Name: Pallavi Suhas Joshi

PRN: 2019BTECS00001

Batch: T3

Date: 2/2/2022

Software Engineering Tools Lab

Assignment 1

Q.1

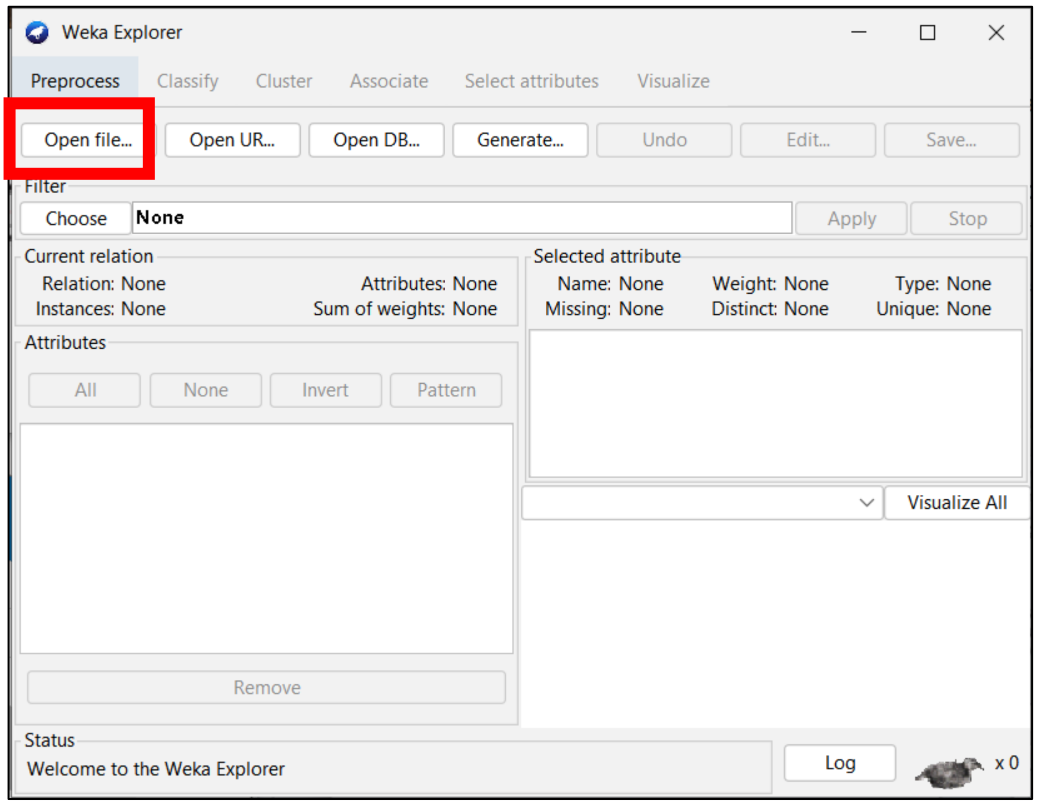
The Iris dataset given

https://drive.google.com/file/d/1A3Fxsfzm6BSfhFZGDrjl47RTe45bSgYP/view

1. Data Pre-processing:

The file in the csv format is converted into arff format. Open the Weka software.

Choose the 'Explorer' tab. Following window will pop-up.



None

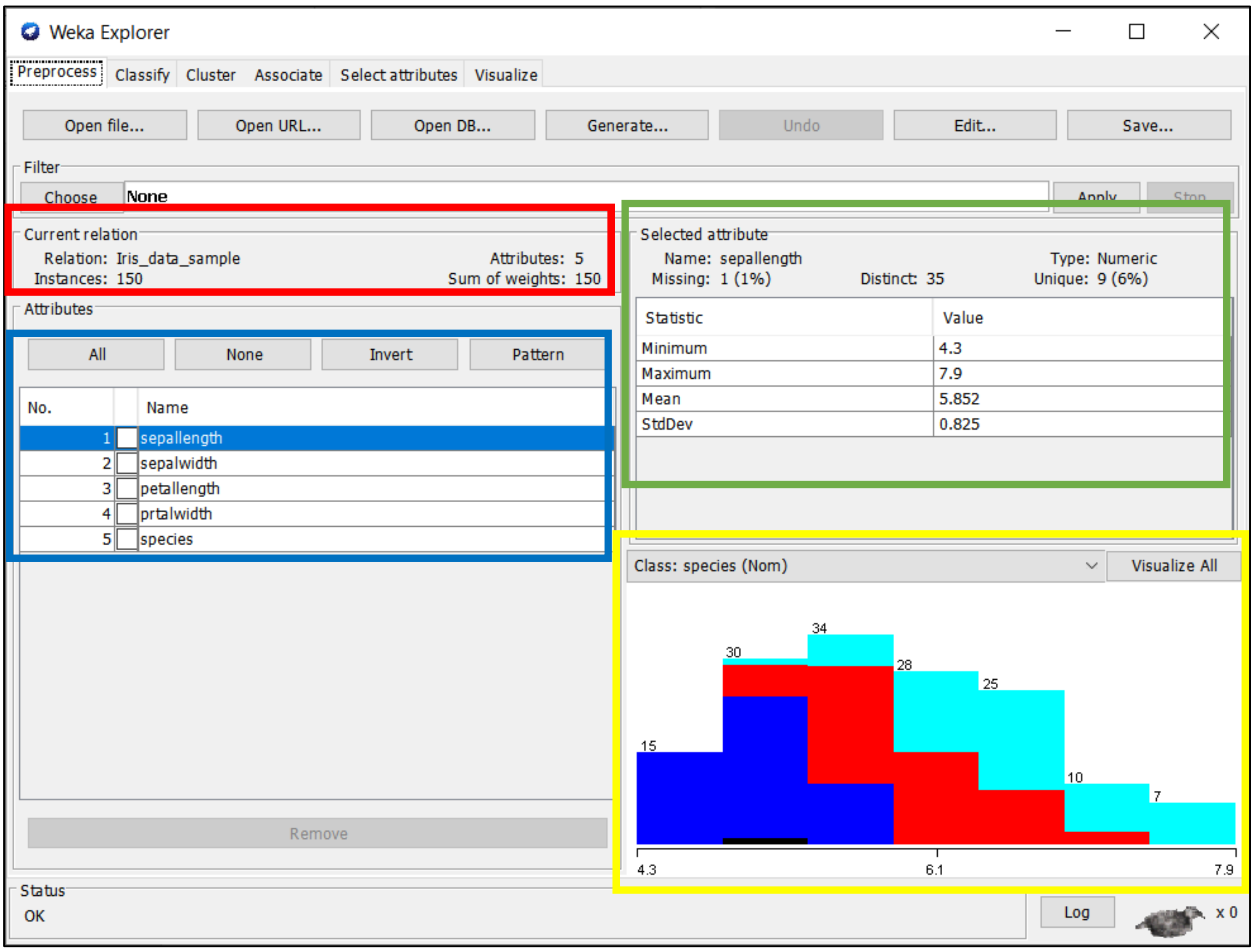
Relation: None
Instances: None
Attributes: None
Sum of weights: None

Name: None
Missing: None
Weight: None
Distinct: None
Type: None
Unique: None

Welcome to the Weka Explorer

Choose the tab 'Open file' and upload the 'Iris_data_sample.arff' file.

When you open the file, your screen looks like as shown here giving information about the loaded data -



Filter: None

Current relation
Relation: Iris_data_sample   Attributes: 5
Instances: 150   Sum of weights: 150

Attributes

No.		Name
1	☐	sepallength
2	☐	sepalwidth
3	☐	petallength
4	☐	prtalwidth
5	☐	species

Selected attribute
Name: sepallength   Type: Numeric
Missing: 1 (1%)   Distinct: 35   Unique: 9 (6%)

Statistic	Value
Minimum	4.3
Maximum	7.9
Mean	5.852
StdDev	0.825

Class: species (Nom)

Status
OK

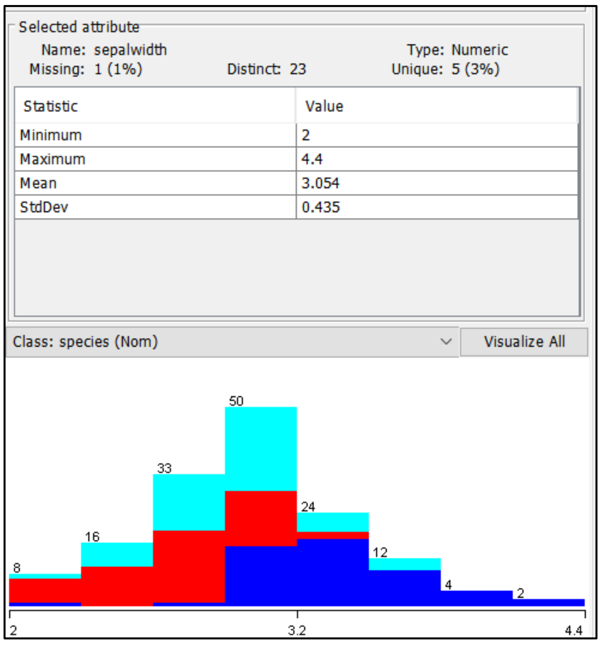
The **Current relation** sub-window (in red box) shows the name of the database loaded. From this window we can see that –

a. There are 150 instances in dataset.
b. The dataset consists of 5 attributes.

Below that **Attributes** sub-window (in blue box) display various fields in dataset.

The dataset consists of 5 fields – sepallength, sepalwidth, petallength, petalwidth, species.

When we click on the attribute, we get detailed information in right hand side sub-window.

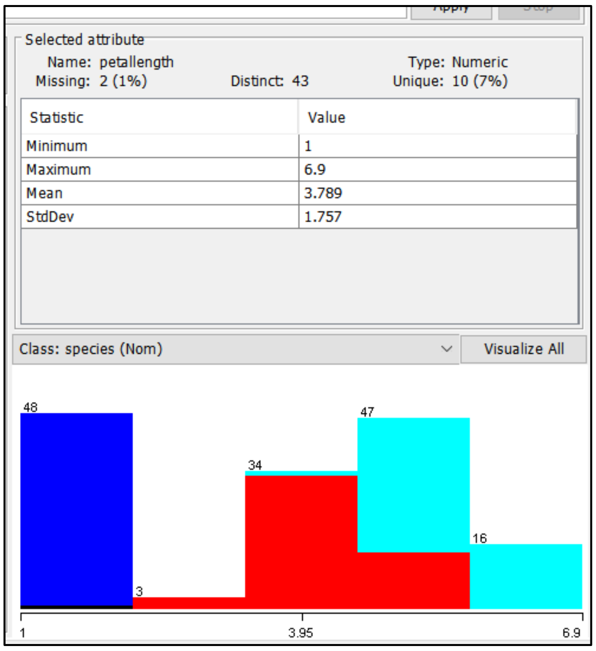


Selected attribute

Name: sepalwidth | Type: Numeric
Missing: 1 (1%) | Distinct: 23 | Unique: 5 (3%)

Statistic	Value
Minimum	2
Maximum	4.4
Mean	3.054
StdDev	0.435

Class: species (Nom)

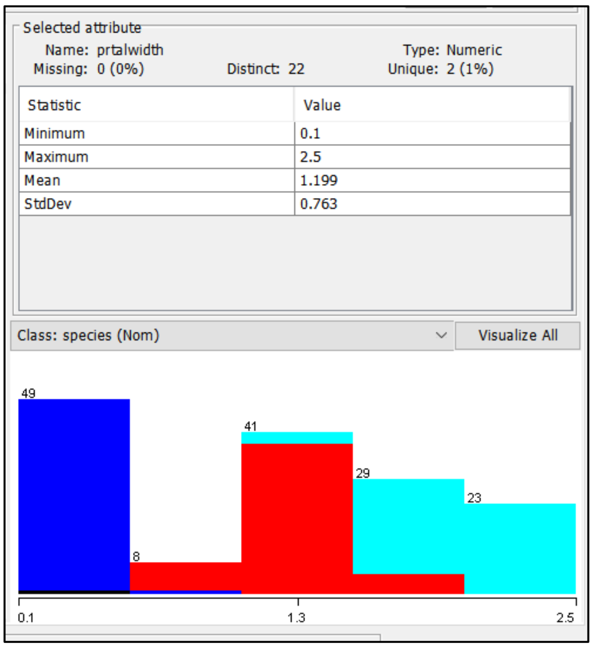


Selected attribute

Name: petallength | Type: Numeric
Missing: 2 (1%) | Distinct: 43 | Unique: 10 (7%)

