Orange Canvas "A Machine Learning Toolkit Tutorial"

1 author:

Hamza Al-Qatawna
Mu'tah University
5 PUBLICATIONS   **0** CITATIONS

Orange Canvas Tutorial

MACHINE LEARNING TOOLKIT

Hamza M. Al-Qatawna

M.Sc
Physics dept. / Mutah University

HAMZA ALQATAWNA
PROJECT FOR ARTIFICIAL INTELLIGENCE COURSE

HMSQV12@GMAIL.COM

Introduction

Scripting languages have recently risen in popularity in all fields of computer science. Within the context of explorative data analysis, they offer advantages like interactivity and fast prototyping by gluing together existing components or adapting them for new tasks.

Python is a scripting language with clear and simple syntax, which also made it popular in education.

Python offers a huge number of extension libraries. Many are related to machine learning, including several general packages like scikit-learn (Pedregosa et al., 2011), PyBrain (Schaul et al., 2010) and mlpy (Albanese et al., 2012).

Orange was conceived in late 1990s and is among the oldest of such tools. It focuses on simplicity, interactivity through scripting, and component-based design.
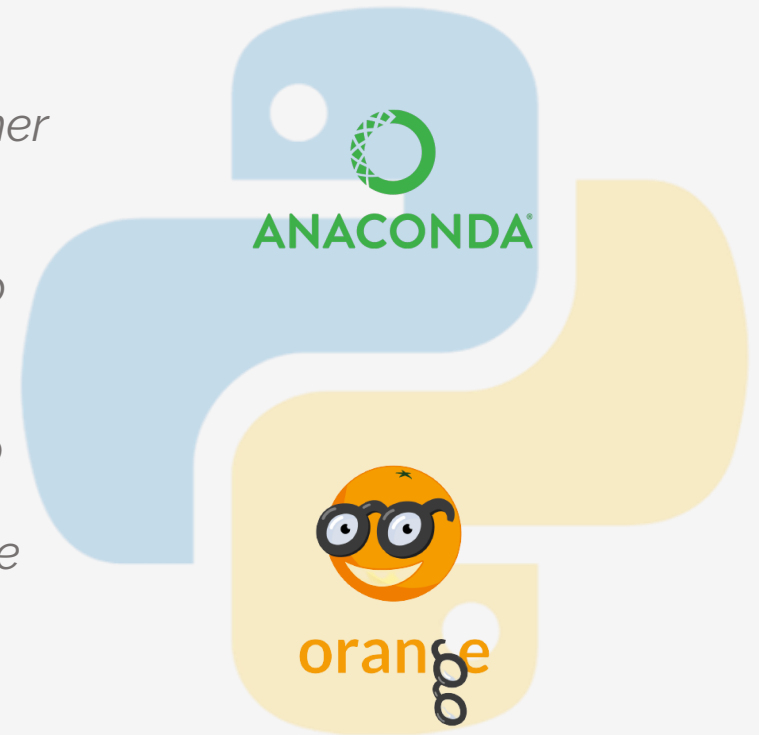
Hamza M. Al-Qatawna

What is Orange ?

> "*Orange is a component-based data mining software. It includes a range of data visualization, exploration, preprocessing and modeling techniques. It can be used through a nice and intuitive user interface or, for more advanced users, as a module for the Python programming language.*"

-GitHub Website

Orange is one of the popular open-source machine learning and **data visualization tool for beginners**. People who don't know more about coding and willing to visualize pattern and other stuff can easily work with Orange.

Nowadays people prefer GUI based tools instead of more coding stuff.

Hamza M. Al-Qatawna

Why Orange?

#Open-Source software

#Code-Free

Hamza M. Al-Qatawna

#Strong data visualization capabilities

Also: *easy-to-use

*lightweight

*drag-and-drop

and most importantly it is FUN

Why Orange? (cont'd)

Orange is an open-source software package released under GPL that powers Python scripts with its rich compilation of mining and machine learning algorithms for data pre-processing, classification, modeling, regression, clustering and other miscellaneous functions.

Orange also comes with a visual programming **environment** and its workbench consists of tools for importing data, dragging and dropping widgets, and links to connect different widgets for completing the workflow.

Orange uses common Python open-source libraries for scientific computing, such as numpy, scipy, and scikit-learn, while its graphical user interface operates within the cross-platform Qt framework.

Hamza M. Al-Qatawna

Orange and Science

Researchers, students, non-developers and business analysts use platforms like Orange to get a good understanding of the data at hand and also quickly build machine learning models to understand the relationship between the data points better.
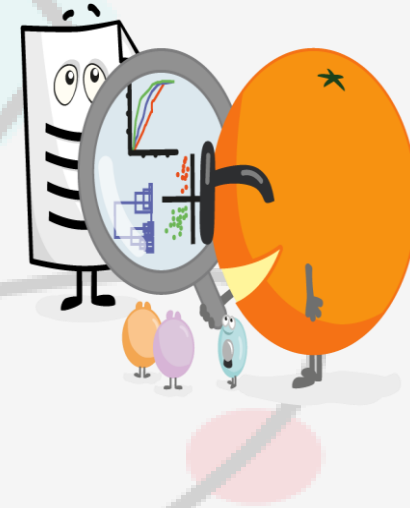
Orange gives a platform for experiment selection, predictive modeling, and recommendation systems and can be used of physics, computational physics, astrophysics, genomic research, biomedicine, bioinformatics, and more.

Orange is always preferred when the factor of innovation, quality, or reliability is involved and that what every scientist and researcher seek.

