classmate
Date
Page

ML Assignment - 1   Ritisha-Gupta, MT22056

Ques 3) a) What is psuedo inverse of a matrix?

Sol^n- If A is a square matrix then its inverse exists and A is known as an invertible matrix

If A is not an invertible matrix, then we find the Moore Penrose psuedo inverse. However, Moore Penrose psuedo inverse is defined for invertible matrix also but in that case both are equal.

Let $A \in \mathbb{R}^{m\times n}$. & let $B \in \mathbb{R}^{n\times m}$ be the psuedo inverse of A. if it satisfies all four conditions :

1) $ABA = A$ ← B is generalized inverse of A.
2) $BAB = B$ ← A is generalized inverse of B.
3) $(AB)^T = AB$ ← AB is symmetric.
4) $(BA)^T = BA$ ← BA is symmetric.

The psuedo inverse of a matrix $A \in \mathbb{R}^{m\times n}$ always exists & is unique. Denoted as (A^+)

The symmetric form of def^n implies $B = A^+$ & $A = B^+$ & thus.
$A = (A^+)^+$.

b)(i) Undetermined System of Equation.
- In such system we have fewer equations than no of variables.
- It cannot have unique sol^n.
- In this case either ∞ many or no sol^n's (consistent).

$\cancel{\boxed{x = (\theta^T\theta)^{-1}\theta^T y}}$ where $\theta^T(\theta^T\theta)^{-1}$ = psuedo inverse.

$\boxed{x = \theta^T(\theta\theta^T)^{-1} y}$

(ii) Overdetermined.

- more equation than no. of variables
- also ∞ many or no solⁿ.
- may have a unique solⁿ.

$$\boxed{x = (O^T O)^{-1} O^T y}$$

where $\underbrace{\quad}_{\text{psuedo inverse.}}$

b) $x_1 + 3x_2 = 17$

$5x_1 + 7x_2 = 19$ $\quad A\vec{X} = \vec{B}$ {Sys of linear equation}

$11x_1 + 13x_2 = 23$

$$A = \begin{bmatrix} 1 & 3 \\ 5 & 7 \\ 11 & 13 \end{bmatrix}_{m \times n} \quad \vec{X} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}_{n \times 1} \quad \vec{B} = \begin{bmatrix} 17 \\ 19 \\ 23 \end{bmatrix}_{m \times 1}$$

(coeff matrix)

$\vec{x}, \vec{B}$ = vector

A = matrix

Here $m > n$ ($m = 3$ & $n = 1$) so ~~the~~ ~~eq~~ⁿ we have over-determined system of linear eq. because we have more ~~not~~ no. of equations than variables.

$AX = B$

Multiply both sides with A^T.

$$A^T A \vec{X} = A^T \vec{B}$$

Multiply both sides with $(A^T A)^{-1}$

$$(A^T A)^{-1} (A^T A) \vec{X} = (A^T A)^{-1} A^T \vec{B}$$

We know that ~~A^T~~ $AA^{-1} = I$ (Identity matrix)

so, $I \cdot \vec{X} = (A^T A)^{-1} A^T \vec{B}$

$$\boxed{\vec{X} = (A^T A)^{-1} A^T \vec{B}} \quad \because A^+ \text{ (psuedo inv)} = (A^T A)^{-1} A^T$$

$$\boxed{\vec{X} = A^+ \vec{B}}$$

First we'll find psuedo inverse. $A^+ = (A^T A)^{-1} A^T$

$$A^T A = \begin{bmatrix} 1 & 5 & 11 \\ 3 & 7 & 13 \end{bmatrix} \begin{bmatrix} 1 & 3 \\ 5 & 7 \\ 11 & 13 \end{bmatrix} = \begin{bmatrix} 147 & 181 \\ 181 & 227 \end{bmatrix}$$

$$(A^TA)^{-1} = \begin{bmatrix} +0.3733 & -0.2976 \\ -0.2976 & 0.2417 \end{bmatrix} = \frac{adj(A^TA)}{|A^TA|}$$

$$(A^TA)^{-1}A^T = \begin{bmatrix} 0.3733 & -0.2976 \\ -0.2976 & 0.2417 \end{bmatrix}_{2\times2} \begin{bmatrix} 1 & 5 & 11 \\ 3 & 7 & 13 \end{bmatrix}_{2\times3}.$$

$$A^+ = (A^TA)^{-1}A^T = \begin{bmatrix} -0.5195 & -0.2167 & 0.2375 \\ 0.4275 & 0.2039 & -0.1315 \end{bmatrix}$$

(psuedo -Inverse.

$$\vec{X} = A^+\vec{B}$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -0.5195 & -0.2167 & 0.2375 \\ 0.4275 & 0.2039 & -0.1315 \end{bmatrix}_{2\times3} \begin{bmatrix} 17 \\ 19 \\ 23 \end{bmatrix}_{3\times1}$$

$$\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} -7.4863 \\ 8.1171 \end{bmatrix}. \quad \underline{\text{Ans}}$$

$\boxed{x_1 = -7.4863}$ $\boxed{x_2 = 8.1171}$

Ques 3) (i) Gradient descent is an iterative method to find the local minima / maxima of a function. It is used to find the best parameters that minimizes the model's cost function. Normal eq^n is a closed form solution for Linear Regression. Using this Normal eq^n, we can directly solve for optimal value of parameter θ in just one step. Optimal parameter can be obtain by just formula.

$$\boxed{\theta = (A^TA)^{-1}A^T\vec{y}}$$

where θ = hypothesis parameter.
A = Input matrix
Y = output

(ii) We prefer iterative methods like Gradient descent rather than normal eqn because in closed form solution of 'N' independent variables requires to find inverse of matrix which takes app $O(n^3)$. Its computational complexicity is very high. whereas in gradient desce-nt, it has a time complexicity that is linear i.e. $O(n)$.

