Linear Regression, Costs & Gradient Descent

Pallavi Mishra
&
Revanth Kumar

Introduction to Linear Regression

- Linear Regression is a predictive model to map the relation between dependent variable and one or more independent variables.
- It is a supervised learning method and regression problem which predicts real valued output.
- The predicted output is done by forming Hypothesis based

$\hat{Y} = \theta_0 + \theta_1 x_1$ (Single Independent Variable)

$\hat{Y} = \theta_0 + \theta_1 x_1 + \theta_2 x_2 + \ldots\ldots + \theta_k x_k$ (Multiple Independent Va

$= \sum_{i=0}^{k} \theta_i x_i$; Where $x_0 = 1$ .....(1)

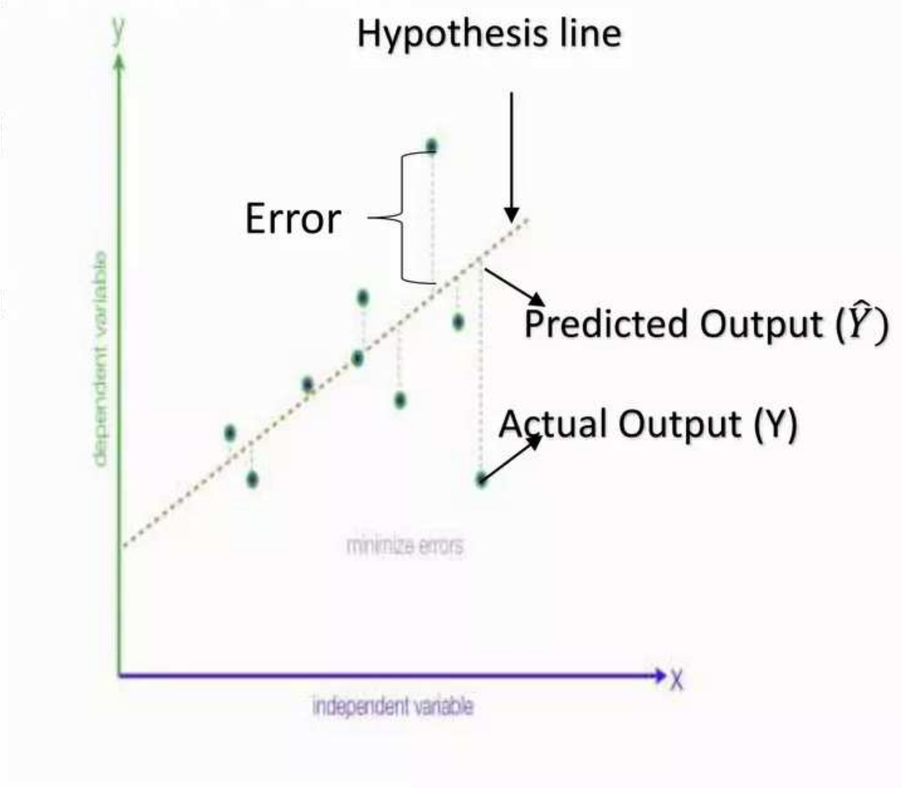
Where θ_i = parameters for i^{th} independent variable(s)

For estimation of performance of the linear model, SSE

Squared Sum Error (SSE) = $\sum_{i=1}^{k} (Y - \hat{Y})^2$

Note: Here, Y is the actual observed output

And, $\hat{Y}$ is the predicted output.