Hamza M. Al-Qatawna

Orange Setup And Installation

First and foremost, there are numerous ways to install Orange. The most common ways are via standalone installer, Anaconda or Pip. We are going to install by Anaconda

If you already have Anaconda, you can start adding conda-forge
to your channels via the following command:

```
conda config --add channels conda-forge
```

Then, run the following command to install orange:

```
conda install orange3
```

The GUI requires some dependencies that are not included or provided by conda-forge.
Let's add them in by typing the following command in the terminal and running it:

```
conda install -c defaults pyqt=5 qt
```

Alternatively, you can find add-on manager in **Options** menu.

Hamza M. Al-Qatawna

Orange Setup And Installation (Cont'd)

This is the best way in my opinion, and who doesn't have Anaconda ! (:
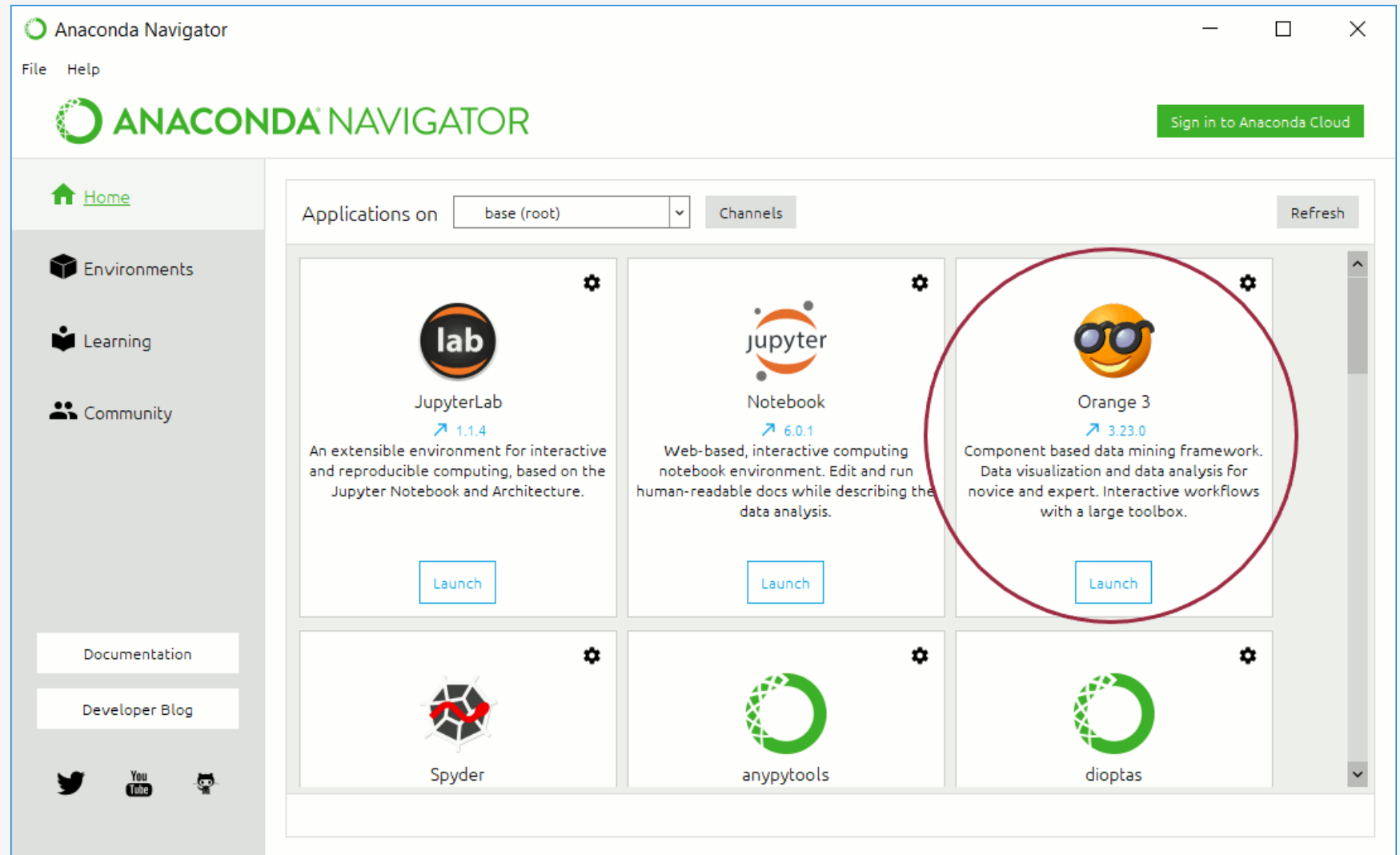
Hamza M. Al-Qatawna

Orange Tabs (Data)

Orange has main four different tabs.

