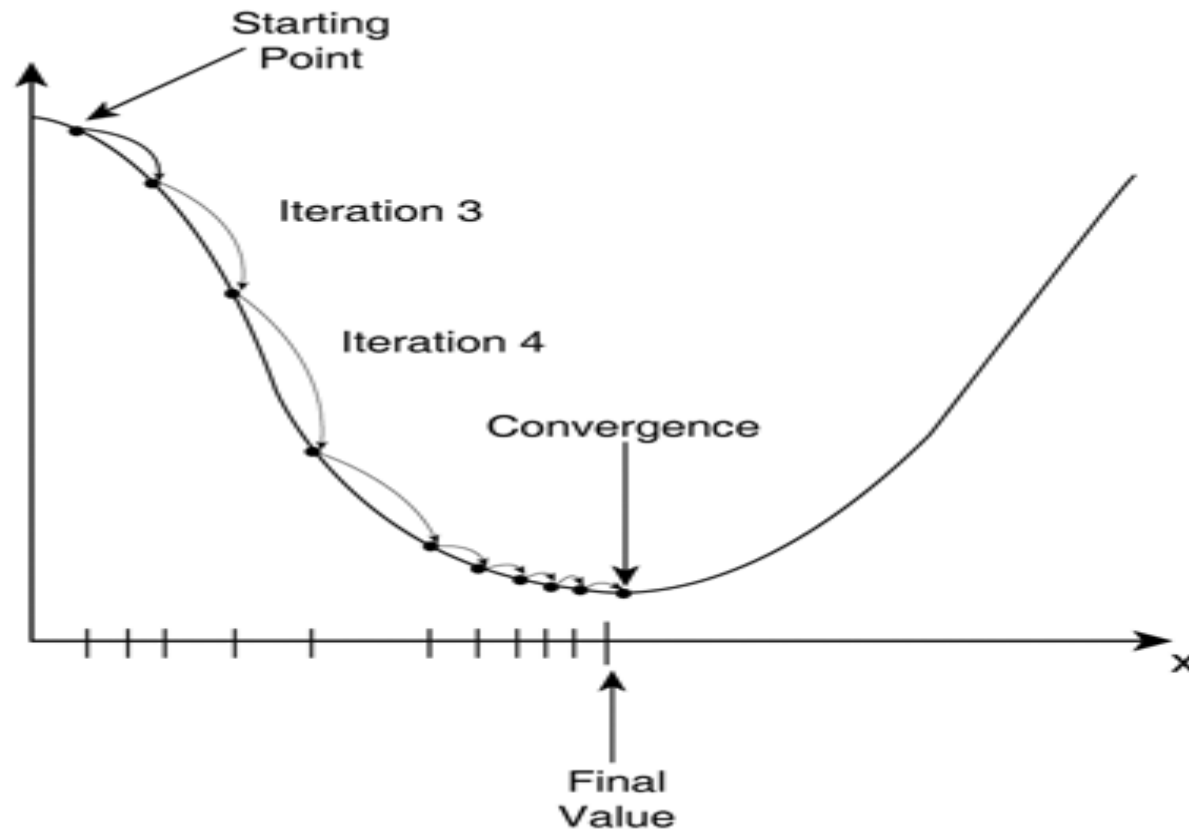


# Linear Regression using Gradient Descent

## Overview



$$\text{Cost Function}(MSE) = \frac{1}{n} \sum_{i=0}^n (y_i - y_{i \text{ pred}})^2$$