Normal eqn is very easy for univariate variables but not in case of multivariate variables.

d) $\text{Tanh } x = \dfrac{e^x - e^{-x}}{e^x + e^{-x}} = g(x)$.

We know that $h_\theta(x)$ = hypothesis.

$$\underbrace{h_\theta(x) = g(\theta^T x)}_{} = \frac{e^{\theta^T x} - e^{-\theta^T x}}{e^{\theta^T x} + e^{-\theta^T x}}.$$

For logistic regression.

Where θ = $\boxed{\text{parameters}}$, x = i/p features

$$\text{Derivative of } \tanh(z) = \frac{\partial}{\partial z}(g(z)) = \frac{\partial}{\partial z}\left(\frac{e^z - e^{-z}}{e^z + e^{-z}}\right).$$

$$\frac{(e^z + e^{-z})(e^z + e^{-z}) - (e^z - e^{-z})(e^z - e^{-z})}{(e^z + e^{-z})^2}$$

$$1 - \frac{(e^z - e^{-z})^2}{(e^z + e^{-z})^2}$$

$$1 - \left(\frac{e^z - e^{-z}}{e^z + e^{-z}}\right)^2 = 1 - \tanh(z)^2 = 1 - (g(z))^2 \quad \text{— (1)}$$

In logistic regression $P(y=1 | x ; \theta) = h_\theta(x) = \dfrac{e^{\theta^T x} - e^{-\theta^T x}}{e^{\theta^T x} + e^{-\theta^T x}}$.

then $P(y=0 | x ; \theta) = 1 - P(y=1 | x ; \theta)$.

$$= 1 - \left(\frac{e^{\theta^T x} - e^{-\theta^T x}}{e^{\theta^T x} + e^{-\theta^T x}}\right)$$

Combining the two equators -

$$P(y|x;\theta) = (h_\theta(x))^y (1-h_\theta(x))^{1-y}.$$

probability of y given x parameterized by θ

We know that likelihood is ~~defined / are~~ represented as $L(\theta)$.

maximising $L(\theta)$ so that we get best fit model.

$$\max (L(\theta)) = \max \{ P = (\vec{y} | x ; \theta).$$

$$= \prod_{i=1}^{m} P(y^i / x^i ; \theta). \quad m = \text{no of i/p features}$$

$$L(\theta) = \prod_{i=1}^{m} h_\theta(x^i)^{y^i} - (1-h_\theta(x_i))^{1-y^i}$$

maximising $L(\theta)$ = ~~maximising $L(\theta)$~~

Taking log on both sides to make our calculation easy. & convert into summation.

$$\sum_{i=1}^{m} \log (h_\theta(x^i))^{y^i} + \sum_{i=1}^{m} \log(1-h_\theta(x^i))^{1-y^i}$$

here $\theta = \theta + \frac{\partial}{\partial\theta} L(\theta)$. $\rightarrow$ maximising gradient-descent.

$$\frac{\partial}{\partial\theta_j} = (y^j \log(h_\theta(x^j)) + (1-y^j) \log(1-h_\theta(x^j))$$

$$= \frac{y^j}{h_\theta(x^j)} \frac{\partial}{\partial\theta_j} h_\theta(x^j) - \frac{1-y^j}{1-h_\theta(x^j)} \frac{\partial}{\partial\theta_j} (h_\theta(x^j))$$

$$\left(\frac{y^j}{h_\theta(x^j)} - \frac{1-y^j}{1-h_\theta(x^j)}\right) \frac{\partial}{\partial\theta} h_\theta(x^j)$$

$$\left(\frac{y^j}{h_\theta(x^j)} - \frac{1-y^j}{1-h_\theta(x^j)}\right) [1-h_\theta(x^j)^2] \times x^j_k$$

$$\left(\frac{y^j - y^j h_\theta(x^j) - h_\theta(x^j) + h_\theta(x^j) y^j}{h_\theta(x^j) \cdot (1-h_\theta(x^j))}\right) (1-h_\theta(x^j)^2) \cdot x^j_k$$

$$\frac{y^j - h_\theta(x^j)}{h_\theta(x^j) \cdot (1-h_\theta(x^j))} \times (1+h_\theta(x^j))\,(1-h_\theta(x^j)) \; \vdots \; x_k.$$

$$= \frac{y^j - h_\theta(x^j)}{h_\theta(x^j)} \times (1+h_\theta(x^j)) \times x_k^j.$$

Thus the updated rule for logistic reg =

$$\left[\theta_k = \theta_k + \alpha \left(\frac{y^j - h_\theta(x^j)}{h_\theta(x^j)} \cdot x_k^j \times (1+h_\theta(x^j)) \right) \right]$$

→ using tanh function as decⁿ boundary.