**EX-8 RANDOM FOREST**  
**DATE**: 14/08/2024  
**NAME**: Ruban s - 71762233039

**Aim**:  
To build a Random Forest ensemble model using Decision Trees to predict employee

attrition based on various features in the provided dataset.

**What is Random Forest?**  
Random Forest is an ensemble learning technique that constructs multiple decision

trees during training and outputs the mode of the classes (classification) or mean

prediction (regression) of the individual trees. Unlike a single decision tree, Random

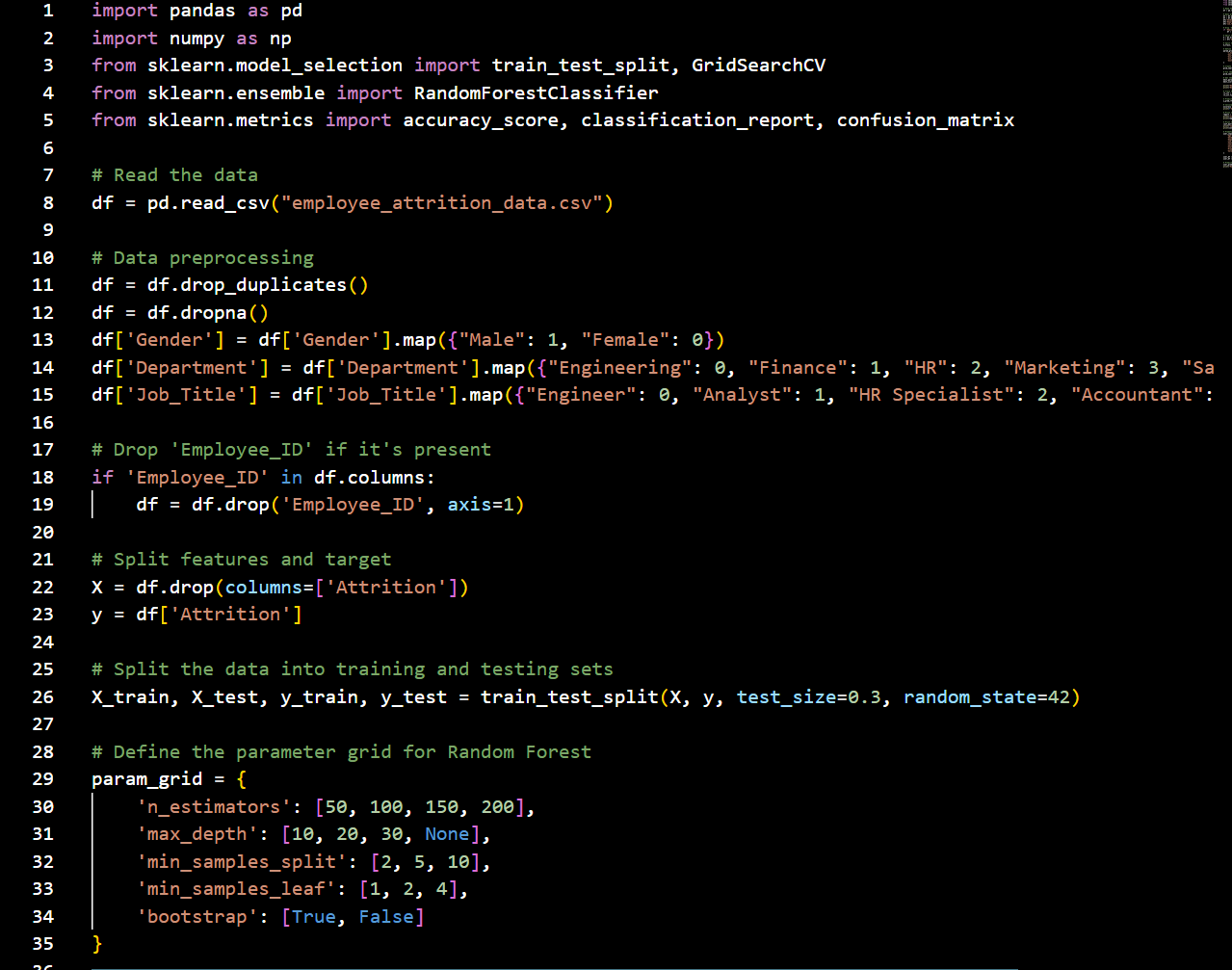
Forest reduces the risk of overfitting by averaging the results of many trees, which

