Regression Analysis

Overview / Contents

- Regression Concepts
  - Types of Regression
- In-depth intuition of OLS
- Lose Functions
- Cost Function
- R Squared Values
- Coding with Python:
  - Implementing Linear Regression
  - Simple ML Project
  - Assignment

- **Regression in Machine Learning:**

Regression is a technique used to predict numerical values based on input features. It models the relationship between a dependent variable (what you want to predict) and independent variables (features).

- **Example: Predicting House Prices:**

Imagine you're predicting house prices based on square footage. The regression model finds a line that best fits the data: Price = 100 * SquareFootage + 50000. Here, 100 is the increase in price for each square foot increase, and $50,000 is the starting price estimate. This model helps estimate prices for different house sizes.

Regression Concepts

Examples

1. Economics: GDP Prediction:
Using historical data, economists can predict a country's future GDP based on factors like inflation rate, unemployment rate, and consumer spending.

2. Healthcare: Patient Outcome:
Doctors can predict a patient's recovery time after surgery based on variables like age, pre-existing conditions, and the complexity of the procedure.

3. Retail: Sales Forecasting:
Retailers can use regression to forecast sales based on parameters like advertising spend, holiday season, and previous sales data.

4. Finance: Stock Price Prediction:
Traders and investors can predict stock prices by analyzing factors like trading volume, historical prices, and economic indicators.

5. Agriculture: Crop Yield Estimation:
Regression helps farmers predict crop yields based on factors like weather conditions, soil quality, and type of crop.

6. Marketing: Customer Lifetime Value:
Marketers use regression to estimate a customer's lifetime value based on purchase history, engagement, and demographic information.

7. Education: Student Performance:
Educators can predict student performance on standardized tests using factors like attendance, study time, and past test scores.

8. Energy: Energy Consumption:
Energy companies can predict household energy consumption based on variables like weather, household size, and appliance usage.

9. Transportation: Fuel Efficiency:
Manufacturers predict a vehicle's fuel efficiency based on engine specifications, weight, and aerodynamics.

10. Real Estate: Property Valuation:
Regression helps in estimating property values based on features like location, square footage, and nearby amenities.

Types of Regression?

Types of Regression

Common types

There are several types of regression techniques, each designed to handle different types of data and relationships between variables. Here are some common types of regression:

1. Linear Regression:
   - Simple Linear Regression: Predicting a continuous dependent variable using a single independent variable.
   - Multiple Linear Regression: Predicting a dependent variable using multiple independent variables.

2. Polynomial Regression:
   - Modeling nonlinear relationships by adding polynomial terms to the regression equation.

3. Ridge Regression:
   - Adding a penalty term to the coefficients to prevent overfitting.

4. Lasso Regression:
   - Similar to ridge regression, but with a penalty that encourages some coefficients to become exactly zero, leading to feature selection.

Types of Regression

Common types

5. Logistic Regression:
   - Used for binary or multinomial classification tasks, predicting the probability of an event occurring.

6. Poisson Regression:
   - Modeling count data, often used in situations where the dependent variable represents counts.

7. Time Series Regression:
   - Modeling time-dependent data, considering temporal patterns and autocorrelation.

9. Nonlinear Regression:
   - Fitting a nonlinear function to the data to capture complex relationships.

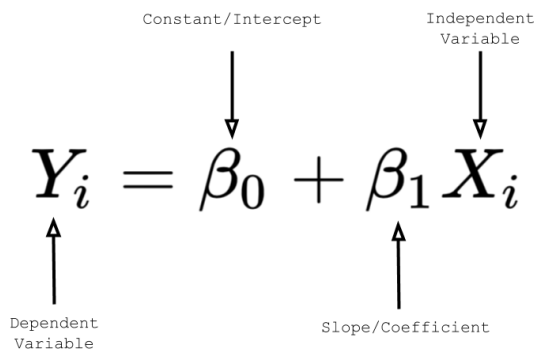
9. Support Vector Regression (SVR):
   - Utilizes support vector machines for regression tasks, particularly suited for high-dimensional spaces.8

All you need to know about Linear Regression!

Linear Regression

Basic Concepts

- Linear regression is a fundamental **supervised machine learning algorithm** used for predicting a continuous numerical value based on one or more input features.
- It models the relationship between the **dependent** variable and the **independent** variables as a linear equation.
- The goal is to find the **best-fitting line** (or hyperplane in higher dimensions) that minimizes the difference between the observed and predicted values.
- This best-fitting line represents the **linear relationship** between the input features and the target variable.



$$Y = ax + b$$

$$Y = mx + c$$

Here,
M = Coefficient of the input feature X
C = Intercept
X = Features
Y = Predicted Output / Label

1. **Simple Linear Regression**:
   - In the equation $y = mx + b$, there's no transpose term because you're dealing with single variables (one input feature x, and one output variable y).
2. **Vectorized Form of Linear Regression**:
   - In the equation $wx + b$, if you're working with multiple features, the weight vector w is often represented as a column vector. However, when you want to perform matrix operations like the dot product, you may need to transpose it to align the dimensions correctly. This is where the transpose term appears.

