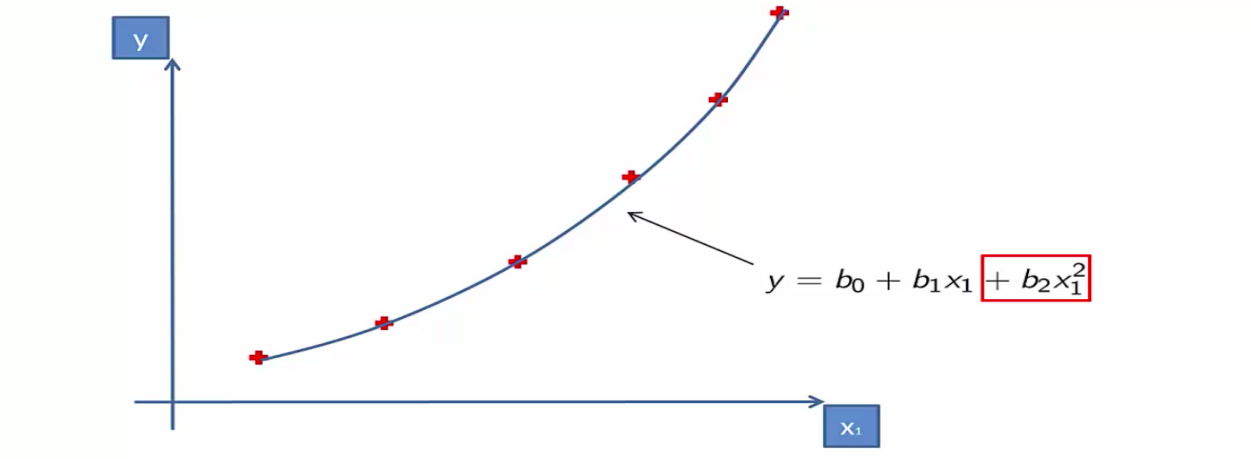
Polynomial Linear Regression

Polynomial regression is a form of regression analysis in which the relationship between the independent variable x and the dependent variable y is modeled as an nth degree polynomial in x.

Polynomial Linear Regression

$$y = b_0 + b_1 x_1 + b_2 x_1^2 + \ldots + b_n x_1^n$$

We use a polynomial regression when the **straight line** (like in Linear Regression) doesn't fit well our obervations and we want to obtain parabolic effect:



Polynomial Regression is a form of linear regression in which the relationship between the independent variable x and dependent variable y is modeled as an *nth* degree polynomial. Polynomial regression fits a nonlinear relationship between the value of x and the corresponding conditional mean of y, denoted E(y |x)

Why Polynomial Regression:

- There are some relationships that a researcher will hypothesize is curvilinear. Clearly, such type of cases will include a polynomial term.
- Inspection of residuals. If we try to fit a linear model to curved data, a scatter plot of residuals (Y axis) on the predictor (X axis) will have patches of many positive residuals in the middle. Hence in such situation it is not appropriate.
- An assumption in usual multiple linear regression analysis is that all the independent variables are independent. In polynomial regression model, this assumption is not satisfied.

Uses of Polynomial Regression:

These are basically used to define or describe non-linear phenomenon such as:

- Growth rate of tissues.
- Progression of disease epidemics

- Distribution of carbon isotopes in lake sediments

Advantages of using Polynomial Regression:

- Broad range of function can be fit under it.
- Polynomial basically fits wide range of curvature.
- Polynomial provides the best approximation of the relationship between dependent and independent variable.

Disadvantages of using Polynomial Regression

- These are too sensitive to the outliers.
- The presence of one or two outliers in the data can seriously affect the results of a nonlinear analysis.
- In addition there are unfortunately fewer model validation tools for the detection of outliers in nonlinear regression than there are for linear regression.


```
import numpy as np
import matplotlib.pyplot as plt
import pandas as pd
```

```
dataset = pd.read_csv("position.csv")
dataset
```

	Position	Level	Salary
0	Business Analyst	1	45000
1	Junior Consultant	2	50000
2	Senior Consultant	3	60000
3	Manager	4	80000
4	Country Manager	5	110000
5	Region Manager	6	150000
6	Partner	7	200000
7	Senior Partner	8	300000
8	C-level	9	500000
9	CEO	10	1000000

```
[10] X = dataset.iloc[:, 1:2].values
     y = dataset.iloc[:, 2].values
```

```
[16] # Fitting Linear Regression to the dataset
     from sklearn.linear_model import LinearRegression
     lin = LinearRegression()

     lin.fit(X, y)
```

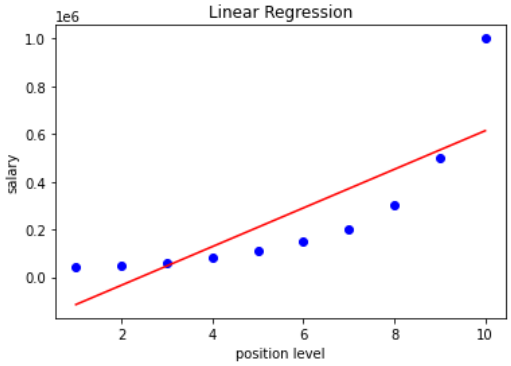
```
     LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

```
[17] # Fitting Polynomial Regression to the dataset
     from sklearn.preprocessing import PolynomialFeatures

     poly = PolynomialFeatures(degree = 4)
     X_poly = poly.fit_transform(X)

     poly.fit(X_poly, y)
     lin2 = LinearRegression()
     lin2.fit(X_poly, y)
```

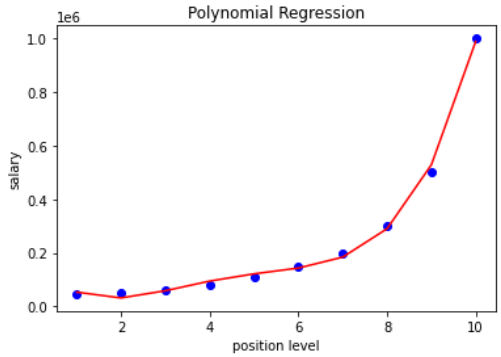


```
     LinearRegression(copy_X=True, fit_intercept=True, n_jobs=None, normalize=False)
```

```
[19] plt.scatter(X, y, color = 'blue')

     plt.plot(X, lin.predict(X), color = 'red')
     plt.title('Linear Regression')
     plt.xlabel('position level')
     plt.ylabel('salary')

     plt.show()
```



```
plt.scatter(X, y, color = 'blue')

plt.plot(X, lin2.predict(poly.fit_transform(X)), color = 'red')
plt.title('Polynomial Regression')
plt.xlabel('position level')
plt.ylabel('salary')

plt.show()
```




```
[24] lin.predict([[7]])
```



```
array([370818.18181818])
```