1. Data

2. Visualize

3. Model

**4. Evaluate**
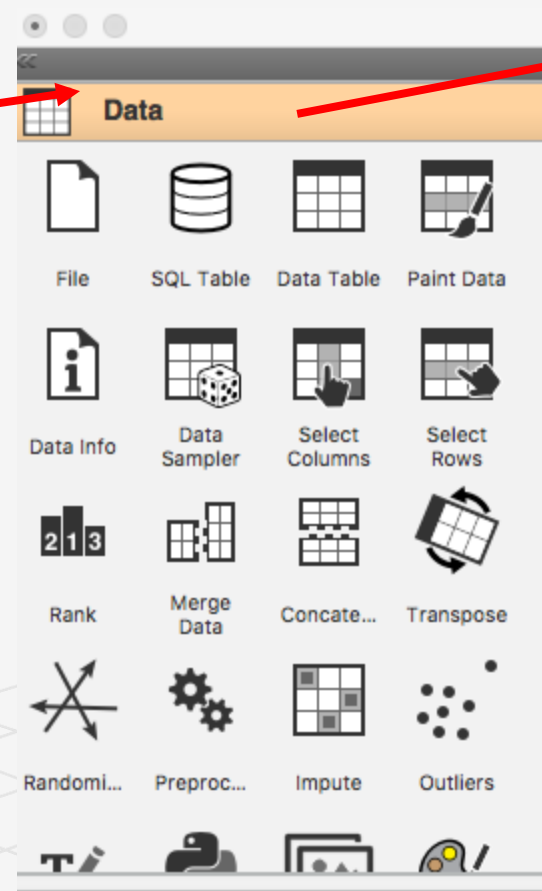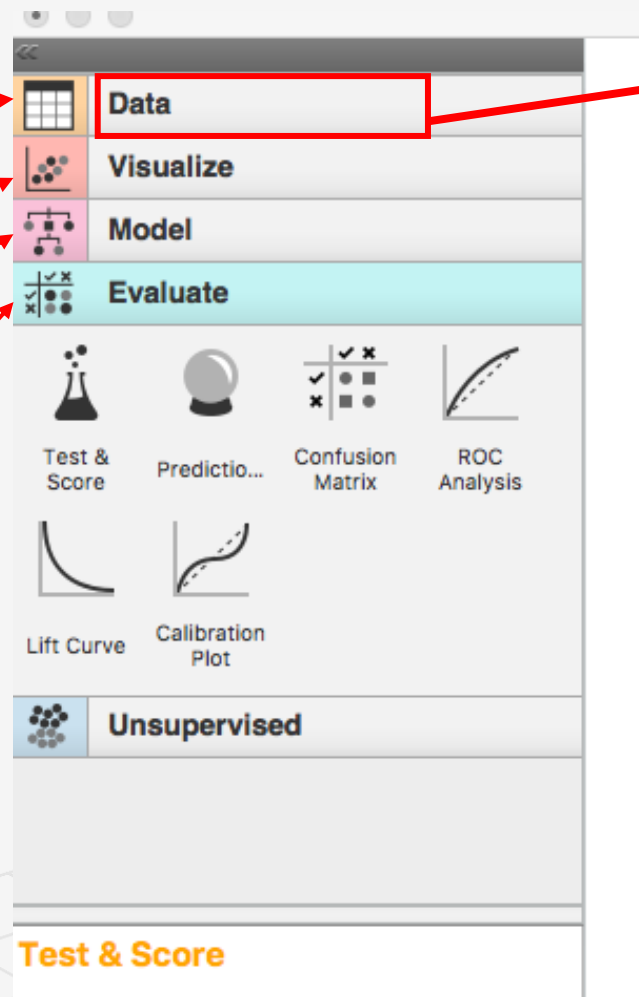
1. Data
It has around 26 different functions. One can extract data from different sources like files, SQL tables, and data tables.
You can paint data, sample, merge, and select data.
You can even construct features, detect outliers, and preprocess data.
The list is long, and a plethora of data-related stuff is available at user's disposal.

Hamza M. Al-Qatawna

Orange Tabs (Visualize)

2. Visualize

Around 15 different types of visualization are available, which can be used to view data across various dimensions.
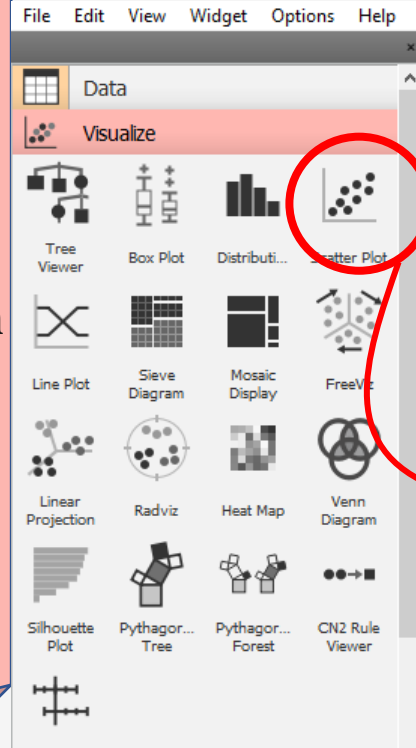
A quick example, for some data, I created a quick scatter plot by connecting the **Paint Data** icon to the scatter plot diagram. In each of the visualizations, there is a handful of functions that can be used for creating marvelous plots.
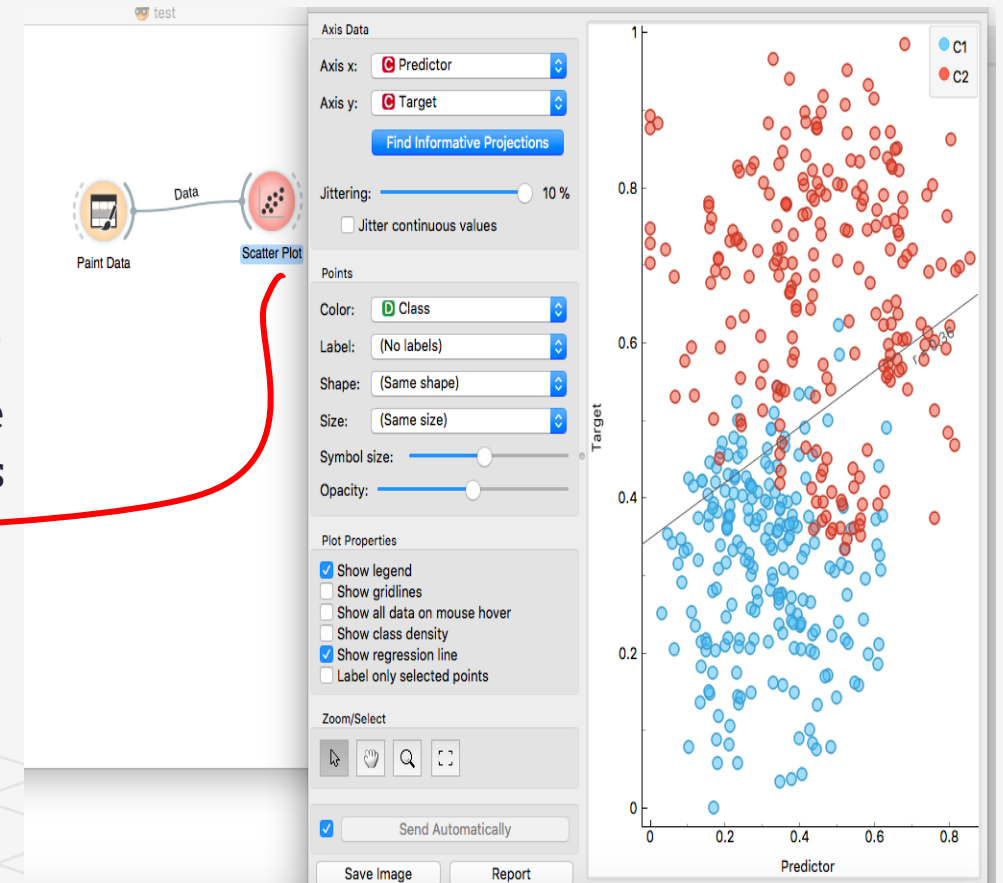
Hamza M. Al-Qatawna