Statistic	Value
Minimum	1
Maximum	6.9
Mean	3.789
StdDev	1.757

Class: species (Nom)

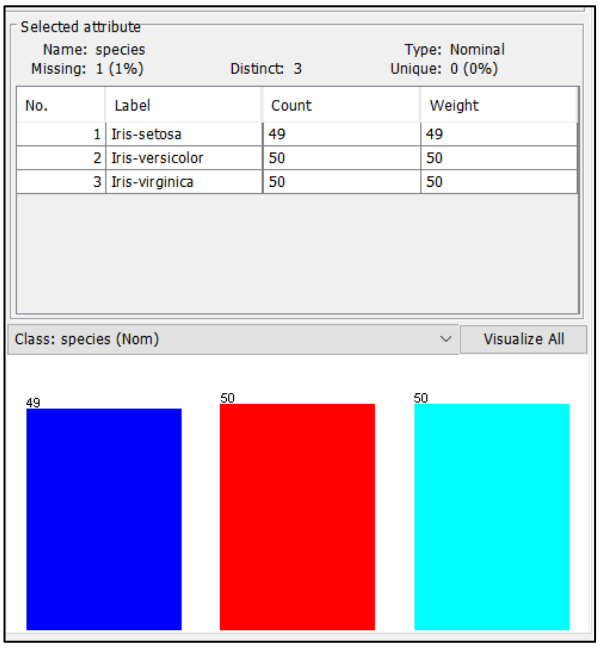


Selected attribute

Name: prtalwidth | Type: Numeric
Missing: 0 (0%) | Distinct: 22 | Unique: 2 (1%)

Statistic	Value
Minimum	0.1
Maximum	2.5
Mean	1.199
StdDev	0.763

Class: species (Nom)



Selected attribute

Name: species | Type: Nominal
Missing: 1 (1%) | Distinct: 3 | Unique: 0 (0%)

No.	Label	Count	Weight
1	Iris-setosa	49	49
2	Iris-versicolor	50	50
3	Iris-virginica	50	50

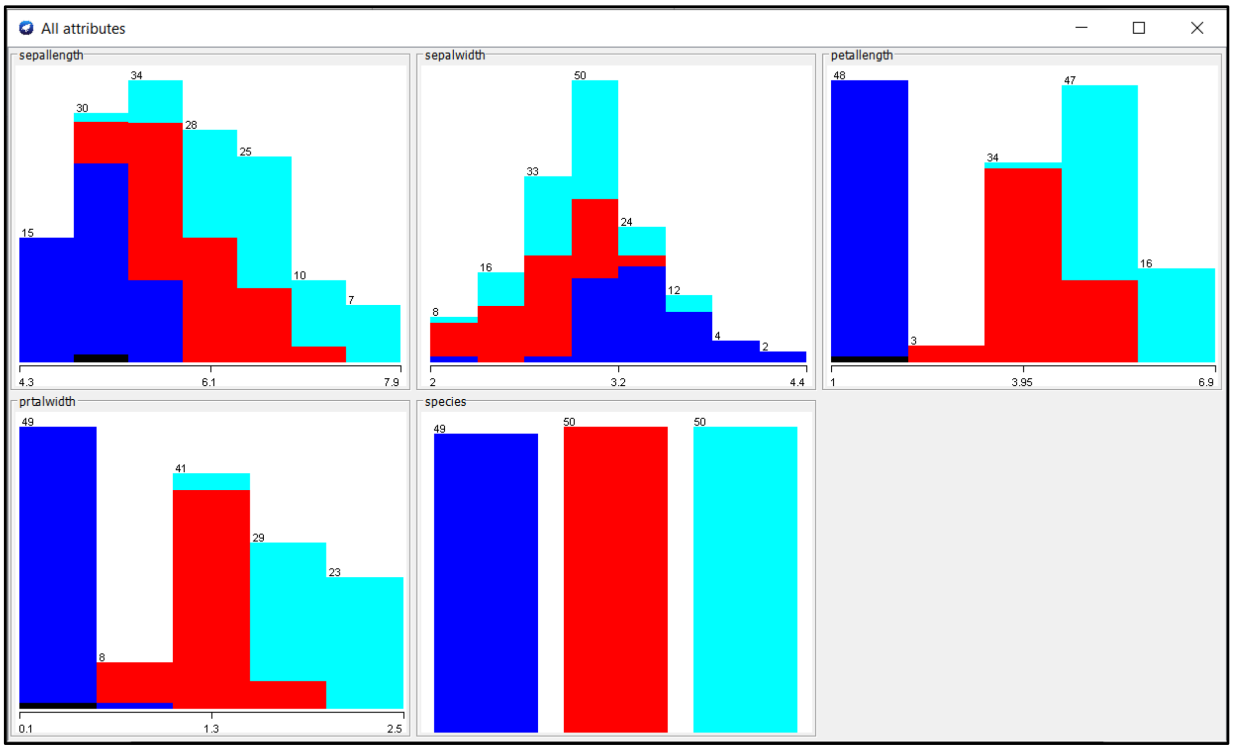
Class: species (Nom)

The **Selected Attribute** sub-window shows –

a. Name and type of attribute
b. Number of missing values, distinct values and unique values.

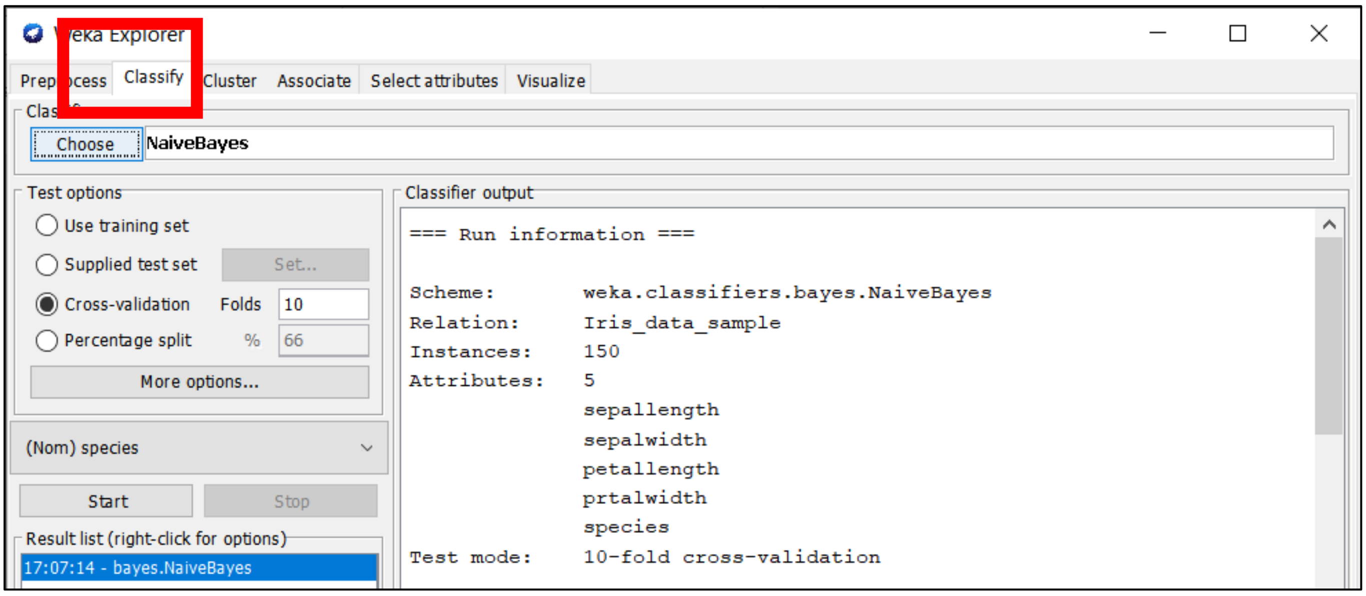
At the bottom of the window, there is the visual representation of the **Class** values.

If we click on **'Visualize All'** all features can be seen in one single window. Following window will pop-up:



2. Data Classification:

Click on the **'Classify'** tab. Following window will pop-up.



NaiveBayes

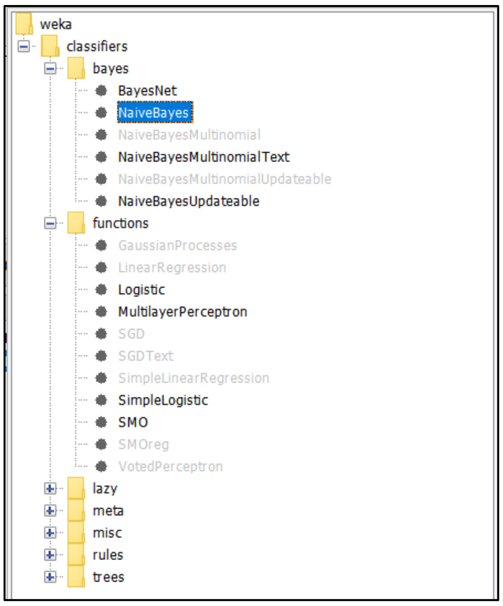
Test options: Cross-validation [✓], Folds 10; Percentage split % 66

(Nom) species

17:07:14 - bayes.NaiveBayes

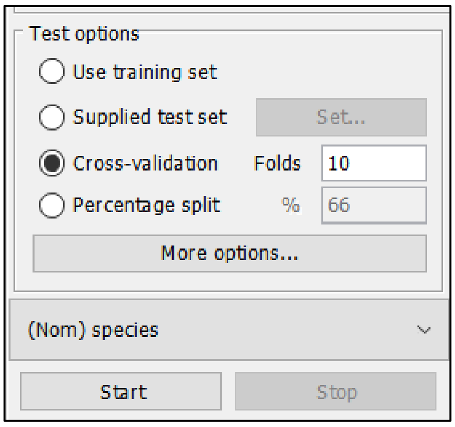
```
=== Run information ===

Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     Iris_data_sample
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              prtalwidth
              species
Test mode:    10-fold cross-validation
```

From there click on **'Choose'** to pick the algorithm for classification. Following are the various algorithms Weka provides.



Here Naive Bayes algorithm is used. Traditionally this algorithm assumes that the input values are nominal, although it numerical inputs are supported by assuming a distribution. Naive Bayes uses a simple implementation of Bayes Theorem (hence naive) where the prior probability for each class is calculated from the training data and assumed to be independent of each other (technically called conditionally independent).

Next is the **'Test options'**. There are 4 options provided. Here Cross-validation is chosen. Then click on start.


Test options

Use training set

Supplied test set

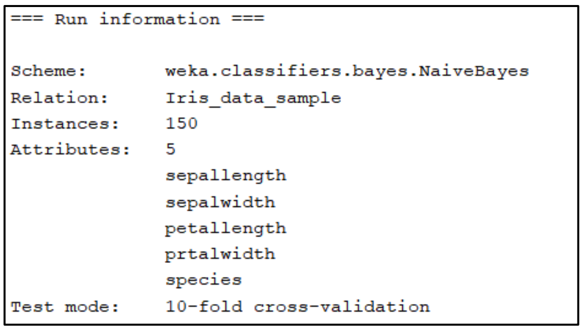
[✓ Cross-validation] Folds 10

Percentage split % 66

(Nom) species

Classification output can be seen on the right hand side of window. It contains various fields.

