**Exploring IRIS Data set with EDA**

**DATA MINING**

CASE: iris data



**Contents**

[**1. Background 3**](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553965)

[**2. Exploratory Data Analysis 3**](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553972)

2.1. [Sample of Data Set 3](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553974)

2.2. [Summary Statistics 4](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553975)

2.2.1. Data [Category 4](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553975)

2.3. [Histogram and Density Plots](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553976) 5

2.4. [Scatter Plots](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553977) 6

**3.** [**Supervised Learning**](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553979) **8**

3.1. [Problem and Approach](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553980) 8

3.2. [Results and Findings](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553981) 8

3.3. [Conclusion 8](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553982)

**4.** [**Unsupervised Learning**](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553979) **10**

4.1. Problem and Approach11

4.2. K-Means Clustering11

4.3. Major Results from K-means Clustering13

4.4. [Results from Hierarchical Clustering](file:///W:\Employee%20Folders\Onward\Projects\ACAE%20Shanghai%20China\Combuster%20Test%20Facility%20SRS.doc#_Toc391553982) 14

#### **BACKGROUND**

The Iris flower data set or Fisher’s Iris data set is a multivariate data set introduced by Sir Ronald Fisher (1936) as an example of discriminant analysis. It is sometimes called Anderson’s Iris data set because Edgar Anderson collected the data to quantify the morphologic variation of Iris flowers of three related species. Two of the three species were collected in the Gaspe Peninsula and all from the same pasture, and picked on the same day and measured at the same time by the same person with the same apparatus.

#### **Exploratory Data Analysis**

**PROBLEM AND APPROACH:**

The iris data set consists of 150 samples of Iris flowers consisting of 3 species viz. Setosa, Versicolour and Virginica. We have 50 samples from each species and their respective sepal length, sepal width, petal length and petal width in centimeters. In our data set, we have one categorical variable i.e. the flower species, which includes 3 categories and 4 other numeric variables.

We performed the analysis in 3 steps: Exploratory data analysis, K-Nearest Neighbor Classification, and K-Means Clustering. In the first step we start exploring our dataset. We divided our dataset into training and testing data, training data containing 80% of the data points and testing data containing 20% of the data points.  
We then summarized the training data using the summary() function to find out where the mean and median lied for the dataset which measures the location of the set of data points.

Below is a sample of our data set:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Sepal.Length** | **Sepal.Width** | **Petal.Length** | **Petal.Width** | **Species** |
| 5.1 | 3.5 | 1.4 | 0.2 | setosa |
| 4.9 | 3 | 1.4 | 0.2 | setosa |
| 4.7 | 3.2 | 1.3 | 0.2 | setosa |
| 4.6 | 3.1 | 1.5 | 0.2 | setosa |
| 5 | 3.6 | 1.4 | 0.2 | setosa |
| 5.4 | 3.9 | 1.7 | 0.4 | setosa |

Table 1: Sample iris dataset

**Summary Statistics:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Statistic** | **Sepal Length** | **Sepal Width** | **Petal Length** | **Petal Width** |
| **Min.** | **4.4** | **2.2** | **1** | **0.1** |
| **1st Qu.** | 5.1 | 2.8 | 1.6 | 0.3 |
| **Median** | **5.75** | **3** | **4.25** | **1.3** |
| **Mean** | 5.84 | 3.06 | 3.75 | 1.19 |
| **3rd Qu.** | **6.4** | **3.33** | **5.1** | **1.8** |
| **Max.** | 7.9 | 4.4 | 6.9 | 2.5 |

Table 2: Summary of the iris training data

In the above table we can see that the measures have similar values for all attributes except for petal length.

In the next step we visualized different attributes among each other to find out their relationships.

