Machine Learning Exercise 1

Michael Mitsios cs2200011

March 25, 2021

2 Vector and Matrices

Tasks

1) The production of the 2 random integer matrices X, Y was by using the function randint(). Because the X, Y ϵ $\mathbb{Z}$ in general we may have negative values also. That's why I use as low border -5 and as high 5 in randint function.
2) The same logic is followed for the vectors. We are using the same function but this time the "size" parameter has only one dimension (because the vectors are actually 1-D arrays).

Computations

1) $g^T z = \begin{bmatrix} 1 & -2 & 4 & -3 \end{bmatrix} \cdot \begin{bmatrix} -5 \\ -1 \\ -3 \\ -1 \end{bmatrix} = 1 \cdot (-5) - 2 \cdot (-1) + 4 \cdot (-3) + 3 = -12$

2)$Xg = \begin{bmatrix} 3 & 3 & 1 & -3 \\ 3 & 2 & -3 & -4 \\ 0 & -1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 1 \\ -2 \\ 4 \\ -3 \end{bmatrix} = \begin{bmatrix} 3-6+4+9 \\ 3-4-12+12 \\ 0+3+3+0 \end{bmatrix} = \begin{bmatrix} 10 \\ -1 \\ -2 \end{bmatrix}$

3)$XY = \begin{bmatrix} 3 & 3 & 1 & -3 \\ 3 & 2 & -3 & -4 \\ 0 & -1 & -1 & 0 \end{bmatrix} \cdot \begin{bmatrix} 2 & -2 & 1 \\ -1 & -2 & 2 \\ 1 & -4 & -2 \\ 0 & 3 & -1 \end{bmatrix} = \begin{bmatrix} 3 \cdot 2 - 3 + 1 + 0 & -6-6-4-9 & 3+6-2+3 \\ 6-2-3+0 & -6-4+12-12 & 3+4+6+4 \\ 0+1-1+0 & 2+4 & -2+2 \end{bmatrix} =$
$\begin{bmatrix} 4 & -25 & 10 \\ 1 & -10 & 17 \\ 0 & 6 & 0 \end{bmatrix}$

4)
$\|X\|_2 = \sqrt{3^2 + 3^2 + 1^2 + (-3)^2 + 3^2 + 2^2 + (-3)^2 + (-4)^2 + 0 + 1 + 1 + 0} = \sqrt{68} = 8.24621125$
$\|\Upsilon\|_2 = \sqrt{2^2 + 2^2 + 1 + 1 + 2^2 + 2^2 + 1^2 + 4^2 + 2^2 + 0 + 3^2 + 1} = \sqrt{49} = 7$
$\|g\|_2 = \sqrt{1 + 2^2 + 4^2 + 3^2} = \sqrt{30} = 5.47722558$
$\|z\|_2 = \sqrt{5^2 + 1^2 + 3^2 + 1} = \sqrt{36} = 6$
$\|g^T z\|_2 = \sqrt{12^2} = 12$
$\|X \cdot g\|_2 = \sqrt{10^2 + 1 + 2^2} = \sqrt{105} = 10.2469508$
$\|X \cdot Y\|_2 = \sqrt{4^2 + 25^2 + 10^2 + 1 + 10^2 + 17^2 + 6^2} = \sqrt{1167} = 34.1613817$

2.1

First of all because A is symmetric we know:

$$A^T = A \Rightarrow \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & & \\ \vdots & & \ddots & \vdots \\ a_{n1} & & \cdots & a_{nn} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{12} & a_{22} & & \\ \vdots & & \ddots & \vdots \\ a_{1n} & & \cdots & a_{nn} \end{bmatrix}$$

Then we have to calculate the f() function. Let's start by calculating the $x^T \cdot A$ value.