Model Representation

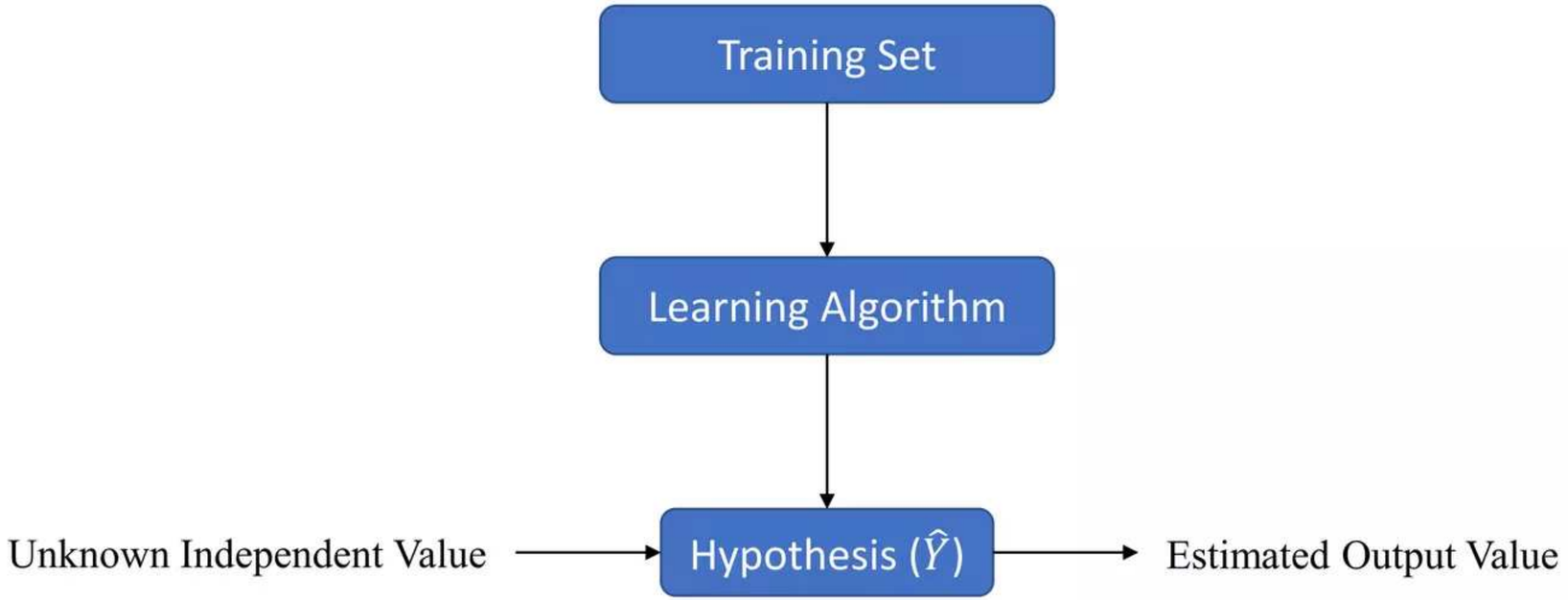


Fig.1 Model Representation of Linear Regression

Hint: Gradient descent as learning algorithm

How to Represent Hypothesis?

- We know, hypothesis is represented by $\hat{Y}$, which can be formulated depending upon single variable linear regression (Univariate Linear Regression) or Multi-variate linear regression.
- $\hat{Y} = \theta_0 + \theta_1 x_1$
- Here, θ_0 = intercept and θ_1 = slope= $\frac{\Delta y}{\Delta x}$ and x_1 = independent variable
- Question arises: How do we choose $\theta_{i's}$ values for best fitting hypothesis?
- Idea : Choose θ_0 , θ_1 so that $\hat{Y}$ is close to Y for our training examples (x, y)
- Objective: min J(θ_0 , θ_1),
- Note: J(θ_0 , θ_1) = Cost Function.
- Formulation of J(θ_0 , θ_1) = $\frac{1}{2m}\sum_{i=1}^{m}(\hat{Y}^{(i)} - Y^{(i)})^2$

Note: m = No. of instances of dataset

Objective function for linear regression

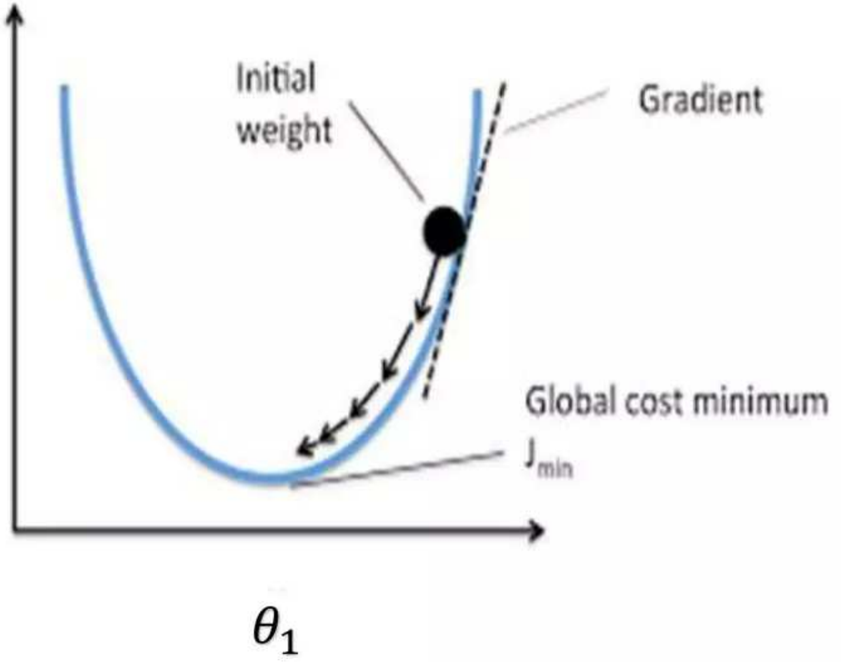
- The most important objective of linear regression model is to minimize cost function by choosing a optimal value for θ_0 , θ_1.
- For optimization technique, Gradient Descent is mostly used in case of predictive models.
- By taking θ_0 = 0 and θ_1 = some random values (in case of univariate linear regression), the graph (θ_1 vs J(θ_1)) gets represented in the form of l