   So, in the vectorized form of linear regression with multiple features, you might have something like:

$$wx + b = \text{np.dot}(w^T, x) + b$$

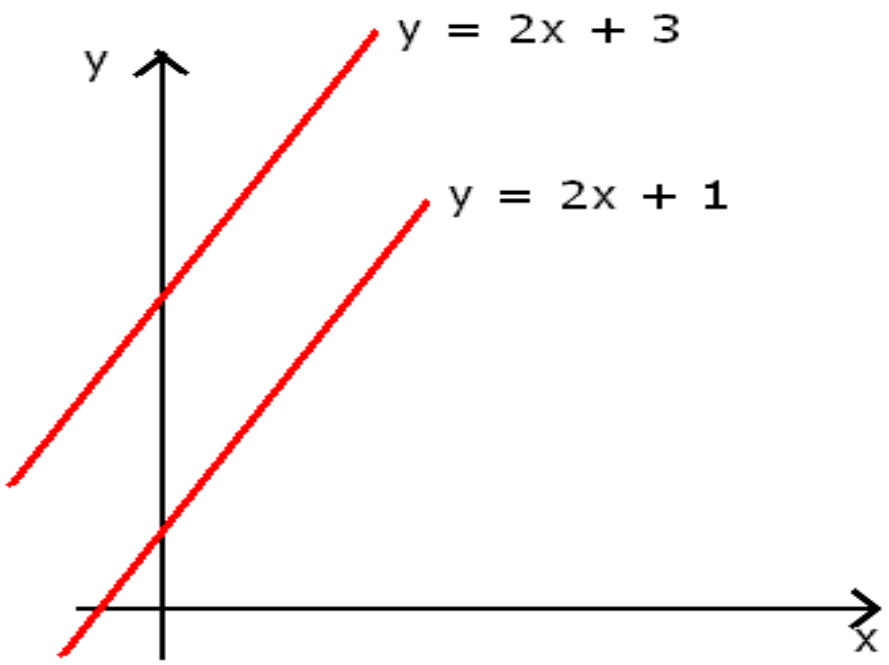


Fig: Straight Line

X = 10, 30, 50
Y = 2*10 + 3 =23
Y = 2*30 + 3 = 63
Y = 2*50 + 3 = 103

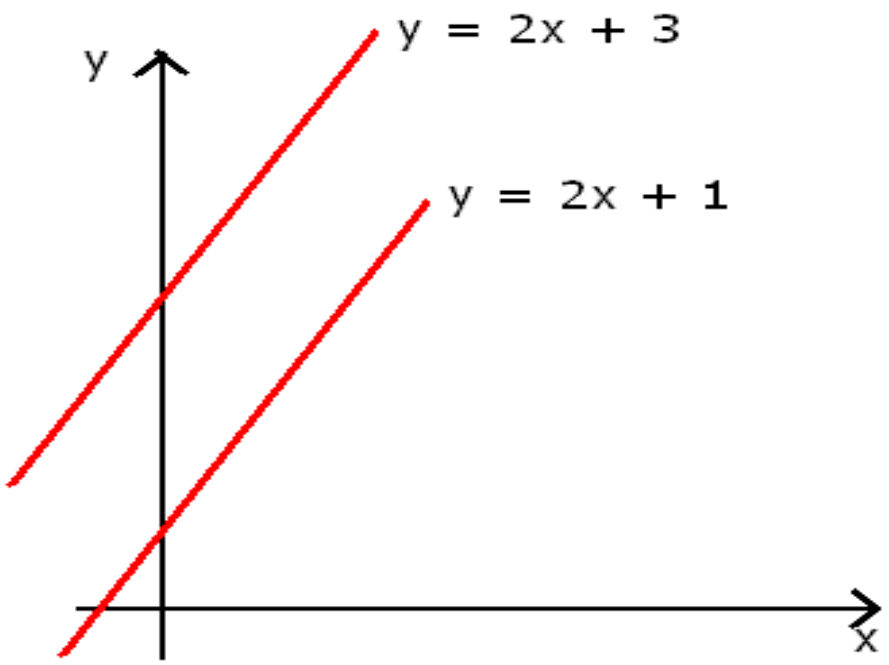


Fig: Straight Line

