"Polynomial Regression"

Polynomial Regression Overview :

given data $X \in R^{n \times m}$ (with n samples and m features) and target $y \in R^n$, polynomial regression try to make a model like : $\boxed{y \approx \Phi(x).\theta}$ where $\Phi(x)$ is a transformation of x to include polynomial features to a specified degree d. θ is the parameter vector to learn.

OLS with Single column $x \left[x \in R^{n \times 1}\right]$

Step-1 Polynomial expansion for degree d.

$$\Phi(x) = \begin{bmatrix} 1 & x_1 & x_1^2 & \dots & x_1^d \\ 1 & x_2 & x_2^2 & \dots & x_2^d \\ \vdots & \vdots & \vdots & & \\ 1 & x_n & x_n^2 & \dots & x_n^d \end{bmatrix} = \Phi(x) \in R^{n \times (d+1)}$$

Step-2

hypo-thesis for a single row of x

$$\hat{y} = \sum_{j=0}^{d} \theta_j . x_j$$

now for n samples MSE will be

$$MSE(\theta) = \frac{1}{n} \sum_{i=0}^{n} (y_i - \hat{y}_i)^2$$

$$MSE(\theta) = \frac{1}{n} \sum_{i=0}^{n} \left(y_i - \sum_{j=0}^{d} \theta_j x_i^j\right)^2$$

Step-3 Define total squared error (Remove $\frac{1}{n}$)

$\frac{1}{n}$ is a constant factor and does not affect the location of minimum while differentiating.

$$\Phi = \begin{bmatrix} 1 & x_1 & x_1^2 & \cdots & x_1^d \\ \vdots & \vdots & \vdots & & \vdots \\ 1 & x_n & x_n^2 & \cdots & x_n^d \end{bmatrix}_{[n \times (d+1)]}$$

$$\theta = \begin{bmatrix} \theta_0 \\ \theta_1 \\ \vdots \\ \theta_d \end{bmatrix}_{[1 \times (d+1)]}$$

Predicted value for each x_i

$$\hat{y}_i = \phi_i \theta = \sum_{j=0}^{d} \theta_j \cdot x_i^j$$

so all prediction (n rows)

$$\boxed{\hat{y} = \phi\theta}$$

now error (residuals) $\boxed{e = y - \hat{y} = y - \phi\theta}$

Step-4 sum of squared errors

$$J(\theta) = \sum_{i=1}^{n} e_i^2 = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2 = \sum_{i=1}^{n} (y_i - \phi\theta)^2 = \|y - \phi\theta\|^2$$

Step-5 expand $J(\theta)$

$$J(\theta) = \|y - \phi\theta\|^2 = (y - \phi\theta)^T \cdot (y - \phi\theta) \quad \left[\|a\| = a^T \cdot a\right]$$

$$\therefore J(\theta) = (y^T - \theta^T \phi^T) \cdot (y - \phi\theta)$$

$$J(\theta) = y^T y - \theta^T \phi^T \cdot y - y^T \phi\theta + \theta^T \phi^T \phi\theta$$

$$\nabla J(\theta) = y^T y - 2\theta^T \Phi^T y + \theta^T \Phi^T \Phi \theta$$ [$\theta^T \Phi^T y$ and $y^T \Phi \theta$ is symetric matrix]

Multivariate polynomial Regression using OLS:

Let, data input matrix: $X \in R^{n \times d}$ where n is the number of rows and d is total columns in X.

Target: $y \in R^{n \times 1}$ ⟶ cause n rows has n target.

Desired polynomial degree P.

Note: we will expand the X matrix by adding (P+1) polynomial term. Then we will apply linear regression on expanded X.

Step-1 polynomial Feature Transform

we want to build all polynomial terms of features up to degree p using combination.

For example: x has d=2 column and degree p=2

For 1 sample, $\Phi(x) = [1, x_1, x_2, x_1^2, x_2^2, x_1 x_2]$

For n col and p degree number of combination $= \frac{(n+p)!}{n!\,p!}$

For n samples $\Phi(x) \in R^{n \times m}$

where m is the polynomial terms, $m = \binom{d+p}{p}$

Step-2 Define the linear model using $\Phi(x)$

hypothesis, $\hat{y} = \Phi(x).\theta$ where:

$\Phi(x) \rightarrow$ transformed x

$\theta \in R^{m \times 1}$ parameter vector

Step-3 define the MSE cost function

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

$$= \frac{1}{n}\sum_{i=1}^{n}(y_i - \Phi(x_i)^T\theta)^2$$

$$= \|y - \Phi(x).\theta\|^2$$

$$= (y - \Phi(x).\theta)^T.(y - \Phi(x).\theta)$$

$$= y^Ty - 2y^T\Phi(x).\theta + \theta^T\Phi(x)^T\Phi(x)\theta$$

Final cost function $\rightarrow$ $J(\theta) = y^Ty - 2y^T(X_{poly}).\theta + \theta^T(X_{poly})^T.X_{poly}.\theta$

$\hat{y}_i = \Phi(x_i).\theta \rightarrow$ scalar

$\hat{y}_i = \Phi(x_i)^T.\theta \rightarrow$ "

$\hat{y}_i = \Phi(x_i).\theta^T \rightarrow$ "

doesn't matter what we write.