Replace  $y_{i \text{ pred}}$  with  $mx_i + c$

$$\text{Cost Function}(MSE) = \frac{1}{n} \sum_{i=0}^n (y_i - (mx_i + c))^2$$

# Linear Regression using Gradient Descent

## Calculus

### Derivative

$$\frac{d}{dx} n = 0$$

$$\frac{d}{dx} x = 1$$

$$\frac{d}{dx} x^n = nx^{n-1}$$

$$\frac{d}{dx} e^x = e^x$$

$$\frac{d}{dx} \ln x = \frac{1}{x}$$

$$\frac{d}{dx} n^x = n^x \ln n$$

$$\frac{d}{dx} \sin x = \cos x$$

$$\frac{d}{dx} \cos x = -\sin x$$

### Integral (Antiderivative)

$$\int 0 \, dx = C$$

$$\int 1 \, dx = x + C$$

$$\int x^n \, dx = \frac{x^{n+1}}{n+1} + C$$

$$\int e^x \, dx = e^x + C$$

$$\int \frac{1}{x} \, dx = \ln x + C$$

$$\int n^x \, dx = \frac{n^x}{\ln n} + C$$

$$\int \cos x \, dx = \sin x + C$$

$$\int \sin x \, dx = -\cos x + C$$

$$\frac{d}{dx} \tan x = \sec^2 x$$

$$\frac{d}{dx} \cot x = -\csc^2 x$$

$$\frac{d}{dx} \sec x = \sec x \tan x$$

$$\frac{d}{dx} \csc x = -\csc x \cot x$$

$$\frac{d}{dx} \arcsin x = \frac{1}{\sqrt{1-x^2}}$$

$$\frac{d}{dx} \arccos x = -\frac{1}{\sqrt{1-x^2}}$$

$$\frac{d}{dx} \arctan x = \frac{1}{1+x^2}$$

$$\frac{d}{dx} \operatorname{arccot} x = -\frac{1}{1+x^2}$$

$$\frac{d}{dx} \operatorname{arcsec} x = \frac{1}{x\sqrt{x^2-1}}$$

$$\frac{d}{dx} \operatorname{arccsc} x = -\frac{1}{x\sqrt{x^2-1}}$$

$$\int \sec^2 x \, dx = \tan x + C$$

$$\int \csc^2 x \, dx = -\cot x + C$$

$$\int \tan x \sec x \, dx = \sec x + C$$

$$\int \cot x \csc x \, dx = -\csc x + C$$

$$\int \frac{1}{\sqrt{1-x^2}} \, dx = \arcsin x + C$$

$$\int -\frac{1}{\sqrt{1-x^2}} \, dx = \arccos x + C$$

$$\int \frac{1}{1+x^2} \, dx = \arctan x + C$$

$$\int -\frac{1}{1+x^2} \, dx = \operatorname{arccot} x + C$$

$$\int \frac{1}{x\sqrt{x^2-1}} \, dx = \operatorname{arcsec} x + C$$

$$\int -\frac{1}{x\sqrt{x^2-1}} \, dx = \operatorname{arccsc} x + C$$

# Linear Regression using Gradient Descent

## Algorithm

### Step: 01

Gradient (m) = 0

Intercept (c) = 0

Learning Rate (L) = ~0.0001

### Step: 02

Calculate the partial derivative of the Cost function with respect to m. Let the partial derivative of the Cost function with respect to m be  $D_m$ .

$$\begin{aligned} D_m &= \frac{\partial(\text{Cost Function})}{\partial m} = \frac{\partial}{\partial m} \left( \frac{1}{n} \sum_{i=0}^n (y_i - y_{i \text{ pred}})^2 \right) \\ &= \frac{1}{n} \frac{\partial}{\partial m} \left( \sum_{i=0}^n (y_i - (mx_i + c))^2 \right) \\ &= \frac{1}{n} \frac{\partial}{\partial m} \left( \sum_{i=0}^n (y_i^2 + m^2 x_i^2 + c^2 + 2mx_i c - 2y_i mx_i - 2y_i c) \right) \\ &= \frac{-2}{n} \sum_{i=0}^n x_i (y_i - (mx_i + c)) \\ &= \frac{-2}{n} \sum_{i=0}^n x_i (y_i - y_{i \text{ pred}}) \end{aligned}$$

### Step: 03

Similarly, let's find the partial derivative with respect to c. Let the partial derivative of the Cost function with respect to c be  $D_c$ .

$$\begin{aligned} D_c &= \frac{\partial(\text{Cost Function})}{\partial c} = \frac{\partial}{\partial c} \left( \frac{1}{n} \sum_{i=0}^n (y_i - y_{i \text{ pred}})^2 \right) \\ &= \frac{1}{n} \frac{\partial}{\partial c} \left( \sum_{i=0}^n (y_i - (mx_i + c))^2 \right) \\ &= \frac{1}{n} \frac{\partial}{\partial c} \left( \sum_{i=0}^n (y_i^2 + m^2 x_i^2 + c^2 + 2mx_i c - 2y_i m x_i - 2y_i c) \right) \\ &= \frac{-2}{n} \sum_{i=0}^n (y_i - (mx_i + c)) \\ &= \frac{-2}{n} \sum_{i=0}^n (y_i - y_{i \text{ pred}}) \end{aligned}$$

# Linear Regression using Gradient Descent

## Algorithm

Step: 04

Update the value of the gradient and intercept.