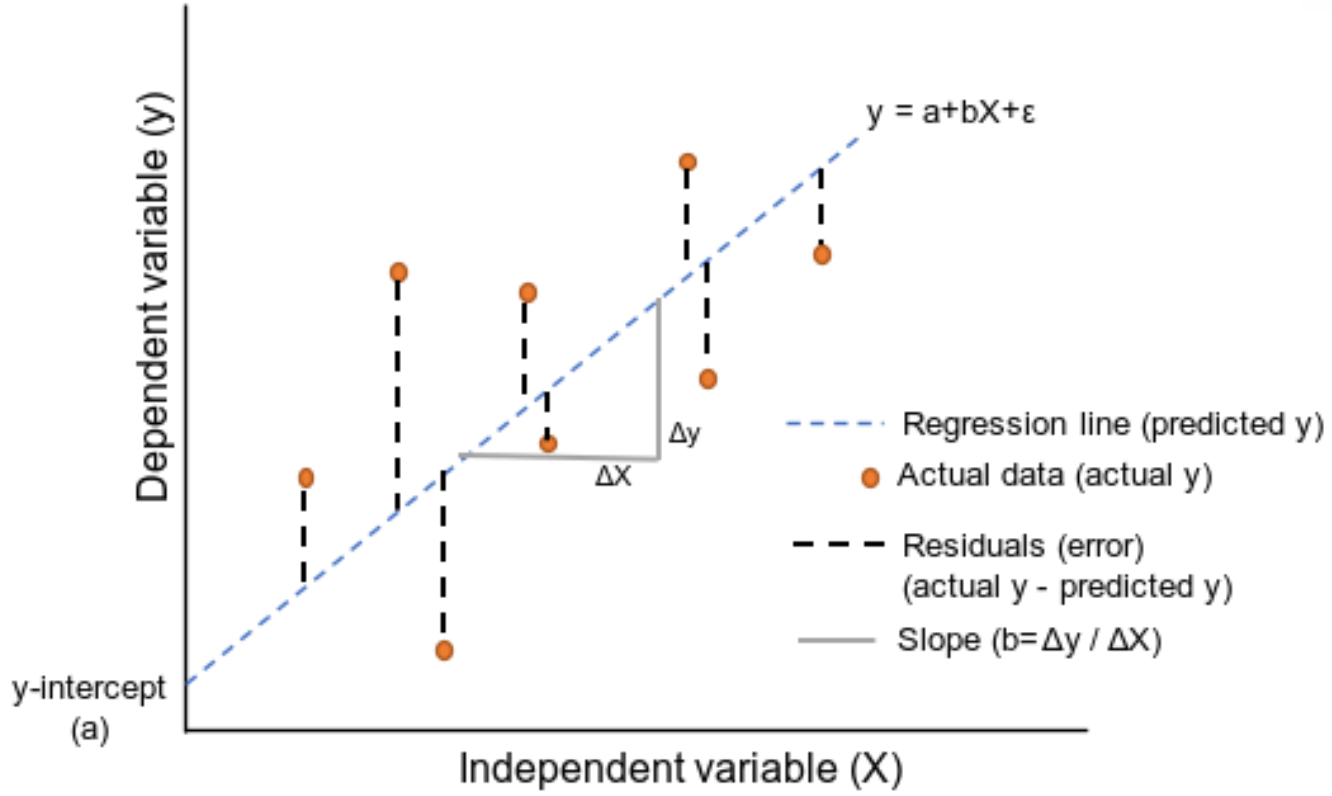
X = 10, 30, 50
Y = 2*10 + 3 =23
Y = 2*30 + 3 = 63
Y = 2*50 + 3 = 103

X	Actual	Predicted
10	25	23
30	60	63
50	100	103

Linear Regression

Basic Concepts

F(x) = mx + c = y = predicted output



$$f(\mathbf{x}) = \sum_{j=1}^{D} w_j x_j + \epsilon = \mathbf{y}$$

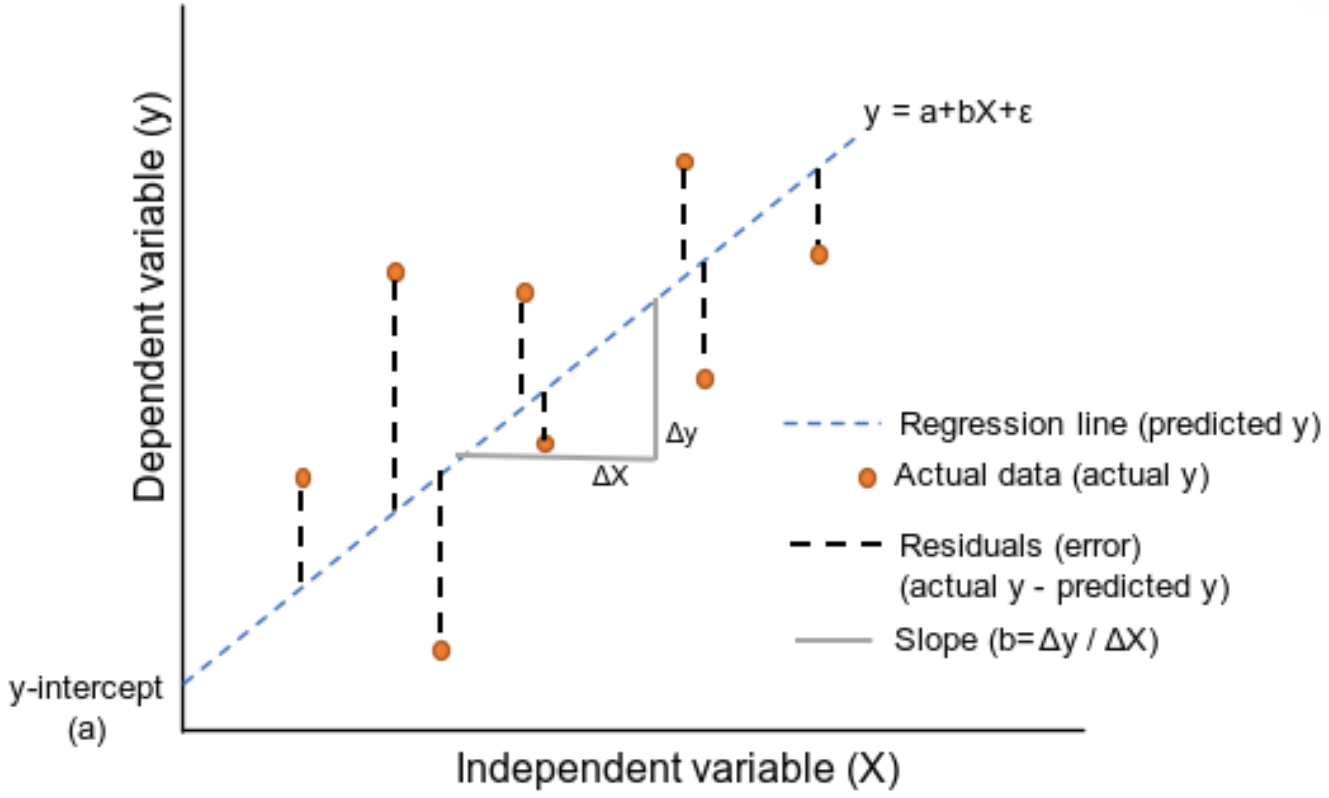
The **model parameters** are:

$$\boldsymbol{\theta} = \{w_0, \ldots, w_n, \sigma\} = \{\mathbf{w}, \sigma\}$$

Linear Regression

Basic Concepts

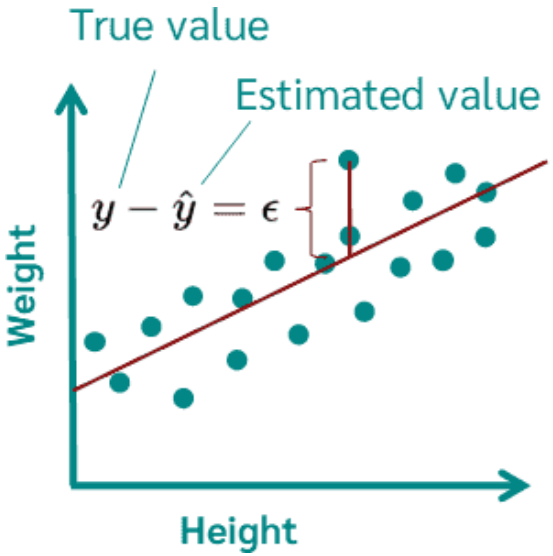
F(x) = mx + c = y = predicted output



$$f(\mathbf{x}) = \sum_{j=1}^{D} w_j x_j + \epsilon = \mathbf{y}$$

The **model parameters** are:

$$\boldsymbol{\theta} = \{w_0, \ldots, w_n, \sigma\} = \{\mathbf{w}, \sigma\}$$



Error epsilon

$$y = b \cdot x + a + \epsilon$$

Formula 01: Slope, $m = \frac{N\sum(xy)-(\sum x)(\sum y)}{N\sum(x^2)-(\sum x)^2}$

Where:

- N is the number of observations.
- $\sum$ denotes the summation symbol.
- x and y are the individual sample points of the independent and dependent variables, respectively.

Linear Regression

Maths

Given Dataset	
x	y
1	2
2	3
3	5
4	4

Plug these values into the formula:

$$m = \frac{4(39)-(10)(14)}{4(30)-(10)^2}$$
$$m = \frac{156-140}{120-100}$$
$$m = \frac{16}{20}$$
$$m = 0.8$$

$$m = \frac{N\sum(xy)-(\sum x)(\sum y)}{N\sum(x^2)-(\sum x)^2}$$

1. Calculate $\sum x$:
   $\sum x = 1 + 2 + 3 + 4 = 10$
2. Calculate $\sum y$:
   $\sum y = 2 + 3 + 5 + 4 = 14$
3. Calculate $\sum xy$:
   $\sum xy = (1 \cdot 2) + (2 \cdot 3) + (3 \cdot 5) + (4 \cdot 4) = 2 + 6 + 15 + 16 = 39$
4. Calculate $\sum x^2$:
   $\sum x^2 = 1^2 + 2^2 + 3^2 + 4^2 = 1 + 4 + 9 + 16 = 30$
5. Calculate N:
   $N = 4$

Read Lecture Notes!

Formula 02: Slope $$m = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sum(x-\bar{x})^2}$$

Explanation of Terms

- N is the number of data points or observations in the dataset.
- $\sum$ denotes summation, meaning that you sum up the values that follow.
- x and y are the individual data points in the dataset for the independent (predictor) and dependent (response) variables.
- $\bar{x}$ and $\bar{y}$ are the mean values of the x and y datasets, respectively. They are calculated as:
  $\bar{x} = \frac{\sum x}{N}$
  $\bar{y} = \frac{\sum y}{N}$

1. Calculate $\bar{x}$ (mean of x):
$$\bar{x} = \frac{\sum x}{N} = \frac{10}{4} = 2.5$$

2. Calculate $\bar{y}$ (mean of y):
$$\bar{y} = \frac{\sum y}{N} = \frac{14}{4} = 3.5$$

Read Lecture Notes!

