Name: Mouselinos Spyridon

Date: January 2020

Machine Learning and Computational Statistics

Homework 2

Exersize 1

Comments about the solution.

- Data Generation:
  In this step i used the functions defined in the supplement code snippets.
  Our X data points derived from a multimodal normal distribution with 0 means and the identity matrix as covariance matrix.

The noise η added came from a normal distribution with 0 mean and 0.05 std.

After plotting the data in 3D we could somewhat identify the plane on which they lied onto.

- Data Transformation
  In this step i sliced our data vector into 2 columns and created a third one by multiplying the first two,

so we could create the $x_1 x_2$ term.
Afterwards, i stacked the columns into a X array again, verifying that the shape was correct.

- Linear Modeling under MSE
  In the third step i added a leading 1 column to our X matrix in order to perform the minimization under MSE steps.
  Afterwards, i created the X^T as well as the X^T and $(X^T X)^{-1}$
  matrices using numpy.
  Finally i solved for θ, and calculated the MSE of our model.

Exersize 2

Comments about the solution.

- Data Generation:
  In this step i decided to use the random uniform distribution from numpy with lower bound -2 and upper bound +2 during the data generation.
  After the creation of the data, the column vector y was created assigning each point that $x_2^2 - x_1^2 > 0$ to the +1 class and -1 elsewhere. Note that the function np.sign was used as a one-liner solution for this task.
  After plotting our data they seem uncapable of beign linearly split, as they create a bowtie-like shape.

- Data Transformation
  In this step i sliced the original X matrix into columns, squared them and reassembled them into a new X matrix.

- Plot and comparison
  In this step i plotted the transformed data and observed almost perfect linear separability between them.

- Linear modeling under MSE
  The final step was to perform again solution under mse following the same steps as in Ex1.

The only difference here was that due to the fact that we perform classification we satisfied the equation $\theta^T x = 0$ that i decided to plot on top of our data in order to show the final decision boundary.

Exersize 3

Consider the following nonlinear model:

$$y = 3x_1^2 + 4x_2^2 + 5x_3^2 + 7x_1x_2 + x_1x_3 + 4x_2x_3 - 2x_1 - 3x_2 - 5x_3 + \eta$$

- Define a suitable function φ that transforms the problem to a space where the problem of estimating the model becomes linear. What is the dimension of the original and the transformed space?

Solution:

First of all let's explore our given model:
It is an instance of the parametric set of 2nd degree polynomials of 3 variables, thus we have an input of **3 dimensions** $x_{input} = [x_1, x_2, x_3]$.
$f_\theta : R^3 \rightarrow R$

A suitable transformation would be that of $\varphi(\chi) = \begin{bmatrix} \varphi_1(x) \\ \varphi_2(x) \\ \varphi_3(x) \\ \varphi_4(x) \\ \varphi_5(x) \\ \varphi_6(x) \\ \varphi_7(x) \\ \varphi_8(x) \\ \varphi_9(x) \end{bmatrix} = \begin{bmatrix} x_1^2 \\ x_2^2 \\ x_3^2 \\ x_1x_2 \\ x_1x_3 \\ x_2x_3 \\ x_1 \\ x_2 \\ x_3 \end{bmatrix}$

Then the above relation could be re-written as a linear combination of the transformed values as:

$$y = 3\varphi_1(x) + 4\varphi_2(x) + 5\varphi_3(x) + 7\varphi_4(x) + \varphi_5(x) + 4\varphi_6(x) - 2\varphi_7(x) - 3\varphi_8(x) - 5\varphi_9(x) + \eta$$

However, that is an instance of the parametric set of 1st degree polynomials of 9 variables, thus we have an input of **9 dimensions**

$x_{input} = [\varphi_1(x), \varphi_2(x), \varphi_3(x), \varphi_4(x), \ldots$
$\ldots, \varphi_5(x), \varphi_6(x), \varphi_7(x), \varphi_8(x), \varphi_9(x)].$

$f'_\theta : R^9 \to R$

Exersize 4

Consider the following two-class nonlinear classification task:

$\mathbf{x} = [x_1, x_2, x_3]^T$
$x_1^2 + 3x_2^2 + 6x_3^2 + x_1x_2 + x_2x_3 > (<) 3 \to \mathbf{x} \in \omega_1(\omega_2)$

Define a suitable function φ that transforms the problem to a space where the problem of estimating the border of the two classes becomes linear. What is the dimension of the original and the transformed space?

Solution:

First of all let's explore our given model:
It is an instance of the parametric set of 2nd degree polynomials of 3 variables, thus we have an input of **3 dimensions** $x_{input} = [x_1, x_2, x_3]$.
$f_\theta : R^3 \rightarrow R$

A suitable transformation would be that of $\varphi(\chi) = \begin{bmatrix} \varphi_1(x) \\ \varphi_2(x) \\ \varphi_3(x) \\ \varphi_4(x) \\ \varphi_5(x) \end{bmatrix} = \begin{bmatrix} x_1^2 \\ x_2^2 \\ x_3^2 \\ x_1 x_2 \\ x_2 x_3 \end{bmatrix}$

Then the above relation could be re-written as a linear combination of the transformed values as:

$$y = \varphi_1(x) + 3\varphi_2(x) + 6\varphi_3(x) + \varphi_4(x) + \varphi_5(x) + \eta$$

However, that is an instance of the parametric set of 1st degree polynomials of 5 variables, thus we have an input of **5 dimensions**
$x_{input} = [\varphi_1(x), \varphi_2(x), \varphi_3(x), \varphi_4(x), \ldots$
$\ldots, \varphi_5(x)]$.

