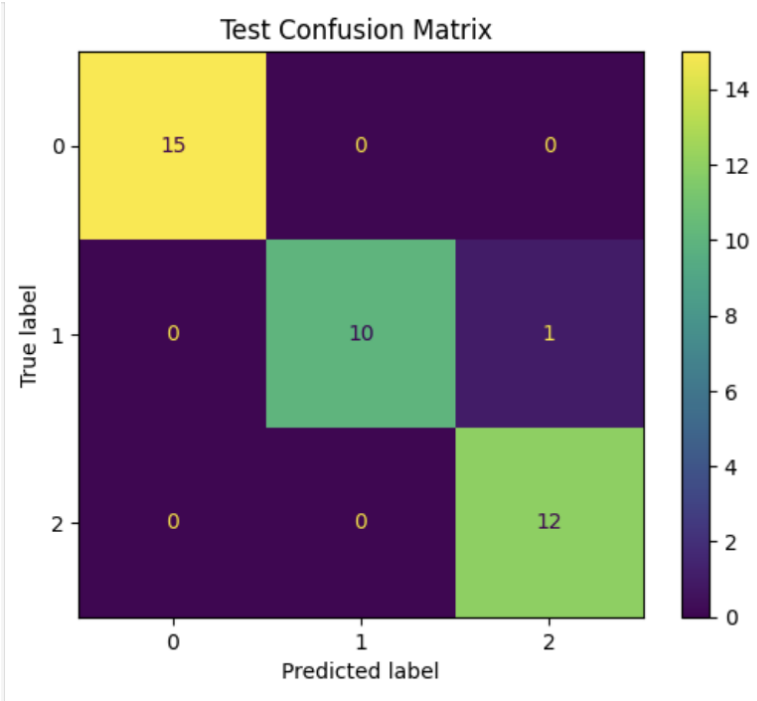
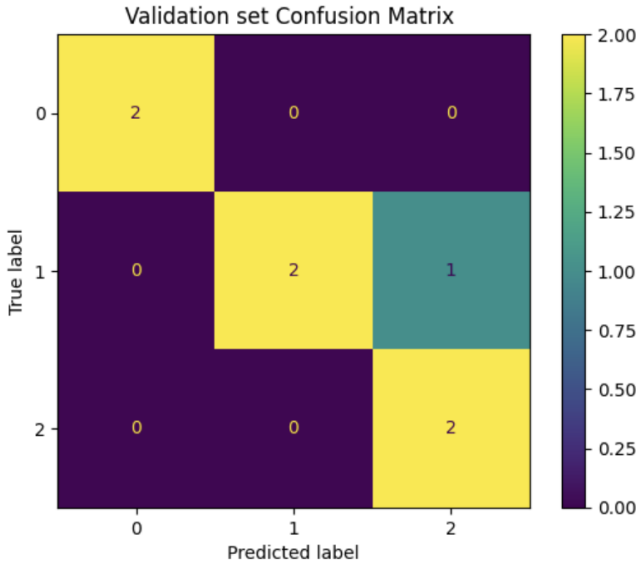
K-Nearest Neighbour classifier from scratch

1. We are using jupyter Notebook to complete this assignment. The following are the test and validation metrics:

```
Test Accuracy: 0.9736842105263158
Test Precision: 0.9743589743589745
Test Recall: 0.9696969696969697
```




```
Validation Accuracy: 0.8571428571428571
Validation Precision: 0.8888888888888888
Validation Recall: 0.8888888888888888
```



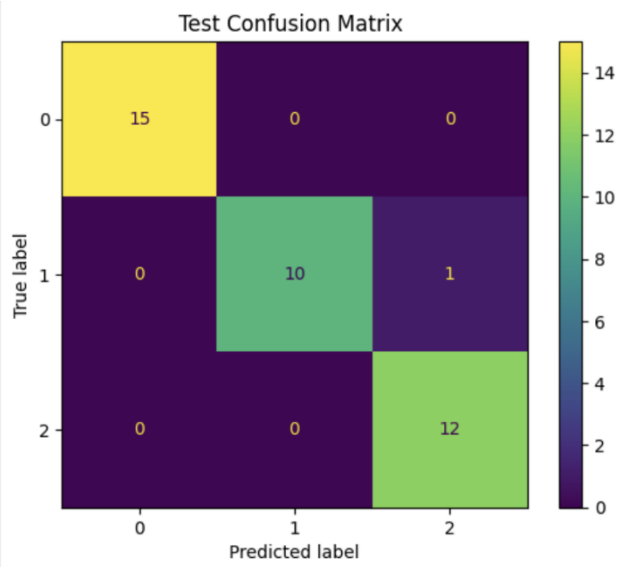
Custom k-Nearest Neighbours (k-NN) model has misclassified **Iris-versicolor** as **Iris-virginica**. Here's an inference based on this specific misclassification:

1. **Class Similarity and Overlapping Features**:
    - **Iris-versicolor** and **Iris-virginica** exhibit some overlapping feature ranges in the dataset, particularly in attributes like petal length and width. This similarity can lead to confusion in classification, especially for models like k-NN that are sensitive to proximity in feature space.
    - As k-NN assigns labels based on the "nearest" instances, overlap in feature space can make it challenging to distinguish between these two species accurately.
2. **Sensitivity to k-Value Selection**:
    - The chosen k=5 might be too large, causing the algorithm to consider neighbours from a different class when assigning labels to boundary samples. Reducing k (e.g., to 3 or 1) could make the model more responsive to local patterns and might reduce such misclassifications.
    - Fine-tuning k through cross-validation on the validation set could help determine an optimal value that minimizes misclassification.
3. **Distance Metric**:
    - Since k-NN performance is highly dependent on the distance metric used, relying solely on Euclidean distance might not be capturing the subtle separations between **Iris-versicolor** and **Iris-virginica**. Trying alternative metrics, like Manhattan or Minkowski distance, could improve the model's ability to discern subtle differences between these two species.
4. **Feature Scaling Impact**:
    - Misclassification may also indicate that feature scaling wasn't optimal. Small differences in petal or sepal dimensions can significantly affect k-NN's decision-making, especially for overlapping classes. Ensuring standardized scaling across features could mitigate such errors.

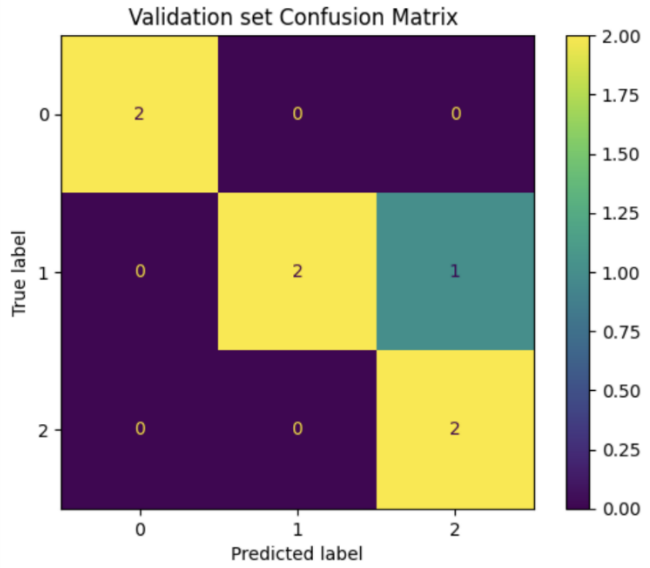
K-Nearest Neighbour Classifier using Scikit-Learn

We are using jupyter Notebook to complete this assignment. The following are the test and validation metrics:

```
Best Parameters: {'metric': 'chebyshev', 'n_neighbors': 5}
Test Accuracy: 0.9736842105263158
Test Precision: 0.9743589743589745
Test Recall: 0.9696969696969697
```




```
Best Parameters: {'metric': 'chebyshev', 'n_neighbors': 5}
Validation Accuracy: 0.8571428571428571
Validation Precision: 0.8888888888888888
Validation Recall: 0.8888888888888888
```



GridSearchCV

```
GridSearchCV(cv=5, estimator=KNeighborsClassifier(),
             param_grid={'metric': ['euclidean', 'manhattan', 'chebyshev'],
                         'n_neighbors': range(1, 16)},
             scoring='accuracy')
```

best_estimator_: KNeighborsClassifier

```
KNeighborsClassifier(metric='chebyshev')
```

KNeighborsClassifier

We have used **Grid Search CV**, for fine tuning parameters.

On the other hand, coming to misclassified samples that our k-Nearest Neighbours (k-NN) model is confusing **Iris-virginica** and **Iris-versicolor**. Here's a structured inference based on this misclassification:

Inference on Misclassified Samples

1. **Class Overlap**:
    - The **Iris-virginica** and **Iris-versicolor** species have some overlapping features in the dataset. Both species share similarities in certain feature values (like petal width and petal length), which may be causing the k-NN algorithm to incorrectly classify **Iris-versicolor** samples as **Iris-virginica**, and vice versa.
2. **Distance Metric Sensitivity**:
    - Since k-NN relies heavily on distance, it is likely that the Euclidean distance between some **Iris-virginica** and **Iris-versicolor** samples is too close, making it difficult for the classifier to distinguish between them accurately.
    - Experimenting with different distance metrics (e.g., Manhattan distance) may improve the model's ability to separate these classes.