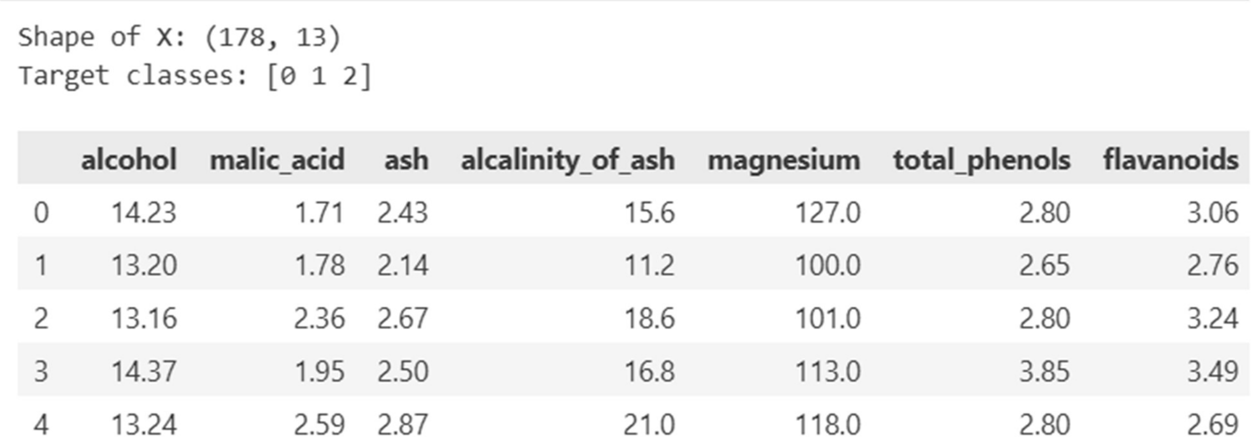
WEEK 6 ASSIGNMENT
Wine Dataset: Machine Learning Model Evaluation & Hyperparameter Tuning

Project Objective

To build and evaluate multiple machine learning models on the Wine dataset, compare their performance using various evaluation metrics, and improve them using hyperparameter tuning techniques. The goal is to find the best-performing model.

Dataset Overview

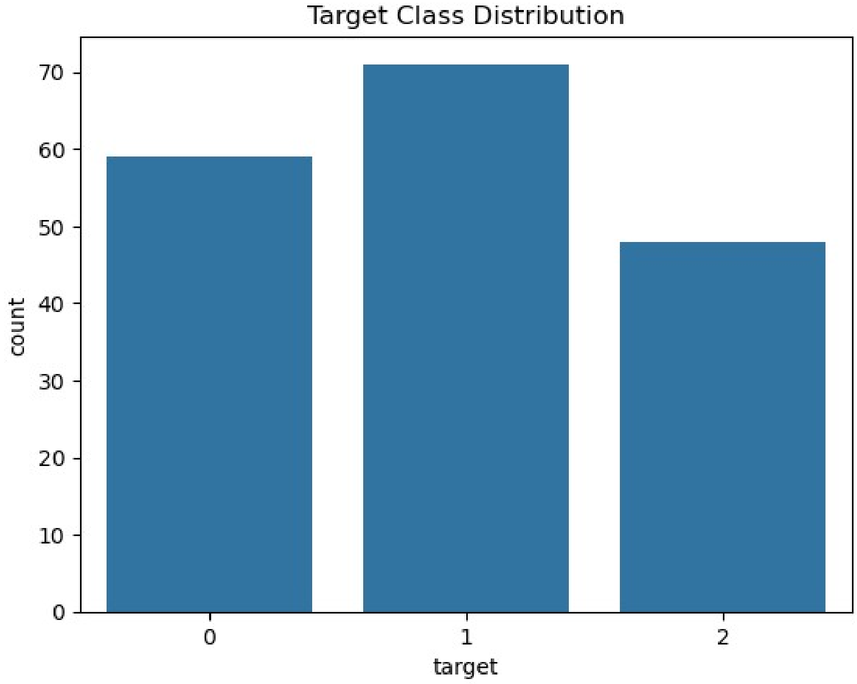
Dataset: UCI Wine dataset from sklearn. datasets
Features: 13 chemical properties of wine
Target: Wine class (3 categories: 0, 1, 2)

