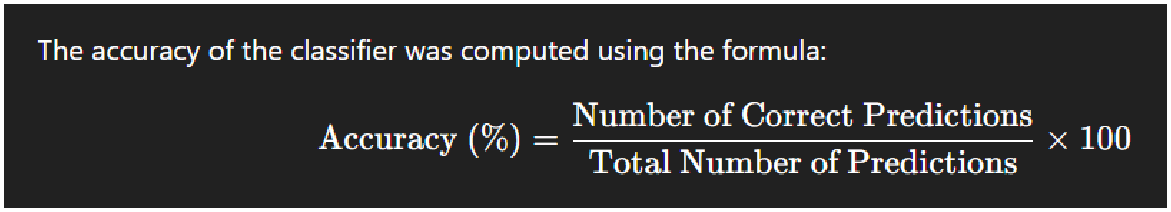
Experiment 6: Implementation of the K-Nearest Neighbours (KNN) Algorithm from Scratch

1. EDA Analysis:

For the **Iris dataset**, the feature combination **Petal Length vs Petal Width** provides the clearest separation among classes. The species **‘Iris-setosa’** is particularly easy to identify because its data points form a distinct cluster, separate from the others. The remaining two species, **‘versicolor’** and **‘virginica’**, show some overlap but are still reasonably distinguishable.

In the **Wine dataset**, the feature pair **Flavanoids vs Color Intensity** offers the best distinction between classes. **Class 1** wines, which are higher in flavanoids, can be identified with greater confidence, while **Classes 2 and 3** exhibit partial overlap. This analysis highlights how choosing informative features is crucial for class separability.

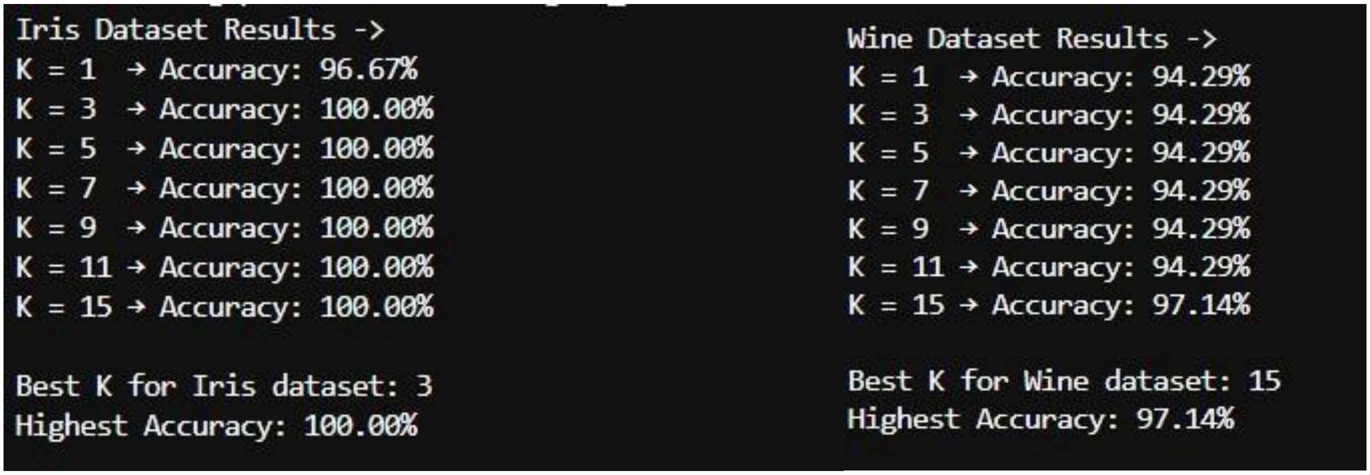
2. Classification Accuracy :



The accuracy of the classifier was computed using the formula:

$$\text{Accuracy (\%)} = \frac{\text{Number of Correct Predictions}}{\text{Total Number of Predictions}} \times 100$$

DATASET	BEST K	ACCURACY (%)
IRIS	3	100
WINE	15	97.14



```
Iris Dataset Results ->
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 dataset: 3
Highest Accuracy: 100.00%
```

