Pavlo Nikolaiev

201ADB090

Practical 3

The repository <https://github.com/Pashost/Practical_2_AI>

A picture containing diagram, line

Description automatically generated

My final work

Contents

[Part1 2](#_Toc135603987)

[Part 2 11](#_Toc135603988)

[Part 3 13](#_Toc135603989)

# Part1

Title name is - Dry Bean Dataset

I got the database with 13K value I deleted some of then and in the end I got 2K values In the discerption 0 information about this so I made this I deleted the random values, this will not affect the operation of the database and in the future in this report I will write what is in this database 2K values.

Link for my database - <https://archive.ics.uci.edu/ml/datasets/Dry+Bean+Dataset>

Authors :

Murat KOKLU

Faculty of Technology,

Selcuk University,

TURKEY.

ORCID : 0000-0002-2737-2360

[mkoklu@selcuk.edu.tr](mailto:mkoklu@selcuk.edu.tr)

Ilker Ali OZKAN

Faculty of Technology,

Selcuk University,

TURKEY.

ORCID : 0000-0002-5715-1040

[ilkerozkan@selcuk.edu.tr](mailto:ilkerozkan@selcuk.edu.tr)

a way how the dataset was collected:

Images of 2019 grains of 7 different registered dry beans were taken with a high-resolution camera. A total of 16 features; 12 dimensions and 4 shape forms, were obtained from the grains.

the number of data objects in the dataset – 2019

a representation of the features (attributes) of the dataset together with their roles in the Orange tool;

Attribute Information:

1.) Area (A): The area of a bean zone and the number of pixels within its boundaries.

2.) Perimeter (P): Bean circumference is defined as the length of its border.

3.) Major axis length (L): The distance between the ends of the longest line that can be drawn from a bean.

4.) Minor axis length (l): The longest line that can be drawn from the bean while standing perpendicular to the main axis.

5.) Aspect ratio (K): Defines the relationship between L and l.

6.) Eccentricity (Ec): Eccentricity of the ellipse having the same moments as the region.

7.) Convex area (C): Number of pixels in the smallest convex polygon that can contain the area of a bean seed.

8.) Equivalent diameter (Ed): The diameter of a circle having the same area as a bean seed area.

9.) Extent (Ex): The ratio of the pixels in the bounding box to the bean area.

10.)Solidity (S): Also known as convexity. The ratio of the pixels in the convex shell to those found in beans.

11.)Roundness (R): Calculated with the following formula: (4piA)/(P^2)

12.)Compactness (CO): Measures the roundness of an object: Ed/L

13.)Class (Seker, Barbunya, Bombay, Cali, Dermosan, Horoz and Sira)

The number of classes in the dataset, the meaning of each class and the way of representing classes (explanation of the labels assigned to classes); if the data set provides several possible data classifications, then the report must identify which classification is considered in the assignment; o the number of data objects belonging to each class;

In this database we have 7 classes - Seker, Barbunya, Bombay, Cali, Dermosan, Horoz and Sir.The name of classes it is the sorts of dry beams.

the number of data objects belonging to each class;

Seker – 87;

Barbunya – 149;

Bombay – 522;

Cali – 514;

Horoz – 289;

Sir – 38;

Dermosan – 418;