$$f(x) = \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix} \cdot \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{12} & a_{22} & & \\ \vdots & & \ddots & \vdots \\ a_{1n} & & \cdots & a_{nn} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} + \begin{bmatrix} b_1 & b_2 & \cdots & b_n \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} =$$

$$\begin{bmatrix} x_1a_{11} + x_2a_{12} + \ldots + x_na_{1n} & x_1a_{12} + x_2a_{22} + \ldots + x_na_{2n} & \ldots & x_1a_{1n} + x_2a_{2n} + \ldots + x_na_{nn} \end{bmatrix} \cdot \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} +$$

$$[b_1x_1 + b_2x_2 + \ldots + b_nx_n] =$$

$$[(x_1a_{11} + x_2a_{12} + \ldots + x_na_{1n})x_1 + (x_1a_{12} + x_2a_{22} + \ldots + x_na_{2n})x_2 + \ldots + (x_1a_{1n} + x_2a_{2n} + \ldots + x_na_{nn})x_n] +$$

$$[b_1x_1 + b_2x_2 + \ldots + b_nx_n]$$

In general we conclude into 2 1-D arrays, which is actually a number.
To compute the final ∇f ()we need to find the derivatives for each $x_1...x_n$

$$\nabla f(x) = \begin{bmatrix} \frac{\partial((x_1a_{11}+x_2a_{12}+\ldots+x_na_{1n})x_1+(x_1a_{12}+x_2a_{22}+\ldots+x_na_{2n})x_2+\ldots+(x_1a_{1n}+x_2a_{2n}+\ldots+x_na_{nn})x_n+(b_1x_1+b_2x_2+\ldots+b_nx_n))}{\partial x_1} \\ \frac{\partial((x_1a_{11}+x_2a_{12}+\ldots+x_na_{1n})x_1+(x_1a_{12}+x_2a_{22}+\ldots+x_na_{2n})x_2+\ldots+(x_1a_{1n}+x_2a_{2n}+\ldots+x_na_{nn})x_n+(b_1x_1+b_2x_2+\ldots+b_nx_n))}{\partial x_2} \\ \vdots \\ \frac{\partial((x_1a_{11}+x_2a_{12}+\ldots+x_na_{1n})x_1+(x_1a_{12}+x_2a_{22}+\ldots+x_na_{2n})x_2+\ldots+(x_1a_{1n}+x_2a_{2n}+\ldots+x_na_{nn})x_n+(b_1x_1+b_2x_2+\ldots+b_nx_n))}{\partial x_n} \end{bmatrix} =$$

$$\begin{bmatrix} 2x_1a_{11} + x_2a_{12} + \ldots + x_na_{1n} + x_2a_{12} + \ldots + x_na_{1n} + b1 \\ x_1a_{12} + x_1a_{12} + 2x_2a_{22} + \ldots + x_na_{2n} + \ldots + x_na_{2n} + b2 \\ \vdots \\ x_1a_{1n} + x_2a_{2n} + \ldots + x_1a_{1n} + x_2a_{2n} + \ldots + x_na_{nn} + b_n \end{bmatrix}$$

Chain Rule

The produced function will have the form below:

$$c(x) = \sin(125x^3)$$

The derivative of the function will follow the chain rule:
$[f(g(x))]' = f'(g(x)) \cdot g'(x)$

$$c(x)' = \left[sin'(125x^3) \cdot (125x^3)\right] = cos(125x^3) \cdot 125 \cdot 3 \cdot x^2 = cos(125x^3) \cdot 375x^2$$

For the last exercise in order to compute the $\nabla_x^2 f(x)$ we just need to find the derivatives of each row of $\nabla f(x)$for each $x_1 \ldots x_n$