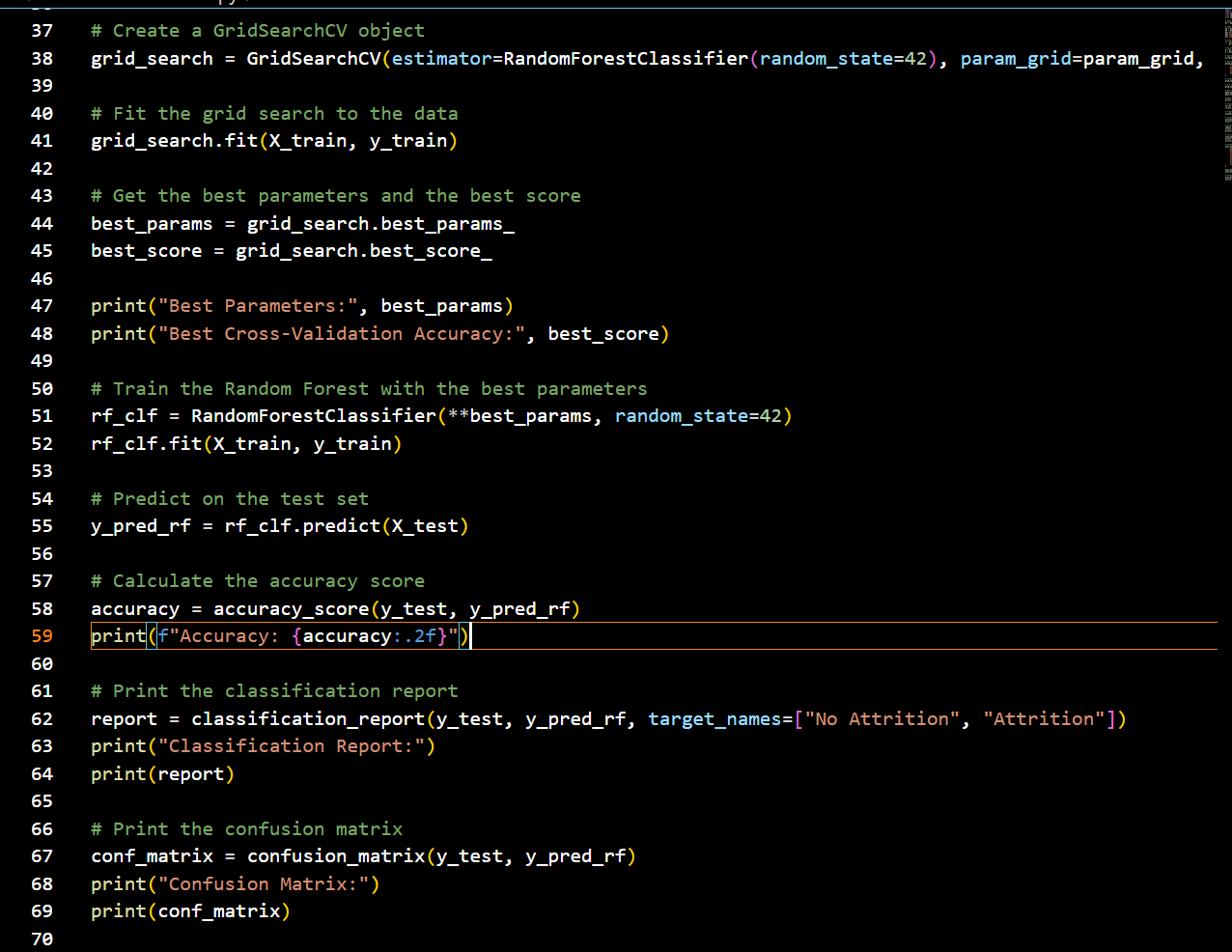
improves generalization and robustness.

