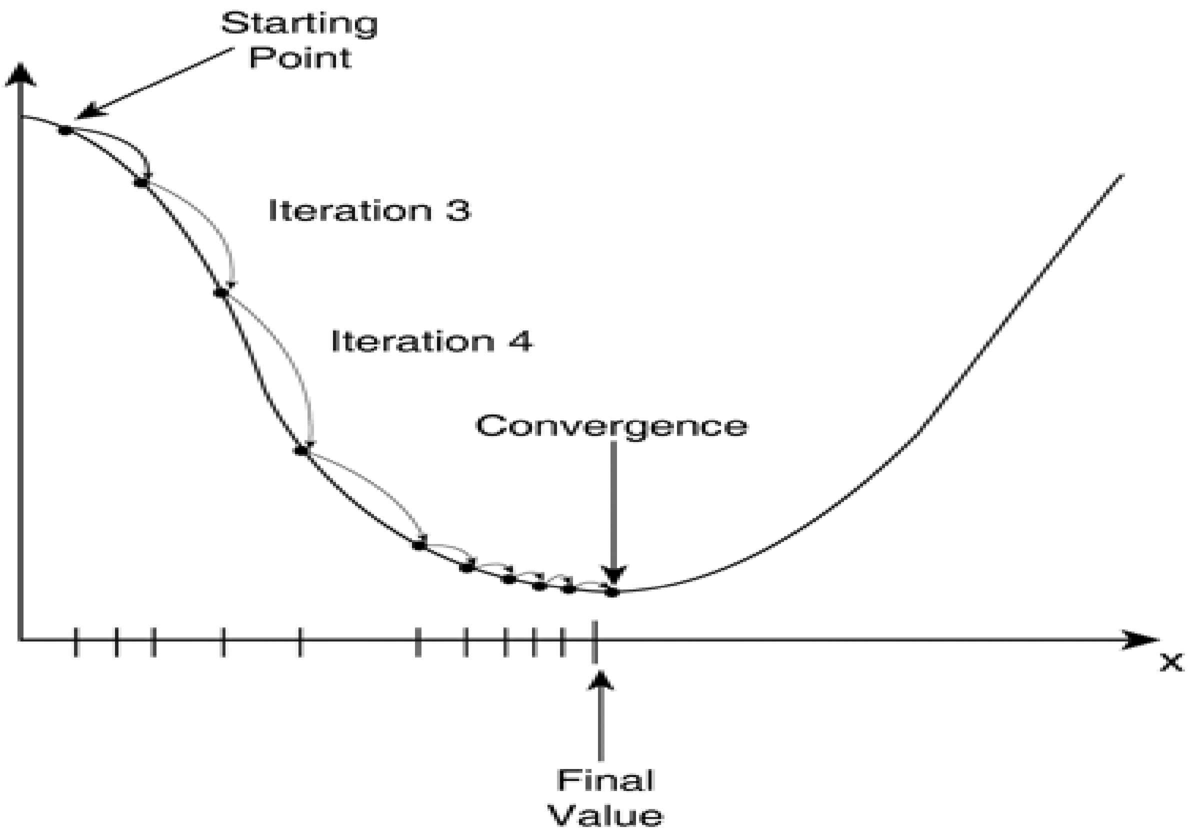
Linear Regression using Gradient Descent