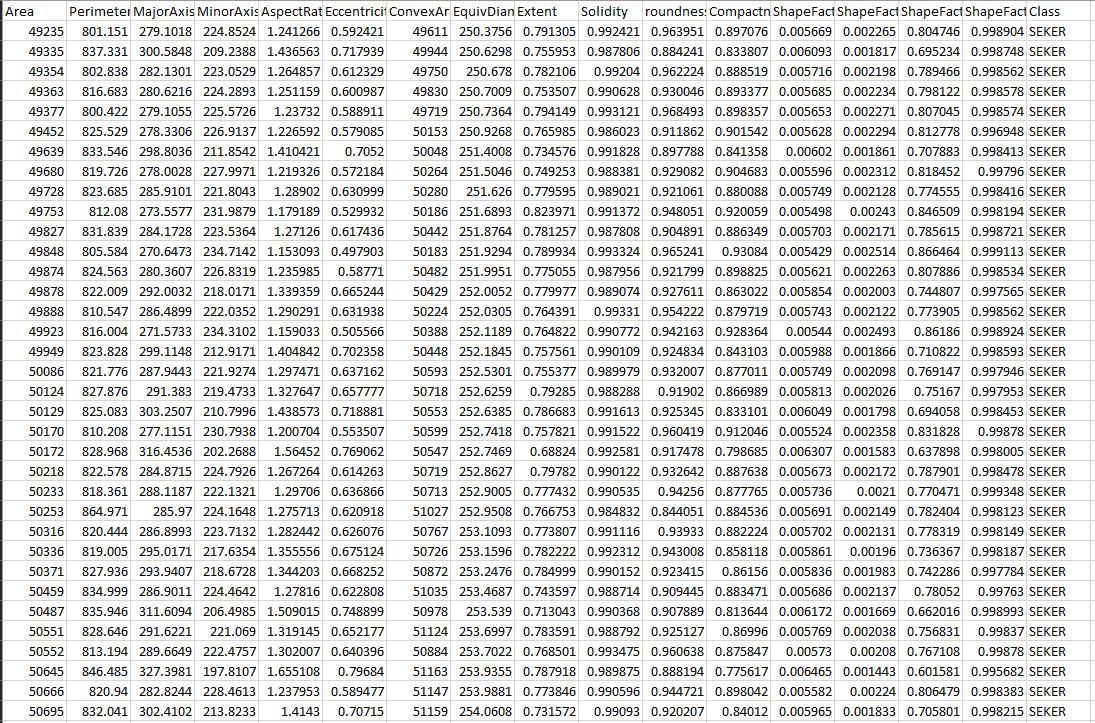
the number and meaning of features in the dataset, as well as their value types and ranges (this information should be presented in a table consisting of the feature representation, its meaning, value type and range of values available in the dataset);

A screenshot of a computer

Description automatically generated with medium confidence

|  |  |  |  |
| --- | --- | --- | --- |
| features | Min | Max | Type |
| Area | 21558 | 254616 | Int |
| Perimeter | 531.318 | 1985.37 | float |
| MajorAxisLength | 187.1686 | 738.8602 | float |
| MinorAxisLength | 122.5127 | 460.1985 | float |
| AspectRation | 1.148116 | 2.387395 | float |
| Eccentricity | 0.491297 | 0.908048 | float |
| ConvexArea | 21808 | 263261 | float |
| EquivDiameter | 165.6759 | 569.3744 | float |
| Extent | 0.581852 | 0.85842 | float |
| Solidity | 0.944568 | 0.994678 | float |
| roundness | 0.607867 | 0.990685 | float |
| Compactness | 0.646562 | 0.932794 | float |