$[(a-b)^T.(a-b) = a^Ta - 2a^Tb + b^Tb]$

Step-4 Take gradient with respect θ to minimize $J(\theta)$

$$\nabla_\theta(y^Ty) = 0$$

$$\nabla_\theta(-2y^TX_{poly}\theta) = -2y^TX_{poly}$$

$$\nabla_\theta(\theta^TX_{poly}^TX_{poly}\theta) = 2X_{poly}^TX_{poly}\theta.$$

Total gradient

$$\nabla_\theta(\exists(\theta)) = -2X_{poly}y^T + 2X_{poly}^T X_{poly}\theta$$

To minimize $\exists(\theta)$ set gradient to 0

$$-2X_{poly}y^T + 2X_{poly}^T X_{poly}\theta = 0$$

$$\Rightarrow X_{poly}^T X_{poly}\cdot\theta = X_{poly}\cdot y^T$$

$$\Rightarrow \boxed{\theta = \left(X_{poly}^T \cdot X_{poly}\right)^{-1} \cdot X_{poly}\cdot y^T}$$

"polynomial Regression using Gradient Descent"

single feature Input:

Step-1 Transform feature X

let, $X \in R^{n\times 1}$ ⟶ a single column input.

and polynomial degree = 2

$$X_{poly} = \phi(x) = \begin{bmatrix} 1 & x_1 & x_1^2 \\ 1 & x_2 & x_2^2 \\ \vdots & \vdots & \vdots \\ 1 & x_n & x_n^2 \end{bmatrix} \in R^{n\times 3}$$

Take the hypothesis: $\boxed{\hat{y} = \phi(x)\theta}$ where $\hat{y} \in R^{n\times 1}$, $\theta \in R^{3\times 1}$

Step-2 Define the cost function (MSE)

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

$$= \frac{1}{2n} \sum_{i=1}^{n} (x^i_{poly}\theta - y_i)^2$$

$\Phi(x)$ and X_{poly} is same thing

$$J(\theta) = \frac{1}{2n} \| X_{poly}\theta - y \|^2$$

Step-3 Gradient of the cost function

$$\nabla_\theta (J(\theta)) = \frac{1}{2n} \cdot 2\, \Phi(x)^T (\Phi(x).\theta - y)$$

$$= \frac{1}{n} X^T_{poly} (X_{poly}.\theta - y)$$

$\nabla_\theta \|A\theta - y\|^2$
$= 2A^T (A\theta - y)$

Step-4 update rule

$$\theta := \theta - \alpha . \nabla_\theta (J(\theta))$$

$$\theta := \theta - \alpha \left[\frac{1}{n} X^T_{poly} (X_{poly}\theta - y) \right]$$

Note $X_{poly}.\theta - y \longrightarrow$ is the vector of prediction error.
shape $(n \times 1)$

Multi-feature Input (using gradient descent)

let $X \in R^{n \times d}$ ⟶ the original data input

after polynomial transformation

$$X_{poly} = \Phi(X) \in R^{n \times m}$$ where $n \rightarrow$ number of rows

$m \rightarrow$ includes all the combinations up to degree p

$$\boxed{m = \frac{(c+p)!}{c!\,p!}}$$ ⟶ where $c \rightarrow$ number of columns in X before transformation

$p \rightarrow$ number of degree.

Define the cost function

mSE cost function

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

$$= \frac{1}{2n} \sum_{i=1}^{n} (\Phi(x).\theta - y)^2 \quad [\hat{y} = \Phi(x).\theta]$$

$$= \frac{1}{2n} \| \Phi(x).\theta - y \|^2$$

calculate the gradient:

$$\nabla_\theta (J(\theta)) = \nabla_\theta \left(\frac{1}{2n} \| \Phi(x).\theta - y \|^2 \right)$$

$$= \frac{1}{n} [\Phi(x)]^T (\Phi(x).\theta - y)$$

update theta

$$\theta := \theta - \alpha . \nabla_\theta (J(\theta))$$

α ⟵ learning rate