Assignment 2

Statistical Learning – Fall 2018

Assignment Date: 1397/08/30

Due Date: Section 1) 1397/09/11, Section 2) 1397/09/18 (no extension)

Note1: For the lab sections, you can use any software which you are familiar with (Python, Matlab or R).

Section 1

1) Consider a two label classification problem in two dimensions with $p(x|\omega_1) \sim N(0,1)$, $p(x|\omega_2) \sim N(1,1)$ and $p(\omega_1) = p(\omega_2) = \frac{1}{2}$.
   a) Calculate the QDA and Bayes decision boundary
   b) Calculate the Bayes decision boundary in the case that $p(\omega_1) = \frac{1}{3}$, $p(\omega_2) = \frac{2}{3}$.

2) Suppose we estimate the regression coefficients in a linear regression model by minimizing :

$$\Sigma_{i=1}^{n}\left(y_i - \beta_0 - \Sigma_{j=1}^{n}\beta_j x_{ij}\right)^2 \; s.t. \; \Sigma_{j=1}^{p}\left|\beta_j\right| \leq s$$

For a particular value of s. For parts (a) through (e), indicate which of i. through v. is correct. Justify your answer.
(a) As we increase “s” from 0, the training RSS will:
i. Increase initially, and then eventually start decreasing in an inverted U shape.
ii. Decrease initially, and then eventually start increasing in a

U shape.
iii. Steadily increase.
iv. Steadily decrease.
v. Remain constant.
(b) Repeat (a) for test RSS.
(c) Repeat (a) for variance.
(d) Repeat (a) for (squared) bias.
(e) Repeat (a) for the irreducible error

3) We have seen that β of ridge regression, can be computed as $\boldsymbol{\beta}^{Ridge} = \left(\mathbf{X}^{\mathbf{T}}\mathbf{X} + \lambda\mathbf{I}_{\mathrm{p}}\right)^{-\mathbf{1}}\mathbf{X}^{\mathbf{T}}\mathbf{y}$**.** Show that the same $\boldsymbol{\beta}$ can be found if we use simple least square regression on the augmented dataset, i.e., where:

$$\mathbf{X}' = \begin{pmatrix} \mathbf{X} \\ \sqrt{\lambda}\mathbf{I}_{\mathrm{p}} \end{pmatrix}, \boldsymbol{y}' = \begin{pmatrix} \boldsymbol{y} \\ \mathbf{0} \end{pmatrix}$$

4) Solve questions 3.2, 3.19 and 3.27 and 3.30 from the “The Elements of Statistical Learning” book.

Section 2

Fist look at the following blog on implementation of fraud-detection algorithm using logistic regression (If you cannot see this page, I have uploaded the PDF version of this page on moodle).

https://ipythonquant.wordpress.com/2018/05/08/from-logistic-regression-in-scikit-learn-to-deep-learning-with-tensorflow-a-fraud-detection-case-study-part-i/

There is also a ipython implementation of all the steps:

https://github.com/mgroncki/IPythonScripts/blob/master/LogisticRegression_Part1.ipynb

The dataset for this part is also uploaded into moodle. Note that, due to the files size limitation, the number of record you see in the database is less than what is reported in the original document.

What you need to do for this assignment, is:

1- Redo the same question but for “sklearn” implementation of logistic regression, use tensorflow and implement you own logistic regression code for classification and finding result.

2- Compare your results with that given by the author (using sklearn package)

3- Use SVM for the same task, report and compare your results. You are allowed to use sklearn implementation of SVM for this part

4- Now look at the following page to see how SVM (and its variants) can be implemented by tensorflow (a PDF version is uploaded) https://medium.com/cs-note/tensorflow-ch4-support-vector-machines-c9ad18878c76

5- Redo part 3, this time in tensorflow; report and compare your results.