Feature Engineering

Feature Engineering Definition:

- Feature engineering is the process of transforming raw data into meaningful input variables (features) that improve the performance of machine learning models.
- It involves creating, modifying, or selecting features to better capture the patterns in the data.

Key Concepts in Feature Engineering:

- Feature Creation → Making new features from existing data (e.g., extracting year from a date column).
- Feature Transformation → Changing the scale, format, or distribution of features (e.g., log transformation).
- Feature Encoding → Converting categorical data into numerical form (e.g., one-hot encoding).
- Feature Scaling → Normalizing or standardizing numerical features.
- Feature Selection → Choosing the most important features to reduce dimensionality and improve model efficiency.

Applications

- Improving model accuracy by providing more informative features.
- Reducing overfitting by removing irrelevant features.
- Enabling algorithms to interpret categorical and text data.
- Handling skewed data distributions for better model training.

***Feature Engineering* Example:**

```
# Creating New Features
# Extracting year from a date column
df['Year'] = pd.to_datetime(df['Order_Date']).dt.year


# One-Hot Encoding
pd.get_dummies(df, columns=['Category'], drop_first=True)


# Label Encoding
from sklearn.preprocessing import LabelEncoder
encoder = LabelEncoder()
df['Category_Code'] = encoder.fit_transform(df['Category'])


# Scaling Numerical Features
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
```

```
13. df['Sales_Scaled'] = scaler.fit_transform(df[['Sales']])


14. # Binning Continuous Variables
15. df['Age_Group'] = pd.cut(df['Age'], bins=[0, 18, 35, 60, 100], labels=['Teen', 'Young Adult',
    'Adult', 'Senior'])


16. # Handling Skewness with Log Transformation
17. import numpy as np
18. df['Log_Sales'] = np.log1p(df['Sales'])


19. # Selecting Important Features
20. from sklearn.feature_selection import SelectKBest, f_classif
21. X_new = SelectKBest(score_func=f_classif, k=5).fit_transform(X, y)
```