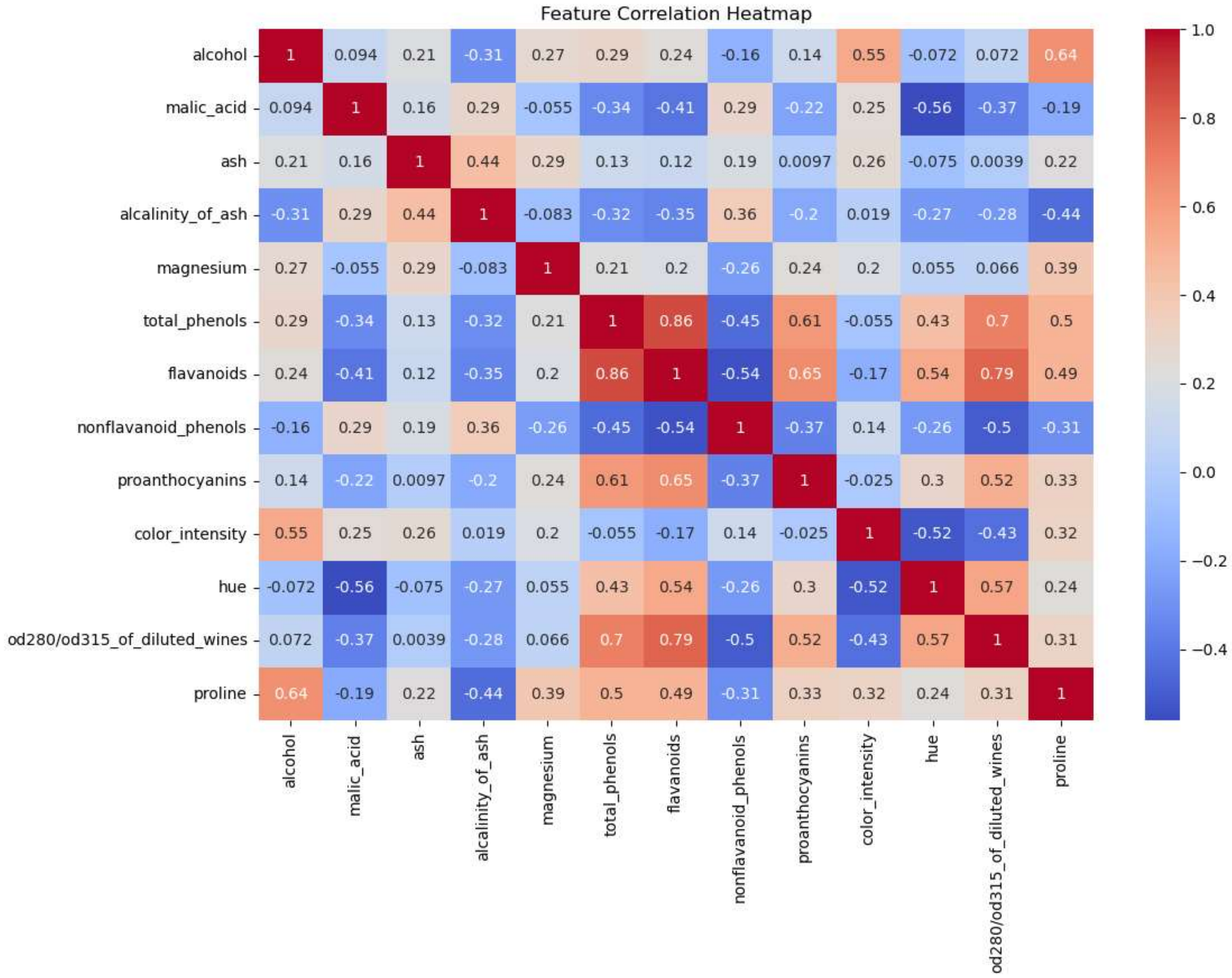


```
Shape of X: (178, 13)
Target classes: [0 1 2]
```

	alcohol	malic_acid	ash	alcalinity_of_ash	magnesium	total_phenols	flavanoids
0	14.23	1.71	2.43	15.6	127.0	2.80	3.06
1	13.20	1.78	2.14	11.2	100.0	2.65	2.76
2	13.16	2.36	2.67	18.6	101.0	2.80	3.24
3	14.37	1.95	2.50	16.8	113.0	3.85	3.49
4	13.24	2.59	2.87	21.0	118.0	2.80	2.69