$$\nabla_x^2 f(x) = \begin{bmatrix} \frac{\partial(2x_1a_{11}+x_2a_{12}+...+x_na_{1n}+x_2a_{12}+...+x_na_{1n}+b1)}{\partial x_1} & \frac{\partial(2x_1a_{11}+...+x_na_{1n}+b1)}{\partial x_2} & \cdots & \frac{\partial(2x_1a_{11}+x_2a_{12}+...+x_na_{1n}+x_2a_{12}+...+x}{\partial x_n} \\ \frac{\partial(x_1a_{12}+x_1a_{12}+2x_2a_{22}+...+x_na_{2n}+...+x_na_{2n}+b2)}{\partial x_1} & \frac{\partial(x_1a_{12}+...+x_na_{2n}+b2)}{\partial x_2} & \cdots & \frac{\partial(x_1a_{12}+x_1a_{12}+2x_2a_{22}+...+x_na_{2n}+...+x}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial(x_1a_{1n}+x_2a_{2n}+...+x_1a_{1n}+x_2a_{2n}+...+x_na_{nn}+b_n)}{\partial x_1} & \frac{\partial(x_1a_{1n}+...+x_na_{nn}+b_n)}{\partial x_2} & \cdots & \frac{\partial(x_1a_{1n}+x_2a_{2n}+...+x_1a_{1n}+x_2a_{2n}+...+x}{\partial x_n} \end{bmatrix}$$

$$\nabla_x^2 f(x) = \begin{bmatrix} 2a_{11} & 2a_{12} & \dots & 2a_{1n} \\ 2a_{12} & 2a_{22} & \cdots & 2a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ 2a_{1n} & 2a_{2n} & \cdots & 2a_{nn} \end{bmatrix}$$

FINAL RESULT!!

3 Linear Regression

3.1 Derivation of the Ordinary Least Squares estimator for multiple regressors

For this section I had 2 approaches:

On the First one I followed the logic that was provided from the slides. To find the RSS we have to compute the sum of all the squares error ($\sum_{i=1}^{N} e_i^2$). If we analyze this a bit more we know that the error comes from the real_y_value-predicted_y_value. So the type becomes $\sum_{n=1}^{N}(y_n - y_n^-)^2$. Analyzing the prediction(estimation) more we have that

$$\sum_{n=1}^{N}(y_n - (b^T \cdot x_n))^2 = \sum_{n=1}^{N}(y_n - (b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np}))^2 \ (1)$$

After that we have to find the values of b that minimize the previous equation (1). In order to do this we have to find the derivative of (1) and equate it to zero.

$$\frac{\partial \sum_{n=1}^{N}(y_n - (b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np}))^2}{\partial x}$$

Because we have Sum we can pass inside the ∂ and we have:

$$\sum_{n=1}^{N} \begin{bmatrix} \frac{\partial((y_n-(b_1x_{n1}+b_2x_{n2}+\ldots+b_px_{np}))^2)}{\partial b_1} \\ \frac{\partial((y_n-(b_1x_{n1}+b_2x_{n2}+\ldots+b_px_{np}))^2)}{\partial b_2} \\ \vdots \\ \frac{\partial((y_n-(b_1x_{n1}+b_2x_{n2}+\ldots+b_px_{np}))^2)}{\partial b_p} \end{bmatrix} =$$

$$\sum_{n=1}^{N} \begin{bmatrix} 2(y_n - (b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np})) \cdot (-x_{n1}) \\ 2(y_n - (b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np})) \cdot (-x_{n2}) \\ \vdots \\ 2(y_n - (b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np})) \cdot (-x_{np}) \end{bmatrix} =$$

$$2\sum_{n=1}^{N} \begin{bmatrix} -x_{n1}y_n + x_{n1}(b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np}) \\ -x_{n2}y_n + x_{n2}(b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np}) \\ \vdots \\ -x_{np}y_n + x_{np}(b_1x_{n1} + b_2x_{n2} + \ldots + b_px_{np}) \end{bmatrix} =$$

$$2\sum_{n=1}^{N} \begin{bmatrix} -x_{n1}y_n + b_1x_{n1}^2 + b_2x_{n1}x_{n2} + \ldots + b_px_{n1}x_{np} \\ -x_{n2}y_n + b_1x_{n1}x_{n2} + b_2x_{n2}^2 + \ldots + b_px_{n2}x_{np} \\ \vdots \\ -x_{np}y_n + b_1x_{np}x_{n1} + b_2x_{np}x_{n2} + \ldots + b_px_{np}^2 \end{bmatrix}$$

