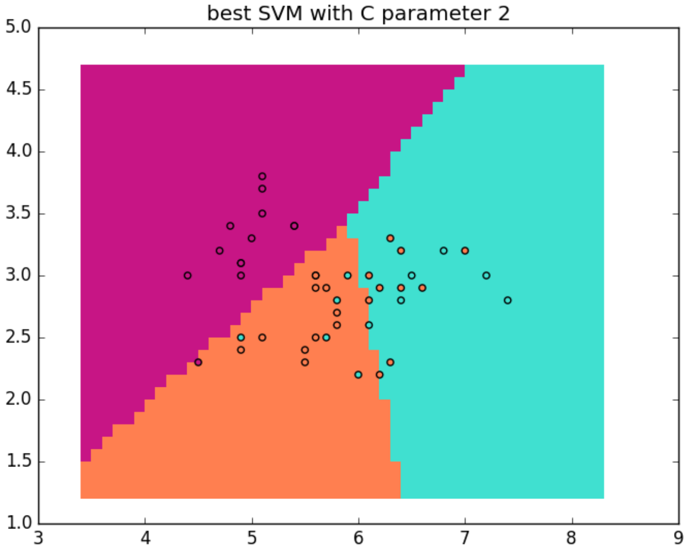
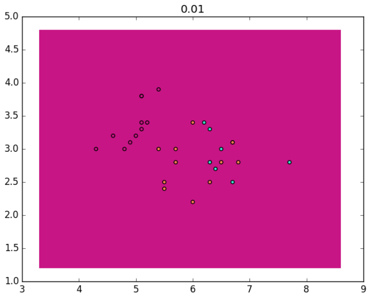


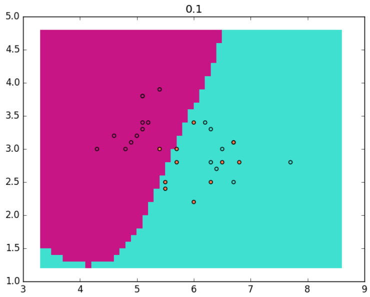
Figure 6

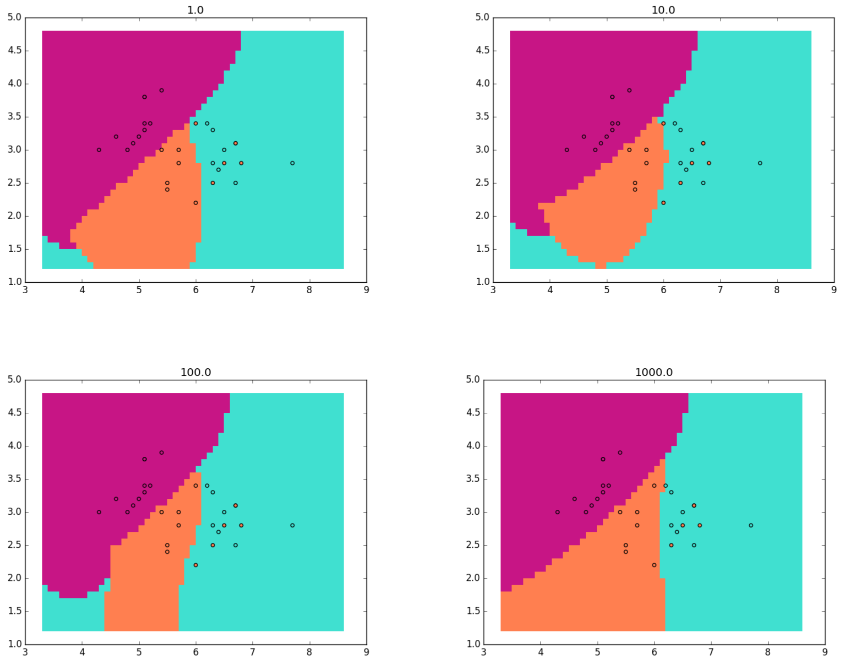
5 Non-linear SVM

5.1 Training with C and gamma

Similarly, I repeated the same procedure with a Radial Basis Function Kernel. In practice, the model can decide smoother boundaries for data classification because of the kernel operation applied on the data in order to bring non-linearity. These figures are the results of a RBF SVM on the dataset with different values for C parameter.







5.2 First scores

As shown in fig(10) the model, identically to the linear experiment, behaves better. with high values of C. In particular, in fig(11) there is the plot of the SVM that has performed better on validation dataset.