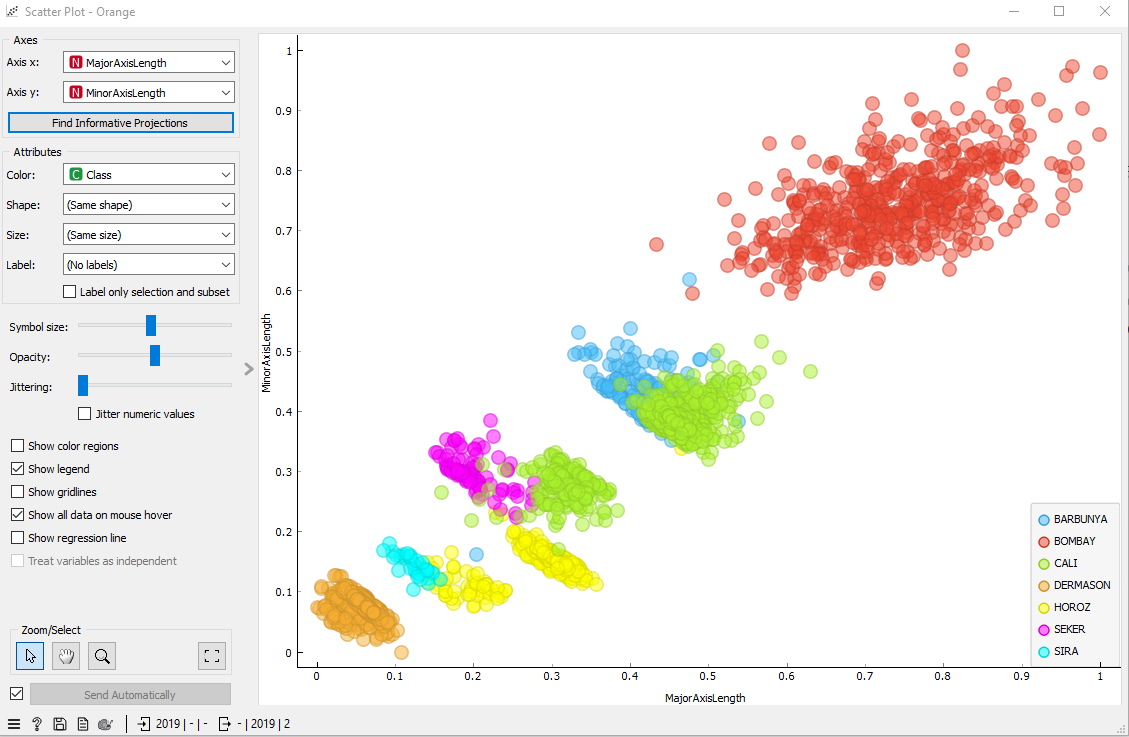
a snippet of the structure of your datafile in which the columns of your datafile and class labels are shown together with some data objects;



Figure

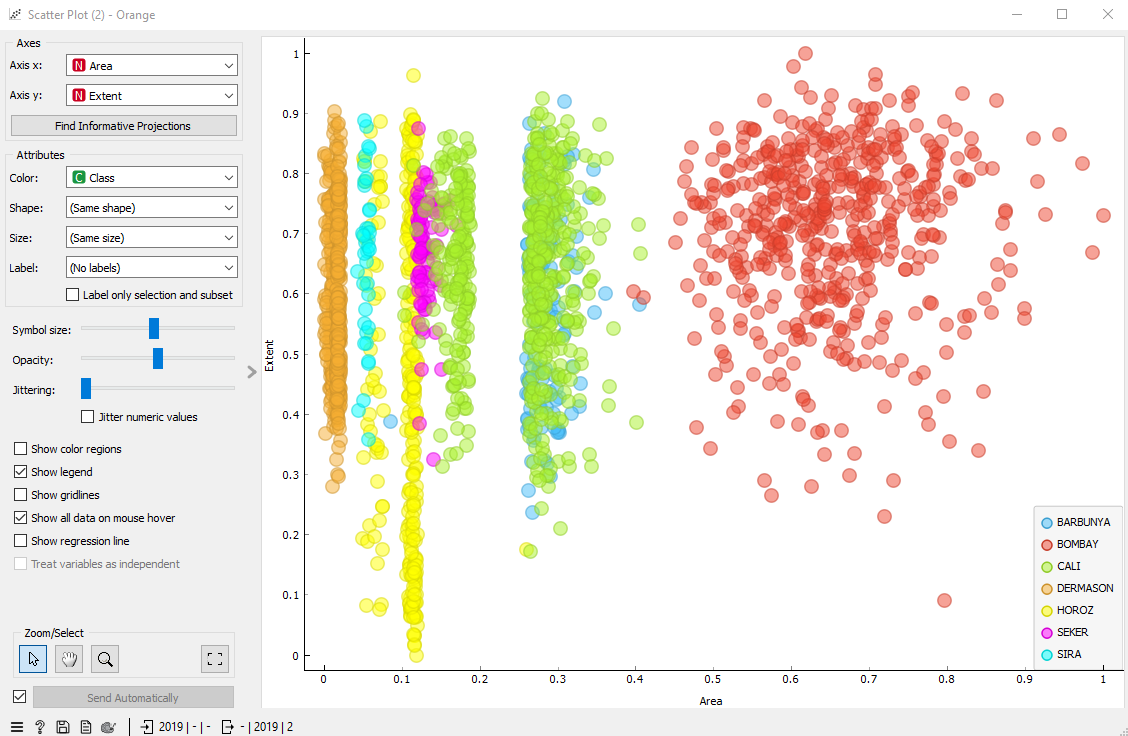
Representing your training dataset visually and statistically:

You must create at least two 2- or 3-dimensional scatter plots illustrating the separability of classes in your dataset based on different features; the student should avoid using the data object ID or a class label as a variable in the scatterplot



Figure

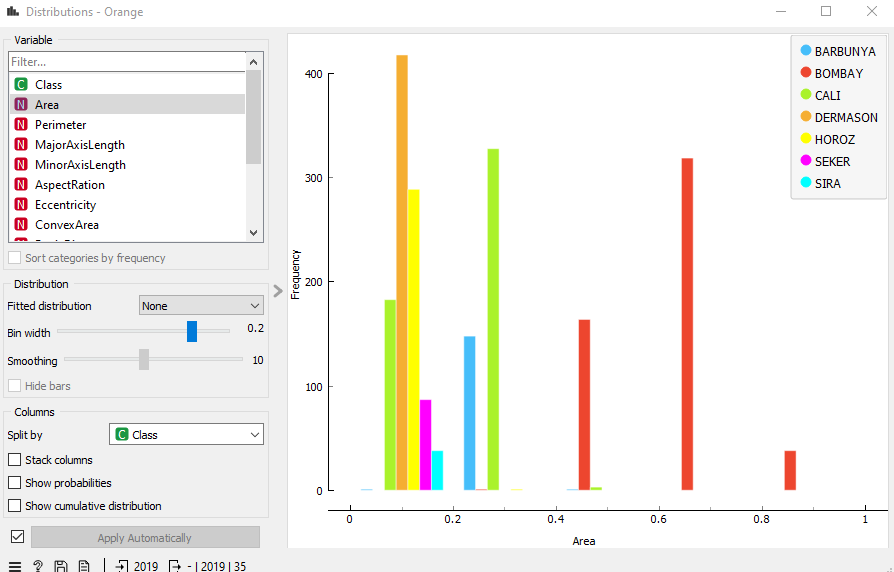
By graphing the major axis length (L) against intensity (I), we gain insight into how bean intensity varies with respect to their major axis length. Such a plot may reveal any patterns or links between these two variables.