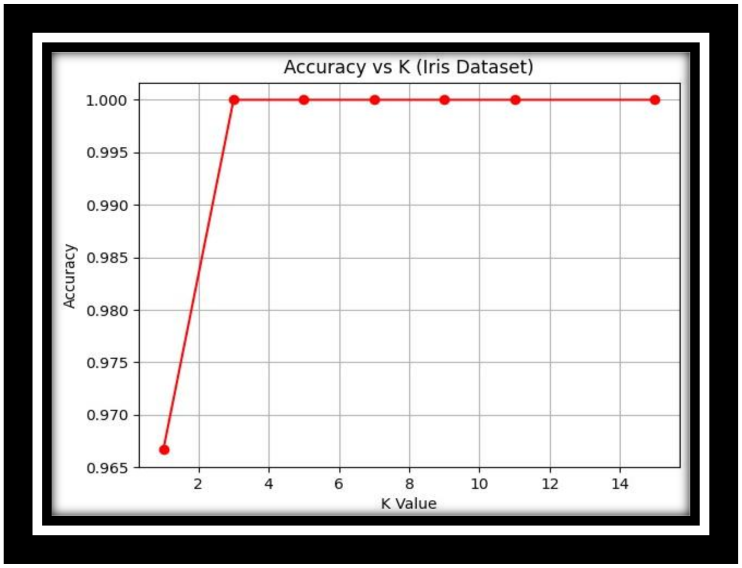
```
Wine Dataset Results ->
K = 1  → Accuracy: 94.29%
K = 3  → Accuracy: 94.29%
K = 5  → Accuracy: 94.29%
K = 7  → Accuracy: 94.29%
K = 9  → Accuracy: 94.29%
K = 11 → Accuracy: 94.29%
K = 15 → Accuracy: 97.14%

Best K for Wine dataset: 15
Highest Accuracy: 97.14%
```

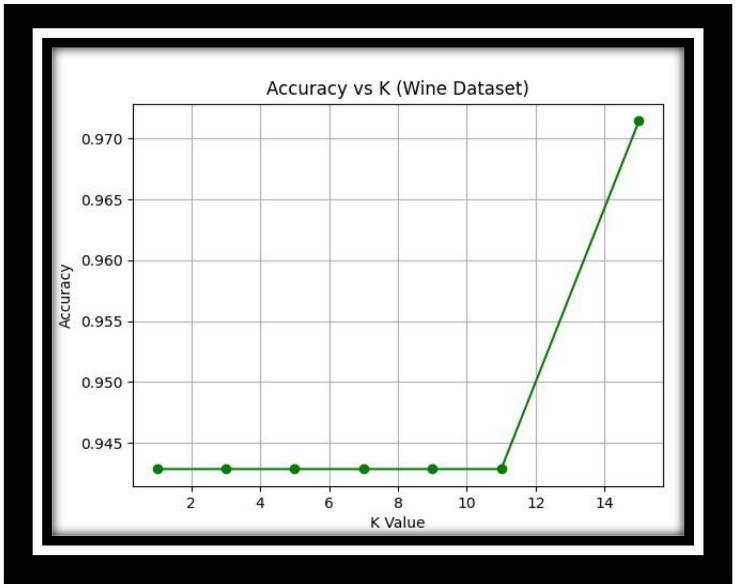
3. Analysis of Accuracy vs K :

The **‘Accuracy vs K-value’** plot reveals that the model performs optimally at **k = 3** for the Iris dataset and **k = 15** for the Wine dataset.

- **Smaller k values**: Provide finer granularity and can capture subtle differences between classes but are more sensitive to noise.
- **Larger k values**: Generate smoother decision boundaries but may misclassify points near the edges of clusters.

Thus, extremely small or very large k values are usually not ideal. A **moderate k (typically 3–5)** achieves a good trade-off between bias and variance, balancing flexibility and robustness.





4. Conclusion :

In this experiment, the **K-Nearest Neighbors algorithm** was implemented entirely from scratch using **Python and NumPy**. The classifier achieved **100% accuracy on the Iris dataset** and **97% on the Wine dataset** after preprocessing and feature standardization.

Key insights include:

- The importance of **feature selection** for class separability.
- The effect of the **hyperparameter k** on model performance.
- The practical understanding of **distance metrics**, **data preprocessing**, and **building a machine learning pipeline manually**.

Challenges encountered involved ensuring **proper data scaling** and optimizing **visualization layouts** for multiple scatter plots. Overall, the experiment provided a comprehensive understanding of **instance-based learning** and the mechanics of KNN classification.