Data Warehousing And Data Mining

Major Project - Phase 3

INSTRUCTOR: Dr. Kamalakar Karlapalem

- Gunjan Karamchandani (20161086), Neal Karpe (20161159)

Dataset Description

We have chosen the iris dataset for this phase.

Attributes of the dataset

Each sample contains 4 features - *sepal length*, *sepal width*, *petal length*, *petal width* of a type of an Iris plant. An Iris plant is of 3 types - **Iris-setosa**, **Iris-versicolor**, **Iris-virginica**. There are 150 records in the dataset. All the lengths and widths are in cms.

Data Visualization

Data Visualization code available here.

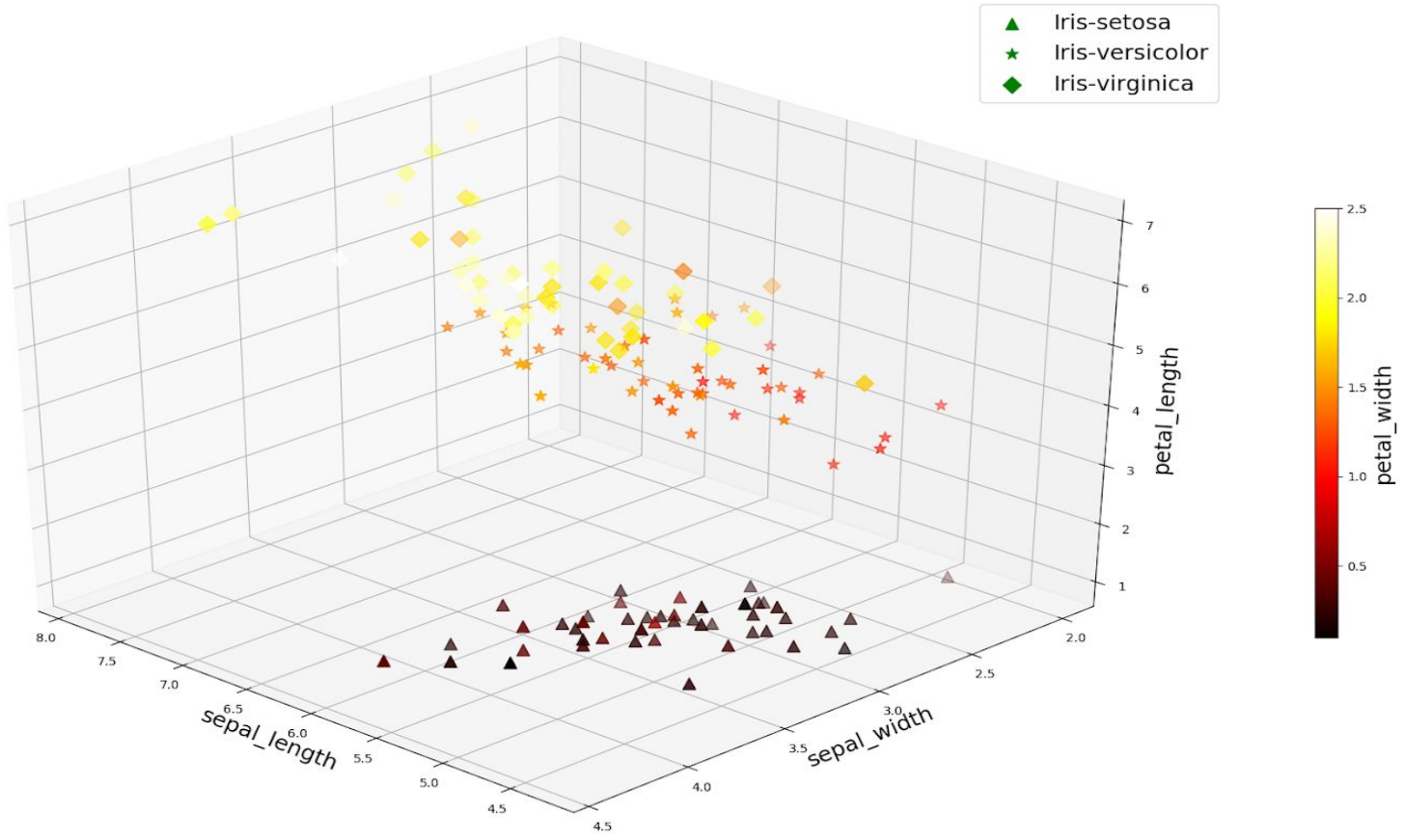

Fig. 1 (Dataset visualization) *Click image to view in browser*

Algorithm used

K-means:

We used K-means clustering algorithm. K-means clustering essentially groups samples into K clusters having similar properties. Each sample belongs to the cluster with the **nearest** mean. The distance function used to measure "nearness" of points is explained below.

Why K-means?