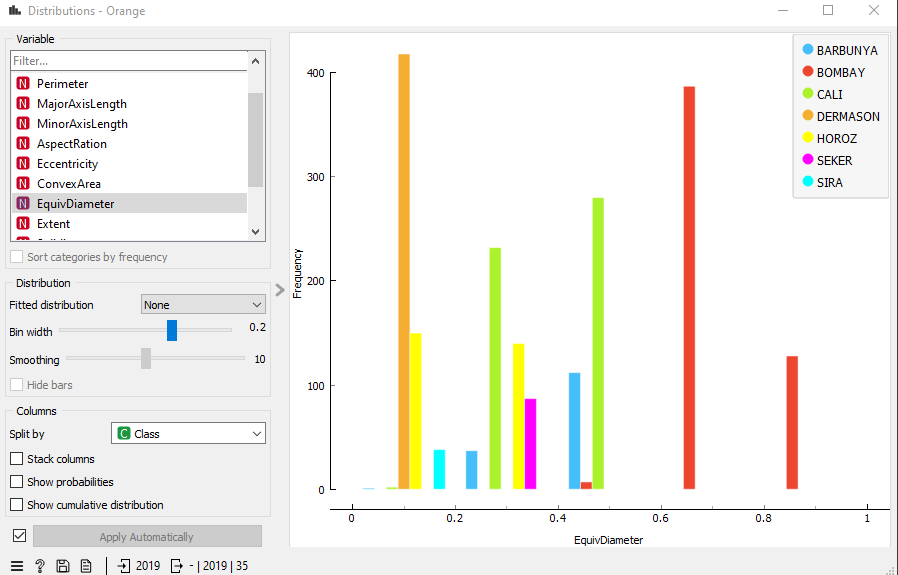
Figure

If we depict Area (A) on the x-axis and Extent (Ex) on the y-axis, a plot can demonstrate how changes in bean size impact their extent. By analyzing pixel ratios within a given bounding box, this visualization may reveal relationships or patterns between these two measured variables.

you must create at least 2 histograms showing the separation of classes for the features of interest;

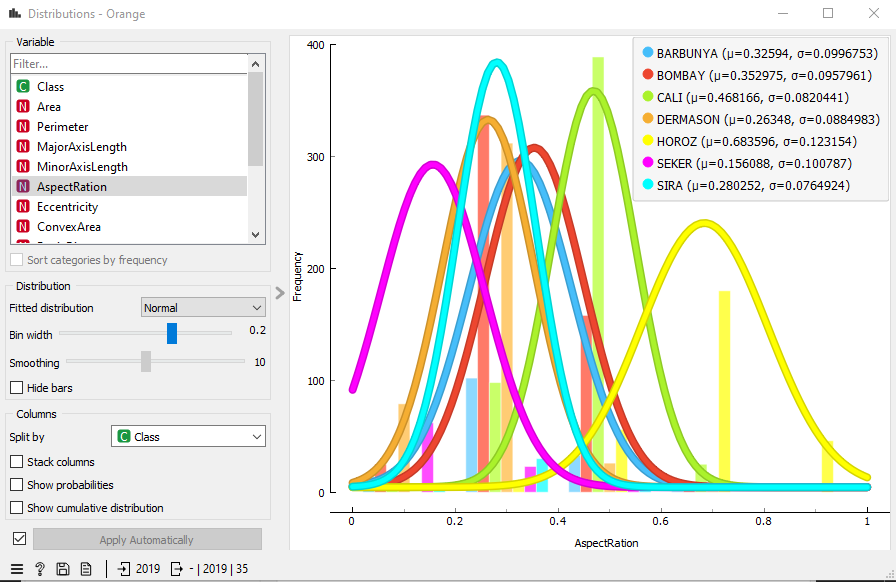


Figure

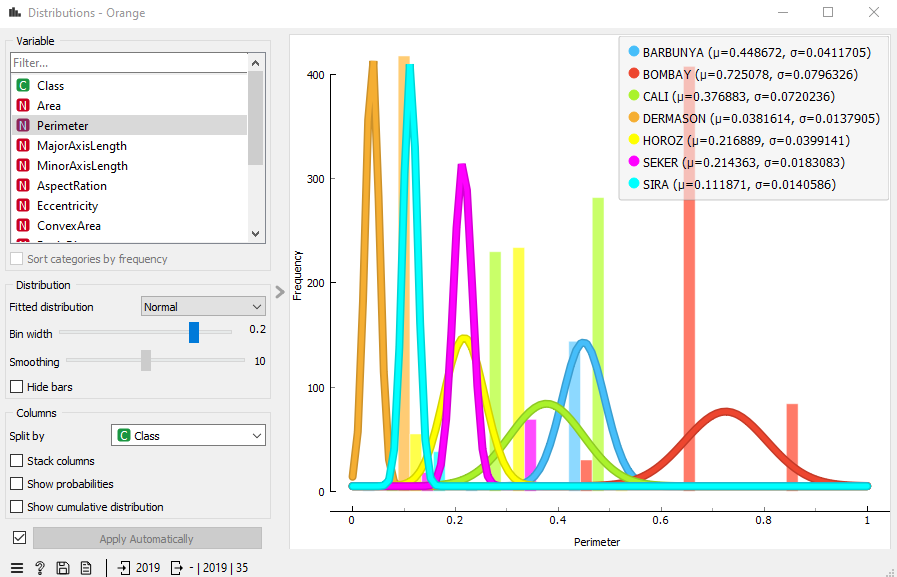


Figure

you must show 2 distributions for the features of interest;