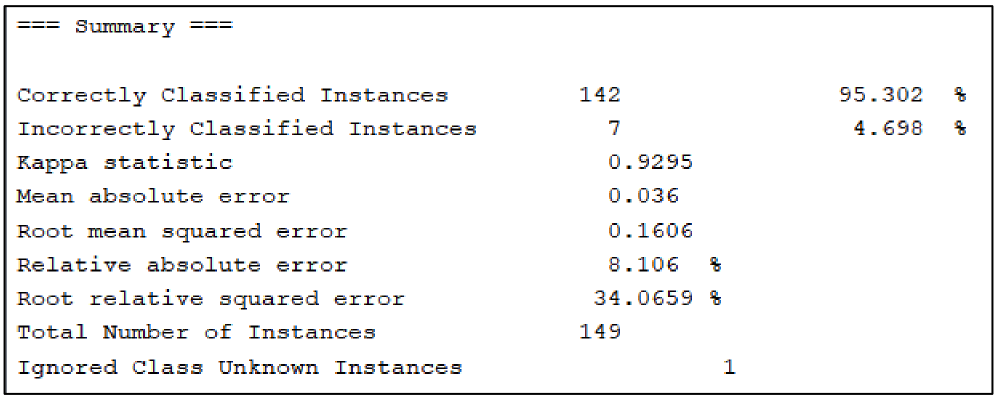
First one is the '**Run information'**. It includes scheme, relation name, number of instances, attributes in the dataset and test mode.



```
=== Run information ===

Scheme:       weka.classifiers.bayes.NaiveBayes
Relation:     Iris_data_sample
Instances:    150
Attributes:   5
              sepallength
              sepalwidth
              petallength
              prtalwidth
              species
Test mode:    10-fold cross-validation
```

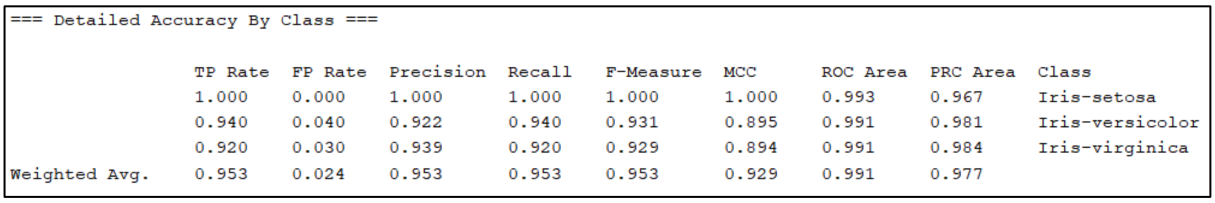
Next one is **'Classifier Model'**. It contains mean, standard deviation, weight sum and precision of all NUMERIC attributes.

Then it gives **'Summary'**. It includes correctly & incorrectly classified instances, mean absolute error, root mean squared error, relative absolute error, etc.



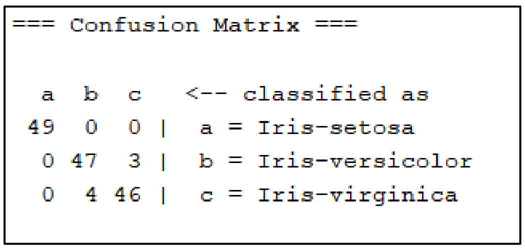
```
=== Summary ===

Correctly Classified Instances         142               95.302  %
Incorrectly Classified Instances         7                4.698  %
Kappa statistic                          0.9295
Mean absolute error                      0.036
Root mean squared error                  0.1606
Relative absolute error                  8.106  %
Root relative squared error             34.0659 %
Total Number of Instances              149
Ignored Class Unknown Instances                  1
```

Next is **'Detailed Accuracy by Class'** which includes precision, F-measure, MCC, ROC area, PRC area, etc. for each class based on NOMINAL attribute and for weighted average.



```
=== Detailed Accuracy By Class ===

                 TP Rate  FP Rate  Precision  Recall   F-Measure  MCC      ROC Area  PRC Area  Class
                 1.000    0.000    1.000      1.000    1.000      1.000    0.993     0.967     Iris-setosa
                 0.940    0.040    0.922      0.940    0.931      0.895    0.991     0.981     Iris-versicolor
                 0.920    0.030    0.939      0.920    0.929      0.894    0.991     0.984     Iris-virginica
Weighted Avg.    0.953    0.024    0.953      0.953    0.953      0.929    0.991     0.977
```

Last one is the **'Confusion Matrix'**.

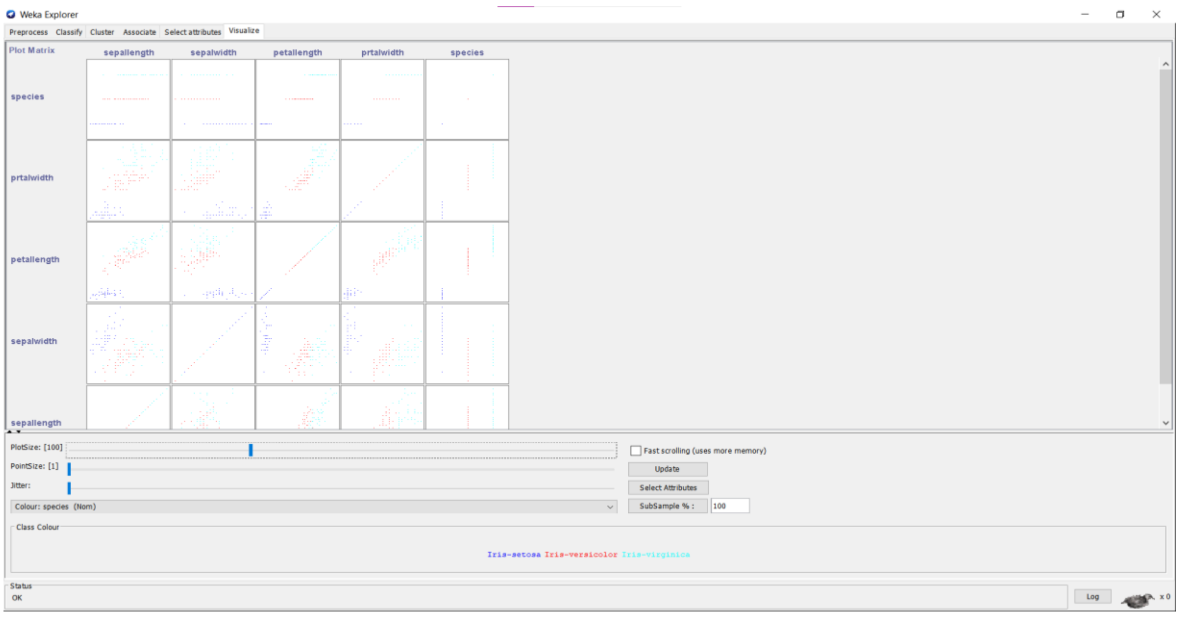


```
=== Confusion Matrix ===

  a  b  c   <-- classified as
 49  0  0 |  a = Iris-setosa
  0 47  3 |  b = Iris-versicolor
  0  4 46 |  c = Iris-virginica
```

3. Data Visualisation:

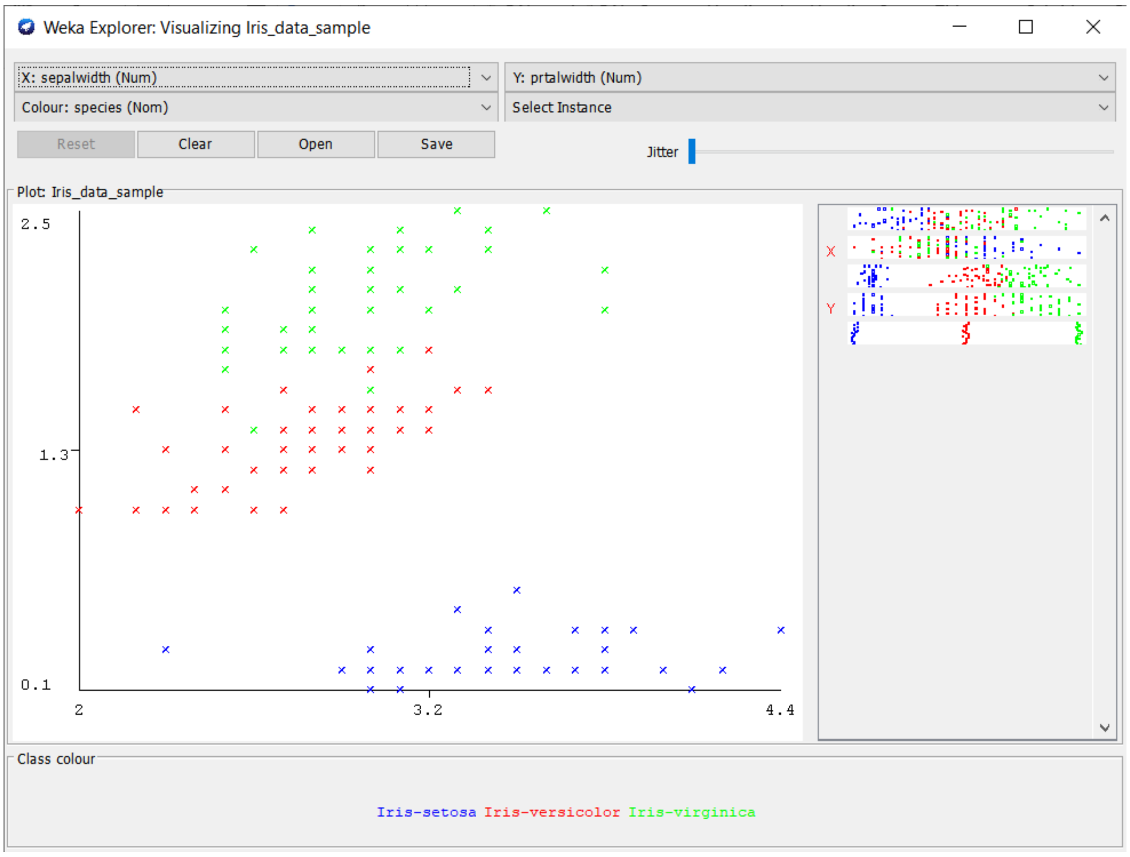
Weka's Visualize panel lets you look at a dataset and select different attributes – preferably numeric ones – for the x- and y-axes.

Choose **'Visualize'** tab. It will display all the plots between the attributes of the dataset.



PlotSize: [100]
PointSize: [1]
Jitter:
Colour: species (Nom)
SubSample % : 100

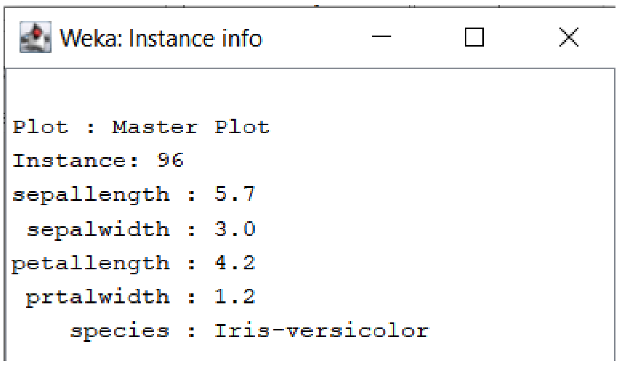
Click on the any plot for detailed view.