# Histograms and Density Plots:

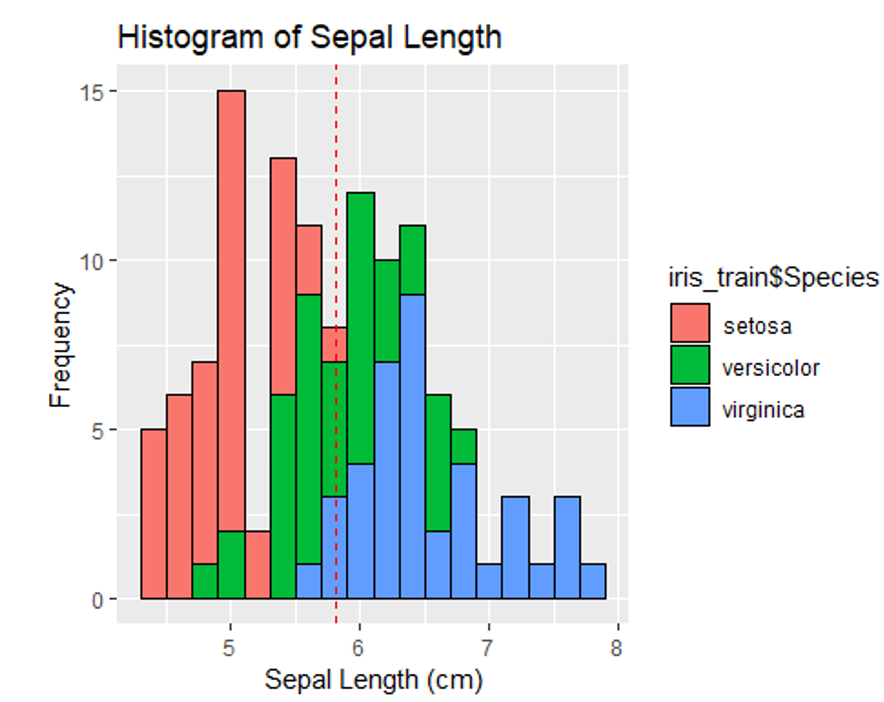


Fig: Histogram of Sepal Length

Looking at the histogram, we see that the sepal length varies of setosa varies from 2 to 6, versicolor from 3 to 7 and virginica from 5.5 to 8cm. We have some flowers with same sepal length among different species. It can also be seen that the highest frequency has sepal length ranging from 4.9 to 5.1cm. It seems that all the sepal length for versicolor and virginica are almost normally distribute while for setosa it seems to be negatively skewed

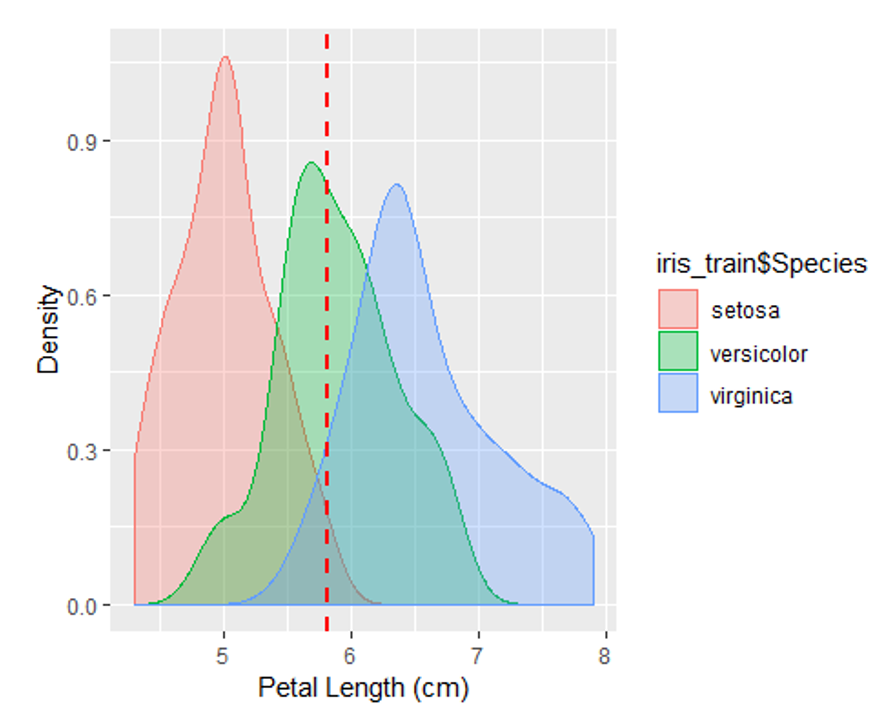


Fig: Density Plot of Petal length

Looking at the density plot we can see that the petal length has some overlapping areas between the three species of iris flowers.