Now that we have simplified out formula its time to equate it with 0.

$$2\sum_{n=1}^{N}\begin{bmatrix} -x_{n1}y_n + b_1x_{n1}^2 + b_2x_{n1}x_{n2} + \ldots + b_px_{n1}x_{np} \\ -x_{n2}y_n + b_1x_{n1}x_{n2} + b_2x_{n2}^2 + \ldots + b_px_{n2}x_{np} \\ \vdots \\ -x_{np}y_n + b_1x_{np}x_{n1} + b_2x_{np}x_{n2} + \ldots + b_px_{np}^2 \end{bmatrix} = 0$$

$$\sum_{n=1}^{N}\begin{bmatrix} -x_{n1}y_n + b_1x_{n1}^2 + b_2x_{n1}x_{n2} + \ldots + b_px_{n1}x_{np} \\ -x_{n2}y_n + b_1x_{n1}x_{n2} + b_2x_{n2}^2 + \ldots + b_px_{n2}x_{np} \\ \vdots \\ -x_{np}y_n + b_1x_{np}x_{n1} + b_2x_{np}x_{n2} + \ldots + b_px_{np}^2 \end{bmatrix} = 0$$

$$\sum_{n=1}^{N}\begin{bmatrix} b_1x_{n1}^2 + b_2x_{n1}x_{n2} + \ldots + b_px_{n1}x_{np} \\ b_1x_{n1}x_{n2} + b_2x_{n2}^2 + \ldots + b_px_{n2}x_{np} \\ \vdots \\ b_1x_{np}x_{n1} + b_2x_{np}x_{n2} + \ldots + b_px_{np}^2 \end{bmatrix} + \begin{bmatrix} -x_{n1}y_n \\ -x_{n2}y_n \\ \vdots \\ -x_{np}y_n \end{bmatrix} = 0$$

$$\sum_{n=1}^{N}\begin{bmatrix} b_1x_{n1}^2 + b_2x_{n1}x_{n2} + \ldots + b_px_{n1}x_{np} \\ b_1x_{n1}x_{n2} + b_2x_{n2}^2 + \ldots + b_px_{n2}x_{np} \\ \vdots \\ b_1x_{np}x_{n1} + b_2x_{np}x_{n2} + \ldots + b_px_{np}^2 \end{bmatrix} - \sum_{n=1}^{N}\begin{bmatrix} x_{n1}y_n \\ x_{n2}y_n \\ \vdots \\ x_{np}y_n \end{bmatrix} = 0$$

$$\sum_{n=1}^{N}\begin{bmatrix} b_1x_{n1}^2 + b_2x_{n1}x_{n2} + \ldots + b_px_{n1}x_{np} \\ b_1x_{n1}x_{n2} + b_2x_{n2}^2 + \ldots + b_px_{n2}x_{np} \\ \vdots \\ b_1x_{np}x_{n1} + b_2x_{np}x_{n2} + \ldots + b_px_{np}^2 \end{bmatrix} = \sum_{n=1}^{N}\begin{bmatrix} x_{n1}y_n \\ x_{n2}y_n \\ \vdots \\ x_{np}y_n \end{bmatrix}$$

We observe that the left part is actual a multiplication of the 2 arrays below, so we separate them.

$$\sum_{n=1}^{N}\begin{bmatrix} x_{n1}^2 + x_{n1}x_{n2} + \ldots + x_{n1}x_{np} \\ x_{n1}x_{n2} + x_{n2}^2 + \ldots + x_{n2}x_{np} \\ \vdots \\ x_{np}x_{n1} + x_{np}x_{n2} + \ldots + x_{np}^2 \end{bmatrix}\begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_p \end{bmatrix} = \sum_{n=1}^{N}\begin{bmatrix} x_{n1} \\ x_{n2} \\ \vdots \\ x_{np} \end{bmatrix} y_n$$

$$\sum_{n=1}^{N}\begin{bmatrix} x_{n1}^2 + x_{n1}x_{n2} + \ldots + x_{n1}x_{np} \\ x_{n1}x_{n2} + x_{n2}^2 + \ldots + x_{n2}x_{np} \\ \vdots \\ x_{np}x_{n1} + x_{np}x_{n2} + \ldots + x_{np}^2 \end{bmatrix} b = \sum_{n=1}^{N} x_ny_n$$