Orange Tabs (Model & Evaluate)

3. Model

There are ten supervised
ML modeling functions.
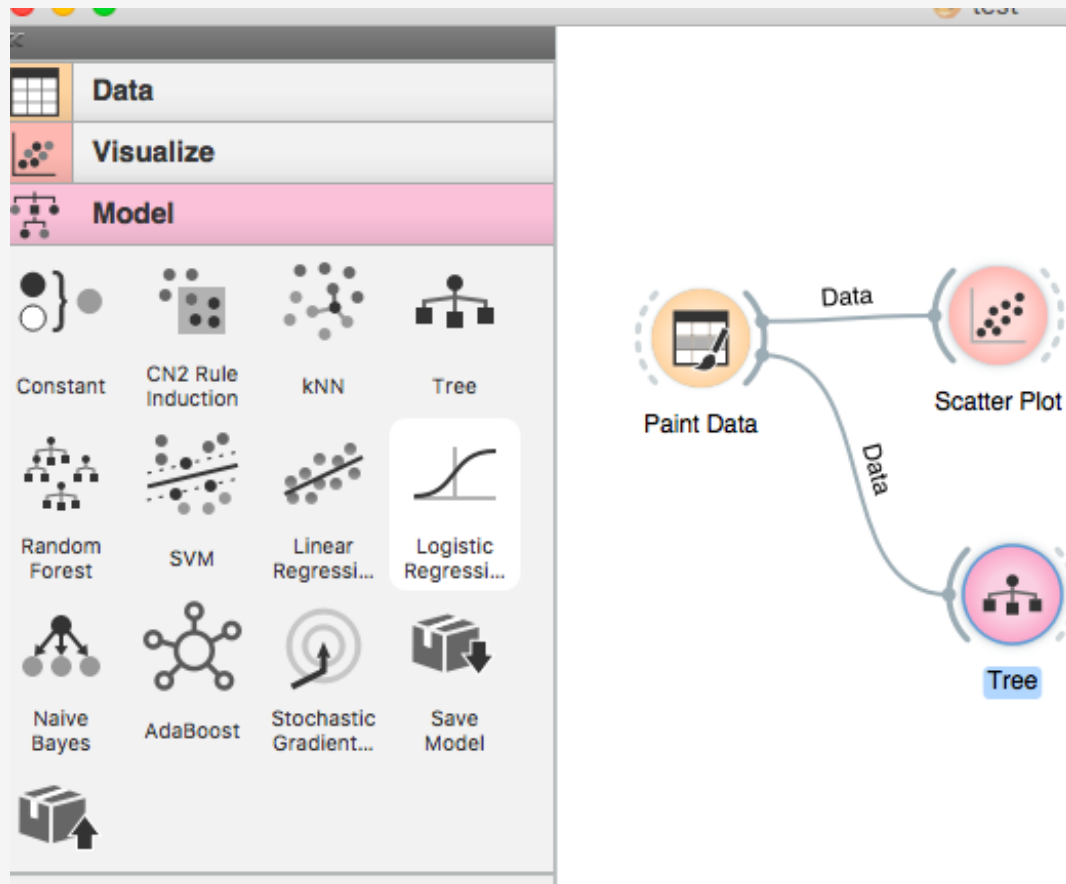
4. Evaluate
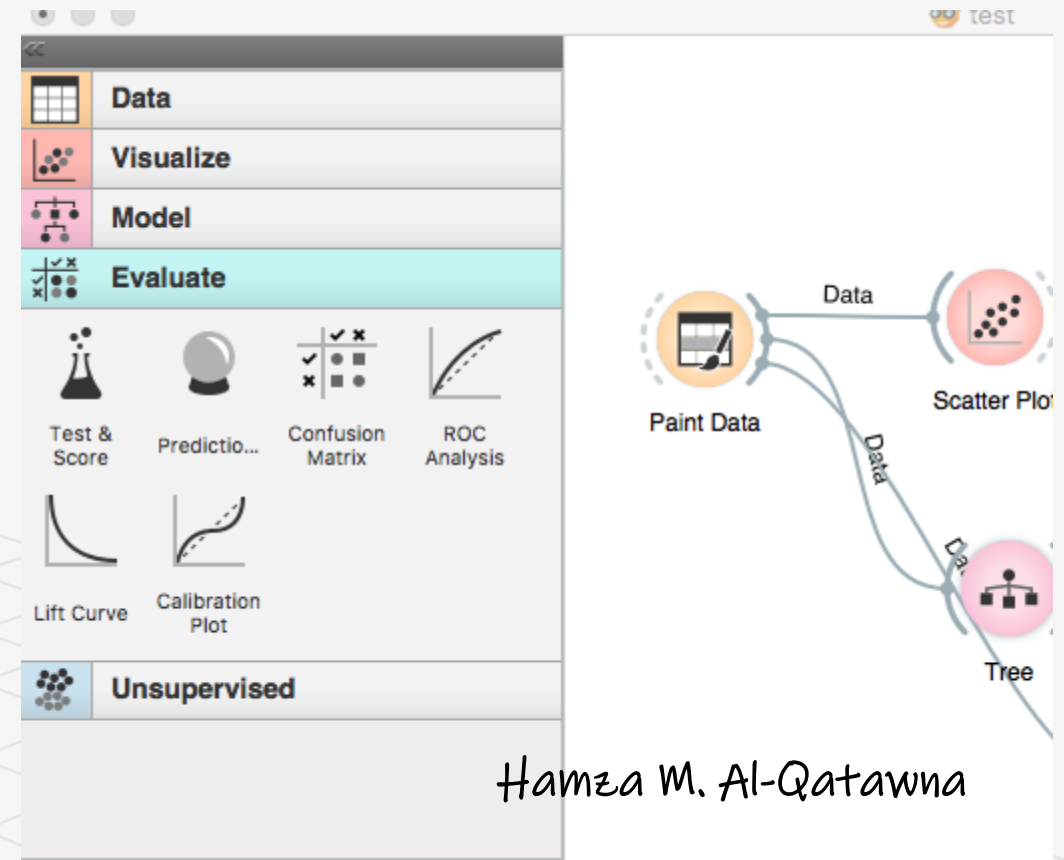
Hamza M. Al-Qatawna

