Lab10 - MultiLR and LogReg

LAB 10 : MultiLR and LogReg (ED5340)
(ED21S014)

DATE 09/10/21

Q1) Using the notation for multivariate Linear Regression, delineate the steps in 'Optimisation' of the cost function, duly explaining the gradient term.

→ Multivariate implies 'n' features are in dataset.
Ground truth data → Input feature/output $(x_1, x_2 \ldots x_n, y)$ are known from dataset.

→ Let each feature & output y have 'm' training samples.
Then we have the set

$(x_1^{(1)}, x_2^{(1)}, \ldots x_n^{(1)}, y^{(1)})$
$(x_1^{(2)}, x_2^{(2)}, \ldots x_n^{(2)}, y^{(2)}) \ldots (x_1^{(m)}, x_2^{(m)}, \ldots, x_n^{(m)}, y^{(m)})$

→ Goal is to fit a hyperplane approximately to the dataset.

→ Let hypothesis function be $h_w(x) = w_0 + w_1 x_1 + w_2 x_2 = w^T x$

→ Goal is to determine weights (w_0, w_1, w_2)
where $w^T = [w_0 \; w_1 \; w_2]$ $x = [x_0 \; x_1 \; x_2]^T$ where $x_0 = 1$

→ Let $i \in (1, m)$ (training samples), $j = \in (1, m)$ (features)

→ Using mean squared error cost function

$$J(w) = \sum_{i=1}^{m} \frac{1}{2m} \left(w_0 x_j^{(i)} + w_1 x_j^{(i)} + w_2 x_j^{(i)} - y^{(i)} \right)^2$$

→ To get critical values for weights we minimise cost function $J(w)$ using gradient descent optimisation

→ We find $\nabla J(w_0, w_1, w_2) = \left(\frac{\partial J}{\partial w_0}, \frac{\partial J}{\partial w_1}, \frac{\partial J}{\partial w_2} \right)$

→ with initial weights w_0, w_1, w_2, we compute $-\nabla J$ at initial weights

→ Update weights $w_{k+1}^* = w_k^* - \alpha_k \nabla J$ (α_k = learning rate = constant)

→ check if $\text{norm}(\nabla J) < \epsilon$ small value.

→ If yes, final/critical values of weights are obtained else continue with iteration

→ The gradient of a function is a vector that points in the direction where the value of function on a curve increases continuously.

→ Hence, for optimisation, we use the $-\nabla J$ direction to reach the optimum value of function.

→ We can use line search to determine α_k values during gradient descent to reach the optimum point faster.

Let $J = (w_0x_0 + w_1x_1 + w_2x_2 - y)^2$

$\therefore \frac{\partial J}{\partial w_0} = 2 \times (w_0x_0 + w_1x_1 + w_2x_2 - y) \times x_0$ where $x_0 = 1$

$\therefore \frac{\partial J}{\partial w_1} = 2 \times (w_0x_0 + w_1x_1 + w_2x_2 - y) \times x_1$

$\therefore \frac{\partial J}{\partial w_2} = 2 \times (w_0x_0 + w_1x_1 + w_2x_2 - y) \times x_2$

$\therefore$ gradients of cost function $J(w)$ w.r.t w_0, w_1, w_2 are:

i) $\frac{\partial J}{\partial w_0} = \frac{1}{m} \sum_{i=1}^{m} (w_0x_0^{(i)} + w_1x_1^{(i)} + w_2x_2^{(i)} - y^{(i)})\, x_0^{(i)}$

ii) $\frac{\partial J}{\partial w_1} = \frac{1}{m} \sum_{i=1}^{m} (w_0x_0^{(i)} + w_1x_1^{(i)} + w_2x_2^{(i)} - y^{(i)})\, x_1^{(i)}$

iii) $\frac{\partial J}{\partial w_2} = \frac{1}{m} \sum_{i=1}^{m} (w_0x_0^{(i)} + w_1x_1^{(i)} + w_2x_2^{(i)} - y^{(i)})\, x_2^{(i)}$

In general,

$$\frac{\partial J}{\partial w_j} = \frac{1}{m} \sum_{i=1}^{m} (w_0x_0^{(i)} + w_1x_1^{(i)} + w_2x_2^{(i)} - y^{(i)})\, x_j^{(i)}$$

where $w_j^{k+1} = w_j^{k} - \alpha_k \frac{\partial J}{\partial w_j}$

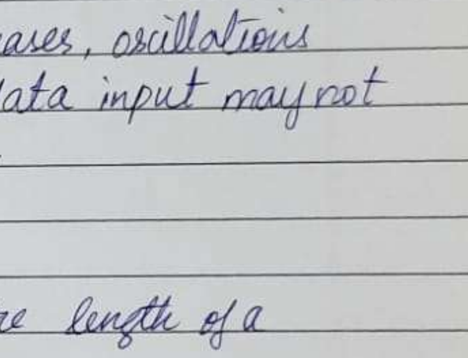
k = iteration step

Q2) Using a suitable example, explain use of polynomial regression.

→ Polynomial regression is a nonlinear regression that gives relationship between a dependent & independent variable where the two are related by n^{th} degree polynomial.
→ The best fit line for given data set is determined by the n^{th} degree of polynomial regression equation.
→ The best fit line is sensitive to outliers, hence dataset needs to be processed to handle outliers.
→ As degree of best fit polynomial increases, oscillations are seen in fitting pattern, new data input may not be result in accurate prediction.

Example:
→ To determine relationship between the length of a bluegill fish and its age.
→ It is well known that with age, length of this fish increases, but polynomial regression helps us to determine the non linear or n^{th} degree relationship between its length & age.

→ Response (y) : length of fish , predictor (x) : age of fish
→ A sample data set shows increase in y with x.
By visualising data set, we can approximate that y is related to x in 2^{nd} degree.



$\therefore h_w(x) = w_0 + w_1 x + w_2 x^2$
is hypothesis function.

→ A suitable cost function can be choosen & optimised to determine the weights w_0, w_1, w_2.