Advantage of Gradient descent in linear regressio

- No scope to stuck in local optima, since there is only

  One global optima position where slope(θ_1) = 0 $J(\theta_1)$

(convex graph)



Normal Distribution $N(\mu, \sigma^2)$

Estimation of mean (μ) and variance (σ^2):

- Let size of data set = n, denoted by $y_1, y_2 \ldots\ldots y_n$
- Assuming $y_1, y_2 \ldots\ldots y_n$ are independent random variables or Independent Identically Distributed (iid), they are normally distributed random variables.
- Assuming no independent variables (x), in order to estimate the future value of y we need to find to find unknown parameters (μ & σ^2).

Concept of Maximum Likelihood Estimation:

- Using Maximum Likelihood Estimation (MLE) concept, we are trying to find the optimal value for value for the mean (μ) and standard deviation (σ) for distribution given a bunch of observed observed measurements.
- The goal of MLE is to find optimal way to fit a distribution to the data so, as to work easily with with data

Continue…

Estimation of μ & σ^2:

- **Density of normal random variable** = f(y) = $\frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-1}{2\sigma^2}(y-\mu)^2}$

L $(\mu,\ \sigma^2)$ is a joint density

Now,

let, L $(\mu,\ \sigma^2)$ = f $(y_1, y_2 \ldots\ldots y_n)$ = $\prod_{i=1}^{n} \frac{1}{\sqrt{2\pi}\sigma} e^{\frac{-1}{2\sigma^2}(y-\mu)^2}$

let, assume $\sigma^2 = \theta$

let, L (μ, θ) = $\frac{1}{(\sqrt{2\pi\theta})^n} e^{\frac{-1}{2\theta}\Sigma(y-\mu)^2}$

taking log on both sides

LL (μ, θ) = log $(2\pi\theta)^{-\frac{n}{2}}$ + log $(e^{\frac{-1}{2\sigma^2}(y-\mu)^2})$

$= -\frac{n}{2}\log(2\pi\theta) - \frac{1}{2\theta}(y-\mu)^2$ (2)

∗LL (μ, θ) is denoted as log of joint density

$* \ log e^x = x$

Continue…

- Our objective is to estimate the next occurring of data point y in the distribution of data. Using MLE we can find the optimal value for (μ, σ^2). For a given trainings set we need to find max LL (μ, θ) .
- Let us assume $\theta = \sigma^2$ for simplicity
- Now, we use partial derivatives to find the optimal values of (μ, σ^2) and equating to zero $LL' = 0$

$$\text{LL}(\mu, \theta) = -\frac{n}{2}\log(2\pi\theta) - \frac{1}{2\theta}(y - \mu)^2$$

- Taking partial derivative wrt μ in eq (2), we get

$$LL'_\mu = 0 - \frac{2}{2\theta}\sum(y_i - \mu)\ (-1)$$

$$=> \sum(y_i - \mu) = 0$$

$$=> \sum y_i = n_\mu$$

* LL'_μ is partial derivative of LL wrt μ

Continue…

$$\hat{\mu} = \frac{1}{n}\sum y_i$$

* $\hat{\mu}$ is estimated mean value

Again taking partial derivatives on eq (2) wrt θ

$$LL'_\theta = -\frac{n}{2}\frac{1}{2\pi\theta}(2\pi) - \frac{-1}{2\theta^2}\sum(y_i - \mu)^2$$