Exersize 5

Consider the following data points:

$$
\begin{aligned}
x_1 &= [1,1]^T, \in class: +1 \\
x_2 &= [1,-1]^T, \in class: +1 \\
x_3 &= [0,0.5]^T, \in class: +1 \\
x_4 &= [-1,1]^T, \in class: -1 \\
x_5 &= [-1,-1]^T \in class: -1
\end{aligned}
$$

Solution:

First of all we have to identify that we have a 2D input of $\mathbf{x} = [x_1, x_2] \in R^2$ and a class set of $\mathbf{\Omega} = (\omega_1(+1), \omega_2(-1))$

We have 5 points in our dataset that will create the following X and y matrices:

$$
X = \begin{bmatrix} \mathbf{x_1^T} \\ \mathbf{x_2^T} \\ \mathbf{x_3^T} \\ \mathbf{x_4^T} \\ \mathbf{x_5^T} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & -1 \\ 1 & 0 & 0.5 \\ 1 & -1 & 1 \\ 1 & -1 & -1 \end{bmatrix}
$$

an 5×3 matrix.

and $y = \begin{bmatrix} 1 \\ 1 \\ 1 \\ -1 \\ -1 \end{bmatrix}$

an 5×1 matrix.

We will solve this problem by assuming a linear model such as:

$$\mathbf{0} = \theta_{\mathbf{0}} + \theta_{\mathbf{1}}\mathbf{x_1} + \theta_{\mathbf{2}}\mathbf{x_2}$$

where $\theta = [\theta_0, \theta_1, \theta_2]^T \in R^3$ is the parameter vector.

In order to solve this under least squares we need to calculate first the following matrices:

$$X^T = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & -1 & -1 \\ 1 & -1 & 0.5 & 1 & -1 \end{bmatrix}$$

$$X^T X = \begin{bmatrix} 5 & 0 & 0.5 \\ 0 & 4 & 0 \\ 0.5 & 0 & 4.25 \end{bmatrix}$$

$$(X^T X)^{-1} = \begin{bmatrix} 0.2024 & 0 & -0.2380 \\ 0 & 0.25 & 0 \\ -0.2380 & 0 & 0.2380 \end{bmatrix}$$

then according to the least squares criterion:

$$\theta = (X^T X)^{-1} X^T Y = [0.1905, 1, 0.0952]^T$$

That means that our hyperplane follows the equation:

$$\mathbf{0 = 0.1905 + x_1 + 0.0952x_2}$$

Exersize 6

Verify the sum, the product and the Bayes rule for the discrete-valued case, using the relative frequency definition of the probability.

Solution:

Let us define two discrete R.V named:

- $\mathbf{x}$ with sample space $X = \{x_1, x_2, ..., x_{nx}\}$
- $\mathbf{y}$ with sample space $Y = \{y_1, y_2, ..., y_{ny}\}$

Now let's conduct n trials of an experiment where:

- n_i^x is the number that x_i occured.
- n_j^y is the number that y_j occured.
- n_{ij} is the number that x_i and y_j occured simultaneously.

In that sense we can define:

- $P(x_i) = \frac{n_i^x}{n}$ and thus $P(\mathbf{x}) = \sum_{i=1}^{nx} P(x_i) = \frac{\sum_{i=1}^{nx} n_i^x}{n}$
- $P(y_j) = \frac{n_j^y}{n}$ and thus $P(\mathbf{y}) = \sum_{j=1}^{ny} P(y_j) = \frac{\sum_{j=1}^{ny} n_j^y}{n}$
- $P(x_i, y_j) = \frac{n_{ij}}{n}$

In this fashion we can say that marginally:
n_i^x , the number that x_i occured, is equal to the number it simultaneously occured alongside $\mathbf{every}$ y_j. So

- $(1)\ n_i^x = \sum_{j=1}^{ny} n_{ij}$

The same applies to n_j^y:

- $(2)\ n_j^y = \sum_{i=1}^{nx} n_{ij}$

Now let's verify the sum Rule:

$P(\mathbf{x} = x_i) = \frac{n_i^x}{n}$ and according to (1) $n_i^x = \sum_{j=1}^{ny} n_{ij}$ so

$$P(\mathbf{x} = x_i) = \frac{n_i^x}{n} = \frac{\sum_{j=1}^{ny} n_{ij}}{n} = \sum_{j=1}^{ny} \frac{n_{ij}}{n} = \sum_{j=1}^{ny} P(x_i, y_j)$$

Now let's verify the prouct Rule:

$$P(\mathbf{x} = x_i, \mathbf{y} = y_j) = \frac{n_{ij}}{n}$$

$P(\mathbf{x} = x_i | \mathbf{y} = y_j) =$ is the number of times
x_i occured on condition that y_j occured $= \frac{n_{ij}}{n_j^y}$

Now $P(\mathbf{x} = x_i | \mathbf{y} = y_j) P(\mathbf{y} = y_j) = \frac{n_{ij}}{n_j^y} \frac{n_j^y}{n} = \frac{n_{ij}}{n} = P(\mathbf{x} = x_i, \mathbf{y} = y_j)$