Week - 3.

Page No.
Date

Normal equation based Approach

⇒ Using linear algebra. All this problem of selecting 'α', how many iteration and other

Advantages of NE over Linear Regression

→ Single step
→ No 'α' selection

Disadvantages

→ when n becomes big more computational power required.

Derivation

$h_\theta(x) = \theta^T x$ in vectorize format.

Lets put x in different format

$$x^i = \begin{bmatrix} 1 \\ x_1^i \\ \vdots \\ x_n^i \end{bmatrix}_{(n+1)*1}$$

$$X = \begin{bmatrix} x_0^1 & x_1^1 & x_2^1 & \cdots \\ x_0^2 & x_1^2 & \cdots & \cdots \\ & & & \end{bmatrix} \equiv \begin{bmatrix} — (x^1)^T — \\ — (x^2)^T — \\ \vdots \\ — x^m — \end{bmatrix}$$

$$\dim(X) = m \times (n+1)$$

Page No.
Date

$$y = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^m \end{bmatrix} \qquad \dim(y) = m \times 1$$

$$y^1 = \theta_0 x_0^1 + \theta_1 x_1^1 + \dots$$

$$y^1 = \theta^T x^1 = x^T \theta$$

$$\left.\begin{aligned} y^1 &= (x^1)^T \theta \\ y^2 &= (x^2)^T \theta \\ &\vdots \\ y^m &= (x^m)^T \theta \end{aligned}\right\} \Rightarrow y = X\theta = Xw$$

$$w = \theta$$

$$\dim(\theta) = (n+1) \times 1$$

$$h_\theta(x) = \theta X \theta = Xw$$

$$\text{error} = e = Xw - y \qquad \left(\overset{\text{Linear}}{\text{Error}} \text{ in each training eg}\right)$$

$$E = e^T e \qquad \text{(Cummulative error)}$$

Also known as MMSE (Minized Mean Squared error.

$$E = (Xw - y)^T (Xw - y) \equiv (Xw)^T - y^T)(Xw - y)$$

$$(n+1 \times 1)^T \times ((n+1) \times 1) = (1 \times 1) = \text{Scalar value.}$$

$(A+B)^T = A^T + B^T$ Property of matrices.

$$E = (Xw)^T (Xw) - (Xw)^T (y) - y^T (Xw) + y^T y.$$

~~$(Xw)^T (y) = ((Xw) y)^T$~~

$y^T (Xw) = ((Xw)^T y)^T$ (Property of Transpose)

Page No.
Date

Now look at dim of matrices.

$$y^T(xw) = ((xw)^T y)$$

$\dim = (xw) = m \times 1$

$\dim y = m \times 1$

$\dim((xw)^T y) = (1 \times m) \times (m \times 1)$

$= 1 \times 1$

Transpose of scalar is equal to original

$x^T = x$ (where x is scalar)

$$E = (xw)^T(xw) - 2(xw)^T y + y^T y$$

$$E = w^T x^T x w - 2(xw)^T y + y^T y$$

$$E = w^T x^T x w - 2 w^T X^T y + y^T y.$$

$$\frac{\partial w^T}{\partial w} = ?$$

$$\frac{\partial 2 w^T X^T y}{\partial w} = 2\left[\begin{bmatrix} x_0^1 & x_1^1 & x_2^1 & x_3^1 & \cdots \\ x_0^2 & x_1^2 & x_2^2 & \cdots \end{bmatrix}\begin{bmatrix} w_0 \\ w_1 \\ w_2 \\ \vdots \end{bmatrix}\right]^T \begin{bmatrix} y \end{bmatrix}$$

$$= \begin{pmatrix} x_0^1 w_0 + x_1^1 w_1 + x_2^1 w_2 \cdots \\ x_0^2 \theta_0 + x_1^2 \theta_1 + \cdots \end{pmatrix}\begin{pmatrix} y_1^1 \\ y_2^2 \\ y^3 \\ \vdots \end{pmatrix}$$