3. Calculate $\sum(x - \bar{x})(y - \bar{y})$:

$$(1 - 2.5)(2 - 3.5) = (-1.5)(-1.5) = 2.25$$
$$(2 - 2.5)(3 - 3.5) = (-0.5)(-0.5) = 0.25$$
$$(3 - 2.5)(5 - 3.5) = (0.5)(1.5) = 0.75$$
$$(4 - 2.5)(4 - 3.5) = (1.5)(0.5) = 0.75$$

$$\sum(x - \bar{x})(y - \bar{y}) = 2.25 + 0.25 + 0.75 + 0.75 = 4$$

Given Dataset	
x	y
1	2
2	3
3	5
4	4

4. Calculate $\sum(x - \bar{x})^2$:

Read Lecture Notes!

$$(1 - 2.5)^2 = (-1.5)^2 = 2.25$$
$$(2 - 2.5)^2 = (-0.5)^2 = 0.25$$
$$(3 - 2.5)^2 = (0.5)^2 = 0.25$$
$$(4 - 2.5)^2 = (1.5)^2 = 2.25$$

$$\sum(x - \bar{x})^2 = 2.25 + 0.25 + 0.25 + 2.25 = 5$$

Plug these values into the formula:

$$m = \frac{4}{5}$$
$$m = 0.8$$

Note: Both formulas are mathematically **equivalent** and will yield the same result when applied correctly. So, slope m=0.8 (Both Formula). You are allowed to apply any of them.

Given Values

- $\bar{x} = 2.5$
- $\bar{y} = 3.5$
- $m = 0.8$

Formula for Intercept c

$$c = \bar{y} - m\bar{x}$$

Calculation

$$c = 3.5 - (0.8 \times 2.5)$$
$$c = 3.5 - 2$$
$$c = 1.5$$

Let’s See Another Math From Scratch

Linear Regression

Raw Calculation

Slope, $m = \Sigma((x - \bar{x}) * (y - \bar{y})) / \Sigma((x - \bar{x})^2)$

Intercept, $c = \bar{y} - m * \bar{x}$

Where:

x is a data point on the independent variable (x-axis).

y is the corresponding dependent variable (y-axis).

$\bar{x}$ is the mean of the independent variable.

$\bar{y}$ is the mean of the dependent variable.

Feature (X)	Class (Y)
5	50
7	65
4	42
8	76
2	23
10	105
7	?

Linear Regression

Raw Calculation

Based on the linear regression calculation using the provided data points, the slope (m) and y-intercept (c) of the best fit line have been determined. Using this line, the predicted value of Y for X = 7 is approximately 69.95. Here is the updated table with the predicted value:

X	Y
5	50.00
7	65.00
4	42.00
8	76.00
2	23.00
10	105.00
7	69.95

Prediction

Linear Regression

Raw Calculation

To predict the value of Y for $X = 7$ using linear regression, we first need to determine the slope (m) and the y-intercept (c) of the best-fit line. Here are the steps:

Calculate the mean of X and Y (denoted as $\bar{X}$ and $\bar{Y}$).

Use the means to calculate the slope (m) of the best-fit line using the formula:

$$m = \frac{\sum (X-\bar{X})(Y-\bar{Y})}{\sum (X-\bar{X})^2}$$

With the slope (m), calculate the y-intercept (c) of the line using the formula:

$$c = \bar{Y} - m\bar{X}$$

Now that we have both m and c, we can make predictions for Y using the linear equation:

$$Y = mX + c$$

For $X = 7$, this gives us:

$$Y = m(7) + c$$

Final Goal

Let's calculate the means, the slope, and the y-intercept with the provided data points, and then use these to predict Y when $X = 7$.

Given data points:

- $X = [5, 7, 4, 8, 2, 10]$
- $Y = [50, 65, 42, 76, 23, 105]$

Means:

- $\bar{X} = \frac{5+7+4+8+2+10}{6}$
- $\bar{Y} = \frac{50+65+42+76+23+105}{6}$

Linear Regression

Raw Calculation

Using these means, we calculate the slope m and y-intercept c, then use the equation $Y = mX + c$ to predict the value of Y for $X = 7$. Let's go through the calculations.

The calculations for the linear regression with the given data points are as follows:

Calculate the mean of X and Y:

- Mean of X, $\bar{X}$ = 6
- Mean of Y, $\bar{Y}$ = 60.166666666666664

Calculate the slope (m) of the best-fit line:

- $m = \frac{\sum (X-\bar{X})(Y-\bar{Y})}{\sum (X-\bar{X})^2}$
- $m = 9.785714285714286$

Calculate the y-intercept (c) of the line:

- $c = \bar{Y} - m\bar{X}$
- $c = 1.452380952380942$

Predict the value of Y for $X = 7$:

- Using the equation $Y = mX + c$
- $Y = 9.785714285714286 \times 7 + 1.452380952380942$
- $Y = 69.95238095238093$

Python Implementation

Step 1: Calculate the means of X and Y

```
mean_x = df['X'].mean()
mean_y = df['Y'].mean()
```

Step 2: Calculate the slope (m) using the formula

```
numerator_m = ((df['X'] - mean_x) * (df['Y'] - mean_y)).sum()
denominator_m = ((df['X'] - mean_x)**2).sum()
slope_m = numerator_m / denominator_m
```

Step 3: Calculate the y-intercept (c) using the formula

```
intercept_c = mean_y - (slope_m * mean_x)
```

Step 4: Predict the value for X = 7 using the regression line equation Y = mX + c

```
x_predict = 7
y_predict = (slope_m * x_predict) + intercept_c
```