Exploratory Data Analysis and Visualization





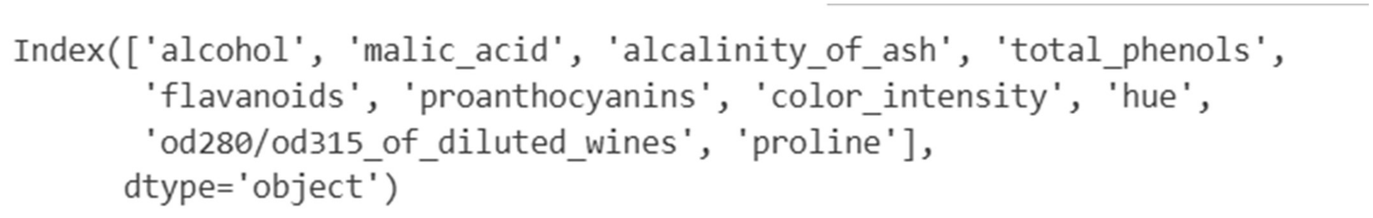
Preprocessing

The dataset was split into training (80%) and test (20%) sets. Features were scaled using StandardScaler to ensure all features had similar ranges. This helps algorithms like KNN and SVM work better.

Feature Selection

We used SelectKBest with ANOVA F-test to select the top 10 most important features. This reduces noise and makes models faster and possibly more accurate.

Selected Features:



```
Index(['alcohol', 'malic_acid', 'alcalinity_of_ash', 'total_phenols',
       'flavanoids', 'proanthocyanins', 'color_intensity', 'hue',
       'od280/od315_of_diluted_wines', 'proline'],
      dtype='object')
```

Model Training

We trained the following 4 models:

1. Logistic Regression
2. Random Forest
3. Support Vector Machine (SVM)
4. K-Nearest Neighbors (KNN)

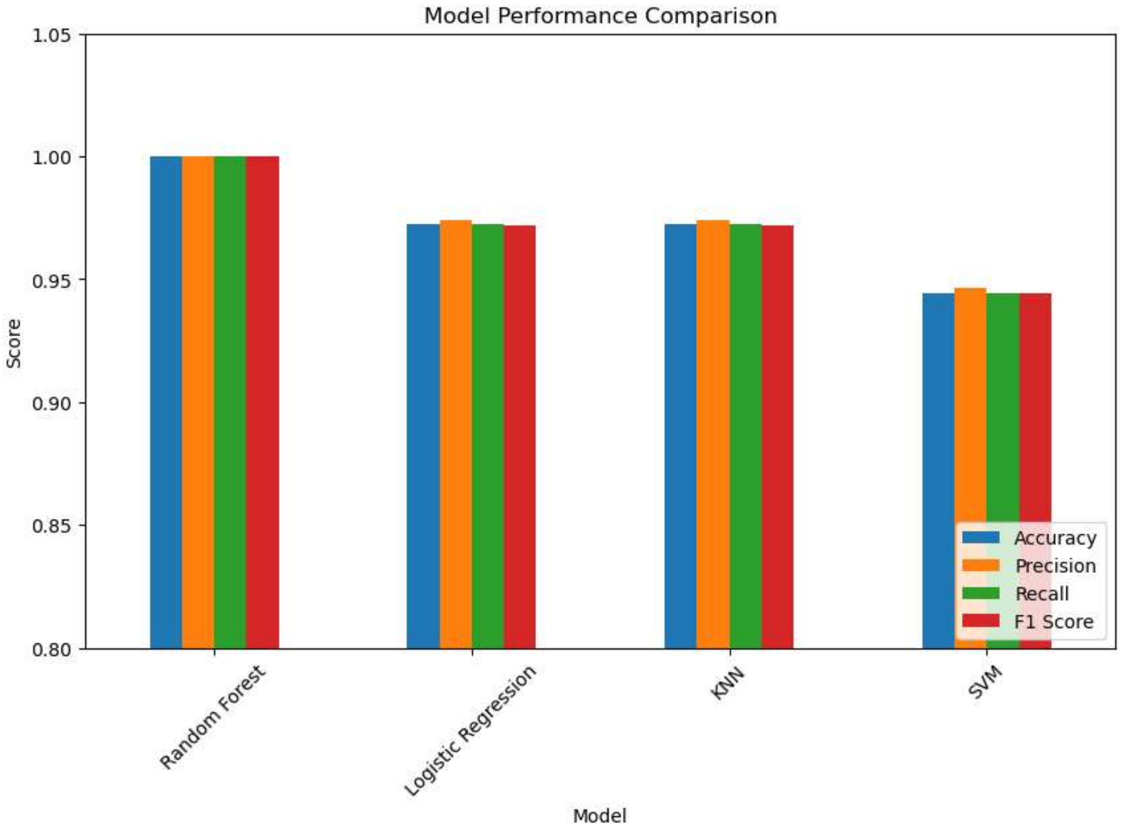
Each model was trained on the training data and evaluated on the test data using:

- Accuracy
- Precision
- Recall
- F1 Score

Initial Model Performance

Model	Accuracy	Precision	Recall	F1 Score
Random Forest	1.00	1.00	1.00	1.00
Logistic Regression	0.97	0.97	0.97	0.97
SVM	0.97	0.97	0.97	0.97
KNN	0.94	0.94	0.94	0.94

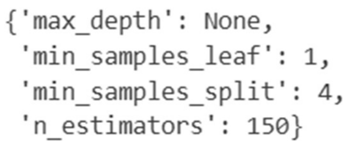
Observation: Random Forest performed perfectly on test data. Logistic Regression and SVM followed closely.



Hyperparameter Tuning

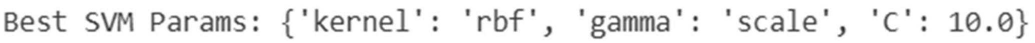
To improve performance and avoid overfitting/underfitting, we fine-tuned two models:

Random Forest (Grid Search):
Tested combinations of: number of trees, max depth, and min samples.



```
{'max_depth': None,
 'min_samples_leaf': 1,
 'min_samples_split': 4,
 'n_estimators': 150}
```

SVM (Randomized Search):
Tested combinations of: C value, kernel type, gamma.

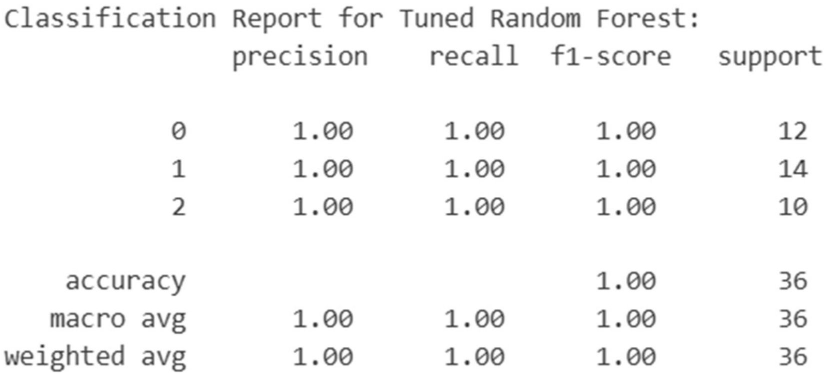


```
Best SVM Params: {'kernel': 'rbf', 'gamma': 'scale', 'C': 10.0}
```