$$m = m - L \times D_m$$

$$c = c - L \times D_c$$

Repeat the steps!  
1000 times

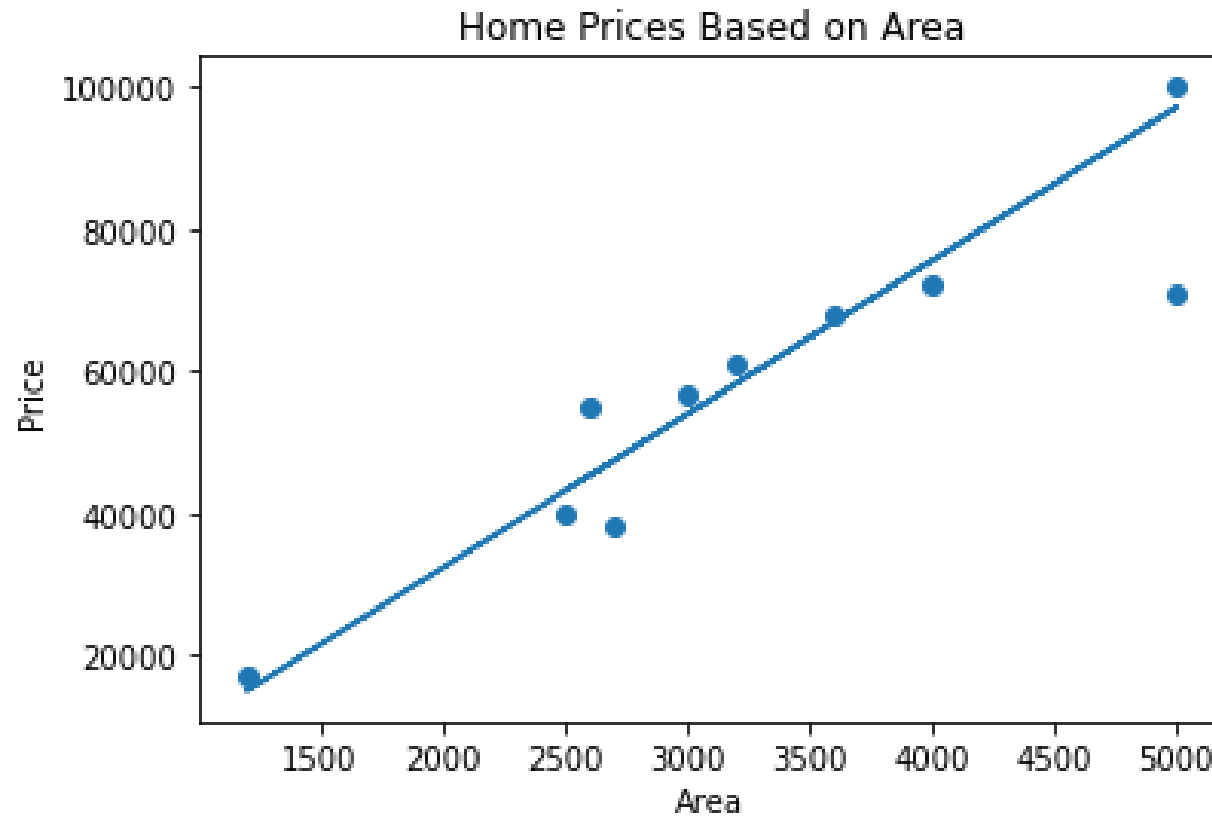
### Overview:

- Single Variable Linear Regression
- Multiple Variable Linear Regression
- Single vs Multiple
- Cost Function
- Gradient Decent
- Accuracy
  - $R^2$  Value
- Implementing with Python



# Linear Regression with Single Variable

## Overview



$$y = mx + b ; \text{ or, } Y = 21.43 * X + 4980.13$$

Coefficient = 21.43  
Intercept = 4980.13

# Linear Regression with Multiple Variables

## Mathematical Representation

predictor, 'x-variable',  
independent variable,  
explanatory variable

coefficient

$$Y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p$$

linear predictor

response, dependent variable,  
observation, 'y-variable'

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \dots + \beta_p x_{ip}$$

**where, for  $i = n$  observations:**

$y_i$  = dependent variable

$x_i$  = explanatory variables

$\beta_0$  = y-intercept (constant term)

$\beta_p$  = slope coefficients for each explanatory variable

# Linear Regression with Single Vs. Multiple Variables

## Mathematical Representation

Single

$$y = b_0 + b_1 * x_1$$

Multiple

Dependent variable (DV)

Independent variables (IVs)

$$y = b_0 + b_1 * x_1 + b_2 * x_2 + ... + b_n * x_n$$

# R Squared Value / Model Accuracy

## Mathematical Calculation

$$\text{R Squared Value} = \sum_{i=0}^{n-1} \frac{\text{Predicted Value (Yp)} - \text{Mean Value } (\bar{Y})}{\text{Actual Value (Y)} - \text{Mean Value } (\bar{Y})}$$

$$= \text{Something} * 100$$

$$\text{Accuracy} = \text{Something} \%$$

Way no: 01

```
reg.score(xtest, ytest)
```

Way no: 02

```
y_pred = reg.predict(xtest) #Predicted y  
from sklearn.metrics import r2_score  
Score = r2_score(ytest, y_pred)
```