Show the calculated slope, intercept, and the predicted value of Y

```
slope_m, intercept_c, y_predict
```

Linear Regression

Best Fit Line with Sklearn

Data Set

	x	y
0	5	50
1	7	65
2	4	42
3	8	76
4	2	23
5	10	105

Using Sklearn Library

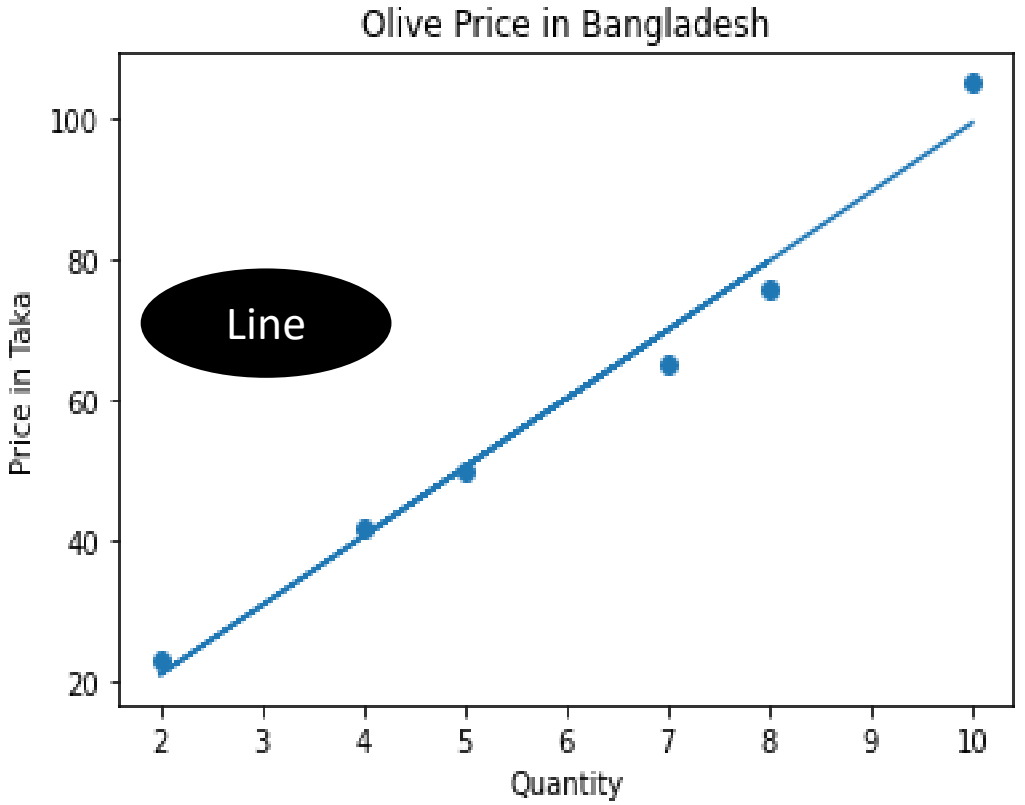
Value of M & C

```
reg.coef_
```

```
array([9.78571429])
```

```
reg.intercept_
```

```
1.4523809523809703
```



Ref: Click the Link

Linear Regression

Visual Prediction

Data Set

	x	y
0	5	50
1	7	65
2	4	42
3	8	76
4	2	23
5	10	105

Value of M & C

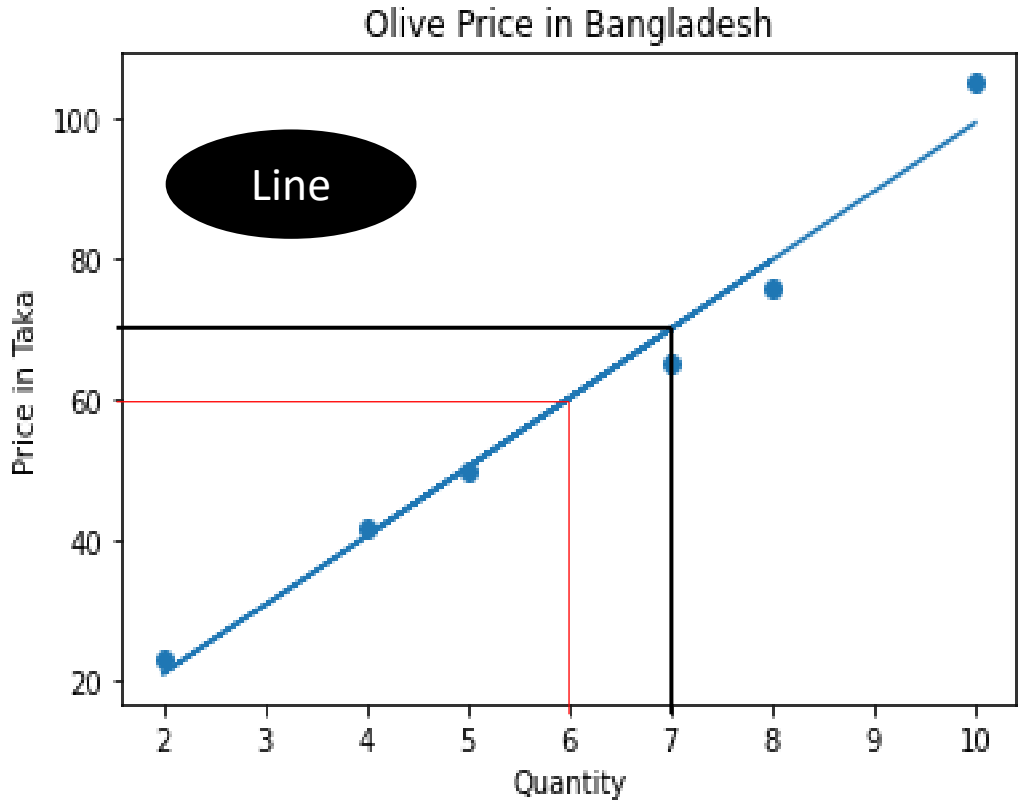
```
reg.coef_
```

```
array([9.78571429])
```

```
reg.intercept_
```