Overview



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

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

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

Calculus

Derivative	Integral (Antiderivative)
$\frac{d}{dx}n = 0$	$\int 0\ dx = C$
$\frac{d}{dx}x = 1$	$\int 1\ dx = x + C$
$\frac{d}{dx}x^n = nx^{n-1}$	$\int x^n\ dx = \frac{x^{n+1}}{n+1} + C$
$\frac{d}{dx}e^x = e^x$	$\int e^x\ dx = e^x + C$
$\frac{d}{dx}\ln x = \frac{1}{x}$	$\int \frac{1}{x}\ dx = \ln x + C$
$\frac{d}{dx}n^x = n^x \ln x$	$\int n^x\ dx = \frac{n^x}{\ln n} + C$
$\frac{d}{dx}\sin x = \cos x$	$\int \cos x\ dx = \sin x + C$
$\frac{d}{dx}\cos x = -\sin x$	$\int \sin x\ dx = -\cos x + C$
$\frac{d}{dx}\tan x = \sec^2 x$	$\int \sec^2 x\ dx = \tan x + C$
$\frac{d}{dx}\cot x = -\csc^2 x$	$\int \csc^2 x\ dx = -\cot x + C$
$\frac{d}{dx}\sec x = \sec x \tan x$	$\int \tan x \sec x\ dx = \sec x + C$
$\frac{d}{dx}\csc x = -\csc x \cot x$	$\int \cot x \csc x\ dx = -\csc x + C$
$\frac{d}{dx}\arcsin x = \frac{1}{\sqrt{1-x^2}}$	$\int \frac{1}{\sqrt{1-x^2}}\ dx = \arcsin x + C$
$\frac{d}{dx}\arccos x = -\frac{1}{\sqrt{1-x^2}}$	$\int -\frac{1}{\sqrt{1-x^2}}\ dx = \arccos x + C$
$\frac{d}{dx}\arctan x = \frac{1}{1+x^2}$	$\int \frac{1}{1+x^2}\ dx = \arctan x + C$
$\frac{d}{dx}\operatorname{arc\,cot} x = -\frac{1}{1+x^2}$	$\int -\frac{1}{1+x^2}\ dx = \operatorname{arc\,cot} x + C$
$\frac{d}{dx}\operatorname{arc\,sec} x = \frac{1}{x\sqrt{x^2-1}}$	$\int \frac{1}{x\sqrt{x^2-1}}\ dx = \operatorname{arc\,sec} x + C$
$\frac{d}{dx}\operatorname{arc\,csc} x = -\frac{1}{x\sqrt{x^2-1}}$	$\int -\frac{1}{x\sqrt{x^2-1}}\ dx = \operatorname{arc\,csc} x + C$

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 Dm.

$$D_m = \frac{\partial(Cost\ Function)}{\partial m} = \frac{\partial}{\partial m}\left(\frac{1}{n}\sum_{i=0}^{n}(y_i - y_{i\,pred})^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial m}\left(\sum_{i=0}^{n}(y_i - (mx_i + c))^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial m}\left(\sum_{i=0}^{n}(y_i^2 + m^2x_i^2 + c^2 + 2mx_ic - 2y_imx_i - 2y_ic)\right)$$

$$= \frac{-2}{n}\sum_{i=0}^{n}x_i(y_i - (mx_i + c))$$

$$= \frac{-2}{n}\sum_{i=0}^{n}x_i(y_i - y_{i\,pred})$$

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 Dc.

$$D_c = \frac{\partial(Cost\ Function)}{\partial c} = \frac{\partial}{\partial c}\left(\frac{1}{n}\sum_{i=0}^{n}(y_i - y_{i\,pred})^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial c}\left(\sum_{i=0}^{n}(y_i - (mx_i + c))^2\right)$$

$$= \frac{1}{n}\frac{\partial}{\partial c}\left(\sum_{i=0}^{n}(y_i^2 + m^2x_i^2 + c^2 + 2mx_ic - 2y_imx_i - 2y_ic)\right)$$

$$= \frac{-2}{n}\sum_{i=0}^{n}(y_i - (mx_i + c))$$

$$\frac{-2}{n}\sum_{i=0}^{n}(y_i - y_{i\,pred})$$

Step: 04

Update the value of the gradient and intercept.

