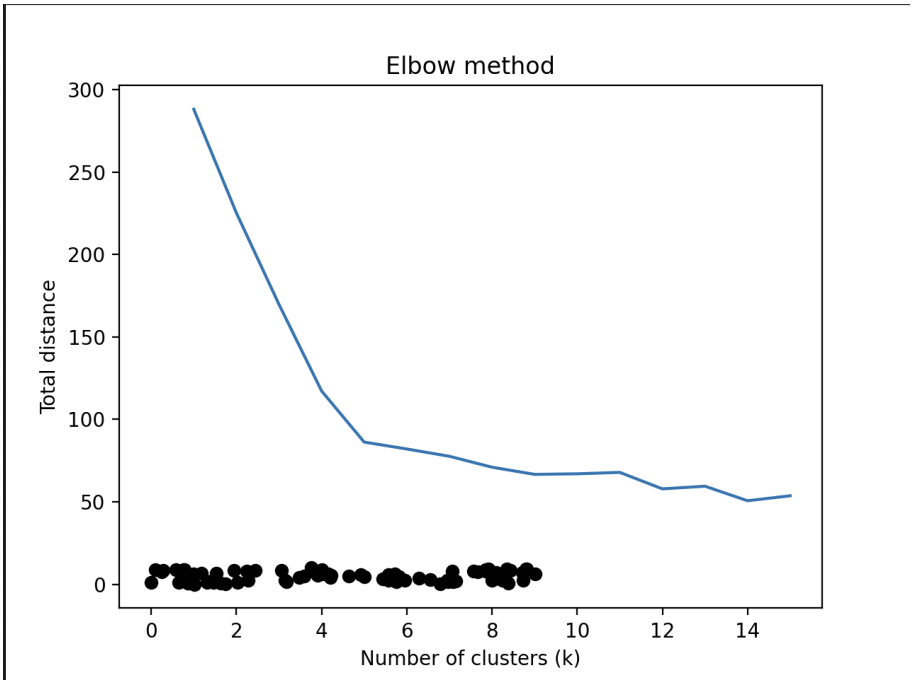
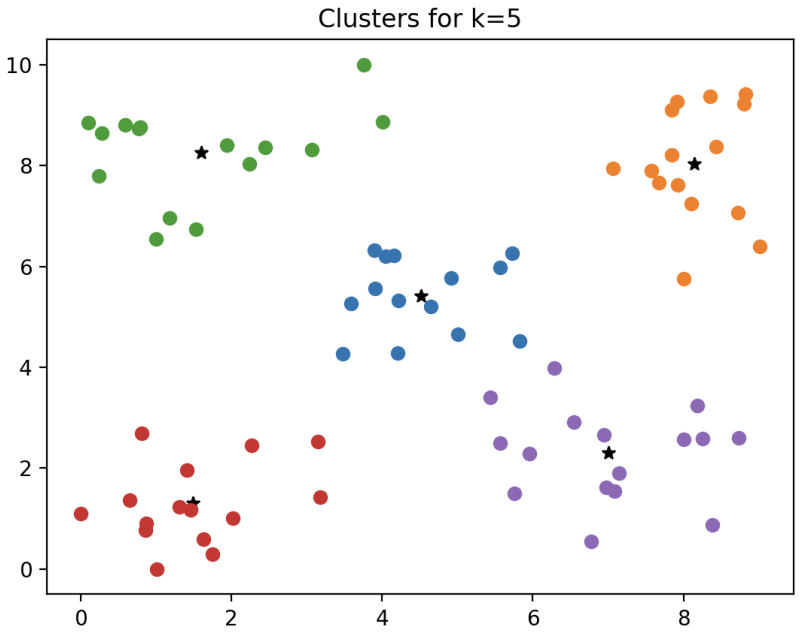
Assignment 3: Machine learning

Part 1 - Clustering

1. With our K-means cluster we iterate our algorithm so the cluster/data points get sorted and organized together. Firstly it starts by choosing random centroids , then the iteration begins. The algorithm gives every data point a centroid and this is based on distance so the closest one, we use the Euclidean distance for this distance check. When all points are allocated to a centroid we take the average of all the data points in its cluster to recalculate the centroid to get a new position for the next iteration. This iteration continues until our maximum number of iteration (“max_iter”) cap has been reached.
2. We consider k = 5 to be optimal for the provided dataset. When k = 5 we can see a clear bend in the plot (elbow).