We also can see that this array $\begin{bmatrix} x_{n1}^2 + x_{n1}x_{n2} + \ldots + x_{n1}x_{np} \\ x_{n1}x_{n2} + x_{n2}^2 + \ldots + x_{n2}x_{np} \\ \vdots \\ x_{np}x_{n1} + x_{np}x_{n2} + \ldots + x_{np}^2 \end{bmatrix}$ is a symmetric array, product of the multiplication of the $x_n \cdot x_n^T = \begin{bmatrix} x_{n1} \\ x_{n2} \\ \vdots \\ x_{np} \end{bmatrix}\begin{bmatrix} x_{n1} & x_{n2} & \cdots & x_{np} \end{bmatrix}$ so we replace it in the equation. The final form of the equation is:

$$\sum_{n=1}^{N}(x^T \cdot x)b = \sum_{n=1}^{N} x_ny_n$$

Then following the slides by using matrix notation we have conclude in:

$$X^TXb = X^TY$$

We multiply the above expression with $(X^TX)^{-1}$ and we get the least squares estimator for b:

$$b = (X^TX)^{-1}X^TY$$

On the **second approach** I followed a logic that found online.

So the general form that our values has are:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_N \end{bmatrix} = \begin{bmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{N1} & x_{N2} & \cdots & x_{Np} \end{bmatrix} \cdot \begin{bmatrix} b_1 \\ b_2 \\ \vdots \\ b_p \end{bmatrix} + \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix}$$

Where the $e_i$are the errors (difference) that our estimation has from the real value.

In order to calculate the $\sum_{i=1}^{N} e_i^2$we have to make the multiplication of

$$e^T \cdot e = \begin{bmatrix} e_1 & e_2 & \cdots & e_N \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ \vdots \\ e_N \end{bmatrix} = \sum_{i=1}^{N} e_i^2$$

In order to minimize the b we have to minimize the $e^Te(1)$. From the definition we know that $e = y - Xb$. By replacing the e in the (1) we get:

$$(y - Xb)^T \cdot (y - Xb) = (y^T - b^TX^T)(y - Xb) =$$

$$y^Ty - b^TX^Ty - y^TXb + b^TX^TXb \text{ (2)}$$

We know that $b^TX^Ty = (b^TX^Ty)^T = y^TXb$ are equals, because the terms are 1x1 dimension the transposition of it will be the same as the original one. So the (2) will become:

$$y^Ty - 2b^TX^Ty + b^TX^TXb$$

Having said that it's time to set the derivative equal to zero, with respect to b.

$$\frac{\partial(y^Ty - 2b^TX^Ty + b^TX^TXb)}{\partial b} \text{ (3)}$$

By following these 2 rules:

$$\frac{\partial a^Tb}{\partial b} = \frac{\partial b^Ta}{\partial b} = a$$

when a and b are Kx1 vectors

$$\frac{\partial b^TAb}{\partial b} = 2Ab = 2b^TA$$

when A is a symmetric matrix.

In our case we have the same form b^TX^TXb because X^TX is actually a symmetric matrix as A.

So by using the above rules the (3) becomes (and also equate it with 0):

$$-2X^Ty + 2X^TXb = 0$$

$$X^TXb = X^Ty \text{ (4)}$$

And in the end we follow the same step as the previous approach which is to multiply the expression (4) with $(X^TX)^{-1}$and we get:

$$b = (X^TX)^{-1}X^TY$$