

1. Introduction to Classification

- Classification is a supervised machine learning technique.
- It predicts a category/label based on input features.
- Real-world examples: spam detection, disease prediction, image recognition.

Basic Concepts:

- Features: Input variables.
- Labels: Output class.
- Train-Test Split: Training data teaches the model; testing evaluates it.
- Evaluation Metrics: Accuracy, Confusion Matrix, Classification Report.

2. Algorithms Used:

- Logistic Regression
- K-Nearest Neighbors (KNN)
- Support Vector Machine (SVM)
- Decision Tree
- Random Forest
- Naive Bayes
- Gradient Boosting (Optional)

3. Full Combined Code (All Algorithms)

```
```python
import numpy as np
import pandas as pd
from sklearn.datasets import load_iris
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report

from sklearn.linear_model import LogisticRegression
from sklearn.neighbors import KNeighborsClassifier
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.naive_bayes import GaussianNB

iris = load_iris()
X = iris.data
y = iris.target
names = iris.target_names

X_train, X_test, y_train, y_test = train_test_split(
 X, y, test_size=0.2, random_state=42, stratify=y)

scaler = StandardScaler()
X_train_s = scaler.fit_transform(X_train)
X_test_s = scaler.transform(X_test)

models = {
```

```
"Logistic Regression": LogisticRegression(),
"KNN": KNeighborsClassifier(n_neighbors=5),
"SVM": SVC(),
"Decision Tree": DecisionTreeClassifier(),
"Random Forest": RandomForestClassifier(),
"Naive Bayes": GaussianNB()
}
```

```
for name, model in models.items():
 model.fit(X_train_s, y_train)
 preds = model.predict(X_test_s)
 acc = accuracy_score(y_test, preds)
 print(f"{name} Accuracy: {acc*100:.2f}%")
 print(classification_report(y_test, preds))
 print("-----")
 ``
```

#### 4. Conclusion

- Iris dataset is useful for understanding classification.
- Different algorithms give different accuracy levels.
- Random Forest & SVM generally perform best.