Problem Set 1

Due Sunday, October 15, 2017 at 11:55pm

How to Submit

Create one .zip file (**not** .rar or something else) of your code and written answers and submit it via `ilearn.ucr.edu`. Your zip file should contain

- Problem 1: ans1.[pdf|txt]
- Problem 2: learnnb.m & prednb.m
- Problem 3: ans3.[pdf|txt], ans3plot1.pdf & ans3plot2.pdf

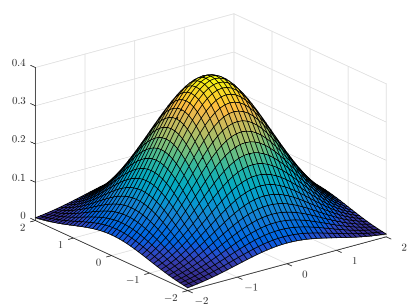
Submit answers in a pdf or ascii text file, not an MS Word document.

Each file should include at the top (in comments if necessary)

- Your name & UCR student ID number
- The date
- The course (CS 171) & assignment number (PS 1)

Problem 1. [5 pts]

Assume that the distribution of x given the class label y (the class-conditional distribution of the features) is a normal distribution. A plot of this distribution is shown below.



Normal distribution with $\sigma = 1$ and $\mu = [0\ 0]^\top$

We will assume a binary classification problem with classes A and B with prior probabilities of p_A and p_B, and that the width or variance of the class-conditional distributions, σ, is the same for both classes. Thus, if μ_A is the center of $p(x \mid y = A)$ and μ_B is the same for $y = B$,

$$p(x \mid y = A) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu_A)^\top (x-\mu_A)} \qquad p(y = A) = p_A$$

$$p(x \mid y = B) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2\sigma^2}(x-\mu_B)^\top (x-\mu_B)} \qquad p(y = B) = p_B\ .$$

If these are the true distributions, what is the Bayes-optimal decision rule?

Demonstrate that this implies a linear decision boundary between the two classes.

Draw the decision boundary for the four 2d cases below.

(a) $p_A = p_B = 0.5$, $\mu_A = [0\ 0]^\top$, $\mu_B = [1\ 2]^\top$, $\sigma = 1$
(b) $p_A = p_B = 0.5$, $\mu_A = [0\ 0]^\top$, $\mu_B = [1\ 2]^\top$, $\sigma = 3$
(c) $p_A = p_B = 0.5$, $\mu_A = [0\ 0]^\top$, $\mu_B = [3\ 2]^\top$, $\sigma = 1$
(d) $p_A = 0.25$, $p_B = 0.75$, $\mu_A = [0\ 0]^\top$, $\mu_B = [1\ 2]^\top$, $\sigma = 1$

Problem 2. [10 pts]

Your task is to implement naïve Bayes learning and testing. You should write two functions:

```
function [priorp,condp] = learnnb(X,Y)

function predY = prednb(X,priorp,condp)
```

`X` is a m-by-n matrix representing m examples, each with n features. For this assignment, you may assume that each feature is one of three values, 0, 1, or 2. `Y` is a m-by-1 vector. For this assignment, you may assume that there are only two classes: 0 and 1.

The first function, `learnnb` estimates the parameters of a naïve Bayes model. In particular, it should return two things:

- `priorp`, a 1-by-2 vector of the prior probabilities of each class.
- `condp`, a 3-by-2-by-n 3d array. The (i, j, k) element should be the estimate of $\mathrm{Prob}(x_k = i \mid y = j)$

The second function, `prednd` should use these estimated parameters to return the predicted class for each of the rows of the input X.

Supplied with the assignment are seven files:

- `runq2.m`, a function which runs the code and outputs the fraction correct
- `loadspdata.m`, a function which will load datasets in the particular format for this assignment
- `toytrain.txt` & `toytest.txt`, two files that encode the example problem from class (you can use to check your calculations)
- `trainspam.txt` & `testspam.txt`, two files that encode a real spam-filtering test. There are 100 features, each corresponding to a word. The word list can be found in `wordlst.txt` and are the 100 most common words in the e-mails, after word endings have been removed. Each example is an e-mail to a linguistics e-mail list. Class 0 is non-spam. Class 1 is spam.
  This data was collected by Ion Androutsopoulos and kindly provided for researchers.

Problem 3. [10 pts]

Consider the following 1-dimensional regression dataset

x	y
0	1
2	-3
2	-2
3	-3
-1	-1
1	-1

Fit a third-degree polynomial to this data using least squares regression. You may use Matlab to do the calculations (matrix inversions, for instance), but show all of your steps. Write the resulting $\hat{f}(x)$ as a third degree polynomial in x

Use Matlab to plot the resulting function and data on the same plot (the data as points, the function as a smooth curve), on the range of $x \in [-1, 4]$.

Now perform the same two steps again (calculate the third-degree polynomial fit and plot the resulting function with the data) for ridge regression with $\lambda = 5$.