Weka Explorer: Visualizing Iris_data_sample
Reset
Clear
Open
Save
Jitter
Plot: Iris_data_sample
2.5
1.3
0.1
2
3.2
4.4
X
Y
Class colour
Iris-setosa Iris-versicolor Iris-virginica

X: sepalwidth (Num)
Y: prtalwidth (Num)
Colour: species (Nom)
Select Instance

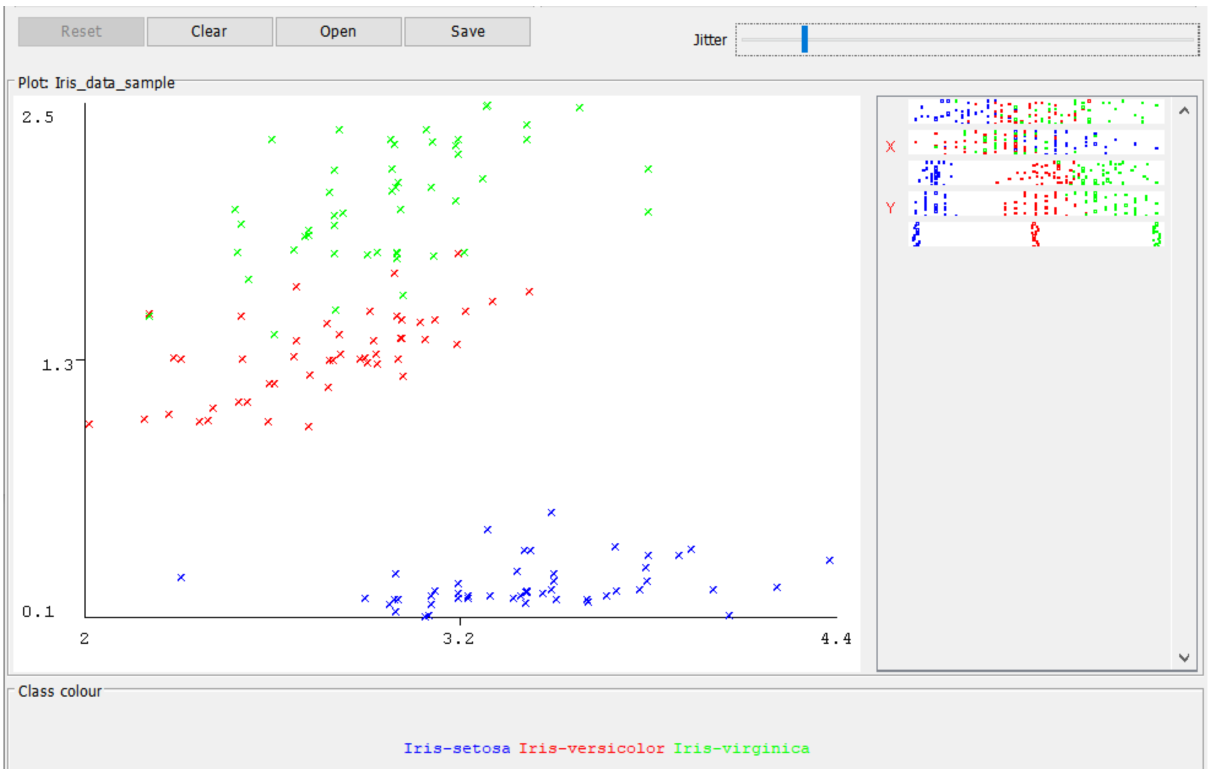
The X and Y-axis attributes can be changed from the right panel in Visualize graph. The user can view different plots.

Instances are shown as points with different colours for different classes. If we click on a point then we get all information about the instance.

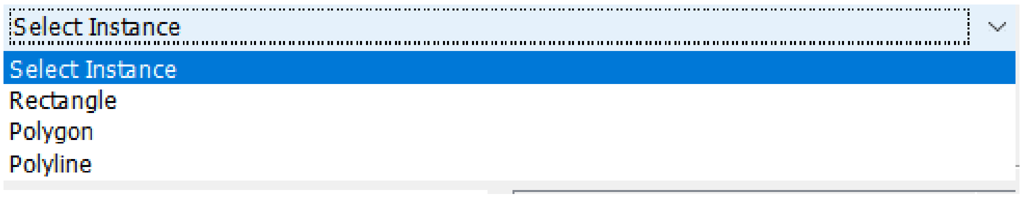


```
Plot : Master Plot
Instance: 96
sepallength : 5.7
 sepalwidth : 3.0
petallength : 4.2
 prtalwidth : 1.2
    species : Iris-versicolor
```

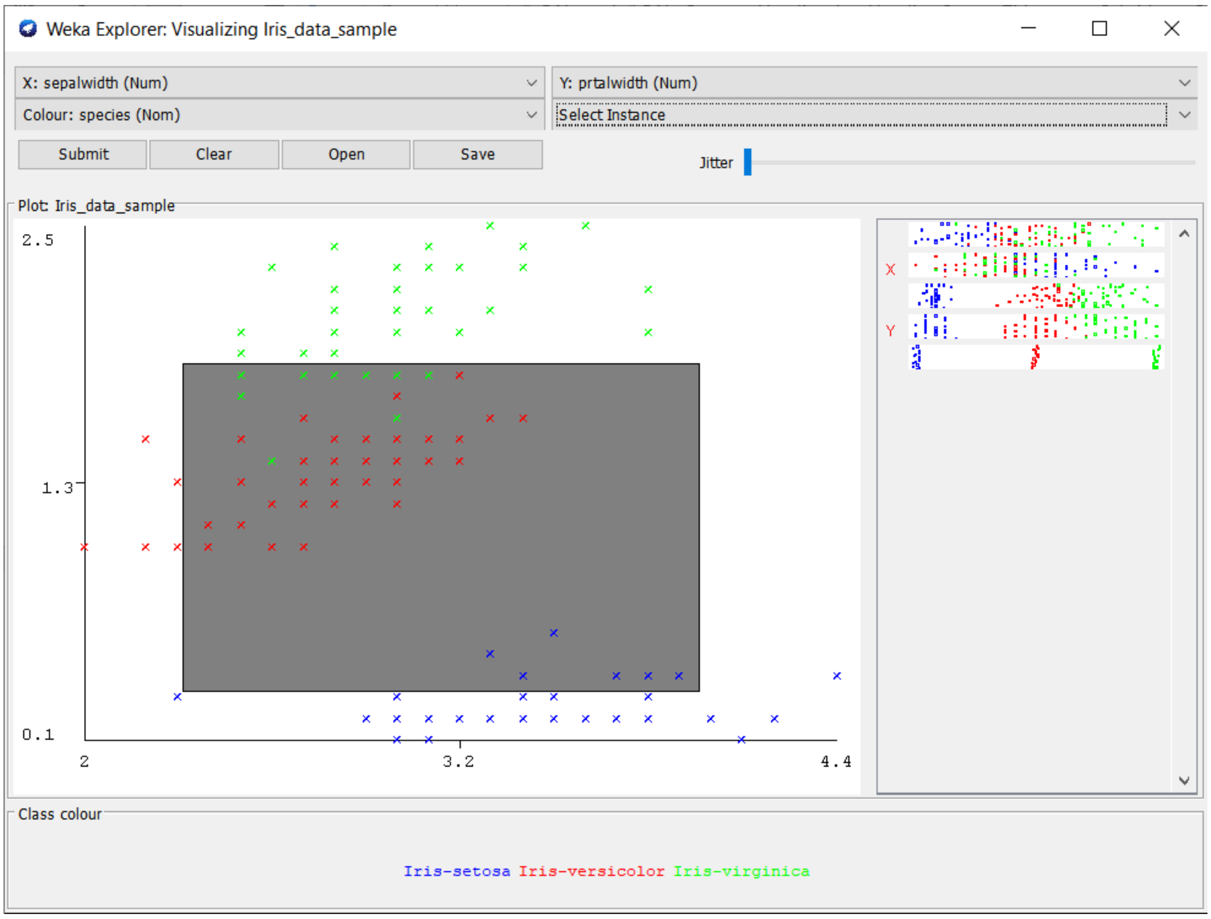
The Jitter is used to add randomness to the plot. Sometimes the points overlap. With jitter, the darker spots represent multiple instances.

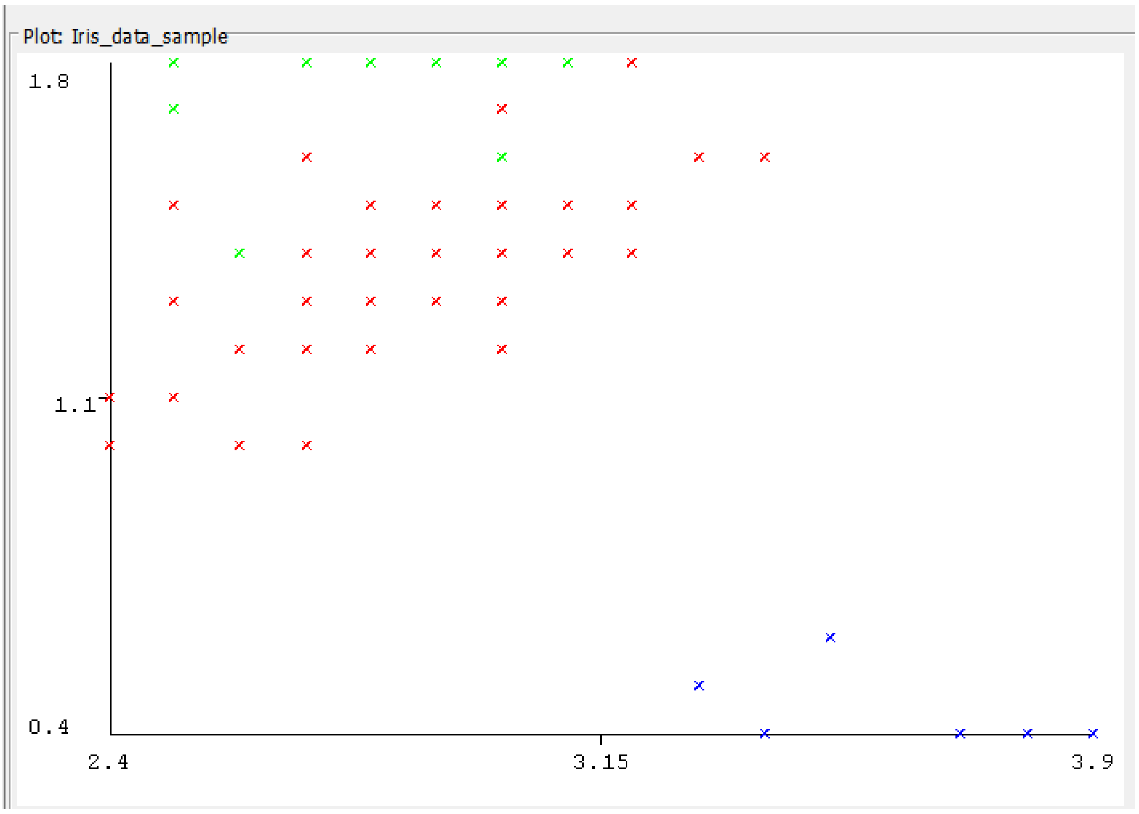


To get a clearer view of the dataset and remove outliers, the user can select an instance from the dropdown. Click on **‘Select Instance’**.



Choose **'Rectangle'**. With this, points in the plot can be selected by plotting a rectangle. Click on **'Submit'**.





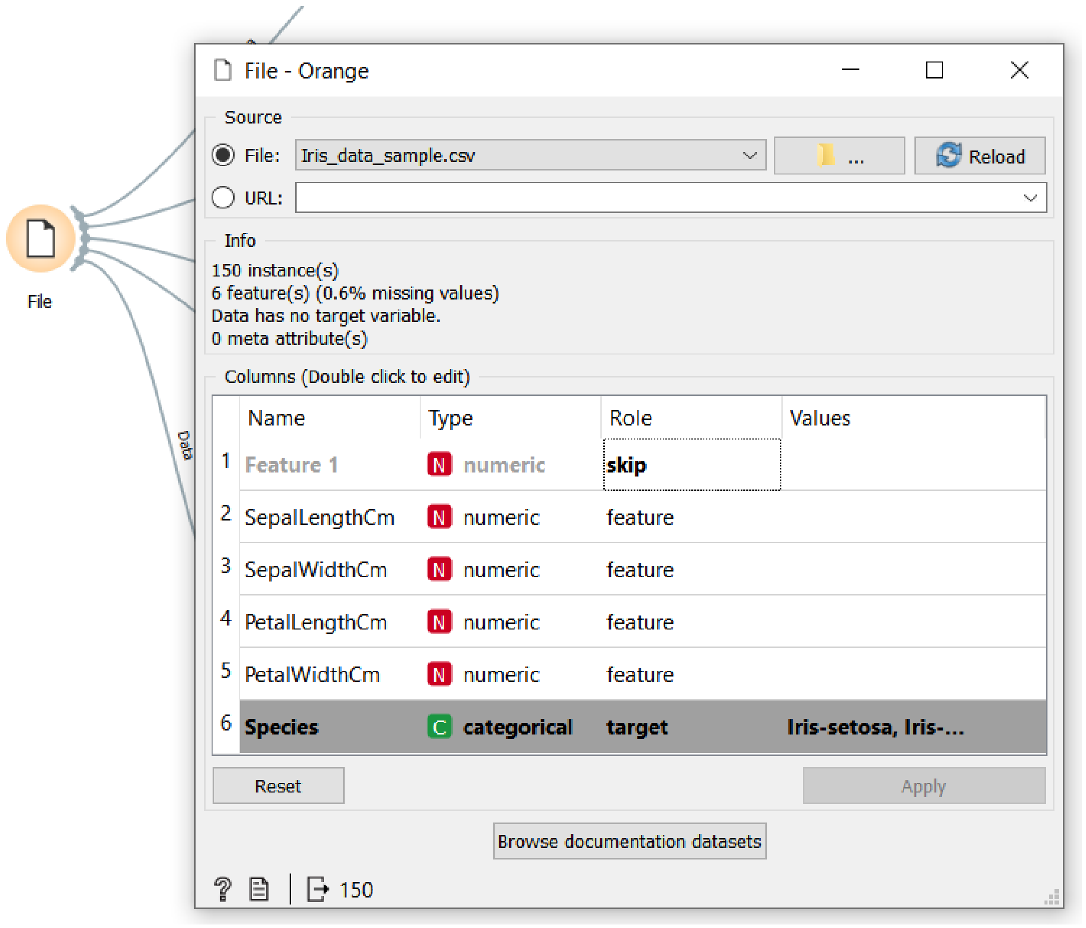
Q.2

The Iris dataset given

https://drive.google.com/file/d/1A3Fxsfzm6BSfhFZGDrjl47RTe45bSgYP/view

1. Data Distribution:

Open the Orange software. Click on the file icon and upload the given dataset. Choose the **Species** target variable.