# Scatter Plots:

In order to find out relation between these variables we plot the variables against each other. We first plot sepal length and sepal width as below:

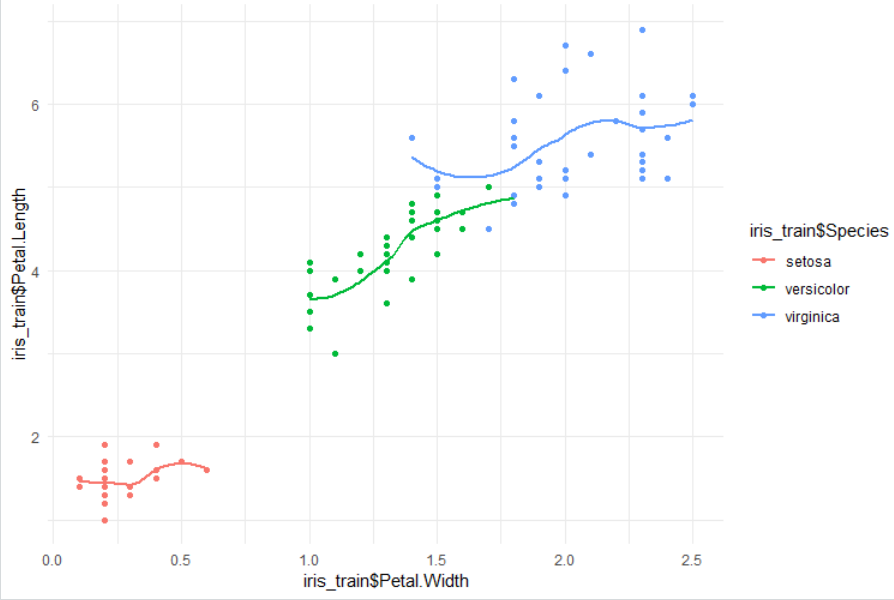


Fig: Scatter Plot of Petal Length and Petal Width

Looking at the above scatter plot, we can predict that the petal length and petal width may not have a strong relationship for virginica, while it has strong correlation for versicolor. We can see that versicolor and virginica have some flowers with similar petal length and petal width while the petal length and width for setosa is comparatively lower that other species. Setosa species flowers have the narrowest petals while virginica has the widest petals.

Also, petal length and petal width seem to have some correlation in each species.

To verify the correlation between the four attributes, we will perform a pair wise correlation plot.

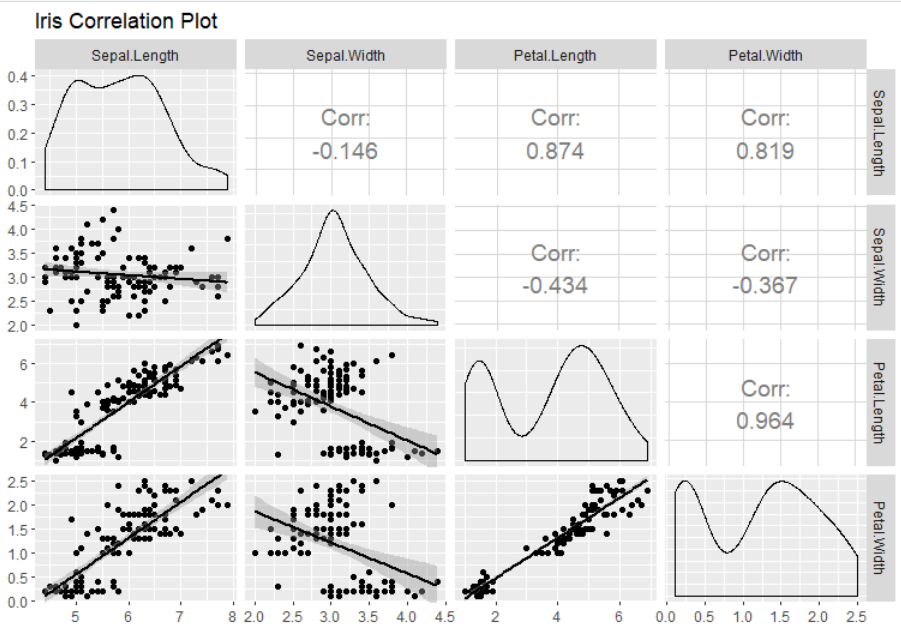


Fig: Iris correlation plot

**RESULTS AND FINDINGS:**

The Petal width and Petal length are highly correlated and has a correlation of 96% and sepal length and petal length have a correlation of 87%. Setosa species flowers have the narrowest petals while virginica has the widest petals. Versicolor has medium sized petals.