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

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

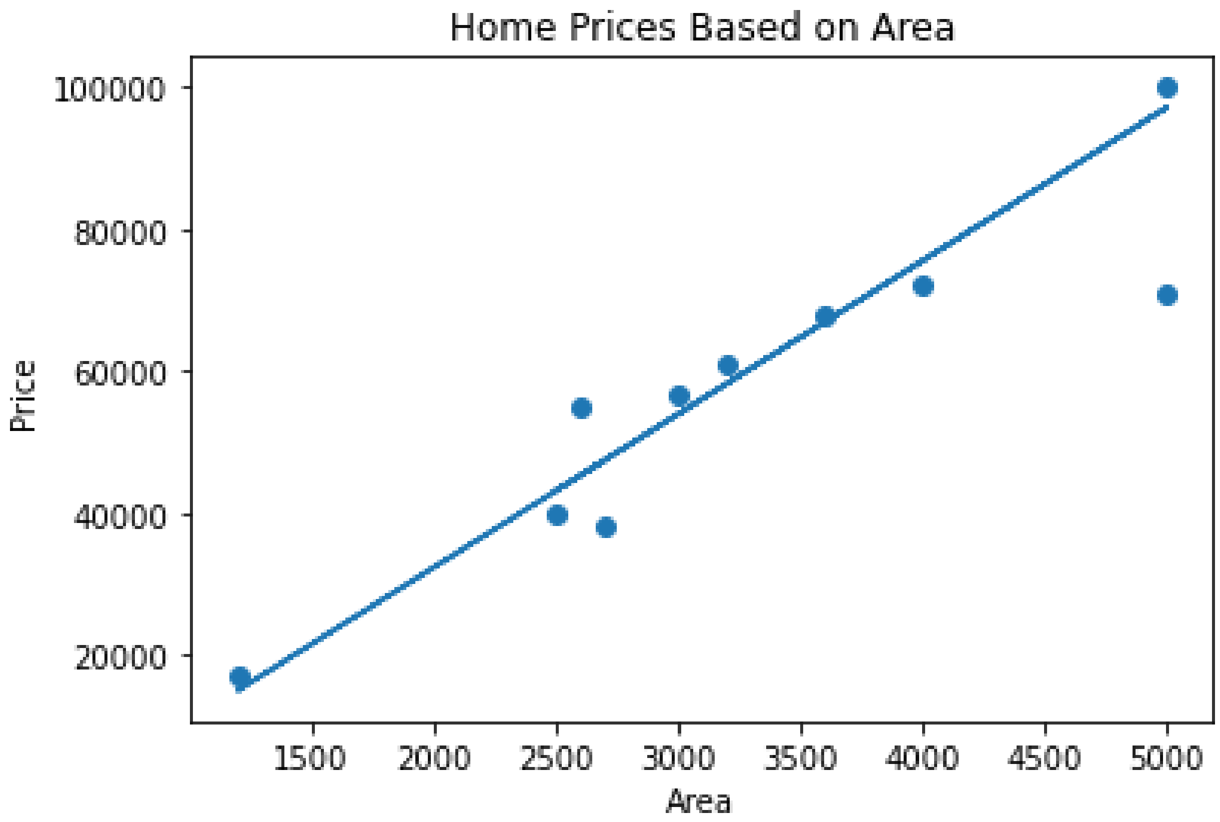
Repeat the steps!
1000 times

Overview:

- Single Variable Linear Regression
- Multiple Variable Linear Regression
- Single vs Multiple
- Cost Function
- Gradient Decent
- Accuracy
    - R2 Value
- Implementing with Python

Linear Regression with Single Variable

Overview



y = mx + b ; 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

$$Y = \underbrace{\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_p x_p}_{\text{linear predictor}}$$

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

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

where, for $i = n$ observations:

y_i = dependent variable

x_i = expanatory variables

β_0 = y-intercept (constant term)

β_p = slope coefficients for each explanatory variable

Mathematical Representation

Single

$$y = b_0 + b_1 * x_1$$

Dependent variable (DV) Independent variables (IVs)

Multiple

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

R Squared Value / Model Accuracy

Mathematical Calculation

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

= Something * 100

Accuracy = Something %

R Squared Value / Model Accuracy

Python Implementation

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)
```