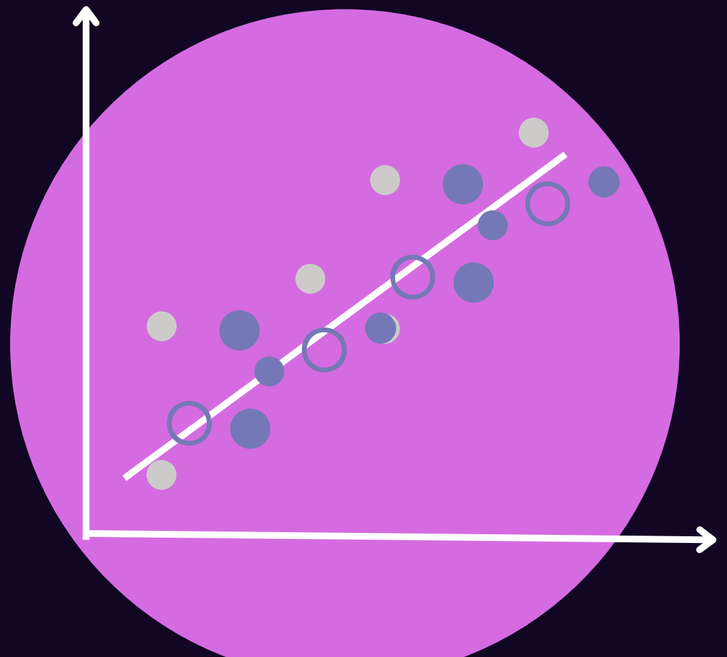
POLYNOMIAL REGRESSION

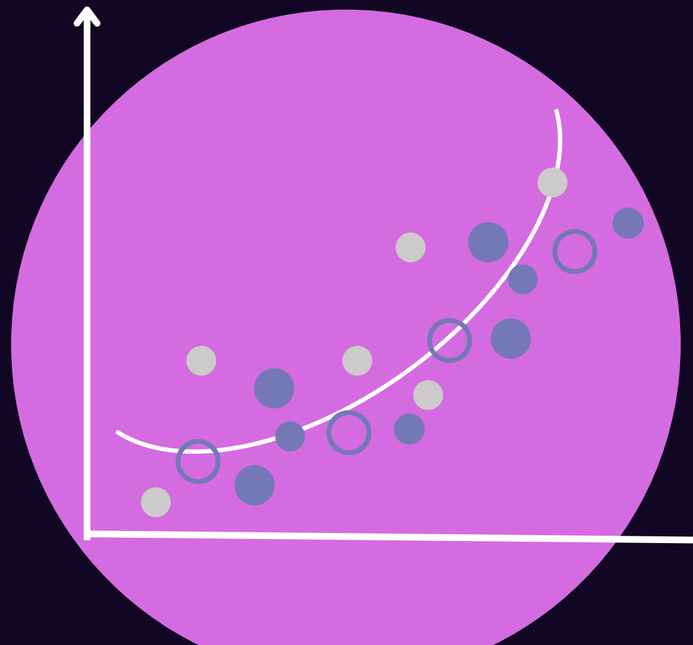
POLYNOMIAL REGRESSION

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)

SIMPLE LINEAR REGRESSION)



POLYNOMIAL REGRESSION



SIMPLE REGRESSION

It is a statistical model that describes the relationship between one independent variable (X) and one dependent variable (Y) using a straight line.

Equation:

$$Y=a+bX+e$$

a = intercept (where the line crosses the Y-axis)
b = slope (how much Y changes when X increases by 1)
e = error term (random noise)

Use it when the relationship between X and Y is linear.

POLYNOMIAL REGRESSION

It is an extension of linear regression that models the relationship between X and Y using a polynomial equation (i.e., includes powers like X^2, X^3, etc.).

Equation:

$$Y=a+b1X+b2X^2+b3X^3+\ldots+e$$

- This model captures non-linear relationships between X and Y

Use it when the data shows a curved trend that a straight line can't fit.

POLYNOMIAL REGRESSION

- Datasets are arranged in a non-linear fashion, then we should use the Polynomial Regression model.
- A Polynomial Regression algorithm is also called Polynomial Linear Regression.
- It does not depend on the variables, instead, it depends on the coefficients, which are arranged in a linear fashion.

Simple Linear Regression equation:

$$Y=b0+b1X$$

Multiple Linear Regression equation:

$$Y=b0+b1X1+b2X2+b3X3+...+bnXn$$

Polynomial Regression equation:

$$Y=b0+b1X+b2X^2+b3X^3+...+bnX^n$$

POLYNOMIAL REGRESSION

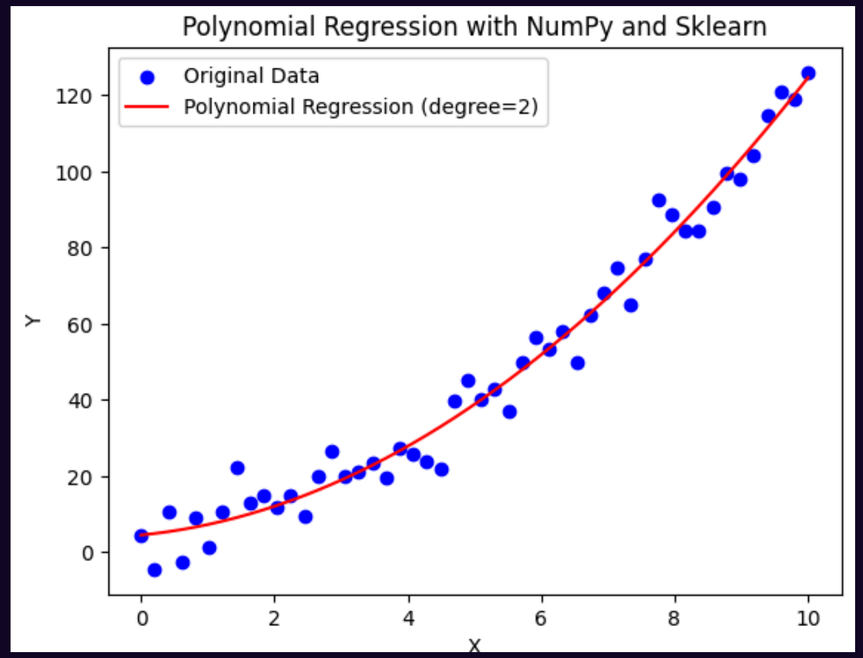
EXAMPLE

```
import numpy as np
import matplotlib.pyplot as plt
from sklearn.linear_model import LinearRegression
from sklearn.preprocessing import PolynomialFeatures


x = np.linspace(0, 10, 50).reshape(-1, 1)
y = 3 + 2*x + x**2 + np.random.randn(50, 1)*5
poly = PolynomialFeatures(degree=2)
x_poly = poly.fit_transform(x)
model = LinearRegression()
model.fit(x_poly, y)

y_pred = model.predict(x_poly)
```

EXAMPLE CONT....

VISUALIZATION

```
plt.scatter(x, y, color='blue', label='Original Data')
plt.plot(x, y_pred, color='red', label='Polynomial
Regression (degree=2)')
plt.legend()
plt.title('Polynomial Regression with NumPy and
Sklearn')
plt.xlabel('X')
plt.ylabel('Y')
plt.show()
```



THANK YOU