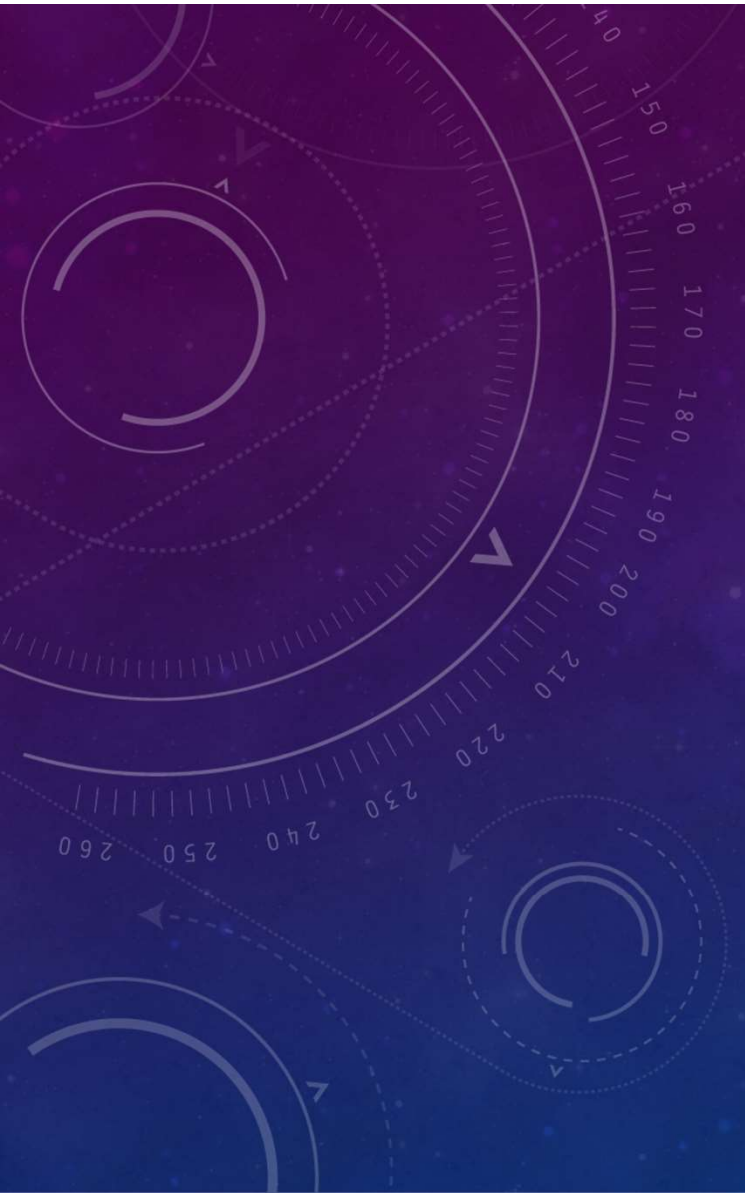


MACHINE LEARNING WITH .NET

Asmaliza Ahzan @ Emma

asmaliza@iverson.com.my

CLUSTERING MODEL

CHAPTER 5

CLUSTERING

- An unsupervised machine learning task that is used to group instances of data into clusters that contain similar characteristics.
- Clustering can also be used to identify relationships in a dataset that you might not logically derive by browsing or simple observation.
- The inputs and outputs of a clustering algorithm depends on the methodology chosen.
- You can take a distribution, centroid, connectivity, or density-based approach.
- ML.NET currently supports a centroid-based approach using K-Means clustering.

