

Multivariate Regression

- Multivariate Regression is a supervised machine learning algorithm involving multiple data variables for analysis.
- A Multivariate regression is an extension of multiple regression with one dependent variable and multiple independent variables.
- Based on the number of independent variables, we try to predict the output.

$$y = m_1.x_1 + m_2.x_2 + \dots + c$$

y: dependent variable

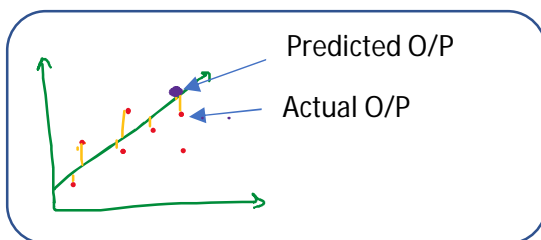
x1, x2: multiple independent variables.

M1, m2: slope of independent variables.

C: intercept.

- **Cost Function:**
 - The cost function is a function that allows a cost to samples when the model differs from observed data.
 - This equation is the sum of the square of the difference between the predicted value and the actual value divided by twice the length of the dataset.
 - A smaller mean squared error implies a better performance. Here, the cost is the sum of squared errors.

$$MSE = \frac{1}{2m} \sum (h_{\theta}(x)^{(i)} - y^i)^2$$



- **Steps of Multivariate Regression analysis:**
 - Feature selection.
 - Normalizing features.
 - Select loss function and hypothesis.
 - Set hypothesis parameter.
 - Minimize the loss function.
 - Test the hypothesis function.

Polynomial Regression

- Polynomial regression is a special case of linear regression where we fit a polynomial equation on the data with a curvilinear relationship between the target variable and the independent variables.

$$Y = m_1x_1 + m_2x_2^2 + m_3x_3^3 + m_4x_4^n + \dots + c$$

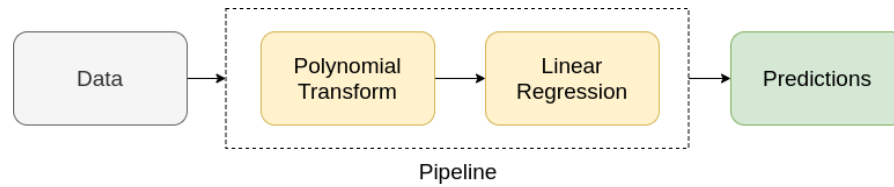
Y: Dependent variable

M1, m2...: slope of independent variable.

X1, x2...: independent variables.

C: intercept.

N: degree of polynomial.



Code:

```
# importing libraries for polynomial transform
from sklearn.preprocessing import PolynomialFeatures
# for creating pipeline
from sklearn.pipeline import Pipeline
# creating pipeline and fitting it on data
Input=[('polynomial', PolynomialFeatures(degree=2)), ('model', LinearRegression())]
pipe=Pipeline(Input)
pipe.fit(x.reshape(-1, 1), y.reshape(-1, 1))
```