1. **Supervised Learning** 
   1. **Problem & Approach**

The last attribute of the data set, Species, will be the target variable which we would be predicting by using the model build using KNN algorithm. The dataset has been divided into training and testing data of which the training data contains 80% of the values while the testing dataset contains 20% of the values. This distribution of the data into training and testing has been done on a random basis. The training set has been used to train the model and the test set has been used to evaluate the trained model.

The result of the summary(iris) function suggests that the Sepal.Length attribute has values between the range 4.3 to 7.9 and Sepal.Width contains values between the range 2 to 4.4, while Petal.Length’s values between the range 1 to 6.9 and Petal.Width between the range 0.1 to 2.5. The values of all attributes are contained within the range of 0.1 and 7.9. So, we do not need to normalize the dataset.

To ensure that all the 3 classes of species are present in the training model, it was required to shuffle the dataset which was achieved using the set.seed() function. This ensured that an equal amount of each of the three species is present in the training and the test dataset in so that the results are not predicted incorrectly in favor any particular species due to its dominance in dataset.

Only the attributes Sepal.Length, Sepal.Width, Petal.Length and Petal.Width were used to build the training and the test data set because the fifth attribute is the one to be predicted using the model. Hence, it was excluded from these two datasets.

After preparing the training and the test data, knn() function was used to find the nearest neighbours for different values of k for the training dataset. This function returned prediction for each row in the test dataset for different values of k. The values were verified using a matrix table which contained the relationship between the actual value and the predicted value. The values on the diagonal of this shows number of correctly classified instances. The values not on the diagonal implies that they have been incorrectly instances.

* 1. **Results and Findings:**

The knn() function was used to predict the values for the test dataset using the trained model. The results for these different k values has been captured in the below Table 1-5 which suggest that trained model was pretty accurate in predicting the values correctly for the trained dataset.

* 1. **Conclusion:**

The trained model predicted correctly for lower values of k using this model. However, for higher values of k, there were prediction errors for one or two observations.(Table 4 and Table 5)

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Table 1** | |  |
| **k=2** | **PREDICTED** | | |
| **TRUE** | setosa | versicolor | virginica |
| setosa | 9 | 0 | 0 |
| versicolor | 0 | 15 | 0 |
| vriginica | 0 | 0 | 6 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Table 2** | |  |
| **k=3** | **PREDICTED** | | |
| **TRUE** | setosa | versicolor | virginica |
| setosa | 10 | 0 | 0 |
| versicolor | 0 | 9 | 0 |
| vriginica | 0 | 0 | 11 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Table 3** | |  |
| **k=5** | **PREDICTED** | | |
| **TRUE** | setosa | versicolor | virginica |
| setosa | 8 | 0 | 0 |
| versicolor | 0 | 10 | 0 |
| vriginica | 0 | 0 | 12 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Table 4** | |  |
| **k=10** | **PREDICTED** | | |
| **TRUE** | setosa | versicolor | virginica |
| setosa | 10 | 0 | 0 |
| versicolor | 0 | 10 | 1 |
| vriginica | 0 | 0 | 9 |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Table 5** | |  |
| **k=16** | **PREDICTED** | | |
| **TRUE** | setosa | versicolor | virginica |
| setosa | 10 | 0 | 0 |
| versicolor | 0 | 9 | 0 |
| vriginica | 0 | 2 | 9 |

**4. Unsupervised Learning**

**4.1** **Problem & Approach**

Unsupervised learning is a branch of machine learning that learns from the data that has not been labeled, classified or categorized. Compared to supervised learning where training data is labeled with the appropriate classifications, models with unsupervised learning must learn relationships between elements in a data set and classify the raw data without “help.”

Clustering is one of the most common unsupervised learning. The organization of unlabeled data into similarity groups is called clusters. A cluster is a collection of data items which are similar between them, and dissimilar to those in other clusters. We aimed to perform clustering with Iris training data while we pretended as the label was unknown by excluding the label variable – Species.