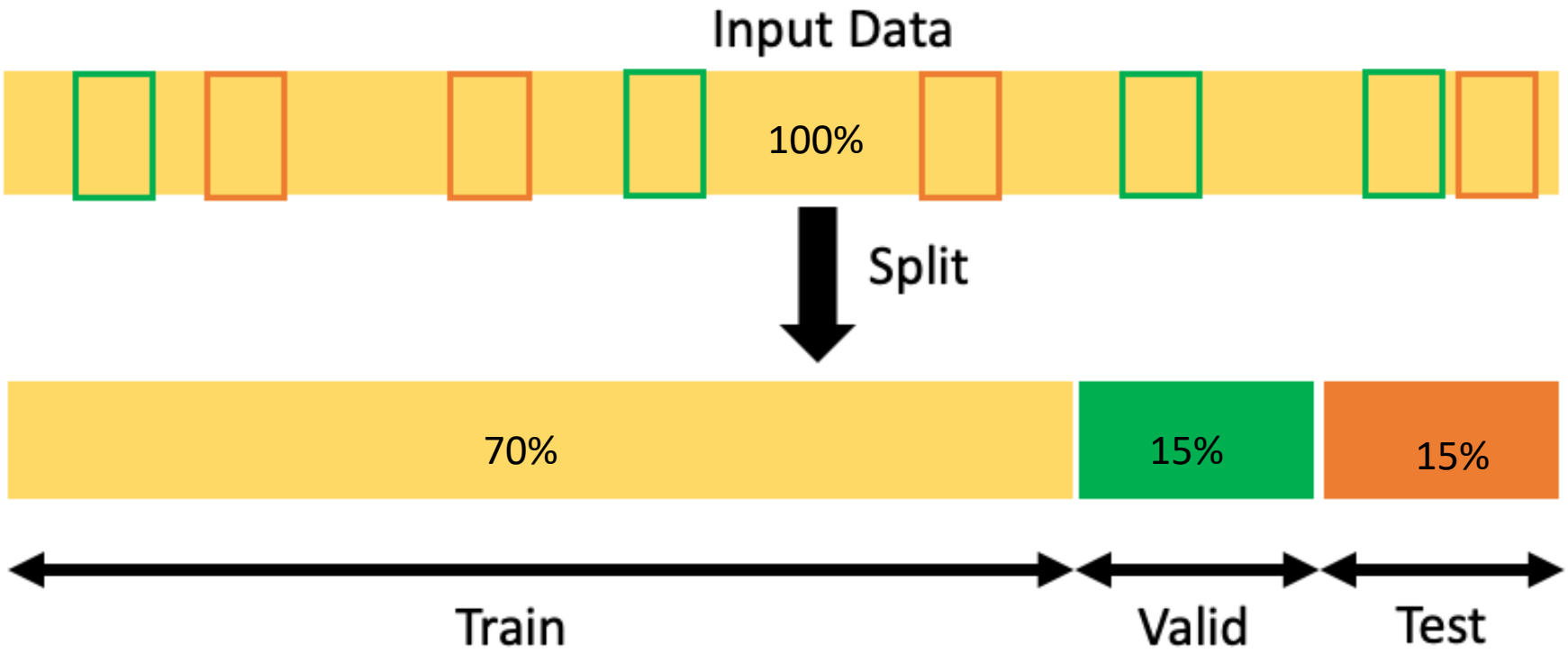
```
1.4523809523809703
```

Ref: Click the Link



Data Splitting Strategies

Splitting Dataset

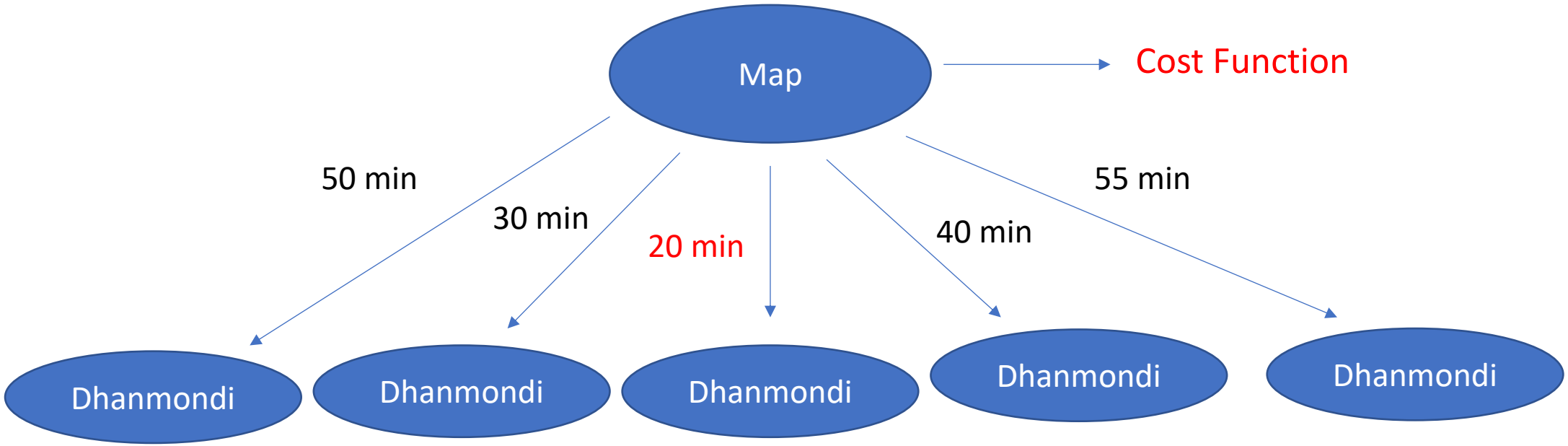


Splitting Dataset

Strategy	Training Set (%)	Validation Set (%)	Test Set (%)	Description
70-30 Split	70%	0%	30%	Standard split where 70% is used for training and 30% for testing.
80-20 Split	80%	0%	20%	80% of data for training, and 20% for testing.
60-20-20 Split	60%	20%	20%	60% for training, 20% for validation, 20% for final testing.
K-Fold Cross-Validation	Varies	Varies	Varies	Data is split into *k* subsets. Each subset is used as a test set once, while the other *k-1* subsets are used for training.
Stratified Split	Depends on user	Depends on user	Depends on user	Ensures that the split maintains the proportion of class labels across the datasets. Useful for imbalanced data.

Lose/Cost Function

The cost function is a function, which is associates a cost with a **decision.**



1. Loss (or Error) for a Single Sample:

- When you calculate the difference between the actual value and the predicted value for a single data point, it's generally referred to as a "loss" or "error" for that specific data point.
- This term is used to describe the discrepancy between the prediction and the true value for a single instance.

2. Cost (or Loss) for the Entire Dataset:

- When you calculate the average or total of these losses/errors across the entire dataset, it's often referred to as the "cost" or "loss" for the dataset.
- The term "cost" or "loss" is used to describe the overall quality of the model's predictions for the entire dataset.

Linear Regression

Residuals

Residual = Observed Value - Predicted Value