Figure

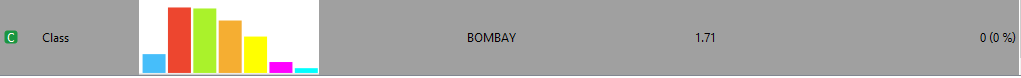


Figure

you must calculate statistics on your data (at least the central tendency and the dispersion of the feature values).



Figure



Figure

1- conclusions coming from the analysis of scatter plots, histograms and distributions (from Step 5 in Part I) about the separability of your classes (remember to include your graphs in the report). Try to answer the following questions:

Are classes in your dataset balanced, or is one class (several classes) prevailing? It is determined by how many data objects belong to each class: In my classes, the entered data from the video camera and they cannot have the same values, in my data, most of the data is in the BOMBAI class, namely 522 values.

2- Does the visual representation of the data allow the structure of the data to be seen? It is a question of whether data objects belonging to different classes are separable.

The diagram shows that the product data and if you make more comparisons, then some varieties will stand out more than others.

3- Yes, the beans differ only visually and there is nothing more.

No ,identification signs are far apart.

Conclusions coming from the analysis of statistical calculations (central tendency and dispersion).

Based on my data, I realized that my data is asymmetric and also that there is a lot of variability and spread in feature values.

# Part 2

Perform at least 3 experiments with Hierarchical clustering, freely moving the cut-off line and analyzing how the number and content of clusters change;

In this algorithm we will define the

A screenshot of a computer

Description automatically generated with medium confidence

Figure

In the first slide we see the cut-off line placing in the end of diagram and do not cut anything and like shown in the screen we have all the results consist of.

A screenshot of a computer

Description automatically generated

Figure

In the second screen shown that cut-off line placing in the 73.8% saving the data which values is bigger then the max in line that’s values will not accepted. And we see the different Class BARBUNYA,CALI,SEKIR and SIRA did not changed because the values was to small.

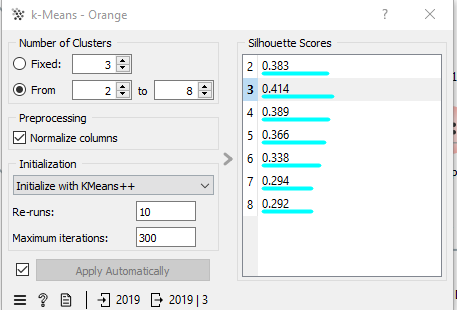
A screenshot of a computer

Description automatically generated with medium confidence

Figure

And the same situation with the second screen ,here the cut-of line saving the 30,1% out of the biggest value. And here we see only 1 class left because this class has the smallest values on dataset.

For the K-means algorithm, calculate the Silhouette Score for at least 5 k values, and analyze the performance of the algorithm.



Figure

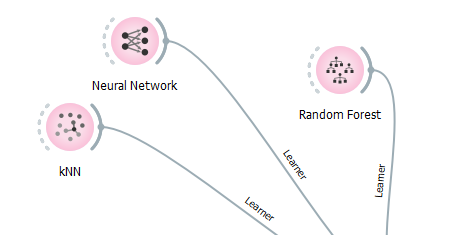
In this algorithm we have 7 clusters 2 to 8 . The Clusters has range -1 to 1 and here we see only positive numbers that shown the clusters work correctly .In the slide shown – That clusters is stable because the different is little bit more than 0.1.The most powerful cluster it is the number 3 then 4,2,5,6,7 and 8.

