📘 Standardization (Z-score Normalization) – Notes

✅ What is Standardization?

Standardization is a scaling technique that transforms data to have:

- **Mean = 0**
- **Standard deviation = 1**

This is also called **Z-score normalization**.

📌 Why is Standardization Important?

Standardization is essential because many **machine learning algorithms** (especially those using distance, gradient descent, or regularization) **assume features are on the same scale**.

Without standardization:

- Features with large scales dominate learning.
- Model performance may degrade.

⚙️ Formula

z=x−μσz = \frac{x - \mu}{\sigma}

Where:

- xx = original value
- μ\mu = mean of the feature
- σ\sigma = standard deviation of the feature

📊 When to Use Standardization?

✅ Use standardization when:

- Data is **normally distributed** (or close to it)
- Algorithms used are **sensitive to scale** like:

Algorithm Type	Examples
Distance-based	KNN, K-Means, SVM
Gradient-based	Logistic Regression, Neural Networks
Regularized	Ridge, Lasso Regression
PCA / LDA	Affected by scale

❌ When Not to Use?

- Tree-based algorithms (e.g., Decision Tree, Random Forest, XGBoost) **don't require standardization**.

🛠️ How to Perform Standardization in Python?

✅ Using Scikit-learn

```
from sklearn.preprocessing import StandardScaler


scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)  # X can be a DataFrame or NumPy array
```

✅ Manually using Pandas

```
X_standardized = (X - X.mean()) / X.std()
```

🧠 Notes to Remember

- **Standardization does not reduce the effect of outliers**, unlike **robust scaling**.
- Always **fit the scaler on training data**, then **transform both training and test data** using the same scaler.

```
scaler.fit(X_train)
X_train_scaled = scaler.transform(X_train)
X_test_scaled = scaler.transform(X_test)
```

✅ Example

Original Data (Age):

[20, 22, 24, 26, 28]

After Standardization:

[-1.41, -0.71, 0, 0.71, 1.41]

Mean = 0, Std Dev = 1 ✅

🔍 Standardization vs Normalization

Feature	Standardization	Min-Max Normalization
Scale	Mean = 0, Std = 1	Range [0, 1]
Formula	(x−μ)/σ(x - \mu)/\sigma	(x−min)/(max−min)(x - min)/(max - min)
Affected by outliers	Yes	Yes
Use Case	Most ML models	Neural networks (sometimes)

🧪 Checking after Standardization

```
print(X_train_scaled.mean())   # ~ 0
print(X_train_scaled.std())    # ~ 1
```

Test set may not have exact mean = 0 or std = 1. That's normal!

✅ Real-world Dataset for Practice

You can apply standardization to:

- **Social_Network_Ads.csv**
  Columns: Age, EstimatedSalary, Target: Purchased
- **Iris Dataset**
  Standardize numeric features (sepal/petal length & width)