EXAMPLES OF CLUSTERING SCENARIOS

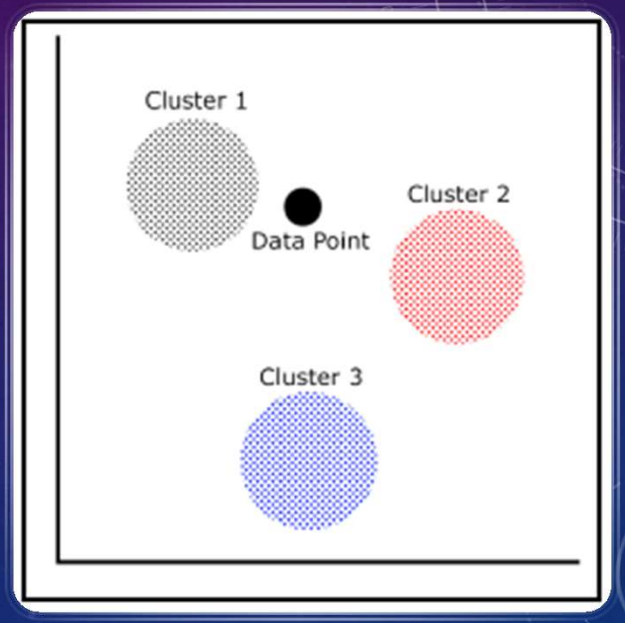
- Understanding segments of hotel guests based on habits and characteristics of hotel choices.
- Identifying customer segments and demographics to help build targeted advertising campaigns.
- Categorizing inventory based on manufacturing metrics.

DIVING INTO THE K-MEANS TRAINER

- The k-means trainer used in ML.NET is based on the Yinyang method as opposed to a classic k-means implementation.
- Like some of the trainers we have looked at in previous chapters, all of the input must be of the Float type.
- In addition, all input must be normalized into a single feature vector.
- Fortunately, the k-means trainer is included in the main ML.NET NuGet package; therefore, no additional dependencies are required.

K-MEANS CLUSTERING

- With k-means clustering (and other clustering algorithms), the distances between the data point and each of the clusters are the measures of which cluster the model will return.
- For k-means clustering specifically, it uses the center point of each of these clusters (also called a centroid) and then calculates the distance to the data point.
- The smallest of these values is the predicted cluster.



K-MEANS ALGORITHMS

1. In the first stage, we need to set the hyperparameter k. This represents the number of clusters (groups) that k-means clustering will create once it is done.
2. K random vectors are picked up in the feature space. These vectors are called centroids. These vectors are changed during the training process and the goal is to put them into the "center" of each cluster.
3. Distances from each input sample x to each centroid is calculated using some metric, like Euclidean distance. The closest centroid is assigned to each sample in the dataset. Basically, this is where the clusters are created.
4. For each cluster average feature vector is calculated using samples that are assigned to it. This value is considered as a new centroid of the cluster.
5. Step 2-4 are repeated for a fixed number of iteration or until the centroids don't change, whichever comes first.

CREATING THE CLUSTERING APPLICATION

- The application we will be creating is a file type classifier.
- Given a set of attributes statically extracted from a file, the prediction will return if it is a document, an executable, or a script.

AVERAGE DISTANCE

- Also referred to as the average score is the distance from the center of a cluster to the test data.
- The value, of type double, will decrease as the number of clusters increases, effectively creating clusters for the edge cases.
- In addition to this, a value of 0, such as the one found in our example, is possible when your features create distinct clusters.
- This means that, if you find yourself seeing poor prediction performance, you should increase the number of clusters.

THE DAVIES-BOULDIN INDEX

- The Davies-Bouldin Index is another measure for the quality of the clustering.
- Specifically, the Davies-Bouldin Index measures the scatter of cluster separation with values ranging from 0 to 1 (of type double), with a value of 0 being ideal (as was the case of our example).

NORMALIZED MUTUAL INFORMATION

- The normalized mutual information metric is used to measure the mutual dependence of the feature variables.
- The range of values is from 0 to 1 (the type is of double)—closer to or equal to 1 is ideal, akin to the model we trained earlier in this chapter.

ACTIVITY

Categorize iris flowers using k-means clustering with ML.Net

THE END