Introduction to AI: assignment 4 - EUR DS4H

Support vector machines

Part I: Questions on a simple dataset

These questions are based on Assignment 9 of the class Machine Learning for Big Data (Polytech Nice-Lionel Fillatre)

A. Linear SVM example

Consider the following training data:

$x^{(1)}$	$x^{(2)}$	Class label
1	1	1
2	2	1
2	0	1
0	0	0
1	0	0
0	1	0

1. Draw a scatter plot of these six data points. Are these classes linearly separable?

2. Draw the prediction border line that separates the two classes with maximum margin and indicate the data points which are the support vectors.

3. If you remove one of the support vectors, does the border line change? If it changes, does the margin increase, decrease or stay the same?

4. The observation with features $x^{(1)} = 1$, $x^{(2)} = 2$ and class label $y = 0$ has been added to the dataset and a soft margin linear SVM classifier has been fitted to this dataset. The parameters of this SVM are $\beta_0 = -3$, $\beta_1 = 2$ and $\beta_2 = 0$.

   Draw the updated dataset and the border line between the classes given by the SVM. Give the value of the SVM margin and, based on this value, draw the margin borders and indicate with circles the support vectors.

Part II: SVM applied to handwritten digit recognition

In this part, we are going to use again the digits dataset from the previous labwork.

A. Binary classification of pairs of digits

We are going to apply SVM classifiers to the 3 datasets considered in the previous labwork, that is, the data for classifying digits 0 *vs.* 1, 3 *vs.* 8, 8 *vs.* 9. As in the previous labwork, classification will be carried out on data after applying principal component analysis with output dimension equal to $d = 2$.

1. Visualize the 3 datasets with scatterplots. For each dataset, which of the following SVM classifiers is the most adequate for classifying the pairs of digits

   - Hard margin linear SVM?
   - Soft margin linear SVM?
   - Kernel SVM?

2. If a hard margin linear SVM seems adequate to classify one of the datasets, train the corresponding classifier with Scikit-learn and evaluate its accuracy.

   To import the SVM method use the command

   ```
   from sklearn import svm
   ```

   and to define a hard margin linear SVM classifier, do the following:

   ```
   svm_hlin = svm.SVC(kernel='linear',C=1e10)
   ```

   Note that all SVM from function *SVC* use soft margins, why do we need to set C to a high value in this case?

   By using the method *decision_function* from the SVM model and *pyplot*'s command contour, draw the border line between the two predicted classes and 2 dashed lines corresponding to the 2 borders of the margin. Draw the scatter plot with the features on top of the contours and indicate with circles which data points are support vectors. The support vectors can be found in the attribute *support_vectors_* of the fitted SVM model. You should obtain a result similar to Fig. 1.

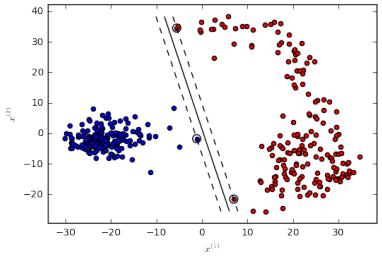

Figure 1: SVM decision line and its margins.

3. Apply a soft margin linear SVM to classify the dataset which is almost linearly separable. Use the parameter value $C = 1$.

   Evaluate the classification accuracy and do a similar drawing as in the previous question illustrating the characteristics of the SVM classifier (decision line, borders of the margin and support vectors).

4. Repeat what you did in the previous exercise for the following values of soft SVM parameter $C \in \{5, 50, 500\} \times 10^{-6}$. What is the overall observed behavior?

5. Apply a soft margin kernel SVM to classify the dataset which does not seem to be close to linear separability. Use a radial basis function kernel (`kernel='rbf'`) and parameters $C = 10^2$ and $\gamma = 10^{-4}$. Display the characteristics of the obtained classifier as in the previous exercises.

6. Test different values of γ, for example, $\gamma \in \{10^{-3}, 10^{-2}, 10^{-1}\}$. What happen when γ increases ?

B. Multi-class classification

In this exercise, we are going to test multi-class classifiers based on one *vs.* one (OVO) and one *vs.* rest (OVR) strategies. We are going to test these strategies on a part of the digits dataset corresponding to digits 1, 3 and 4. We will apply PCA with $d = 2$ to this dataset as we did in the previous exercises.

1. Test OVO and OVR strategies using a soft margin linear SVM for the underlying binary classification. Look into *sklearn* documentation[1] to see how to apply these strategies with SVM. For the binary soft SVM classifier you can use parameter value $C = 1$.

   Evaluate the accuracy of both strategies and draw with different colors the corresponding predicted classes regions behind a scatterplot of the data. For OVO strategy, you should obtain a graphic similar to Fig. 2. To draw the different predicted classes regions as different background colors you can use the function *pcolormesh* from *matplotlib.pyplot*.

   Is there a difference in performance between the strategies? Why?

2. Repeat the previous exercise but use a kernel (soft margin) SVM with radial basis function kernel. You can use the following parameters for the SVM classifiers $C = 1$ and $\gamma = 10^{-3}$. Is there a difference in performance between strategies in this case? Why?

[1] `https://scikit-learn.org/stable/modules/multiclass.html`

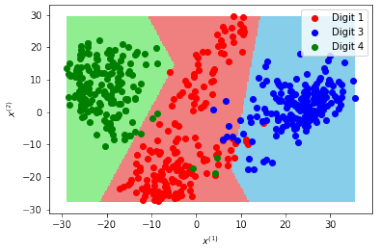

Figure 2: Scatter plot with colored predicted classes regions obtained with OVO strategy and soft margin linear SVM.