# Part 3

Choose at least two supervised learning methods suitable for the classification task. You can use the methods

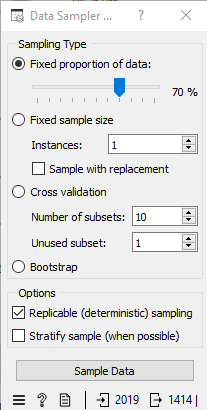
considered in class and other algorithms intended for classification

I chouse to use the kNN algorithm (k-nearest neighbors algorithm) this is one of the simplest analysis algorithms, regression problems also sometimes arise. The neural network this is an algorithm that tries to artificially replicate the work of the brain. And Random Forest - using this algorithm, you can solve classification and regression problems, an algorithm that uses the results of several trees for one result.



Figure

Divide your dataset into training and test sets.



Figure

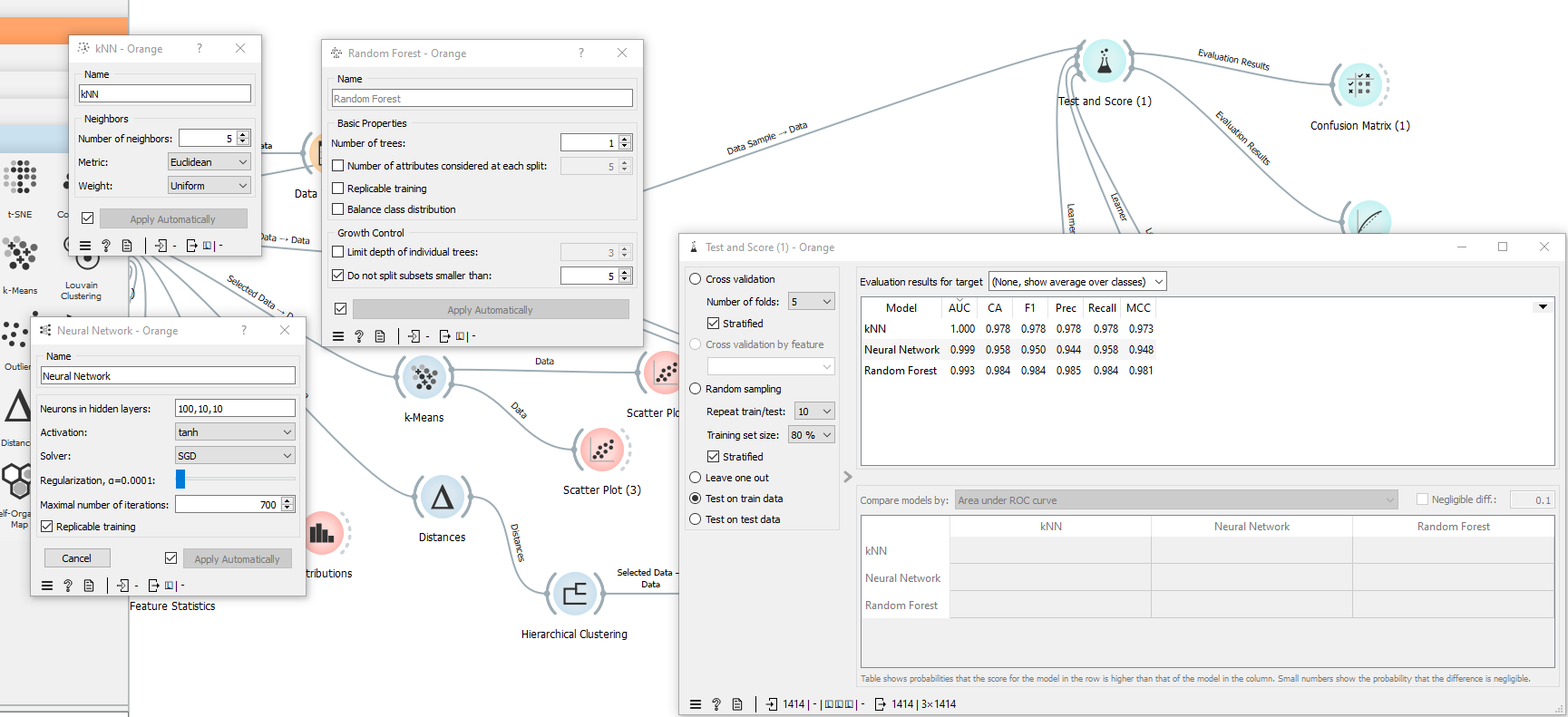
A screenshot of a computer

Description automatically generated with medium confidence

Figure

In the slides seems that I divided the database in to the training and test sets. The 30% will work like training data and 70% will be test data.

For each algorithm, perform at least 3 experiments using the training dataset, changing the values of the algorithm hyperparameters and analyzing the algorithm performance metrics.



Figure

A screenshot of a computer

Description automatically generated with medium confidence

Figure

A screenshot of a computer

Description automatically generated with medium confidence

Figure

For each algorithm, choose the trained model that provides the best algorithm performance.

kNN

The best ,was values from the first slide

Number of neighbors = 5;

Metrix = Euclidean

Neural Network

The best, was values from the first slide

Neurons hidden layers = 100,10,10

Activation = tanh

Solder = SGD

Maximum number of iterations = 700

Random forest

The best ,was values from the last slide

Number of trees = 5;

The parameters for hyperparameters was taken randomly.

Apply the trained model of each algorithm to the test dataset

A screenshot of a computer

Description automatically generated

Figure

Test dataset.

A screenshot of a computer

Description automatically generated with medium confidence

Figure

Training dataset.

Evaluate and compare the performance of the trained models.

From the first test model that the performance of kNN and Neural Network are powerful then Random forest.

