

# Polynomial Regression

- Polynomial regression is an extension of Linear Regression.

## 1 Polynomial Regression क्या है?

- सबसे पहले समझते हैं **Linear Regression**:
- जब हमारे डेटा का रिश्ता **सीधी रेखा (Straight Line)** जैसा होता है।

फॉर्मूला:

$$y = mx + c$$

- लेकिन **असली दुनिया** में ज़्यादातर डेटा सीधी लाइन में नहीं होता।
- Problem:**
- अगर डेटा **Curve (मोड़दार)** है तो Linear Regression fail हो जाती है ✗ यहीं पर **Polynomial Regression** काम आता है।
- Polynomial Regression **अलग algorithm नहीं है Linear Regression + Feature Engineering है**
- Trick:** हम input feature को power में बदल देते हैं:  $x = [1, 2, 3]$  तो Polynomial (degree = 2) में बदल जाएगा:
- $x, x^2$

# Work of Polynomial Regression

- Polynomial Regression क्या करता है?
- डेटा को **Curve (U-shape, S-shape)** में फिट करता है, Linear model को **Non-linear relationship** सीखने लायक बनाता है
- फॉर्मूला:
$$y = b_0 + b_1x + b_2x^2 + b_3x^3 + \dots$$
- 👉 यहाँ  $x^2, x^3$  जैसे terms आते हैं
- 👉 इसी वजह से इसे **Polynomial** कहते हैं



# Why we use Polynomial Regres.

- Polynomial Regression क्यों use करते हैं?
- Data graph **सीधा नहीं है**, Relation **curve** में है, Linear regression **underfit** कर रही है
- **Real Life Examples:**
  - **Salary vs Experience** (शुरुआत में salary तेज़ी से बढ़ती है, फिर slow हो जाती है) **Car**
  - **Speed vs Fuel Consumption**
  - **Temperature vs Chemical Reaction**



# How we can use in sklearn

- sklearn में **direct Polynomial Regression नहीं होता**
- हम use करते हैं: PolynomialFeatures , LinearRegression
- **Step 1: Libraries import**

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

# How we can use in sklearn

- **Step 2: Data बनाना**

```
X = np.array([1, 2, 3, 4, 5]).reshape(-1, 1)
y = np.array([1, 4, 9, 16, 25])
```

- **Step 3: Polynomial Features बनाना**

```
poly = PolynomialFeatures(degree=2)
X_poly = poly.fit_transform(X)
```

अब  $X_{\text{poly}}$  में:  $[1, x, x^2]$

- **Step 4: Linear Regression Model**

```
model = LinearRegression()
model.fit(X_poly, y)
```



# How we can use in sklearn

- Step 5: Prediction

```
y_pred = model.predict(X_poly)
```

- Step 6: Graph देखना

```
plt.scatter(X, y, color='red')  
plt.plot(X, y_pred, color='blue')  
plt.show()
```

-  Blue line = Polynomial curve
-  Red dots = Original data



# Benefits of Polynomila Reg.

- Curve data को अच्छी तरह fit करता है
- Polynomial regression सिर्फ **छोटे और clean data** पर use करें

# What

- **Machine Learning** = Data + Learning + Prediction



# What

- **Machine Learning** = Data + Learning + Prediction

# What

- **Machine Learning** = Data + Learning + Prediction



# What

- **Machine Learning** = Data + Learning + Prediction



# What

- **Machine Learning** = Data + Learning + Prediction

# What

- **Machine Learning** = Data + Learning + Prediction

# What

- **Machine Learning** = Data + Learning + Prediction