File: Iris_data_sample.csv
URL:

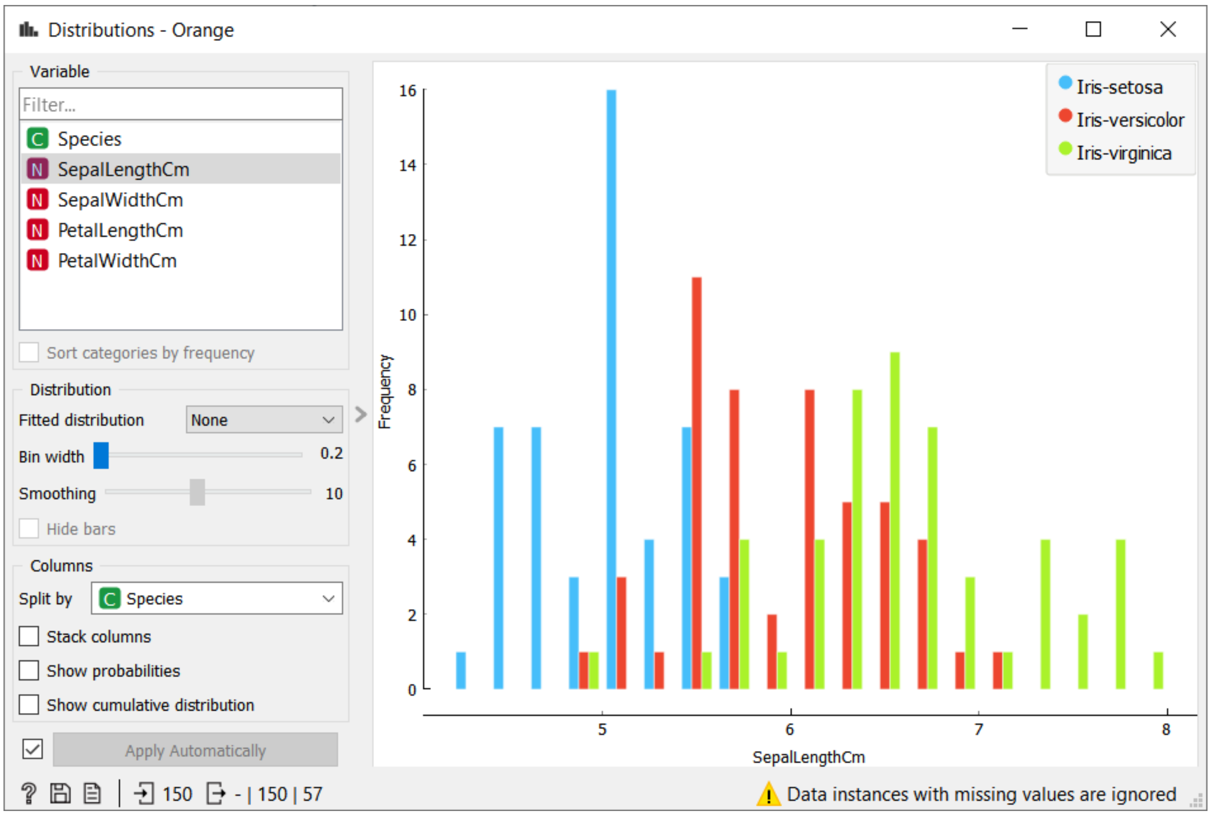
Info
150 instance(s)
6 feature(s) (0.6% missing values)
Data has no target variable.
0 meta attribute(s)

Columns (Double click to edit)

	Name	Type	Role	Values
1	Feature 1	N numeric	skip	
2	SepalLengthCm	N numeric	feature	
3	SepalWidthCm	N numeric	feature	
4	PetalLengthCm	N numeric	feature	
5	PetalWidthCm	N numeric	feature	
6	**Species**	C **categorical**	**target**	**Iris-setosa, Iris-...**

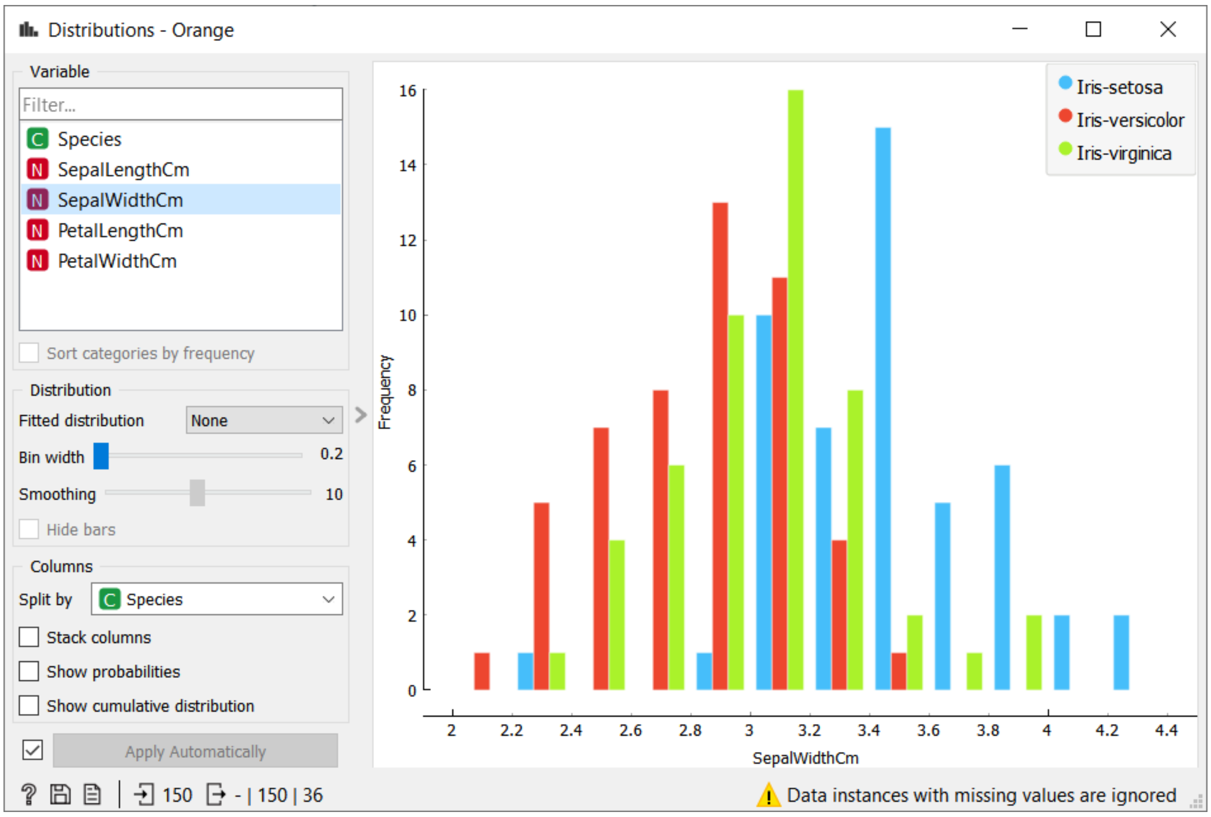
Choose the **'Distribution'** widget.

The graph shows how many times (e.g., in how many instances) each attribute value appears in the data.



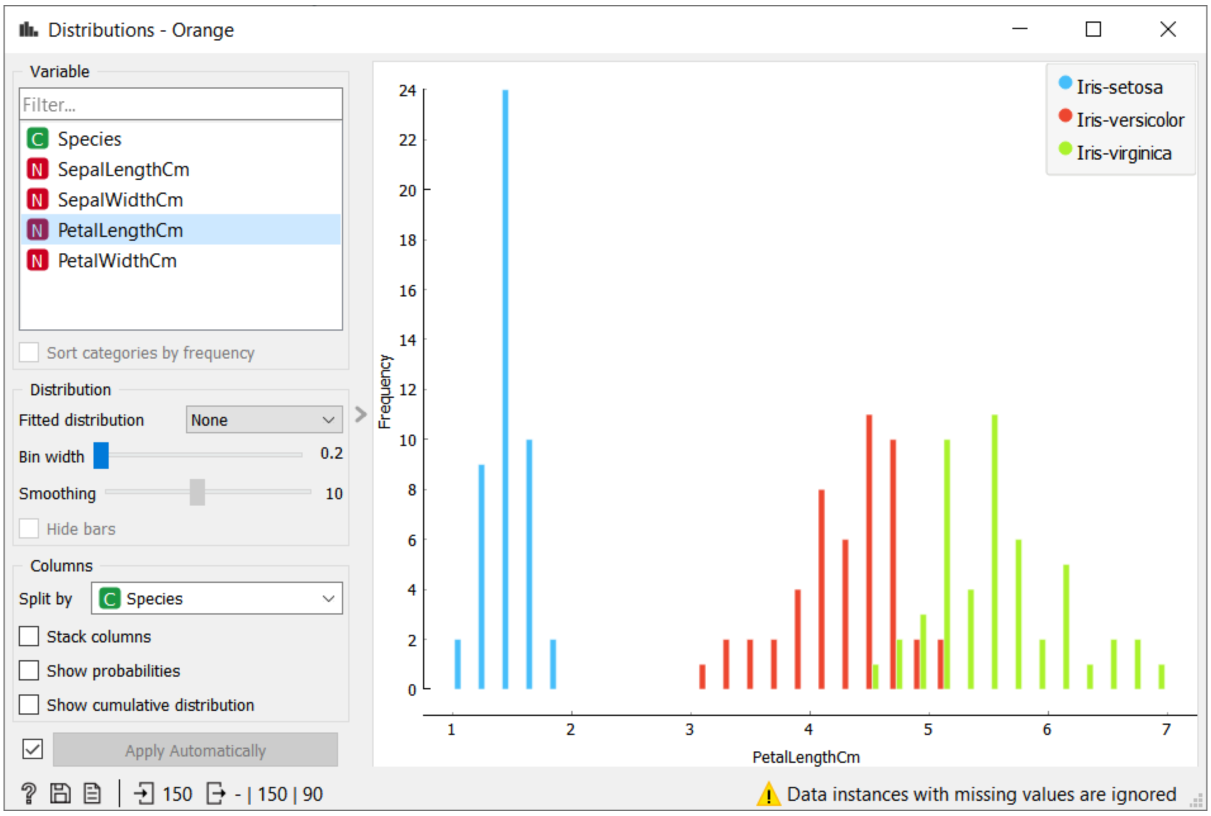
Fitted distribution: None
Bin width: 0.2
Smoothing: 10
Split by: Species
150 | - | 150 | 57
Data instances with missing values are ignored