How to add widget to the canvas

There are 3 ways to add widget to the canvas:

1 ₁Double click on the widget.

2 ₂Drag the widget to the canvas.

3 ₃Right click on the canvas for the widget menu.

💡 You can notice that the toolbox is on your left and the white working area is the canvas. The toolbox contains all the widgets that can be moved to the canvas.

Hamza M. Al-Qatawna
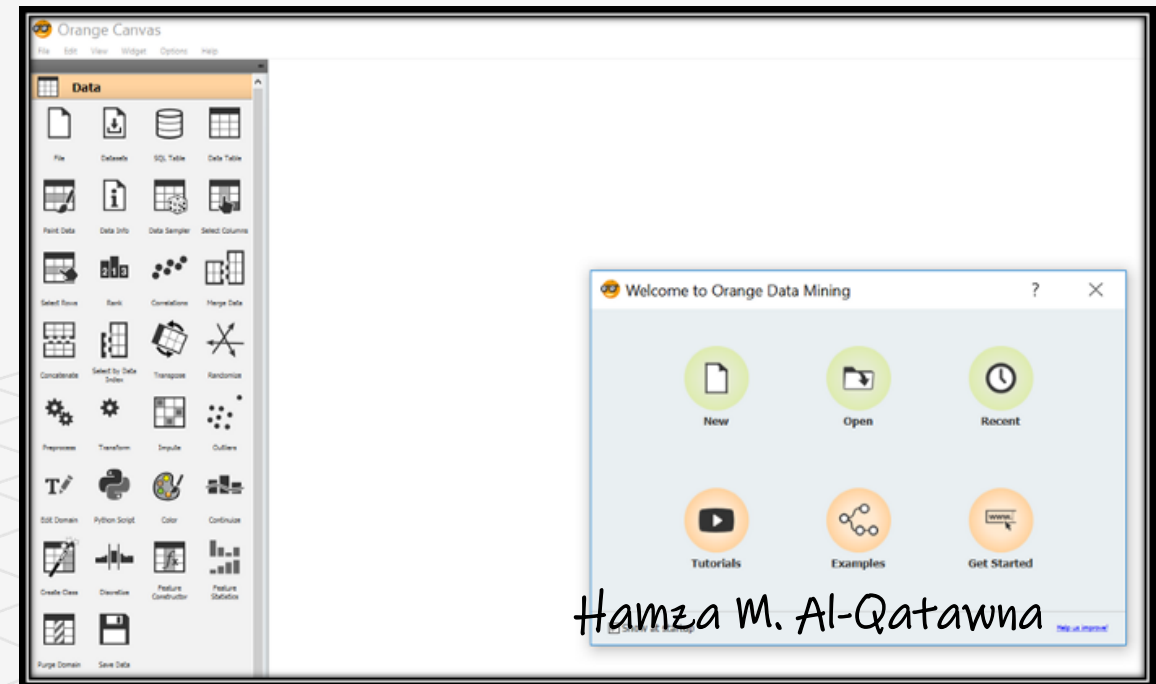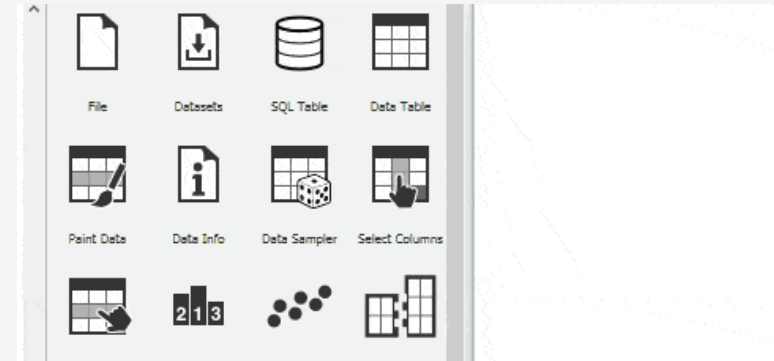
Connecting Widgets

