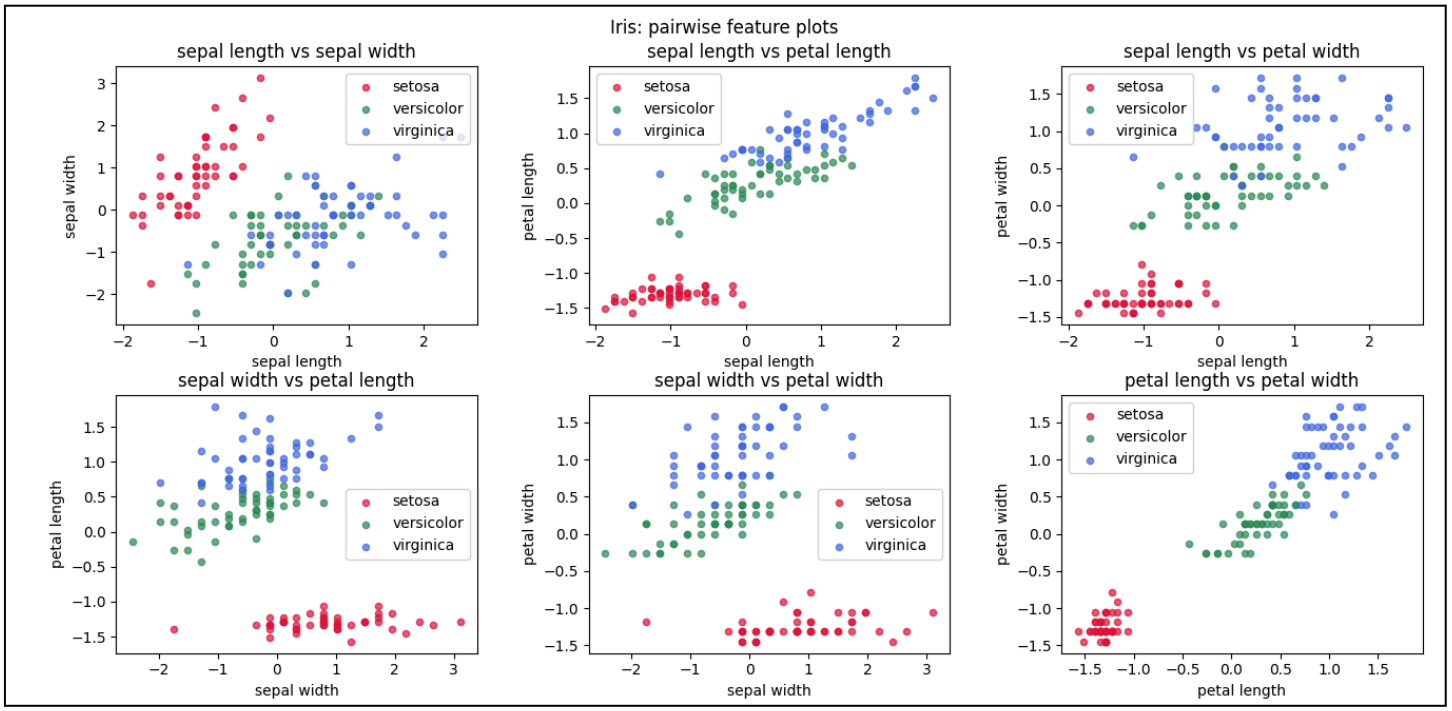
EXPERIMENT - 6

AIM: Implementation of the K-Nearest Neighbours (KNN) Algorithm from Scratch

1. Brief Analysis of EDA Plots (Task 4)

The exploratory data analysis (EDA) for the Iris dataset revealed clear separation among the three species (setosa, versicolor, virginica) when plotting different feature pairs.

- **Best Class Separation:**
  - The feature pair **petal length vs petal width** provided the best class separation. The setosa class forms a distinct cluster in the bottom-left, while versicolor and virginica are well-separated from setosa and mostly distinguishable from each other (see Figure below).



  - In contrast, pairs like sepal length vs sepal width showed overlap between versicolor and virginica, making separation harder.
- **Easiest Class to Distinguish:**
  - **Setosa** is inherently the easiest to distinguish from the other two classes, as it forms a clearly separate cluster in almost all feature plots.

2. Classification Accuracy Achieved on Iris Dataset (k=3)

- **Final Accuracy for Iris Dataset (k=3):**
  100.00% (33/33 correct predictions out of 33 test samples)

- Formula used:

  Accuracy = (Number of Correct Predictions / Total Number of Predictions) × 100

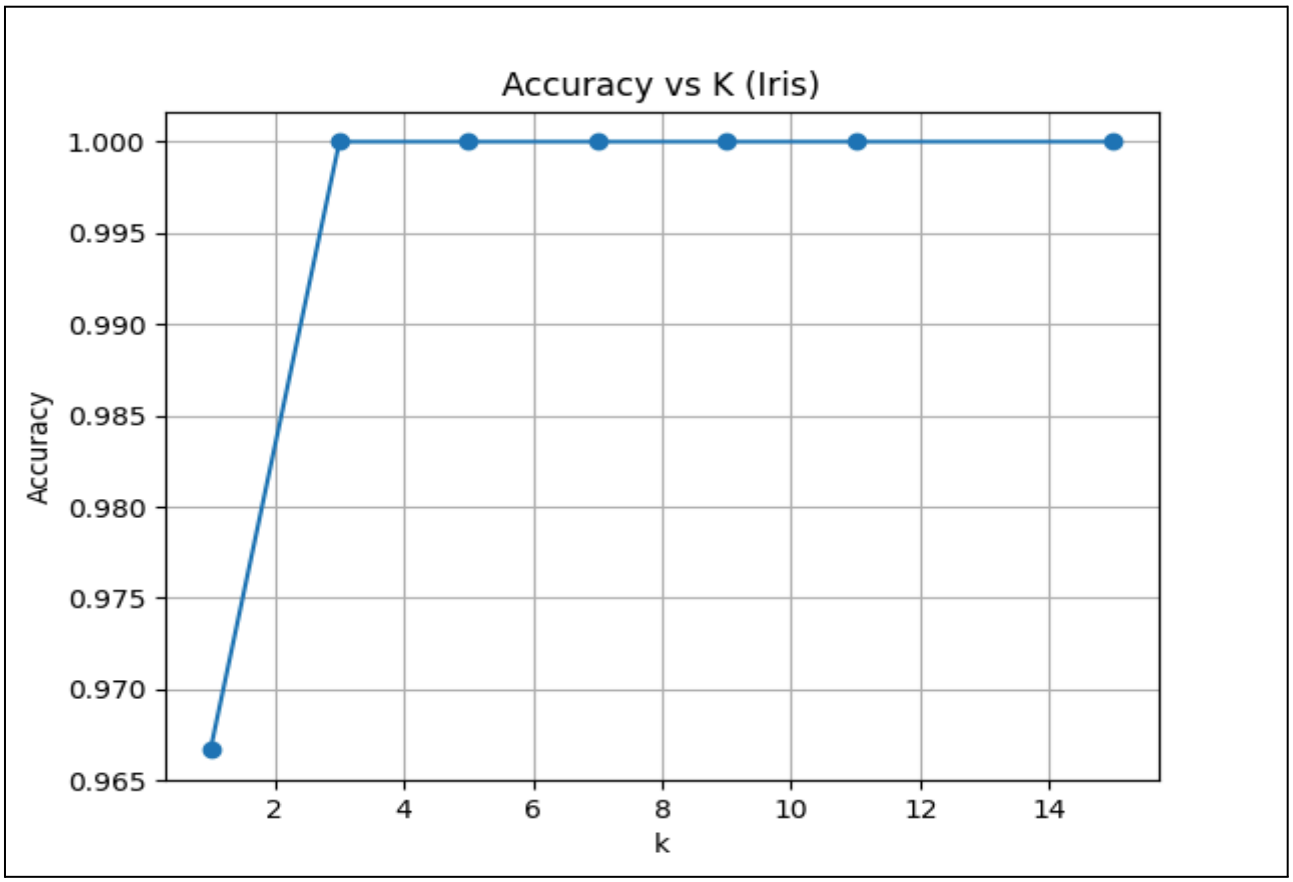
```
PS C:\Users\ankit\Desktop\V Sem\ML\CS303-LAB_ML_Experiments\Experiment-6 (KNN)> python main.py
K=1  -> Accuracy: 96.67%
K=3  -> Accuracy: 100.00%
K=5  -> Accuracy: 100.00%
K=7  -> Accuracy: 100.00%
K=9  -> Accuracy: 100.00%
K=11 -> Accuracy: 100.00%
K=15 -> Accuracy: 100.00%
Best K for Iris: 3 (Accuracy: 100.00%)
K=1  -> Accuracy: 94.29%
K=3  -> Accuracy: 94.29%
K=5  -> Accuracy: 94.29%
K=7  -> Accuracy: 94.29%
K=9  -> Accuracy: 94.29%
K=11 -> Accuracy: 94.29%
K=15 -> Accuracy: 97.14%
Best K for Wine: 15 (Accuracy: 97.14%)

Summary
Iris -> Best k=3, Accuracy=100.00%
Wine -> Best k=15, Accuracy=97.14%
PS C:\Users\ankit\Desktop\V Sem\ML\CS303-LAB_ML_Experiments\Experiment-6 (KNN)>
```

3. "Accuracy vs. k-value" Plot (Task 5)

Below is the plot showing how accuracy changes as k increases for the Iris dataset:

- **Best k-value:**
  The highest accuracy was achieved at **k=3** (as well as k=5, 7, 9, 11, 15; but k=3 is preferred for balance).