SepalLengthCm

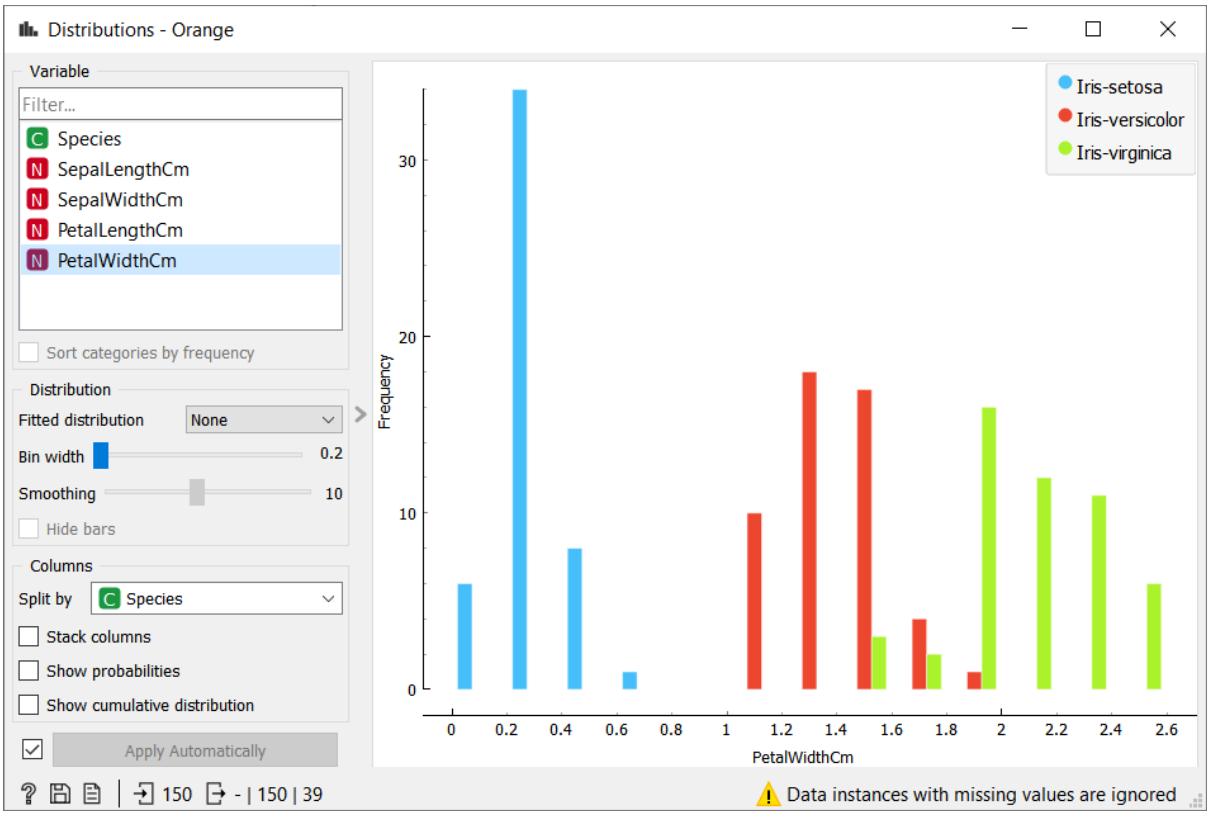


Fitted distribution: None
Bin width: 0.2
Smoothing: 10
Split by: Species
150 | - | 150 | 36
Data instances with missing values are ignored

SepalWidthCm



PetalLengthCm



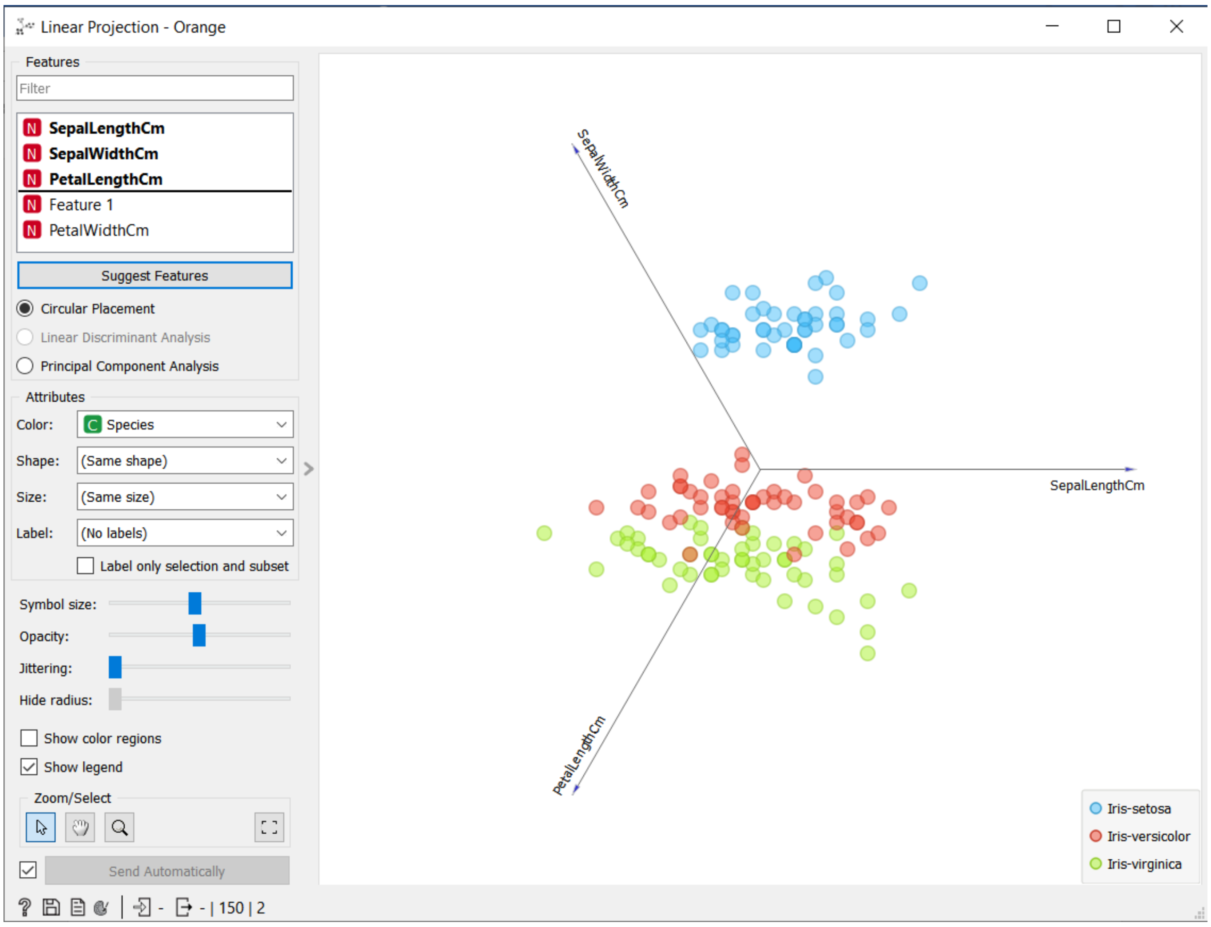
PetalWidthCm

2. Linear Projection:

Choose the **'Linear Projection'** widget. It supports various types of projections such as circular, linear discriminant analysis, and principal component analysis.

Here **SepalLengthCm, SepalWidthCm, PetalLengthCm** attributes are selected. Hence the projection is in two-dimension.

It is the sepal width and sepal length that already separate **Iris setosa** from the other two, while the petal length is the attribute best separating **Iris versicolor** from **Iris virginica**.


Linear Projection - Orange

Features

Filter

- **SepalLengthCm**
- **SepalWidthCm**
- **PetalLengthCm**
- Feature 1
- PetalWidthCm

Suggest Features

- [✓ Circular Placement]
- Linear Discriminant Analysis
- Principal Component Analysis

Attributes

Color: Species

Shape: (Same shape)

Size: (Same size)

Label: (No labels)

☐ Label only selection and subset

Symbol size:

Opacity:

Jittering:

Hide radius:

☐ Show color regions

☑ Show legend

Zoom/Select

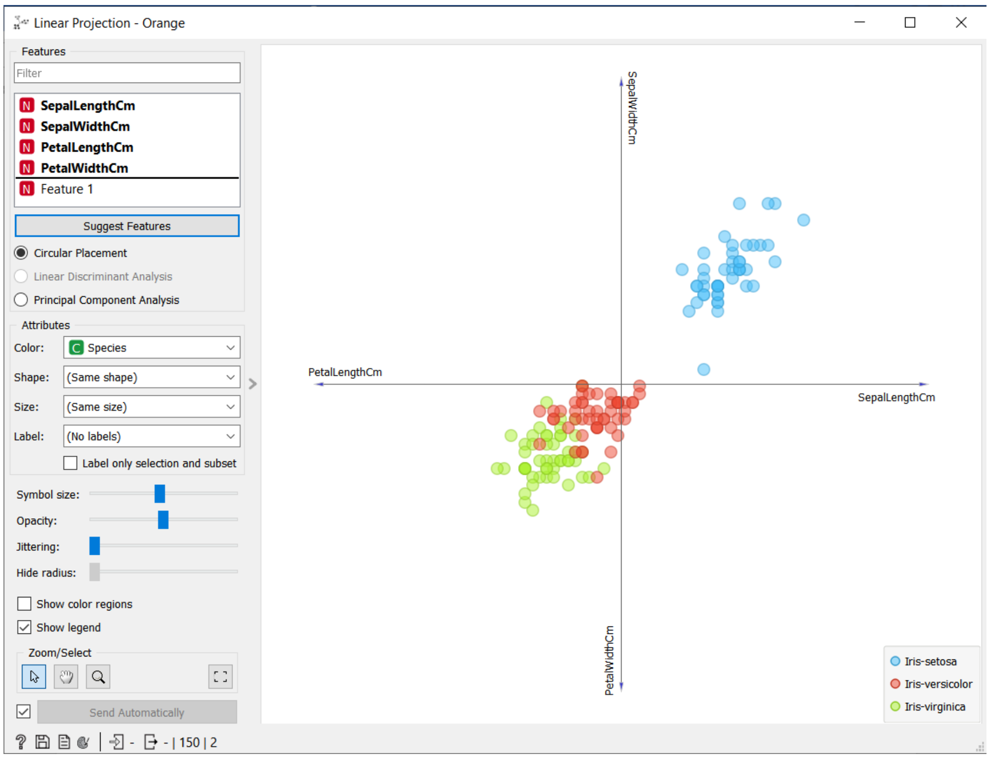
☑ Send Automatically

150 | 2

Axes in the projection and other available axes are displayed.

We can add or remove the features as we need.

Below window shows linear projection with all attributes.


Linear Projection - Orange

Features

Filter

- N SepalLengthCm
- N SepalWidthCm
- N PetalLengthCm
- N PetalWidthCm
- N Feature 1

Suggest Features

- [✓ Circular Placement]
- Linear Discriminant Analysis
- Principal Component Analysis

Attributes

Color:	Species
Shape:	(Same shape)
Size:	(Same size)
Label:	(No labels)

☐ Label only selection and subset

Symbol size:
Opacity:
Jittering:
Hide radius:

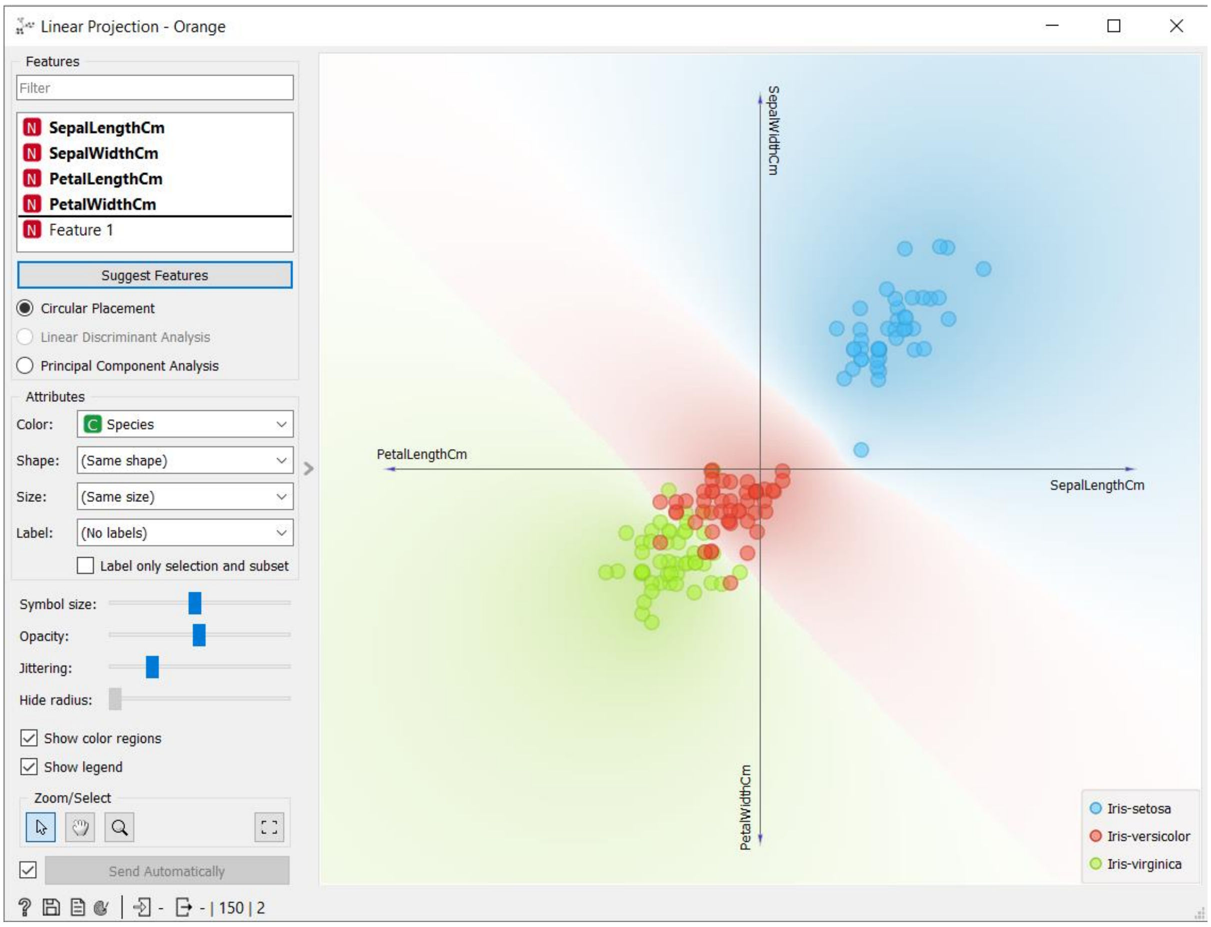
☐ Show color regions
☑ Show legend

Zoom/Select

☑ Send Automatically

150 | 2

Jittering is used to prevent the dots from overlapping (especially for discrete attributes). Show class density colours the graph by class (see the screenshot below)


Linear Projection - Orange

Features

Filter

- N SepalLengthCm
- N SepalWidthCm
- N PetalLengthCm
- N PetalWidthCm
- N Feature 1

Suggest Features

- [✓ Circular Placement]
- Linear Discriminant Analysis
- Principal Component Analysis

Attributes

Color:	Species
Shape:	(Same shape)
Size:	(Same size)
Label:	(No labels)

☐ Label only selection and subset

Symbol size:
Opacity:
Jittering:
Hide radius:

☑ Show color regions
☑ Show legend

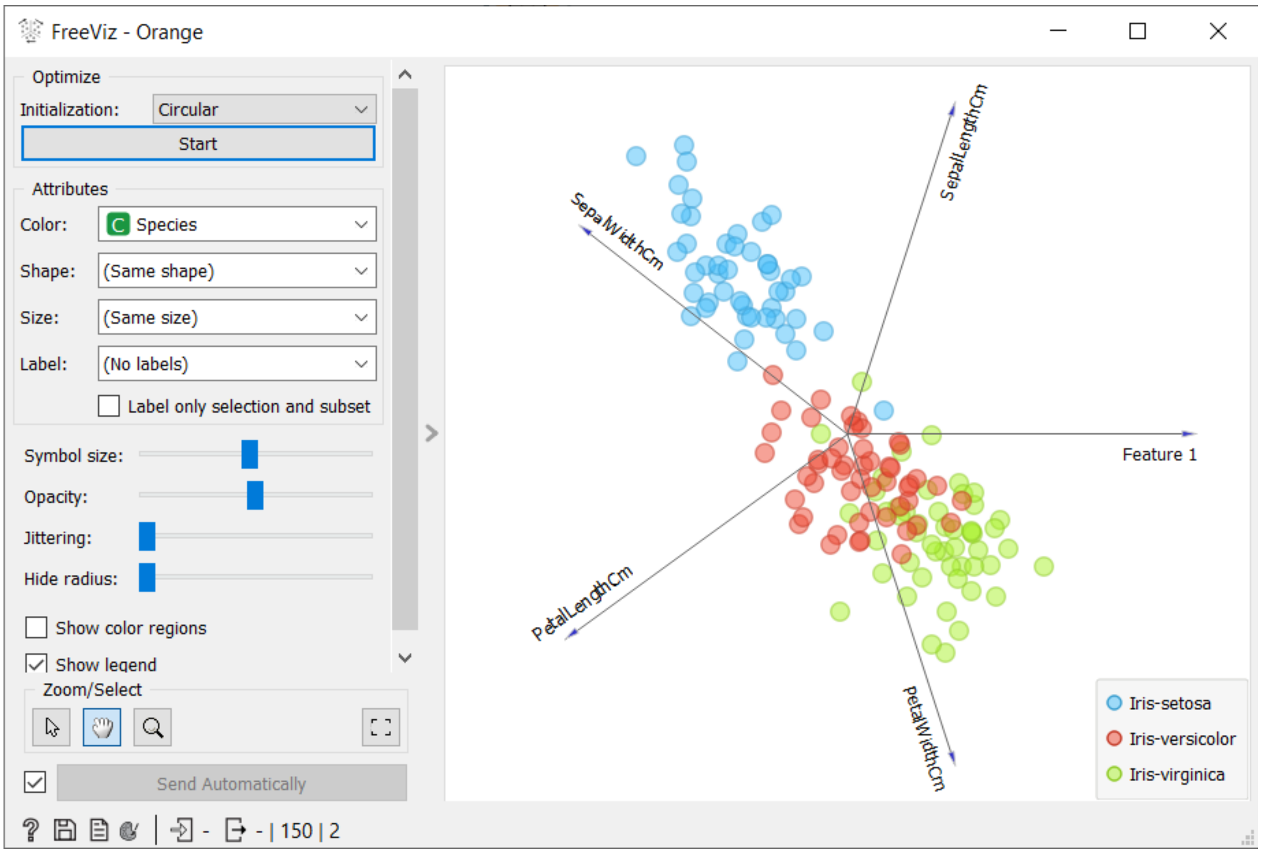
Zoom/Select

☑ Send Automatically

150 | 2

3. FreeViz:

Choose the **'FreeViz'** widget. It is a multivariable projection method. In FreeViz points in the same class attract each other, those from different class repel each other, and the resulting forces are exerted on the anchors of the attributes.


FreeViz - Orange

Optimize
Initialization: Circular
Start

Attributes
Color: Species
Shape: (Same shape)
Size: (Same size)
Label: (No labels)
Label only selection and subset

Symbol size:
Opacity:
Jittering:
Hide radius:

Show color regions
Show legend

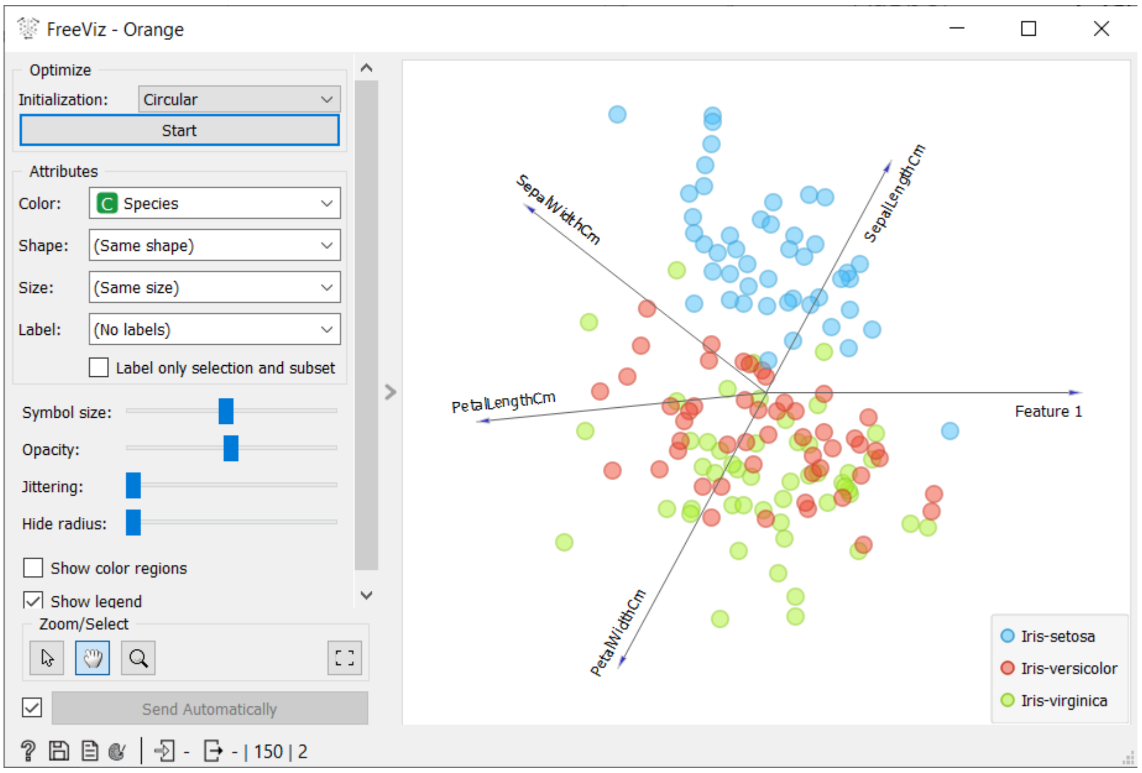
Zoom/Select

Send Automatically

150 | 2

The points cannot move (are projected in the projection space), but the attribute anchors can. Anchors can be moved manually.

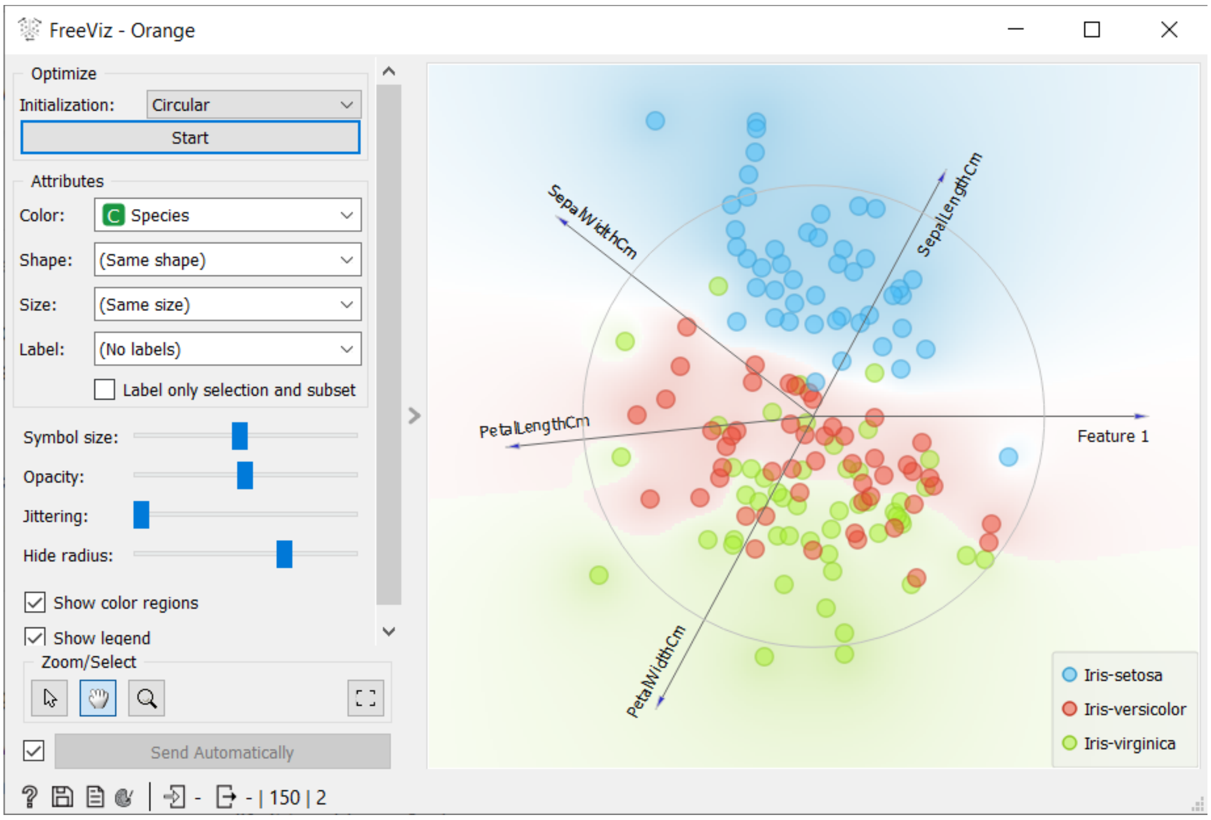
Use a mouse pointer and hover above the end of an anchor. Then move selected anchor where ever you want and the positions of points will change accordingly as shown in following window.