You can connect two widgets together if they are compatible.

Let's try it:

1 Drag the **File** widget to the canvas.

2 Drag the **Data Table** widget to the canvas.

3 At the right side of the **File** widget, there is a semi-circular shape. Mouse down on it and drag it to the **Data Table** widget.

4 Notice that there is a link between both widget with the word *Data* on top.

5 Do not worry about the error mark on top of the **File** widget as we have yet to load the data.
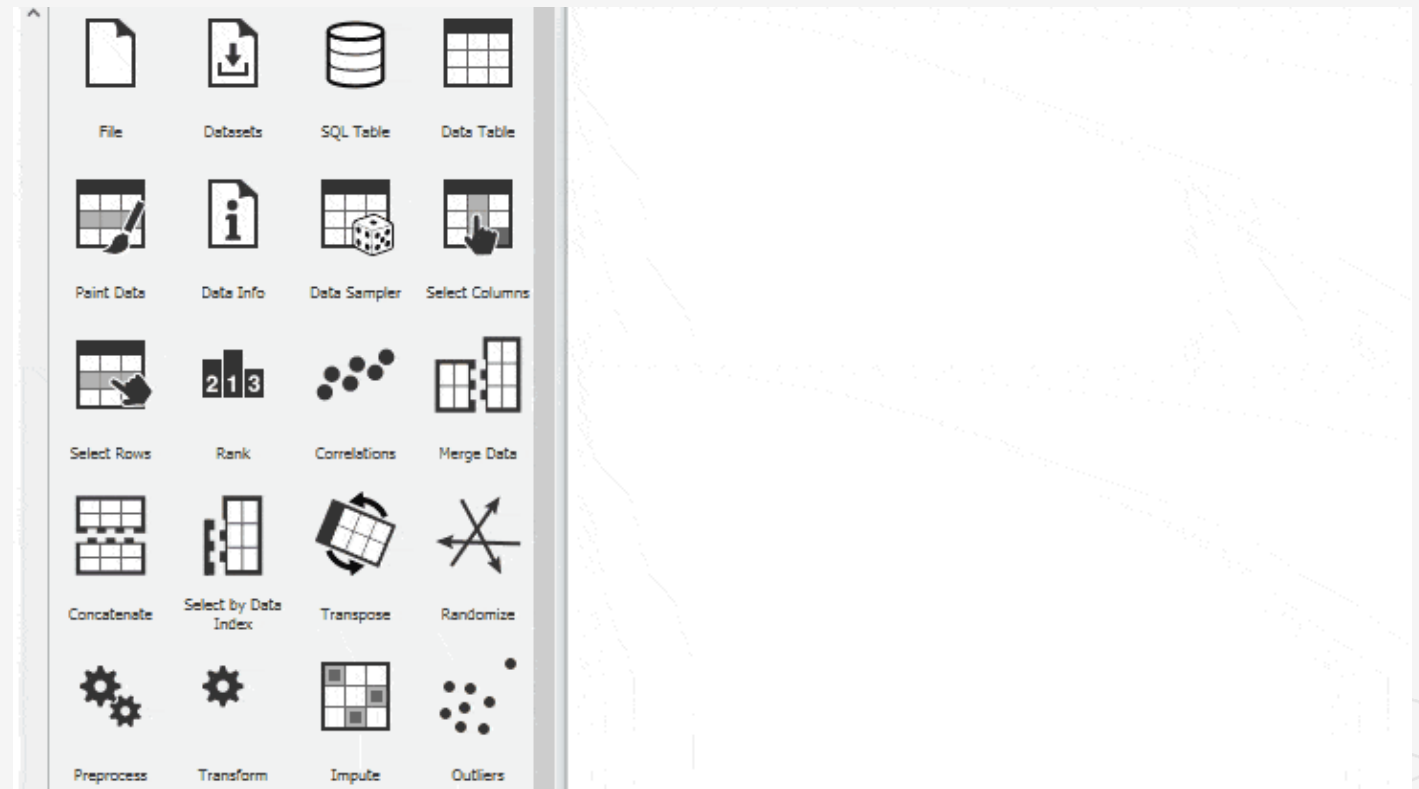
Hamza M. Al-Qatawna

