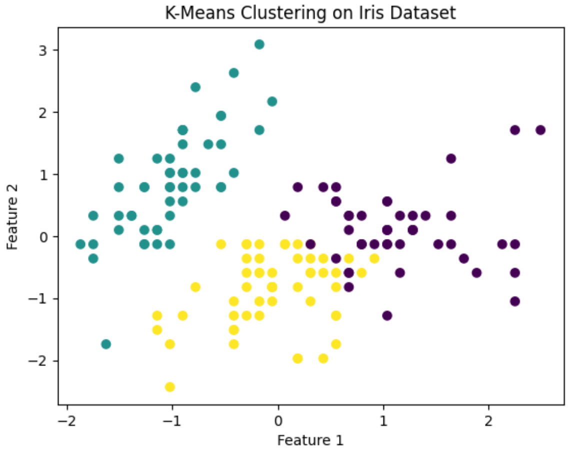
Unit 6: Clustering with Python
Unit 6 Artefact

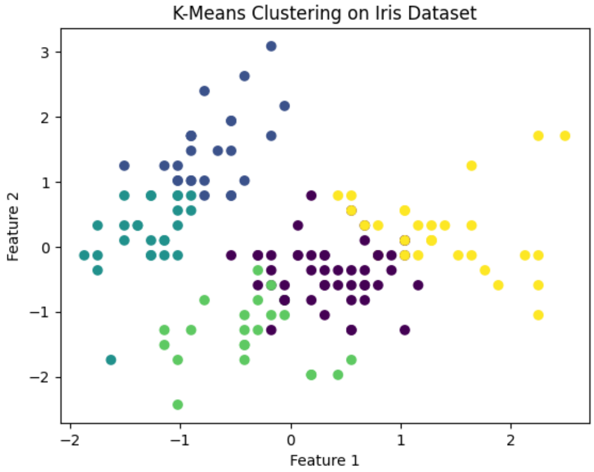
We had out final meetings and push for the team project submission and completed the report and peer template submissions that also on the portfolio under Unit 6.

I also did some more work to learn more about k-means clustering, an unsupervised machine learning algorithm used to partition a set of data points into distinct, no-overlapping subsets called clusters.

I did some self-directed learning to understand this topic in more detail. For this, I used the Iris dataset which contains measurements of different features of iris flowers. I worked with Google Colab and internet guides to understand how to preprocess t0 data which included standardising the features using scaling. This ensures that each feature contributes equally to the distance calculations the k-means algorithm. I initially chose n=3 for the number of clusters (Silhouette Score: 0.460)



I then repeated the programme with n=5 clusters and the Silhouette Score went down to 0.342, meaning it was less well-defined than with n=3 clusters.



Programme:

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.datasets import load_iris
from sklearn.preprocessing import StandardScaler
from sklearn.cluster import KMeans
from sklearn.metrics import silhouette_score

# Load the Iris dataset
iris = load_iris()
X = iris.data
y = iris.target

# Standardise the features
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Apply K-Means clustering
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans.fit(X_scaled)

# Get the cluster labels
labels = kmeans.labels_

# Evaluate the clustering using silhouette score
silhouette_avg = silhouette_score(X_scaled, labels)
print(f'Silhouette Score: {silhouette_avg:.3f}')

# Visualize the clustering
plt.scatter(X_scaled[:, 0], X_scaled[:, 1], c=labels, cmap='viridis', marker='o')
plt.title('K-Means Clustering on Iris Dataset')
plt.xlabel('Feature 1')
plt.ylabel('Feature 2')
plt.show()

# Show the cluster centres
print("Cluster Centres:")
print(kmeans.cluster_centres_)
```