3. Here are the clusters when k = 5.

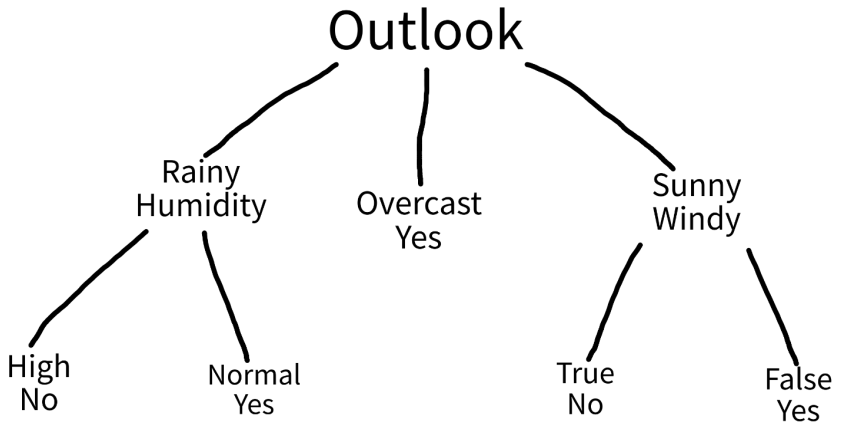


4. When we start with random centroids every time we run the k-means algorithm, they have completely different locations each time. The k-means algorithm updates clusters to reduce the sum of distances from points to their nearest centroid, which means that every time we run the algorithm it can find a different local optima. This is why we can get different clustering results depending on where the initial centroids are placed.

Part 2 - Classification

1. The ID3 algorithm that we created seeks to separate the data into groups that are the most pure using the entropy, by continuously splitting the data and the goal is to build a decision tree to predict whether to play golf Yes or No depending on outlook. In the algorithm firstly we have done checks if all the target values are the same and returning also the other check is if the maximum height is reached or no more features available to split. If the stopping condition is not met then it continues with calculating the current dataset's entropy. Then for each possible attribute for example outlook, temp, humidity etc it goes though the all the values for example for outlook it checks rainy, sunny, overcast. It computes and checks the entropy of each subset and calculates the information gain and the best feature is the one with the most information gain. Lastly our algorithm checks if a split is better and if checks if no attribute improves the information gain the node should then become a leaf and that means there isn't a useful split.
2. Here is the decision tree constructed by the algorithm and a sketch to see it more clearly we made to show it graphically.

```
Node<1>
  Non leaf node - Parent: None
  Split variable: Outlook
    Child_node: 2, split_value: Rainy
    Child_node: 5, split_value: Overcast
    Child_node: 6, split_value: Sunny
Node<2>
  Non leaf node - Parent: 1
  Split variable: Humidity
    Child_node: 3, split_value: High
    Child_node: 4, split_value: Normal
Node<5>
  Leaf node - Parent: 1, Decision: Yes
Node<6>
  Non leaf node - Parent: 1
  Split variable: Windy
    Child_node: 7, split_value: False
    Child_node: 8, split_value: True
Node<3>
  Leaf node - Parent: 2, Decision: No
Node<4>
  Leaf node - Parent: 2, Decision: Yes
Node<7>
  Leaf node - Parent: 6, Decision: Yes
Node<8>
  Leaf node - Parent: 6, Decision: No
```



3. When setting the maximum height to a smaller amount for example 2, the tree has fewer splits which means the accuracy will be lower. It will only capture the two most dominant features for classification. When setting it to a larger amount, nothing happens to our implementation but this is probably because we use a very small dataset. If it would have been a large dataset, we could see a better accuracy the higher maximum height. But if it's too high it could be overfitted if tested on unseen data.
4.

Outlook	**Temp**	**Humidity**	**Windy**	**Play Golf**	Predicted	Actual
Rainy	Hot	High	False	No	No	No
Rainy	Hot	High	True	No	No	No

Overcast	Hot	High	False	Yes	Yes	Yes
Sunny	Mild	High	False	Yes	Yes	Yes
Sunny	Cool	Normal	False	Yes	Yes	Yes
Sunny	Cool	Normal	True	No	No	No
Overcast	Cool	Normal	True	Yes	Yes	Yes
Rainy	Mild	High	False	No	No	No
Rainy	Cool	Normal	False	Yes	Yes	Yes
Sunny	Mild	Normal	False	Yes	Yes	Yes
Rainy	Mild	Normal	True	Yes	Yes	Yes
Overcast	Mild	High	True	Yes	Yes	Yes
Overcast	Hot	Normal	False	Yes	Yes	Yes
Sunny	Mild	High	True	No	Yes	No

	Positive	Negative
Positive	9 TP	0 FP
Negative	1 FN	4 TN

Underfitting happens when the maximum height is set too low, like 2 or 1 which we could see in the previous question. The model then becomes too simplistic and fails to capture important patterns. Overfitting happens when the maximum height is set too high which will make the model memorize patterns that are too specific and makes the model potentially lose generalizability.