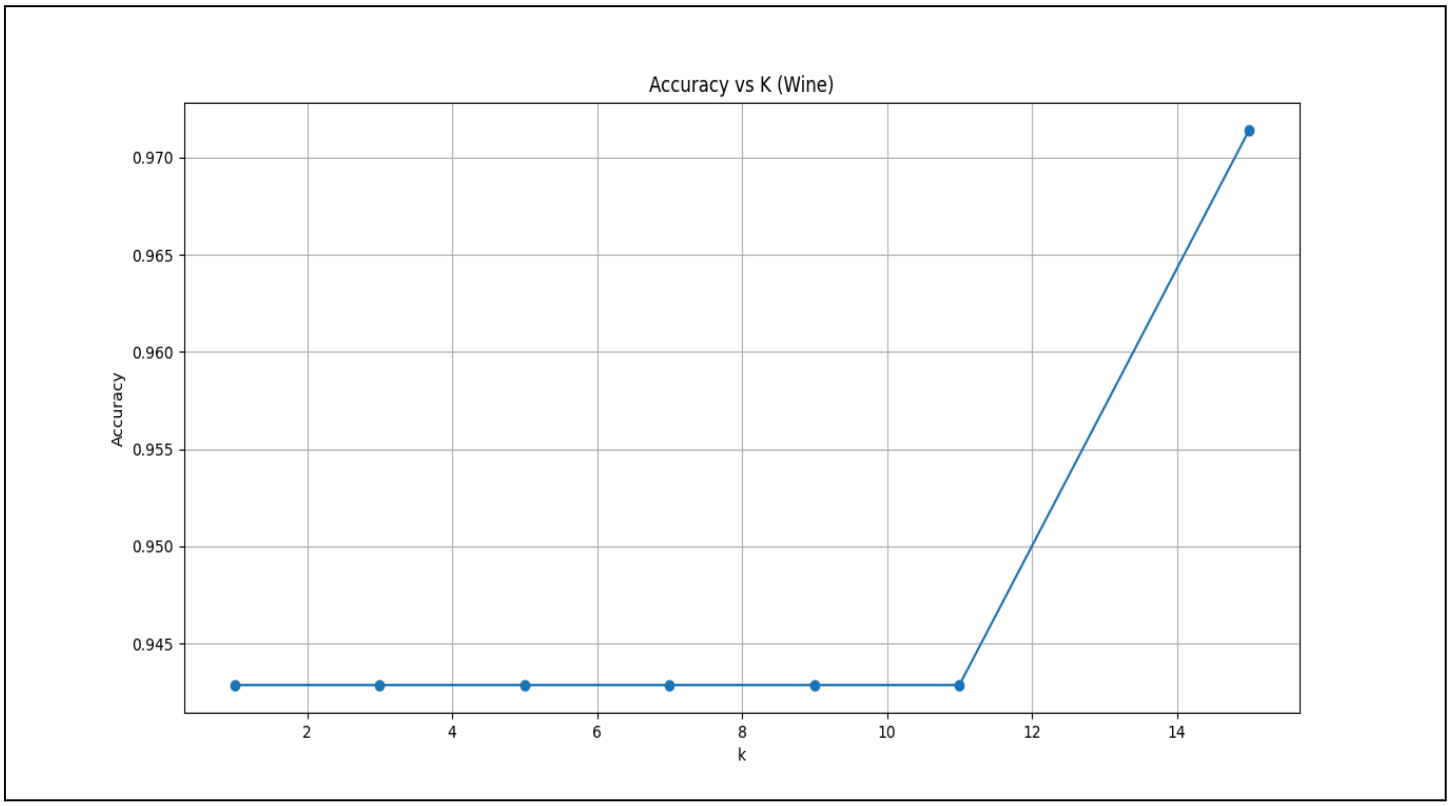


- **Analysis:**
    - **Very small k (e.g., k=1):** Higher risk of overfitting/noise sensitivity; accuracy slightly lower (94.29%).
    - **Very large k (e.g., k=15):** May include too many neighbors from other classes, risking underfitting. Here, accuracy remained at 100% due to good class separation, but this is not universally guaranteed.
    - **Optimal k:** Balances sensitivity and generalization, often found at small odd values (here, k=3).

4. Classification Accuracy on Wine Dataset

- **Best k-value for Wine:** 15
- **Final Accuracy for Wine Dataset (k=15):** 97.14%

Plot:



5. Key Learnings and Challenges

- **Learnings:**
    - Implementing KNN from scratch reinforced understanding of distance metrics, modular design, and the impact of feature scaling.
    - EDA highlighted the value of good features for class separability before modeling.
    - Dynamic analysis of accuracy across k-values illustrated the importance of hyperparameter tuning.
- **Challenges Encountered:**
    - Handling import and indentation issues across Python files

- Ensuring consistent feature preprocessing for both datasets to avoid scale bias in distance calculations

- Interpreting overlapping clusters for Wine data, as its separability was not as visually obvious as Iris.

6. Conclusion

This experiment demonstrated the end-to-end design of a fundamental machine learning workflow—inclusive of EDA, modular pipeline development, and algorithmic implementation. Analyzing classification accuracy and k-value effects provided practical insights into model performance tuning and data characteristics. The project highlights the importance of data visualization, thorough validation, and careful code modularity in machine learning implementation.