**Procedure**:

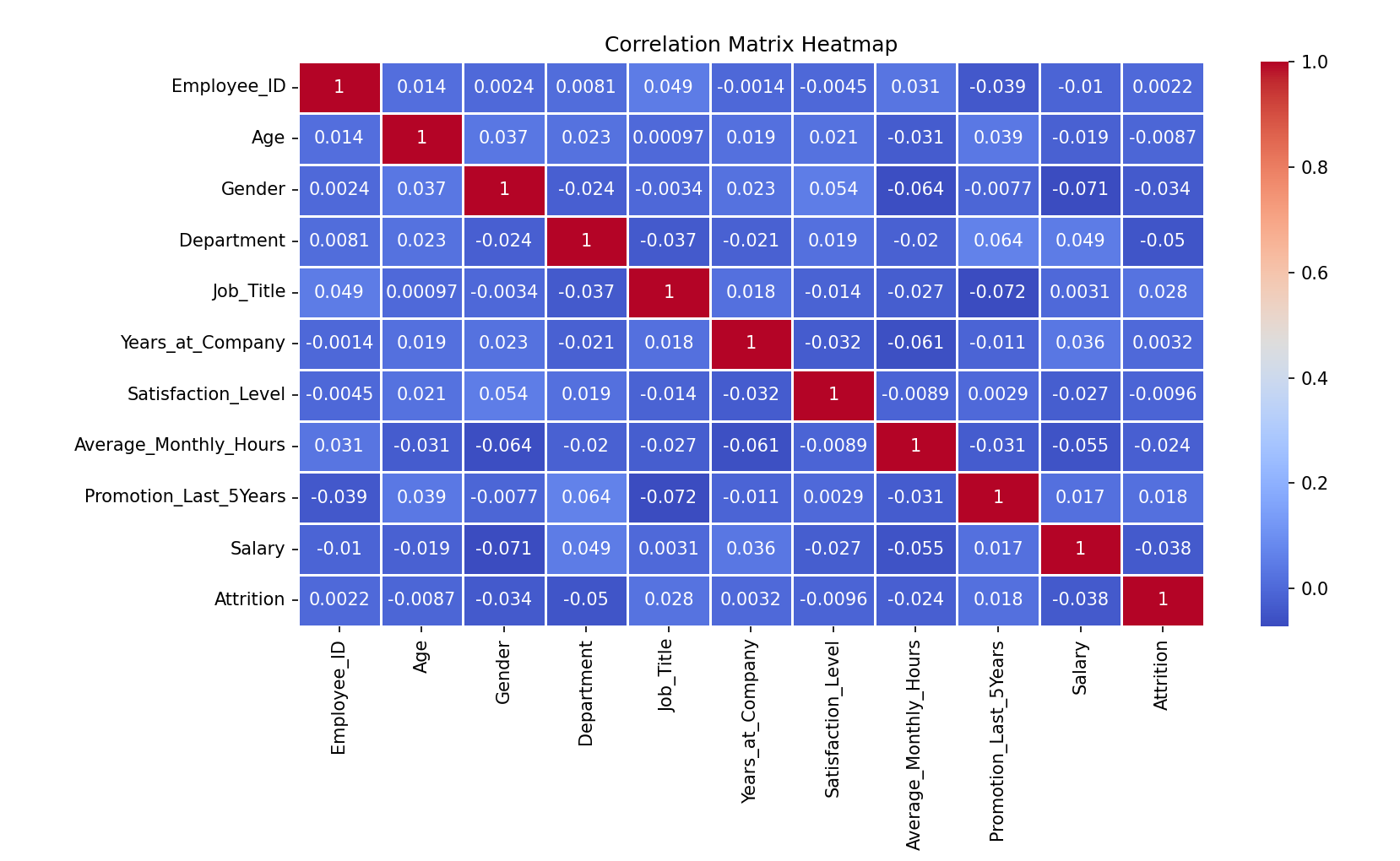
1. **Import Necessary Libraries and Load the Dataset**:
   * Import libraries such as pandas, numpy, scikit-learn, matplotlib, and seaborn.
   * Load the employee attrition dataset into a DataFrame.
2. **Encode Categorical Data**:
   * Use LabelEncoder or mapping to transform categorical columns (such as Gender, Department, Job\_Title) into numerical values.
3. **Prepare the Data**:
   * Select features like Age, Gender\_encode, Department\_encode, Job\_Title\_encode, Years\_at\_Company, Satisfaction\_Level, Average\_Monthly\_Hours, Promotion\_Last\_5Years, and Salary, and define the target variable (Attrition).
   * Split the data into training and testing sets using train\_test\_split.
4. **Define the Base Model**:
   * Use DecisionTreeClassifier as the base estimator for building the Random Forest model.
5. **Implement Random Forest**:
   * Use RandomForestClassifier to create the Random Forest model with the DecisionTreeClassifier as the base estimator.
   * Define the number of trees (n\_estimators), maximum tree depth (max\_depth), and random state (random\_state) for reproducibility.
   * Train the Random Forest model on the training data.
6. **Evaluate the Model**:
   * Make predictions on the test data.
   * Evaluate the model using metrics such as accuracy score, classification report, and confusion matrix.
7. **Plot Feature Importances**:
   * Use seaborn or matplotlib to plot the importance of different features in the Random Forest model.
8. **Plot Confusion Matrix**:
   * Plot the confusion matrix for the Random Forest model to visualize the classification performance.

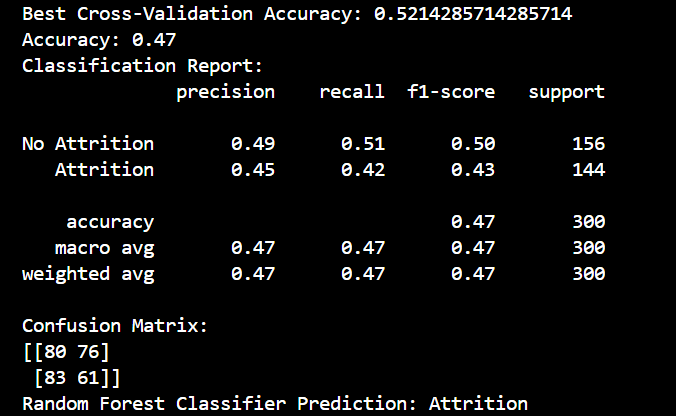
Code :



**Output:**





**Conclusion**:

Random Forest is a powerful ensemble technique that improves model performance by

averaging the predictions of multiple decision trees. This method reduces the risk of

overfitting and often results in more accurate and robust models compared to single

decision trees. However, careful tuning of hyperparameters, such as the number of

trees and tree depth, is essential to ensure the model generalizes well to unseen data.