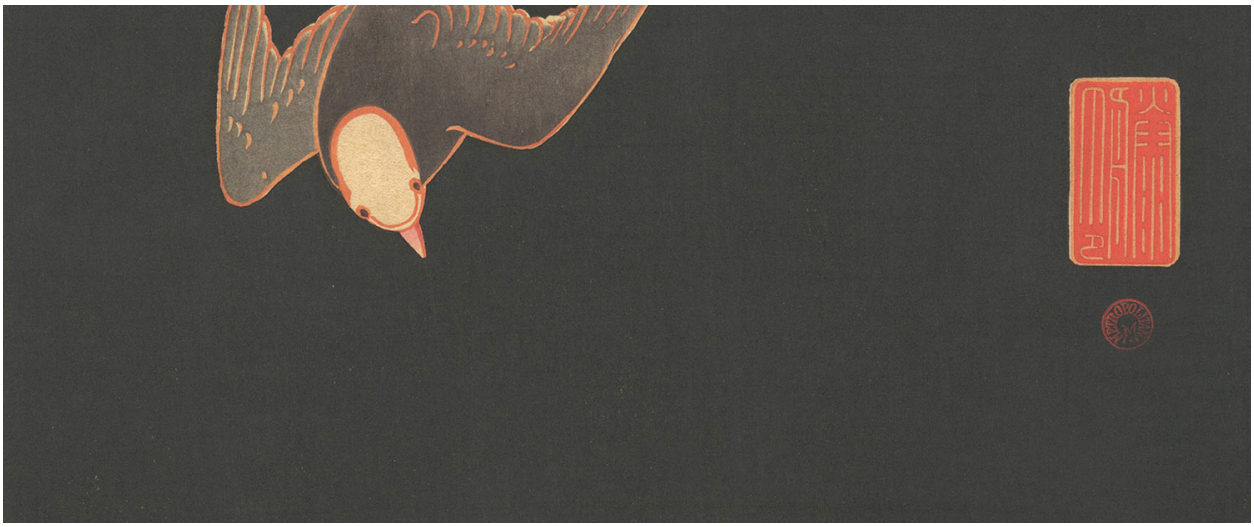

Linear Regression

Introduction

A statistical technique method used to define a linear relationship between a dependent variable and an independent variable. There are some notations given the dependent and independent variable as follows,

$$y = dependent\ variable$$
$$x = independent\ variable$$

Instead of using ‘linear regression method’ we will use ‘linear regression model’ which is a standard model defined in machine learning terminology. The main aim of this model it to fit a perfect straight line through a set of data points. This straight line is used to predict new value of y for given values of x. In this straight line relationship there are errors which are also distributed evenly in the model.

Little Math behind the linear regression

When we talk about a line we usually visualise a 2 dimensional graph with X and Y coordinates. When we plot the data in this graph we don’t no how this data is spread but we can say that it’s not perfectly spread. So, to fit a perfectly straight line we need to

transform the line by moving it up or down between the data to fit the best. For this we need to know the slope / slope intercept of the line. Which is found using the formula below.

When we are modelling the linear regression there are key concepts which we need to know. They are,

- Regression line
- Slope (m)
- Slope intercept (b)
- Predicted value (y)
- Residual / Error(e)
- Sum of error's squared(SE)
- What is a cost function then?

The Regression line

$$y = mx + b$$

Where m is the slope and b is the slope intercept of the line which we are transforming.

Now, let's break down how the m and b values are calculated. It's easy don't get scared by the equations.

The slope(m)

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

Here the slope m represents the change of y with respect to change is x. And, $\overline{x}\ and\ \overline{y}$ are the means of x and y respectively. This is usually used to calculate sum of their differences from their means.

Slope Intercept(b)

$$b \ = \ \overline{y} - m\overline{x}$$

This equation gives the where the line crosses the y axis by the values of slope(m) and means($\overline{x}$ and $\overline{y}$).

Once we get to know the values of m and b we substitute it in the regression line formula to get our predictions.

$$Regression\ line: \ y = mx + b$$
$$Predicted: \ \overline{y} = mx + b$$

Error(e)

We got to know how to model a linear regression to predict values but how do we know how well our model is performing? How do we calculate the error in our predictions? That is where the residual also known as (error) comes. If we want to know how correct is something, we usually subtract the actual value with the value we got thus giving us the error. Same thing here also,

$$e \ = \ y - \hat{y}$$

where y is the actual value and the $\hat{y}$ is predicted value for each point.

Sum of Squared residuals(SSE)

$$SSE = \sum e^2$$

If we expand the above formula it would look something like this,

$$SSE = \sum (yi - \hat{y}i)^2$$

Now what's the difference between $y - \hat{y}$ and $(yi - \hat{y}i)$?

The formula of Error(e) $e = y - \hat{y}$ is only for one value of y. In reality we will be working with more than one value of x so i indicates the i^{th} data point.And, to get the value in positive we square the difference value.

R-Squared(R^2)

R^2 is nothing but the measure of how god it fits the regression line with the actual data. It is calculated using the formula below,

$$R^2 = 1 - SSE/SST$$

Here,

SSE is Sum of Squared Errors(Residuals) which we learnt in previous point.

SST is Total sum of Squares. This shows the total variance with respect to y variable.

The SST is calculated as below,

$$SST = \sum (yi - \overline{y})^2$$

Here again the yi is the data point at i^{th} value and $\overline{y}$ is the mean of y. The value of R^2 lies between 0 to 1 can be represented as: $0 \leq R^2 \leq 1$. In simple terms,

If $R^2 = 0$ the variance is 0% and model doesn't fit well.

If $R^2 = 1$ the variance is 100% and model fits perfectly well.

Cost function.

Cost function is also known as loss function which is used to quantify the error of a model's predictions over a set of data. The goal os model training is to minimise the cost function to make better predictions.

In our case the cost functions are the ***Sum squares residuals*** and R^2 ***error.***

The code

```
import statistics

class LinearRegression:

  def __init__(self):
```

```
    self.m = None
    self.b = None

  def fit(self, X, Y):
    """
    Fit linear regression model to data
    - Calculate slope (m) and intercept (b)
    - Calculate R-squared score
    """
    # Calculate statistics
    x_mean = statistics.mean(X)
    y_mean = statistics.mean(Y)
    x_squared_mean = statistics.mean([x**2 for x in X])
    xy_mean = statistics.mean([x*y for x,y in zip(X,Y)])

    # Calculate slope (m) and intercept (b)
    self.m = (x_mean * y_mean - xy_mean) / (x_mean**2 - x_squared_mean)
    self.b = y_mean - self.m * x_mean

    # Calculate R-squared
    y_predicted = [self.m*x + self.b for x in X]
    ss_res = sum((y - y_pred)**2 for y, y_pred in zip(Y, y_predicted))
    ss_tot = sum((y - y_mean)**2 for y in Y)
    self.r_squared = 1 - (ss_res / ss_tot)

    return self

  def predict(self, X):
    """
    Make predictions on data using fitted model
    """
    y_predicted = [self.m*x + self.b for x in X]
    return y_predicted

  def score(self):
    """
    Return R-squared score of model fit
    """
    return self.r_squared

if __name__ == "__main__":

  # Sample data
  X = [1, 2, 3, 4, 5]
  Y = [2, 4, 6, 8, 10]

  # Fit model
  model = LinearRegression().fit(X, Y)

  # Make predictions
  predictions = model.predict([7, 8, 9, 10])

  # Score model
  r_squared = model.score()
```

```
print("Predictions:", predictions)
print("R-squared:", r_squared)
```