INF264 - Exercise 2

August 2019

1 Instructions

- Deadline: 20/09/2019 , 23:59
- Submission place: https://mitt.uib.no/courses/19532/assignments
- Format: Your answers are to be returned in a single pdf report. You can also return scanned pages for your calculations. For results, your answers must include any values and plots that are requested in the Notebook.

2 Linear SVM

First load the dataset "svmlin.csv". In the first part of the exercise, you are required to find two separating lines that classify the X point on the plot with coordinates (0.6,2.1) in a different class. (Once in the yellow class and once in the red class). Then you should compute the margin using the formula below:

$$margin = \frac{|x2 + m * x1 + b|}{\sqrt{m^2 + 1}} \tag{1}$$

- what values of m and b do you use for your line?
- Plot the line. Which line gets a better margin? Why?

Next fit a linear SVM to the dataset and plot the boundary.(use $C = 1^{-}10$ with linear kernel)

- Plot the decision boundary and indicate the support vectors. How many support vectors are in the plot? Explain why they are chosen as the support vectors.

3 Kernel SVM

Load the dataset "svmnonlin.csv". First try to fit a linear SVM to the data set and plot the boundaries.

- What is the result of fitting a linear SVM to the data? Why?

Next, you should complete the function for computing the function of the SVM classifier for kernel SVM. Predictions are made using the formula below:

$$f(x) = sign(\sum_{i}^{N} \alpha_i y_i K(x_i, x) + b) \tag{2}$$

for N number of training samples, where α_i are the learned weights (positive only for the support vectors).

Then you should fit a SVM with RBF kernel to the dataset and plot the function inside the sign function in 3D and the decision boundary in the input space (2D)

- Where are the red and yellow datapoints mainly located on the surface?

Next, fit a SVM with second degree polynomial kernel to the datapoints. Plot the 3D surface and 2D decision boundaries for this kernel.

- How is this surface different than the RBF surface? which one classifies the data better?

4 Softening the margins

Load the dataset 'svmmargin.csv'. This dataset is not linearly separable. To handle this case, the SVM implementation has a bit of a fudge-factor which "softens" the margin: that is, it allows some of the points to creep into the margin if that allows a better fit. The hardness of the margin is controlled by a tuning parameter, most often known as C. For very large C, the margin is hard, and points cannot lie in it. For smaller C, the margin is softer, and can grow to encompass some points.

Fit a linear SVM model with two values of 0.1 and 10 for C and see the effect of the C on SVM boundry and support vectors.

- What is different about the plot with $C = 10$ and $C = 0.1$? How does the value of C effect the number of support vectors?

Next, We need to choose the best value of C for classifying Iris dataset.

- Load the Iris dataset and split the data into training and test sets.

- Determine the best value of C using cross validation and measure the value of the accuracy score on the test set. What value of C do you get? What is the accuracy?

5 Kernel Trick

The kernel trick avoids transforming data vectors explicitly to high dimensions, thus avoiding computationally intensive operations.

We will now try to elaborate this point with an example. Each of the x or z (d dimensional data) can be transformed into higher demensions mapping of second degree polynomial kernel through

$$\phi(x) = (x_d^2, ..., x_1^2, \sqrt{2}x_d x_{d-1}, ..., \sqrt{2}x_d x_1, \sqrt{2}x_{d-1}x_{d-2}, ..., \sqrt{2}x_{d-1}x_1, ..., \sqrt{2}x_d, ..., \sqrt{2}x_1, 1) \quad (3)$$

and the kernel is computed through

$$K(x, z) = \phi(x)^T \phi(z) \quad (4)$$

Also the formula for second degree polynomial kernel is as:

$$K(x, z) = (1 + x^T z)^2 \quad (5)$$

- compute the value $\phi(x)^T \phi(z)$ by explicitly transforming the data into higher dimension and also using kernel K.
- Measure the time of each procedure and plot the running time with the increasing number of dimensions.
- Compare the values of the kernel in both of the procedures.

6 Naïve Bayes Classification

Consider the data which shows if tennis was played on a Saturday based on weather, temperature, humdity, and wind conditions of the day in Table 1.

- First, calculate $P(x|PlayTenis)$ for every x in the table.

  (Hint: To calculate, let's say, $P(Sunny|Yes)$ and $P(Sunny|No)$, we first make a table of **Outlook** and **PlayTennis** entries, and Estimate these probabilities using Maximum Likelihood.)

- Suppose that the forecast for the coming Saturday is:
  - weather: Sunny
  - temperature: Hot
  - humidity: Normal
  - wind: False

  Based on the data for the 14 previous Saturdays, how likely it is that tennis will be played the coming Saturday using Naïve Bayes?

Day #	Outlook	Temperature	Humidity	Wind	PlayTennis
0	Rainy	Hot	High	False	Yes
1	Rainy	Hot	High	True	No
2	Overcast	Hot	High	False	Yes
3	Sunny	Mild	High	False	Yes
4	Sunny	Cool	Normal	False	No
5	Sunny	Cool	Normal	True	No
6	Overcast	Cool	Normal	True	Yes
7	Rainy	Mild	High	False	No
8	Rainy	Cool	Normal	False	Yes
9	Sunny	Mild	Normal	False	Yes
10	Rainy	Mild	Normal	True	Yes
11	Overcast	Mild	High	True	Yes
12	Overcast	Hot	Normal	False	Yes
13	Sunny	Mild	High	True	No

Table 1: Tenis dataset

- What is the probability that there won't be a tennis match with following conditions?
  - weather: Rainy
  - temprature: Cool