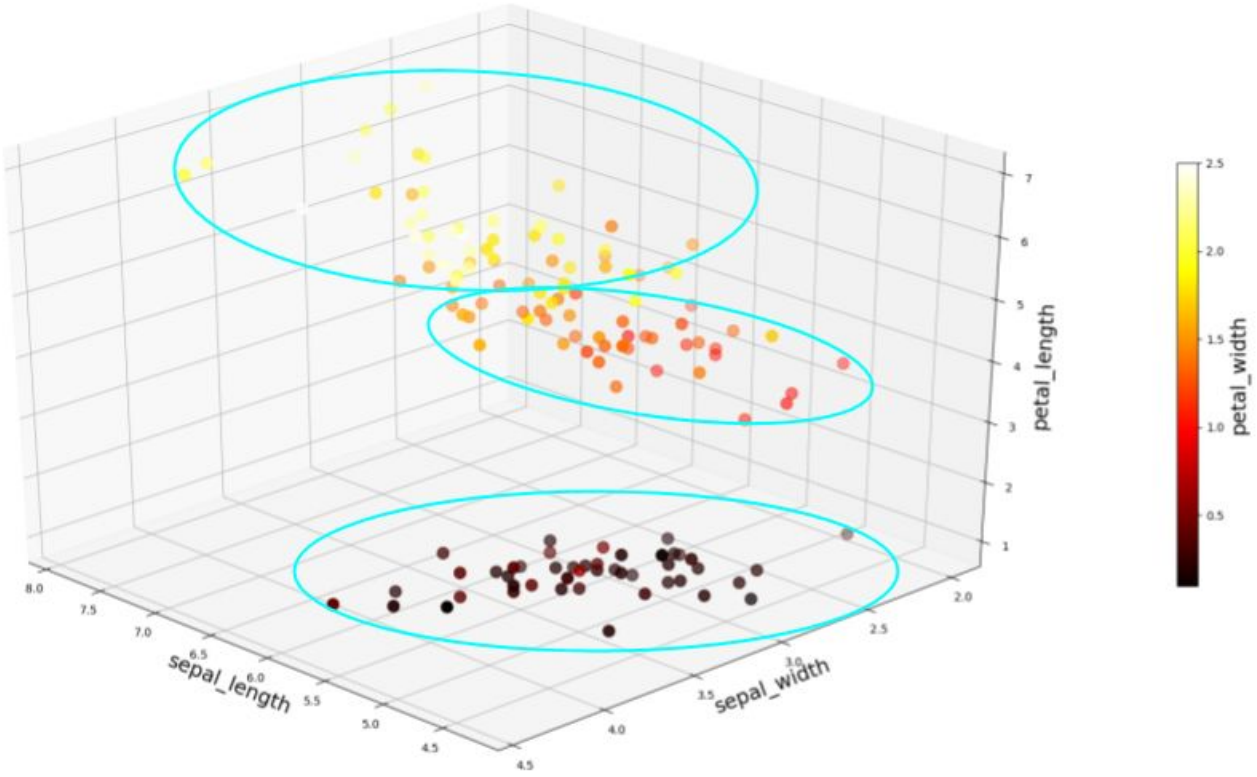

Fig. 2 (Intuitive clusters) *Click image to view in browser*

On visualizing the data (as shown above), we can intuitively see that the samples can be roughly grouped into 3 clusters. Hence K-means clustering with K = 3 was the best suited for this task (given that the number of clusters are known beforehand).

Implementation

We implemented the algorithm using python3, completely from scratch (without the use of external tool). The code is available as ***clustering.py*** in this folder.

Distance Function used

To calculate the distance between 2 vectors, we first **rescale** the vectors followed by which we calculate the **euclidean distance** between the 2 vectors.

Rescaling

We used **min-max normalization** to rescale each feature x as:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)}$$

Why?

In our data, all 4 attributes have different value ranges. When Minkowski distance is used, the attribute with higher value range tends to over-power the attributes with lower value ranges. This gives higher weightage/preference to certain attributes, whereas for our dataset we want equal weightage to all attributes in distance calculation. Hence, we normalize each attribute to the range [0,1].

Euclidean Distance

Post normalization, distance between 2 vectors x and y (which are both 4-d in our case) is :

$$d = \sqrt{(x1' - y1')^2 + (x2' - y'2')^2 + (x3' - y3') + (x4' - y4')^2}$$

Both Euclidean distance and Manhattan distance gave roughly the same results on our data (since we normalized the attribute ranges - attribute overpowering is not an issue). However, we finally chose Euclidean distance because it gave slightly less clustering error.