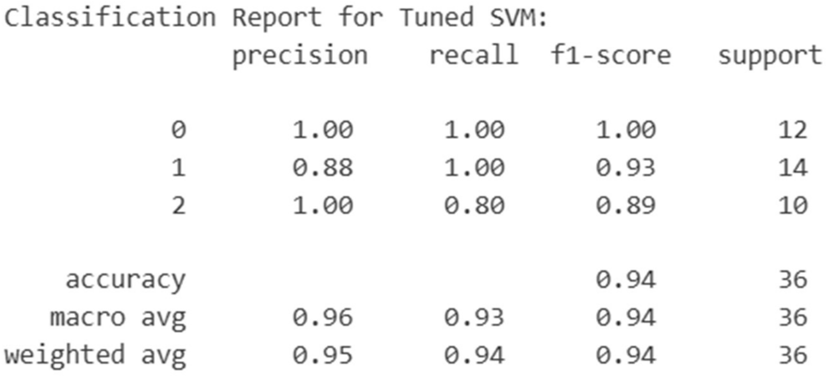
Final Evaluation (Tuned Models)

Model	Accuracy	Precision	Recall	F1 Score
Tuned Random Forest	1.00	1.00	1.00	1.00
Tuned SVM	0.97	0.97	0.97	0.97

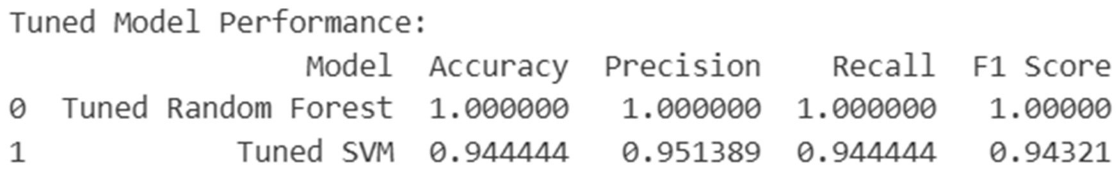
Observation: Even after tuning, Random Forest remained the best model with perfect performance.



```
Classification Report for Tuned Random Forest:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        12
           1       1.00      1.00      1.00        14
           2       1.00      1.00      1.00        10

    accuracy                           1.00        36
   macro avg       1.00      1.00      1.00        36
weighted avg       1.00      1.00      1.00        36
```



```
Classification Report for Tuned SVM:
              precision    recall  f1-score   support

           0       1.00      1.00      1.00        12
           1       0.88      1.00      0.93        14
           2       1.00      0.80      0.89        10

    accuracy                           0.94        36
   macro avg       0.96      0.93      0.94        36
weighted avg       0.95      0.94      0.94        36
```