$= x_0^1 \theta_0$

$$\frac{\partial}{\partial w}\left[(x_0^1 w_0 + x_1^1 w_1 + \cdots) y_1 + \cdots + (x_0^m w_0 + \cdots) y^m\right]$$

$$= \frac{\partial}{\partial w}$$

$$P = 2 \sum_{k=1}^{m} y_k^k (x_0^r w_0 + x_1^r w_1 - \cdots)$$

$$P = 2 \sum_{r=1}^{m} y^r \sum_{c=0}^{n} x_c^r w_c$$

$$\frac{\partial P}{\partial w_0} = 2(x_0^1 y_1 + x_0^2 y_2 \cdots)$$

$$\frac{\partial P}{\partial w_1} = 2(x_1^1 y_1 + x_1^2 y_2 \cdots)$$

$$\frac{\partial P}{\partial w_n} = 2(x_n^1 y_1 + x_n^2 y_2 \cdots)$$

$$\frac{\partial 2 w^T x^T y}{\partial w} = 2 x^T y$$

$$P_2 = w^T x^T x w$$

$$= (w_0 \; w_1 \; w_2 \cdots w_n) \begin{pmatrix} x_0^1 & x_0^2 & x_0^3 & \cdots \\ x_0^2 & & & \\ x_2^1 & & & \\ \vdots & & & \end{pmatrix}$$

$$\begin{pmatrix} x_0^1 & x_1^1 & x_2^1 & \cdots \\ x_0^2 & & & \\ x_0^3 & & & \\ \vdots & & & \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ \vdots \\ w_n \end{pmatrix}$$

$$= (w_0 \; w_1 \; w_2 \cdots w_n) \begin{pmatrix} (x_0^1)^2 w_0 + \cdots & & (x_n^1)^2 w_n \\ \vdots & & \\ (x_0^n)^2 w_0 \cdots & & (x_n^n)^2 w_n \end{pmatrix}$$

$$P2 = ((x_0^1)^2 w_0 + \dots (x_n^1)^2 w_n) + \dots$$
$$(w_n (x_n^n)^2 w_0 \dots (x_n^n)^2 w_n)$$

$$\frac{\partial P2}{\partial w_0} = 2(x_1^1)^2 + 2 w_0 (x_{02}^1)^2 + \dots 2 o_n x_n^1$$

$$\frac{\partial P2}{\partial w} = 2X^T X w$$

$$\frac{\partial E}{\partial w} = 2X^T X w - 2X^T y$$

Now for minimum error derivative should be zero at that point

$$\frac{\partial E}{\partial w} = 0 \qquad \frac{\partial y^T y}{\partial w} = 0$$ As there is no 'w' in it

$$X^T X w = X^T y$$

$$w = (X^T X)^{-1} X^T y$$

As Dimension increases time taken, or no. of computation required to process to find inverse is increased so for high Dimension (Generally > 2048) it takes high amount of time so we prefer Regression Model there.

Page No.
Date

When normal equation is preferred

~~when dimension < 1024 to~~

dim(w) < 1024

NE is preffered

Clustering

Unsupervised Learning

In Supervised learning we are given a label associated with each data point, which helps in Classification of data.

In unsupervised we don't have any labels associated with it and tasked to find some structure

One such approach could be to divide the given data into various distinct cluster based on some properties

Application

→ Use of to find pattern in stars

→ Dividing Customer base into segment on some criteria

Extra

Can we labeled data for clustering?

The thing is that we can use whatever we want, to use, what matters is results. & results for labelled data is not as good as that of some Algorithm for labelled data such as regression. They perform better.

Page No.
Date

Why?
Because Clustering tries to find relation b/w data point where as in labelled data we want relation b/w data point and its label. So they perform poorly as they fail to do so.

for eg. if some person visits some ~~webs~~ e-commerce

label when ~~the~~ buy something

Classifier tries to predict wheather he will buy this time or not (or what he will buy)
whereas Clustering may try to cluster it type of user with its parameter based ~~on them~~ data points

K-Means Clustering

This is an iterative algorithm which perform algorithm ~~which~~ two step in each.

first: randomly takes k - data point as center
~~Second~~ Then
for every iteration, colour the points based on its nearest centroid
Then move centroid to center of all colored points
Then repeat until not a single ~~centroid~~ point change its color.

Page No.
Date

Can k-means only segment well separated data?

Even where it is not seen to segment data. k-means clustering separate our data points. This use heavily on market segmentation

k-means clustering always give non overlapping k-segments

In linear Regression we have cost function to reduce. In Clustering we take similar approach to minimize the avg. distance of all the cluster centroids with the datapoints in that cluster

Formally let

$c^{(i)}$ = index of cluster to which data point $x^{(i)}$ is assigned

μ_k = cluster centroid

$\mu_{c^{(i)}}$ = centroid of cluster to which $x^{(i)}$ is assigned

$$J = \frac{1}{m} \sum_{i}^{m} \| x^{(i)} - \mu_{c^{(i)}} \|^2$$

And our objective is to minimize this cost function

This cost function is also called distortion.