Results

Observations

With initial centroids as the mean of first 50 sample, mean of middle 50 samples and mean of last 50 samples and with k=3, we got the following clusters:

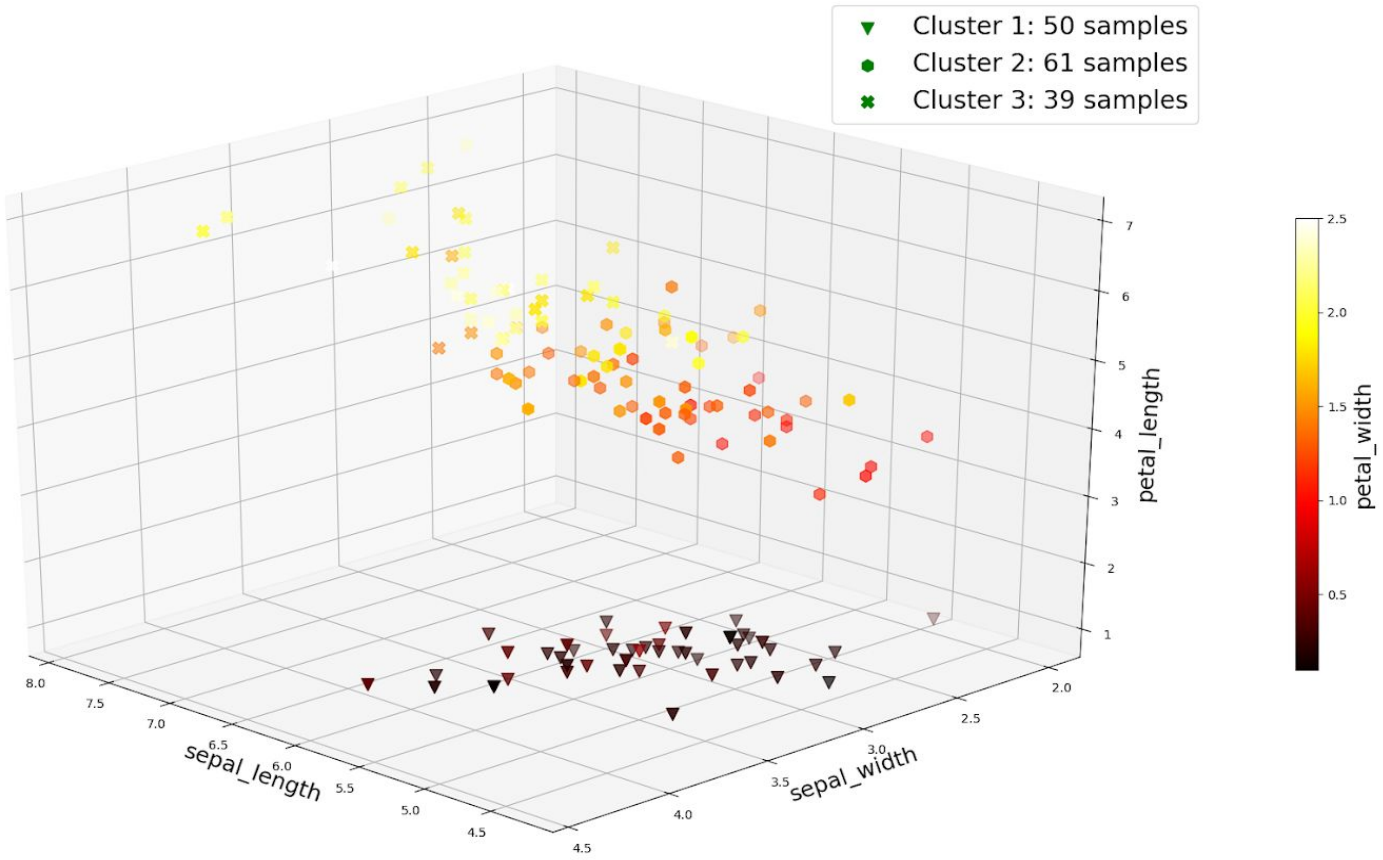

Fig. 3 (Clusters obtained from K-means) *Click image to view in browser*

Explanation

Cluster 1: Iris-setosa - 50
Cluster 2: Iris-versicolor - 47, Iris-virginica - 14
Cluster 3: Iris-versicolor - 3, Iris-virginica - 36

- **Cluster1** contains all *Iris-setosa* plants. In the dataset, all *Iris-setosa* plants have very similar feature values & their feature values are very different from *Iris-versicolor* and *Iris-virginica* plants. In Fig. 1, it can be seen that all *Iris-setosa* samples have roughly the same petal_length and petal_width, but have some variations in sepal dimensions (very little variation along z-axis & c-axis, some variation along x-axis and y-axis).
- **Cluster2** contains almost all the *Iris-versicolor* plants as well as some *Iris-virginica* plants. It can be seen in Fig. 1 that *iris-versicolor* plants typically have medium petal_width and medium petal_length. *Iris-virginica* plants typically have higher petal_width & petal_length. However, there are some *Iris-virginica* samples (14) that have lower petal_width & petal_length, and hence they ended up being clustered along with the *Iris-versicolor* samples.
- **Cluster3** contains the remaining *Iris-virginica* samples and 3 outliers from the *Iris-versicolor* samples.

Since each of the 3 clusters roughly contain samples from the 3 classes, we can say that the results we got from clustering are meaningful. Since the first class *(Iris-setosa)* samples are isolated from the others, it is clear that they all got clustered into a single cluster. The samples of second and third class show some intermixing, so it is not surprising to see samples of class3 in cluster2 and vice versa.