Load Your Data
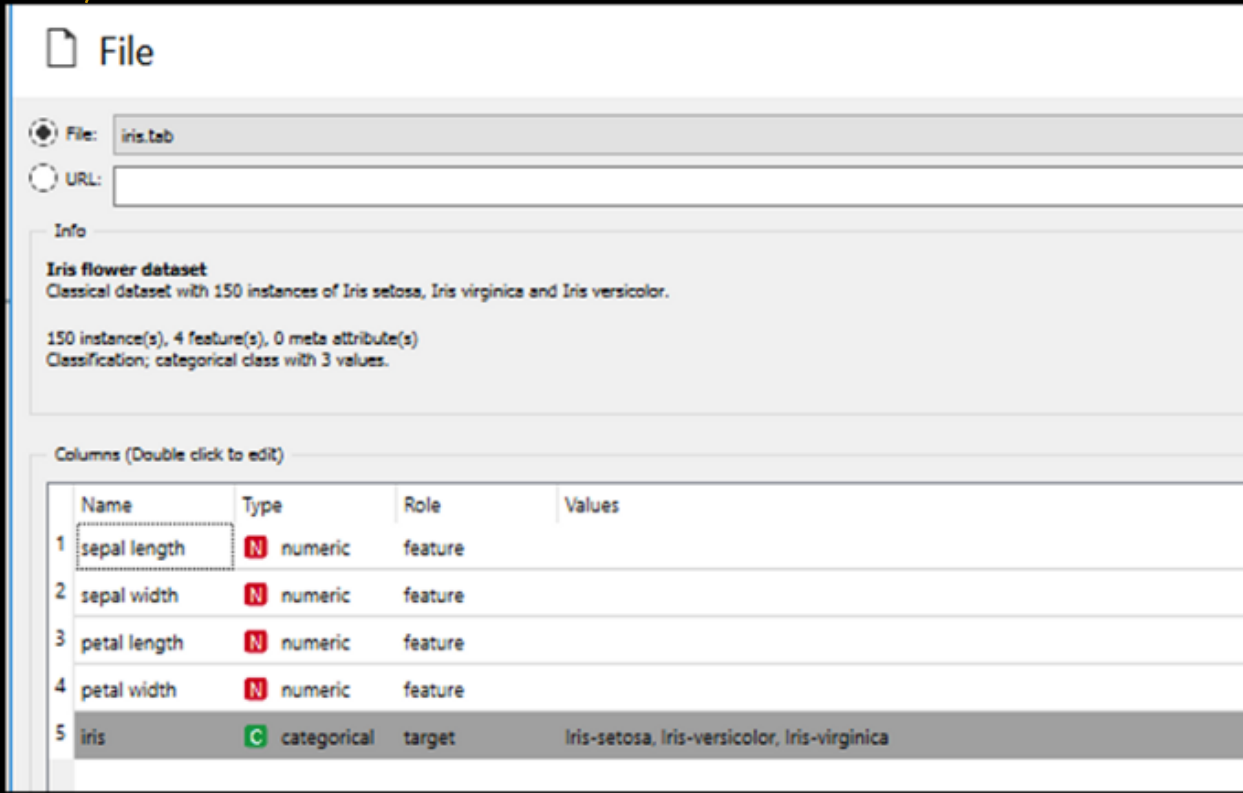
In the canvas, double click on the File widget to open it. Then, you can load your own dataset or browse it from the custom documentation dataset.

Let's try it with iris.tab via the custom documentation dataset. Orange accepts any of the following format:

- Tab-separated value
- Comma-separated value
- Basket file
- Microsoft Excel spreadsheet
- Pickled Orange data

Hamza M. Al-Qatawna

You should be able to see this screen

Data Table

Once you are done, close the pop up using the X mark at the top right. Do not be alarmed if you noticed that the **Apply** button is grey out. It is meant to apply changes after you have modified or edited the **Type** or **Role** of the data. Next, double click on the Data Table widget. You should be able to see the dataset. You can check the variables at the left-hand side to visualize the numbers. You should be able to see the following screen once you have checked all the options
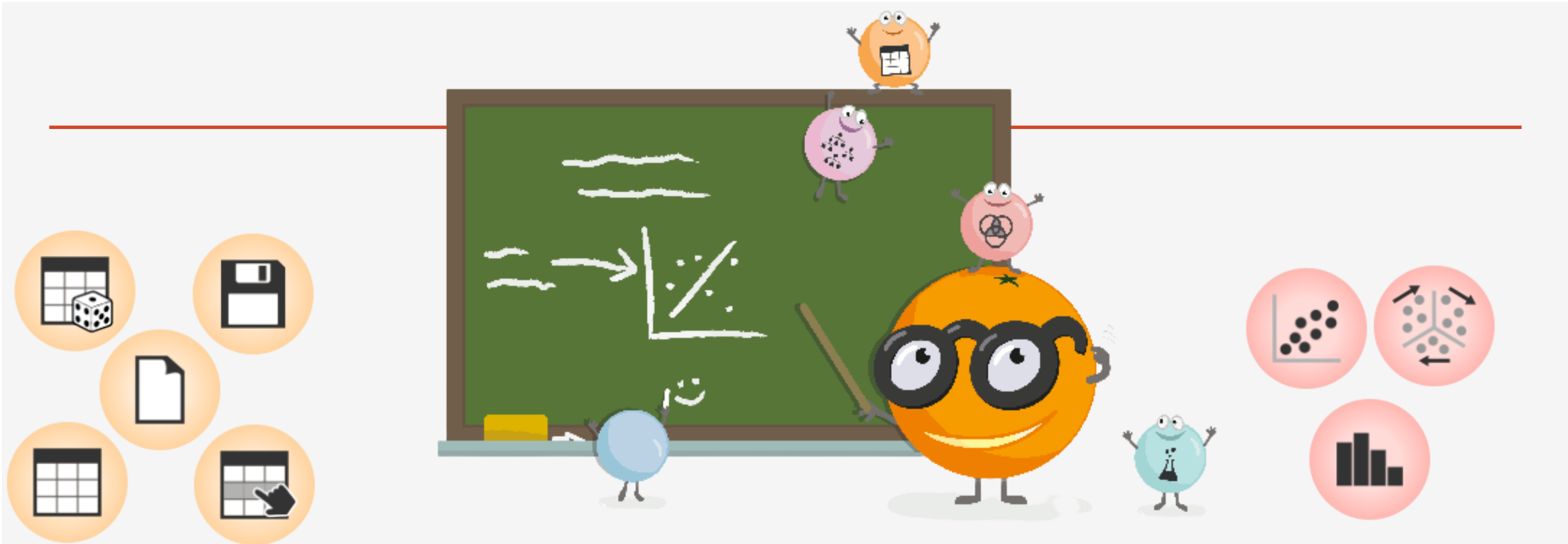