Setting above to zero, we get

$$\Rightarrow \frac{1}{2\theta}\sum(y_i - \mu)^2 = \frac{n}{2}\frac{1}{\theta}$$

Finally, this leads to solution

$$\widehat{\sigma^2} = \hat{\theta} = \frac{1}{n}\sum(y_i - \mu)^2$$

* $\widehat{\sigma^2}$ is estimated variance

After plugging estimate of

$$\widehat{\sigma^2} = \frac{1}{n}\sum(y - \bar{y})^2$$

Continue…

- Above estimate can be generali $\widehat{\sigma^2} = \frac{1}{n}\sum error^2$ ror^2 * error = y − $\bar{y}$

- Finally we estimated the value of mean and variance in order to predict the future occurrence of y ($\hat{y}$) data points.
- Therefore the best estimate of occurrence of next y ($\hat{y}$) that is likely to occur is $\hat{\mu}$ and the solution is arrived by using SSE ($\widehat{\sigma^2}$)

Optimization & Derivatives

$$J(\theta) = \frac{1}{2n}\sum_{i=1}^{i=n}(y_i - \sum_{j=1}^{j=k} x_{ij}\theta_j)^2$$

$$Y=\begin{pmatrix} y_1 \\ y_2 \\ \dots \\ y_n \end{pmatrix}; \qquad X=\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1k} \\ x_{21} & x_{22} & \dots & x_{2k} \\ \dots & \dots & \dots & \dots \\ x_{n1} & x_{2n} & \dots & x_{nk} \end{bmatrix}; \qquad \theta=\begin{pmatrix} \theta_1 \\ \theta_2 \\ \dots \\ \theta_k \end{pmatrix}$$

$\sum_{j=1}^{j=k} x_{ij}\theta_j$ is simple multiplication of i^{th} row of matrix X and vector θ . Hence

$$= \frac{1}{2n}\sum_{i=1}^{i=n}(Y - X\theta)^2$$

Continue…

$= \left(Y - \hat{Y}\right)'(Y - \hat{Y})$ $\therefore \ \hat{Y} = X\theta$

$\text{J}(\theta) = \frac{1}{2n}(Y - X\theta)'(Y - X\theta)$

$= Y'Y \ - Y'X\theta \ - YX\theta' - X\theta'X\theta$

Now, Derivative with respect to θ

$\frac{\partial}{\partial\theta} = 0 - 2\text{XY} + 2X^2\theta$

$= \frac{1}{2n}(-\ 2\text{XY} + 2X^2\theta)$

$= -\ \frac{2}{2n}(\text{XY} - X^2\theta)$

$= -\ \frac{1}{n}(\text{XY} - X'X\theta)$

$= \ -\ \frac{1}{n}X'(\text{Y} - \hat{Y})$

$\text{J}(\theta) = \frac{1}{n}X'(\hat{Y} - Y)$

How to start with Gradient Descent

- The basic assumption is to start at any random position x_0 and take derivative value.
- 1^{st} case: if derivative value > 0 , increasing
- Action : then change the θ_1 values using the gradient descent formula.
- $\theta_1 = \theta_1 - \alpha \frac{d\, J(\theta_1)}{d\theta_1}$
- here, α = learning rate / parameter

Gradient Descent algorithm

- Repeat until convergence { $\theta_1 := \theta_1 - \alpha \frac{d\, J(\theta_1)}{d\theta_1}$ here, assuming $\theta_0 = 0$ for univariate linear regression }

For multi variate linear regression:

- Repeat until convergence { $\theta_j := \theta_j - \alpha \frac{d\, J(\theta_0,\theta_1)}{d\theta_j}$ }