Anchors inside a circle are hidden. Circle radius can be changed using a slider.

Set jittering to prevent the dots from overlapping (especially for discrete attributes).

Show class density colours the graph by class (see the screenshot below).



Q.3

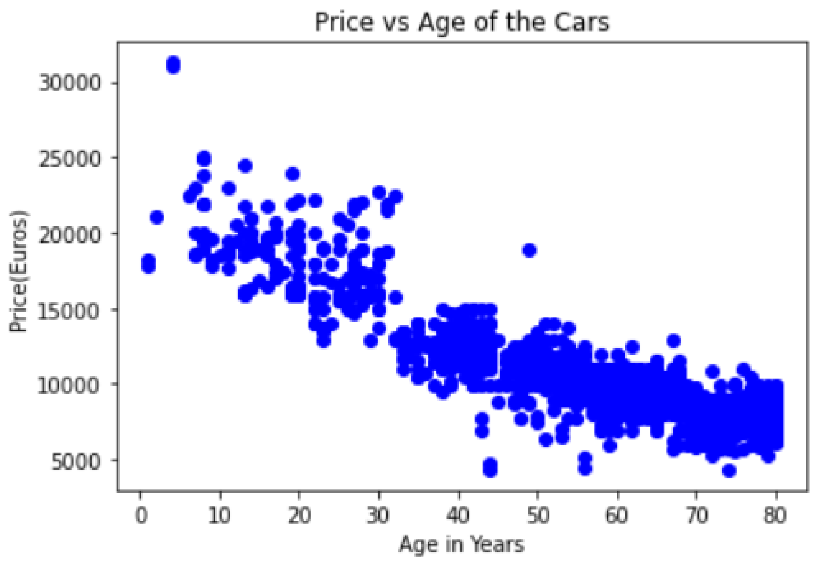
Free Software	Open Source Software	Proprietary Software
Free software means software that respects users’ freedom and community.	Open source software is a computer software whose source code is available openly in internet.	Proprietary software is a computer software where the source codes are not publicly not available
Users have the freedom to run, copy, distribute, study, change and improve the software.	Programmers can modify it to add new features and capabilities without any cost.	only the company which has created can modify the software.
Software freedom translates to social freedom. Freedom is a value that is more important than any economic advantage.	Software is just software. There are no ethics associated directly to it. Ethics are to be associated to the people not to the software.	This software is managed by a closed team of individuals or groups that developed it. No freedom to users.
The term “free software” is sometimes misunderstood—it has nothing to do with price. It is about freedom.	Users can get open software for free of charge. Users do not need to have any authenticated license to use this software.	Users have to pay to get this software and its commercial support if available for maintenance. A valid and authenticated license is required.
E.g., Linux kernel, the BSD and Linux operating systems, the GNU Compiler Collection and C library; the MySQL relational database; the Apache web server	E.g., Android, Linux, Firefox, Open Office, GIMP, VLC Media player etc.	E.g., Windows, MacOS, Internet Explorer, Google earth, Microsoft Office etc.

Q.4

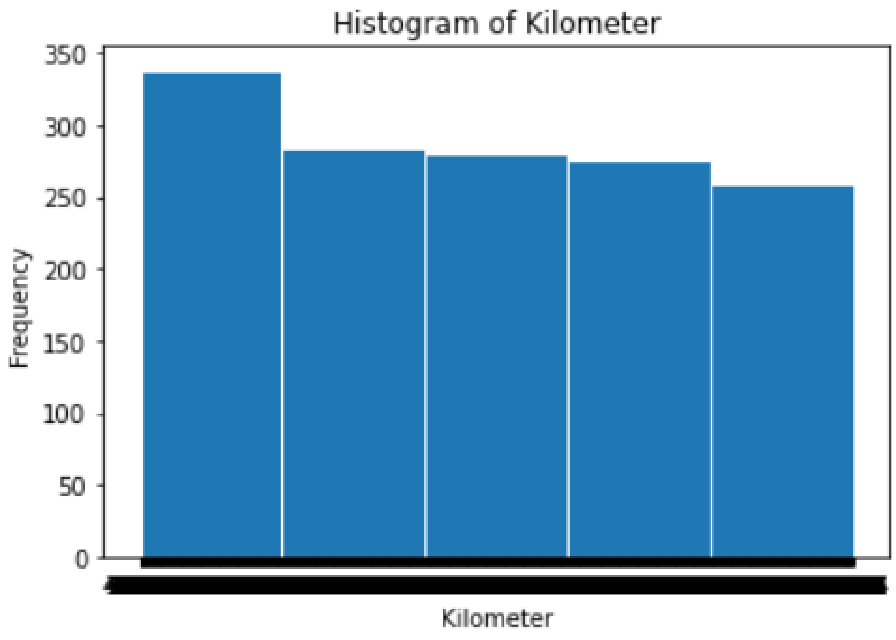
Dataset given:

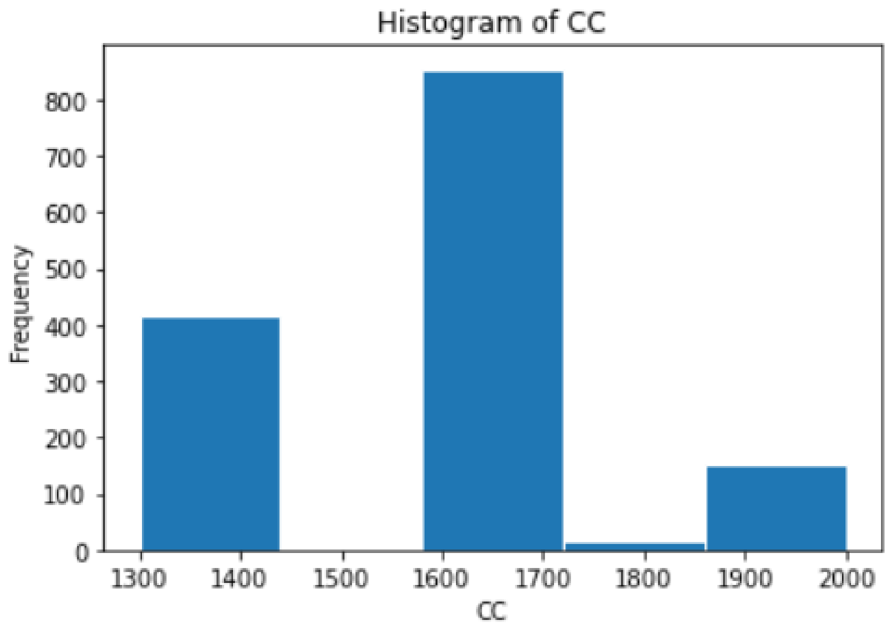
https://drive.google.com/file/d/1i11BZFe8Xj9kNq7eeE9KOa_Iz1KhEdXJ/view

a. Scatter plot- Scatter plot of Price Vs Age

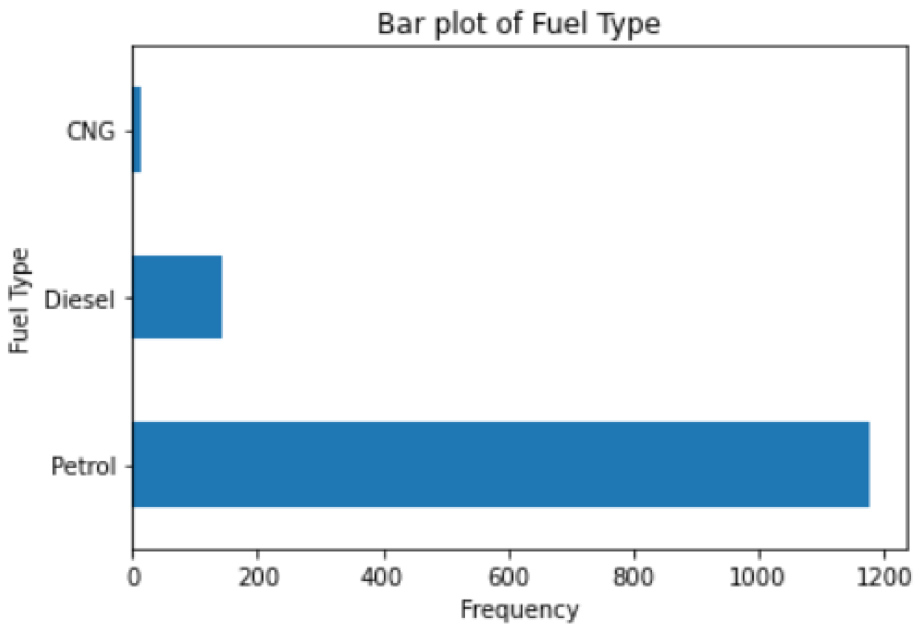


b. Histogram- for Kilometre and CC





c. Bar plot- Bar plot for different fuel types



Q.5

<u>Free Open Source Software:</u>

Purpose:

1. A software that anyone is freely licensed to use, copy, study, and change in any way, and the source code is openly shared so that people are encouraged to voluntarily improve the design of the software.
2. "Free and open-source software" (FOSS) is an umbrella term for software that is simultaneously considered both free software and open-source software.
3. FOSS (free and open-source software) allows the user to inspect the source code and provides a high level of control of the software's functions compared to proprietary software.

Properties:

1. Initially an individual or few volunteers involve in the project. Once the project is successful then a community of project is established to contribute to the project.
2. The FOSS model is inherently collaborative and transparent.
3. The Concurrent Versions System (CVS) helps in distributed development of FOSS.
4. Globally distributed software development by virtual teams promises the flexibility, responsiveness, lower costs, and improved resource utilization.
5. Due to distributed nature of FOSS, its design is modular that can easily incorporate into the main system.
6. Modular design supports abstraction, increased understanding of the system and concurrent development.
7. It provides reusability. FOSS licenses grants the rights to the developer to obtain the source code, inspect it, modify it, and distribute it.
8. Wide ranges of licensing options, such as GPL, LGPL, BSD, ISC, Artistic License, etc., are available for FOSS distribution.
9. Business model of FOSS is getting success. Sources of income range from donations to providing services such as consulting, integration, support and training.
10. Free and Open Source Software model ensures security, availability, reliability, quality, and efficiency of software development.

Example:

i. Linux kernel
ii. GNOME Desktop
iii. Free Berkley Software Distribution

Proprietary Software:

Purpose:

1. A computer software for which the software's publisher or another person reserves some licensing rights to use, modify, share modifications, or share the software.
2. Proprietary software is also known as ‘Non-free Software’, ‘Closed-Source Software’ or ‘Commercial Software’.
3. Non-free software sometimes includes patent rights.

Properties:

1. It has a license which is the property of a developer, company or the owner.
2. Every user requires a proper license for use.
3. The owners can restrict the number of computer systems that a user can use to run the software.
4. The owners do not provide the source code of the software. Moreover, users have no right to modify the software.
5. The users have no right to distribute or share the software.
6. The software is stable because the development is totally the responsibility of the owner.
7. The software have their own protocols and codes which are not compatible with other software.
8. Some software restrict particular hardware for use. Such as the macOS can run only on Apple devices.
9. Developers provide a good user experience. Therefore, this software is easy to use.
10. The users are paying for the software. Hence, it is the duty of the vendors to provide any technical support.

Example:

- **i.** macOS
- **ii.** Adobe Suite
- **iii.** Microsoft Windows Professional Edition.