There are 5 classes of clustering methods: 1) partitioning methods, 2) Hierarchical clustering, 3) density-based clustering, 4) model-based clustering, and 5) Fuzzy clustering. Out of these, we aimed to perform, k-means clustering which is one of the most well known partition method, and Hierarchical clustering. For visualization in k-means lustering, Principal Components were used for dimensionality reduction. Elbow Method and Silhouette Method will be used to determine optimal number of clusters.

***4.2* K-Means Clustering**

**Visualization of Principal Component**

There were 4 quantitative variables in the iris data and dimensionality reduction was useful for visualization. Figure 4-1. showed the first Principal Component captured 74.7% of the total variance in the data and the second Principal Component captured 21.4%. Furthermore, it displayed Petal.Length, Petal.Width, and Sepal.Length tended to increase together, but Sepal.Width inversely related.

|  |
| --- |
|  |

**Figure 4-1. Contribution of Variables to Dimensions**

K-means algorithm is to cluster n objects into k partitions, based on attributes/features, where the number of clusters (k) to be generated should be pre-determined (k< n). The grouping is done by minimizing the sum of squares of distances between data and the corresponding cluster centroid. In general, k-means can be applied to data that has a smaller number of dimensions, and variables are numeric and continuous.

K-means has found wide spread usage in lot of fields, market segmentation, computer vision, geostatistics, astronomy and agriculture.

The table below (Figure 4-2.) displayed k-means clustering with several different values of number of clusters (k). “dc1” in x-axis and “dc2” in y-axis denoted two principal components. Although each plot described each scenario with a specific k, it didn’t tell what the optimal number of clusters was for our Iris data.

|  |  |
| --- | --- |
| 1. **Clustering with 10 clusters** | 1. **Clustering with 8 clusters** |
|  |  |
| 1. **Clustering with 5 clusters** | 1. **Clustering with 4 clusters** |
|  |  |
| 1. **Clustering with 3 clusters** | 1. **Clustering with 2 clusters** |
|  |  |

**Figure 4-2. K-means Clustering with Various Partitioning Clusters**

***Determining Optimal Number of Clusters: Elbow Method and Silhouette Method***

First approach we took to identify the optimal number of clusters was the Elbow method. In Elbow method, within-cluster sum of squares at each number of clusters was calculated and graphed. An Elbow method curve displayed a change of slope from steep to shallow to determine the optimal number of clusters as increasing the number of clusters. The Elbow method did not provide an exact cut-off value, but still very useful to make decision. Figure 4-3. (A) indicated the slop did not change much after 3 clusters. In other words, adding additional clusters beyond third became less meaningful.

|  |  |
| --- | --- |
| 1. **Elbow Method** | **(B) Silhouette Method** |

**4-3. Elbow vs Silhouette Methods**

Next approach to determine the optimal number of clusters was called the Silhouette method. Silhouette method calculated the average silhouette of observations for different values of k. The optimal number of clusters was the one that optimize the average silhouette over a range of possible values of k. Based on Figure 4-3. (B), suggested an optimal of 2 clusters. Silhouette method

**4.3 Major Results from K-Means Clustering**

The plots in Figure 4-4. display k-means with two optimal numbers (k=2 and 3), and a number at each dot represents the respective row number in Iris data. Elbow method suggested 3 clusters in iris training data and Silhouette method suggested only 2 clusters.

|  |
| --- |
|  |

**Figure 4-4. k-means with 3 vs 2 Clusters**

**4.4 Results from Hierarchical Clustering**

Hierarchical clustering is a method of cluster analysis, and it builds a hierarchy of clusters. A dendrogram represents a diagram of tree, and initially all data is in the same cluster, and the largest cluster is split until every object is separate. The plots in Figure 4-5. display dendrograms with two optimal numbers (k=2 and 3).

|  |  |
| --- | --- |
| 1. **With 3 clusters** | 1. **With 2 clusters** |

**Figure 4-5. Hierarchical Clustering with 3 vs 2 Clusters**

It was easier to decide on the number of clusters by looking at the dendrogram. However, each data point in an end node was not distinguishable because of too many data points. Perhaps, Hierarchical clustering should be more suitable for small data set.