Simultaneous update of $\theta_{0,}\theta_1$

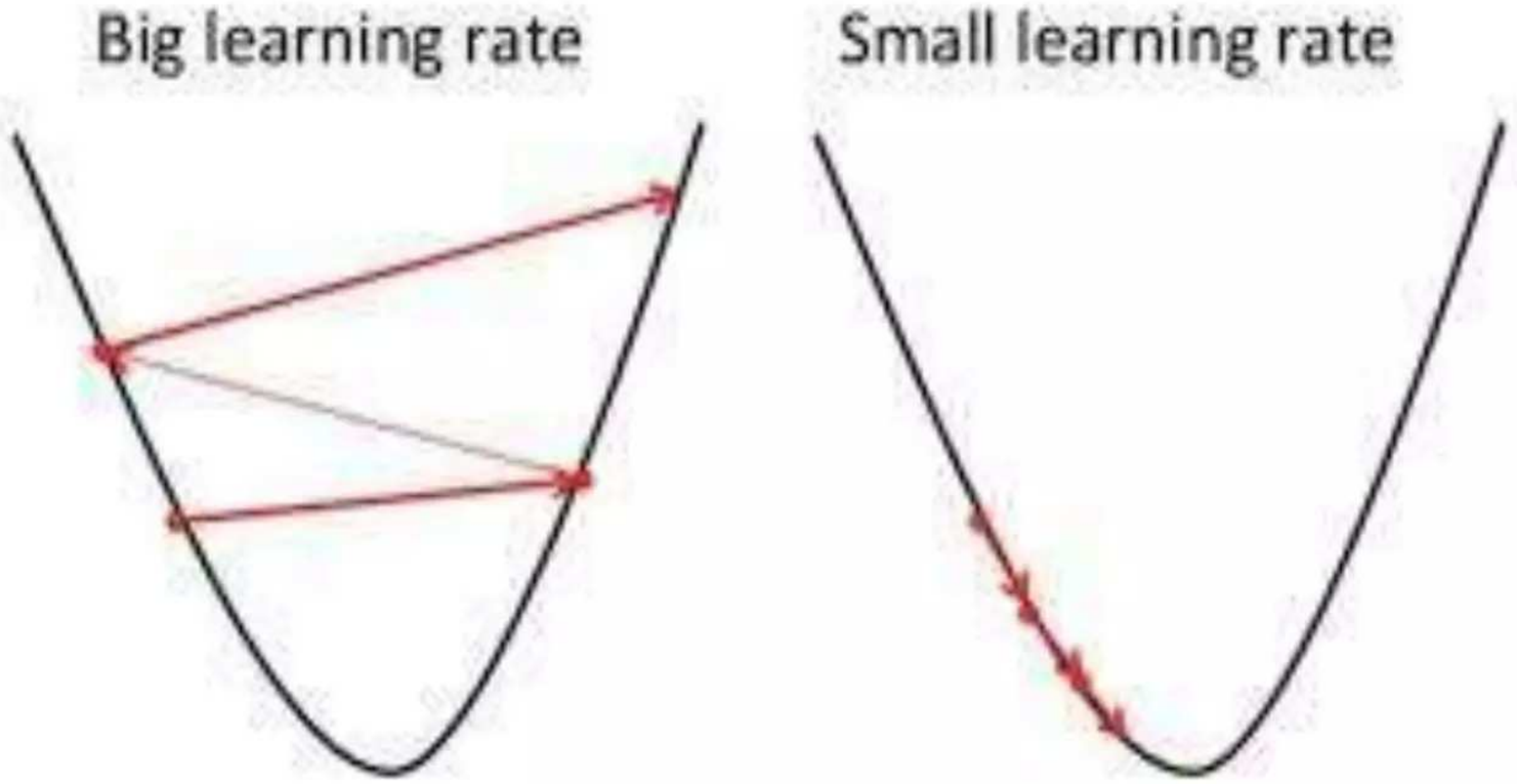
Temp 0 := $\theta_o := \theta_0 - \alpha \frac{d\, J(\theta_0,\theta_1)}{d\theta_0}$

Temp 1 := $\theta_1 := \theta_1 - \alpha \frac{d\, J(\theta_0,\theta_1)}{d\theta_1}$

$\theta_o :=$ Temp 0

$\theta_1 :=$ Temp 1

Effects associated with varying values of learning rate (α)



Continue:

- In the first case, we may find difficulty to reach at global optima since large value of α may overshoot the optimal position due to aggressive updating of θ values.
- Therefore, as we approach optima position, gradient descent will take automatically smaller steps.

Conclusion

- The cost function for linear regression is always gong to be a bow-shaped function (convex function)
- This function doesn’t have an local optima except for the one global optima.
- Therefore, using cost function of type $J(\theta_{0,}\theta_1)$ which we get whenever we are using linear regression, it will always converge to the global optimum.
- Most important is make sure our gradient descent algorithms is working properly .
- On increasing number of iterations, the value of $J(\theta_{0,}\theta_1)$ should get decreasing after every iterations.
- Determining the automatic convergence test is difficult because we don't know the threshold value.