Because the random forest need to use as many trees as possible for getting the better results like shown in the slide number 3 but for kNN the smaller the number of neighbors will be better and it shown on the results. For Neural Network for mine dataset - as many neurons possible then better algorithm will work. Because in my dataset I have to many values and it is a reason to have many neurons to work faster and better.

1. I chose the kNN algorithm because of its simplicity and clarity. Its principle of operation is first it finds the closest neighbors to a given data point in the feature space and assigns the label of the majority class among neighboring labels to the data point.
2. (https://en.wikipedia.org/wiki/K-nearest\_neighbors\_algorithm)
3. I chose a random forest algorithm because it can efficiently process data with a large number of values, it works according to the principle - it automatically selects random features for each tree, reducing the risk of overfitting and improving the ability of the model to generalize
4. (https://www.ibm.com/topics/random-forest#:~:text=Random%20forest%20is%20a%20commonly,both%20classification%20and%20regression%20problems.).

Description of the hyperparameters available in the Orange tool and their meaning for each algorithm:

kNN

Number of neighbored – the number of neighbored which will include in tests.

Weight - has parameters Uniform which means that all neighbors will be equal or By Distances which means that preference will be given to the closest.

Random Forest

Number of trees – Numbers of trees which will work on the algorithm.

the total number of data objects added to the test and training datasets (by number and %);

to the test - 605 – 30%

to the training 1414 – 70%

to classes - 30% to test and 70% to training and I did not found how many data in each class.

Conclusion

In this experiment, I followed the data from -

(https://orangedatamining.com/widget-catalog/evaluate/testandscore/) and came to the conclusion that I should look at the AUC and F1 values (https://darwin.unmc.edu/dxtests/roc3.htm) this information helped to decide which of the algorithms works better and which hyperparameters are best used to improve the results of the program.

Refences

<https://www.wikipedia.org/>

<https://orangedatamining.com/>

<https://darwin.unmc.edu/dxtests/roc3.htm>

<https://www.ibm.com/topics/random-forest#:~:text=Random%20forest%20is%20a%20commonly,both%20classification%20and%20regression%20problems>

Thanks for watching my work.