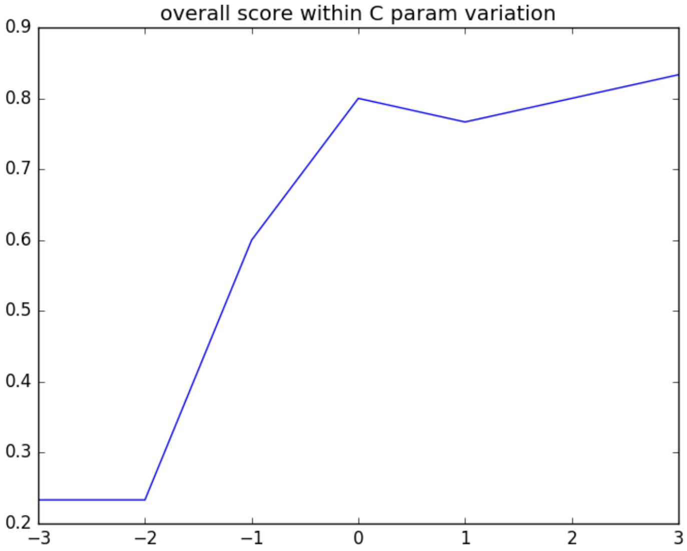


Figure 10

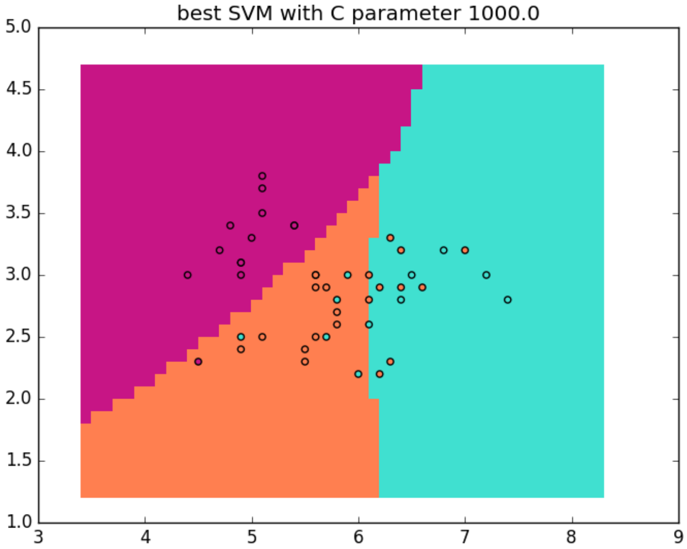


Figure 11

5.3 Grid search with K-fold validation

In order to chose the best classifier parameters, as shown in 1 I merged together train and validation datasets and I set up an iterator that will be used to perform a k-fold validation. Subsequently, I managed to build up a grid search to find the best parameter combination of C and gamma.

```
X_train_merged, X_test_merged, y_train_merged, y_test_merged=
    train_test_split(X, y,  test_size=split_rate_merged,
    random_state=20)

C_range = np.logspace(-3, 6, 10)
gamma_range = np.logspace(-6, 3, 10)
param_grid = dict(gamma=gamma_range, C=C_range)
cv = StratifiedShuffleSplit(n_splits=5, test_size=0.2, random_state=
    42)
grid = GridSearchCV(SVC(), param_grid=param_grid, cv=cv)
grid.fit(X_train_merged, y_train_merged)
```

Code 1: Kfold wihin a grid search

Gamma parameter seems to modify the way in which the classifier consider the significance of a data point according to his distance from the classification separator.

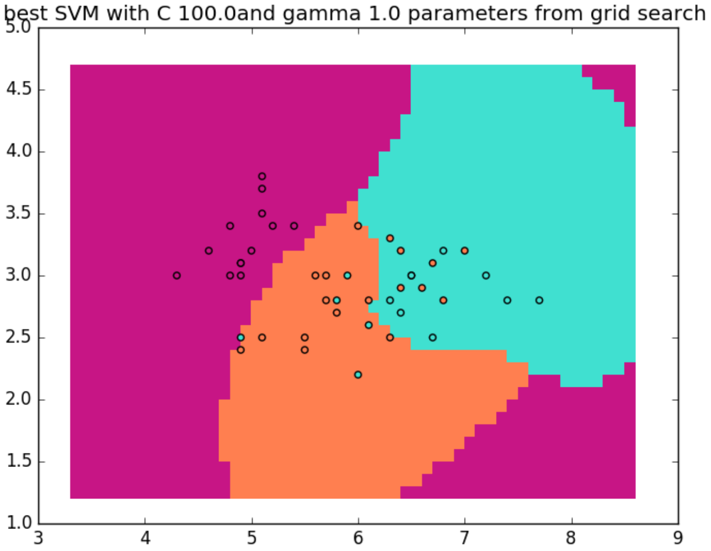


Figure 12

Part III

Conclusions

To sum up, in fig(12) is shown how a smart choice of parameters, in the case of a RBF classifier, C and gamma, could improve the score quality of the model.

This particular model with C =100 and gamma = 1 has performed a score of 0.86 on the validation set and of 0.68 on the merged set with further reliability due to the 5-fold validation.