x	y	$\hat{y}$ (Predicted)	Residuals ($y - \hat{y}$)
1	2	2.3	-0.3
2	3	3.1	-0.1
3	5	3.9	1.1
4	4	4.7	-0.7

Read Lecture Notes!

- Residual for $x = 1$: $2 - 2.3 = -0.3$
- Residual for $x = 2$: $3 - 3.1 = -0.1$
- Residual for $x = 3$: $5 - 3.9 = 1.1$
- Residual for $x = 4$: $4 - 4.7 = -0.7$

Predicted Result:

- Pred = 0.8 * X + 1.5
- Y_pred = (0.8*1)+ 1.5 = 2.3
- Y_pred = (0.8*2)+ 1.5 = 3.1
- Y_pred = (0.8*3)+ 1.5 = 3.9
- Y_pred = (0.8*4)+ 1.5 = 4.7

1. **Mean Absolute Error (MAE):**

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i|$$

2. **Mean Squared Error (MSE):**

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

3. **Root Mean Squared Error (RMSE):**

$$\text{RMSE} = \sqrt{\text{MSE}}$$

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n}|Y_i - \hat{Y}_i|$$

1. **MAE Calculation:**

$$\text{MAE} = \frac{1}{4}\left(|2 - 2.3| + |3 - 3.1| + |5 - 3.9| + |4 - 4.7|\right)$$

$$\text{MAE} = \frac{1}{4}\left(0.3 + 0.1 + 1.1 + 0.7\right)$$

$$\text{MAE} = \frac{1}{4} \times 2.2$$

$$\text{MAE} = 0.55$$

$$\text{MSE} = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2$$

2. **MSE Calculation:**

$$\text{MSE} = \frac{1}{4}\left((2-2.3)^2 + (3-3.1)^2 + (5-3.9)^2 + (4-4.7)^2\right)$$

$$\text{MSE} = \frac{1}{4}\left((-0.3)^2 + (-0.1)^2 + (1.1)^2 + (-0.7)^2\right)$$

$$\text{MSE} = \frac{1}{4}\left(0.09 + 0.01 + 1.21 + 0.49\right)$$

$$\text{MSE} = \frac{1}{4} \times 1.8$$

$$\text{MSE} = 0.45$$

3. **RMSE Calculation:**

$$\text{RMSE} = \sqrt{\text{MSE}}$$

$$\text{RMSE} = \sqrt{0.45}$$

$$\text{RMSE} \approx 0.67$$

NOTE: To verify the results, please read the Python lecture notes.

We are Looking for your Questions!

Thanks for your patience!
Let’s Implement with Python