Q2) a)

Pseudo-Inverse (A^+)

Let there be a matrix A.

The pseudo-inverse (A^+) matrix is a generalisation of the m[...] inverse when the matrix might not be invertible ([...]

If A is invertible, $A^+ = A^{-1}$ else its also defined even when A is not invertible.

i) For underdetermined system of eqns,

$[A]_{n\times m}$: $n < m$ and rank$(A) = n$

$$A^+ = A^T (AA^T)^{-1}$$

ii) For overdetermined system of equations,

$[A]_{n\times m}$: $n > m$ and rank$(A) = m$

$$A^+ = (A^TA)^{-1}A^T$$

Q2(b) Solve:

$$x_1 + 3x_2 = 17$$
$$5x_1 + 7x_2 = 19$$
$$11x_1 + 13x_2 = 23$$

Converting these eqns to matrix form:

$$\underbrace{\begin{bmatrix} 1 & 3 \\ 5 & 7 \\ 11 & 13 \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} x_1 \\ x_2 \end{bmatrix}}_{x} = \underbrace{\begin{bmatrix} 17 \\ 19 \\ 23 \end{bmatrix}}_{b}$$

$Ax = B$

To calculate x_1, x_2 we need to find A^{-1} as $\underline{x = A^{-1} b}$.

As it is over determined system of eqns because $A_{3\times 2}$;
no. of rows > no. of col. and rank$(A) = 2$
Hence the pseudo-inverse of matrix is

$A^{+} = (A^T A^{-1}) B$ where

$A \rightarrow$ original matrix
$A^{+} \rightarrow$ Pseudo Inverse
$A^T \rightarrow$ Transpose of A

Now formula for x would be :

$$\boxed{x = (A^T A)^{-1} A^T \cdot b}$$

$$A^T A = \begin{bmatrix} 1 & 5 & 11 \\ 3 & 7 & 13 \end{bmatrix}_{2\times3} \begin{bmatrix} 1 & 3 \\ 5 & 7 \\ 11 & 13 \end{bmatrix}_{3\times2} = \begin{bmatrix} 1+25+121 & 3+35+143 \\ 3+35+143 & 9+49+169 \end{bmatrix}$$

$$= \begin{bmatrix} 147 & 181 \\ 181 & 227 \end{bmatrix}$$

Using the property $X^{-1} = \frac{Adj(X)}{|X|}$ to calculate $(A^T A)^{-1}$.

$$|A^T A| = (147 \times 227) - (181 \times 181) = 608$$

Now $Adj(A^T A) = \begin{bmatrix} 227 & -181 \\ -181 & 147 \end{bmatrix}$

$$(A^T A)^{-1} = \frac{1}{608} \begin{bmatrix} 227 & -181 \\ -181 & 147 \end{bmatrix}$$

Now, we use these values to calculate A^+

$$A^+ = (A^T A)^{-1} A^T$$

$$= \frac{1}{608} \begin{bmatrix} 227 & -181 \\ -181 & 147 \end{bmatrix}_{2\times2} \begin{bmatrix} 1 & 5 & 11 \\ 3 & 7 & 13 \end{bmatrix}_{2\times3}$$

$$= \frac{1}{608}\begin{bmatrix} 227\times1-181\times3 & 227\times5-181\times7 & 227\times11-181\times13 \\ -181\times1+147\times3 & -181\times5+147\times7 & -181\times11+147\times13 \end{bmatrix}$$

$$= \frac{1}{608}\begin{bmatrix} -316 & -132 & 144 \\ 260 & 124 & -80 \end{bmatrix}$$

Putting this value of A^+ in eqn ①, we get

$$x = \frac{1}{608}\begin{bmatrix} -316 & -132 & 144 \\ 260 & 124 & -80 \end{bmatrix}\begin{bmatrix} 17 \\ 19 \\ 23 \end{bmatrix}$$

$$= \frac{1}{608}\begin{bmatrix} -316\times17 - 132\times19 + 144\times23 \\ 260\times17 + 124\times19 - 80\times23 \end{bmatrix}$$

$$= \frac{1}{608}\begin{bmatrix} -4568 \\ 4936 \end{bmatrix}$$

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \frac{1}{608}\begin{bmatrix} -4568 \\ 4936 \end{bmatrix} = \begin{bmatrix} -7.513 \\ 8.118 \end{bmatrix}$$

Q2(c)

(i) Closed form solⁿ (using normal eqns) to solve a linear regression problem is :-

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

where :- $(X^T X)^{-1} X^T$: pseudo-inverse of matrix X ;

y : target value

θ : learning parameter

(ii) Gradient descend like iterative methods are used over closed form solutions to reduce computational time & space complexity to solve linear regression problem for large datasets (having Lakh⁺ or million⁺ data points)

This is because closed form soln might run over the entire dataset to calculate just 1 learning parameter, which has to be repeated for no. of times eql. to no. of features for every learning parameter.

Space complexity will also be huge as we would require to solve the matrix $(m \times n)$, where $m, n \approx 10^5$ or more but gradient descend iterates over 1 data point at a time & learns gradually.

Q2(d)

MAE:

Mean absolute error is the average of the absolute values of diffrence of actual vs predicted values of the target

$$\boxed{MAE = \sum_{i=1}^{n} \frac{|\hat{y}_i - y_i|}{n}}$$

where $\hat{y}_i$ = prediction , $\hat{y}_i$: true value

i : i^{th} data point

n : total no of data points

MSE: the average of the squared values of diffrence of actual vs predicted values of the target for the entire test dataset

$$\boxed{MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2}$$

RMSE: the quadratic mean of the differences between each of the actual vs predicted values of the target

$$\boxed{RMSD = \sqrt{\frac{\sum_{i=1}^{N}(\hat{y}_i - y_i)^2}{N}}}$$

Q(3)(C)

$$b_{new} = b_{old} - \eta \frac{\partial L(b)}{\partial b}$$ → gradient descend update eq^n:

where $L(b) = (y^* - h_b(u))^2$,

η is learning rate

and $u = xb$

$$\frac{\partial L(b)}{\partial b} = 2(y^* - h_b(u)) \frac{\partial}{\partial b}\left[y^* - h_b(u)\right]$$

$$= 2(y^* - h_b(u)) \, -\frac{\partial}{\partial b}(h_b(u)) \cdot \frac{\partial(xb)}{\partial b}$$

$$= 2(y^* - h_b(u))\left[-\frac{\partial}{\partial b}\left(h_b(u)\right)\right] \cdot x$$

given $h_b(u) = \tanh(u)$

hence $\frac{\partial}{\partial b}\left(h_b(u)\right) = \frac{\partial}{\partial b}\tanh(u)$

$$= 1 - \tanh^2(u)$$

$$= 1 - \left(h_b(u)\right)^2$$

$$= \left(1 - h_b(u)\right)\left(1 + h_b(u)\right)$$

$$\boxed{\text{so, } \frac{\partial}{\partial b}\left(L(b)\right) = 2\left(y^* - h_b(u)\right)\left(1 - h_b(u)\right)\left(1 + h_b(u)\right) \cdot x}$$