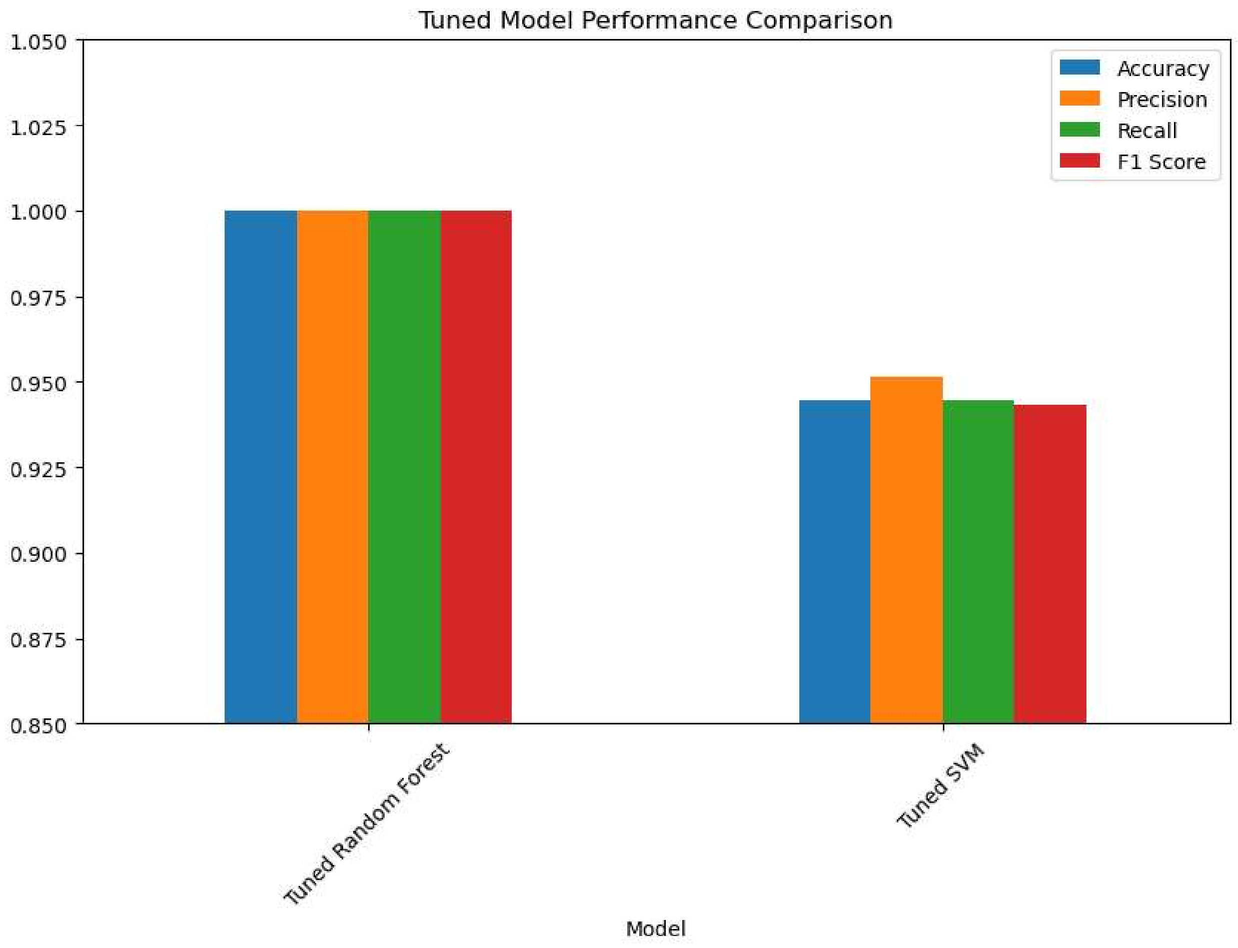
```
Tuned Model Performance:
                  Model  Accuracy  Precision    Recall  F1 Score
0  Tuned Random Forest  1.000000   1.000000  1.000000   1.00000
1            Tuned SVM  0.944444   0.951389  0.944444   0.94321
```

Visual Comparison

Two bar plots were created:

1. Initial Model Comparison — to see how all models performed before tuning.
2. Tuned Model Comparison — to compare Random Forest and SVM after tuning.

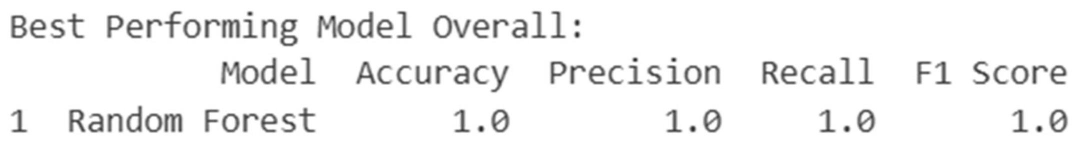
These helped visualize the differences in F1-scores and confirm that Random Forest performed best.



Final Conclusion

Best Model	Accuracy	Precision	Recall	F1 Score
Random Forest	1.00	1.00	1.00	1.00

The Random Forest Classifier was the best-performing model for the Wine dataset. It achieved 100% accuracy, precision, recall, and F1-score after tuning. This suggests the dataset is clean, and Random Forest captures patterns in the data very well.



```
Best Performing Model Overall:
           Model  Accuracy  Precision  Recall  F1 Score
1  Random Forest       1.0        1.0     1.0       1.0
```

Model Export (Optional)

The best model was saved to a file using joblib so it can be used later without retraining.