Hamza M. Al-Qatawna

In the next slides we are going to discuss some Data and Visualize Widgets, how to apply and make use of them step by step;
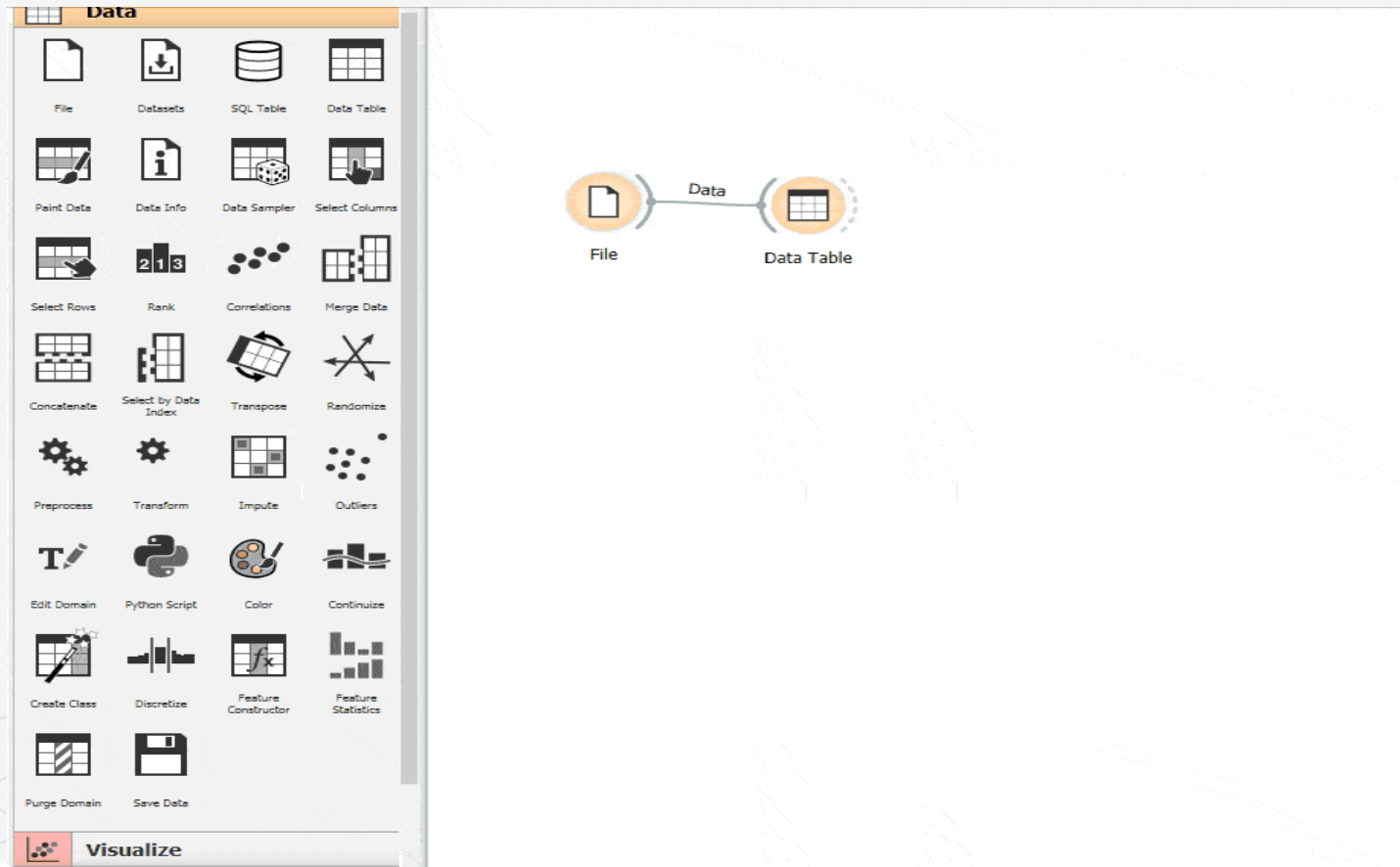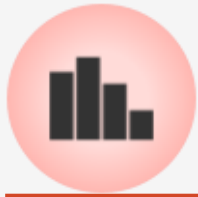
So, let's go

Hamza M. Al-Qatawna

Distribution Widget

We can visualize the data easily via some of the Visualize widgets.
Distribution is one of the best widget to identify important features for the dataset. You can easily visualize whether the dataset are separated nicely or not. Let's continue from the previous step.
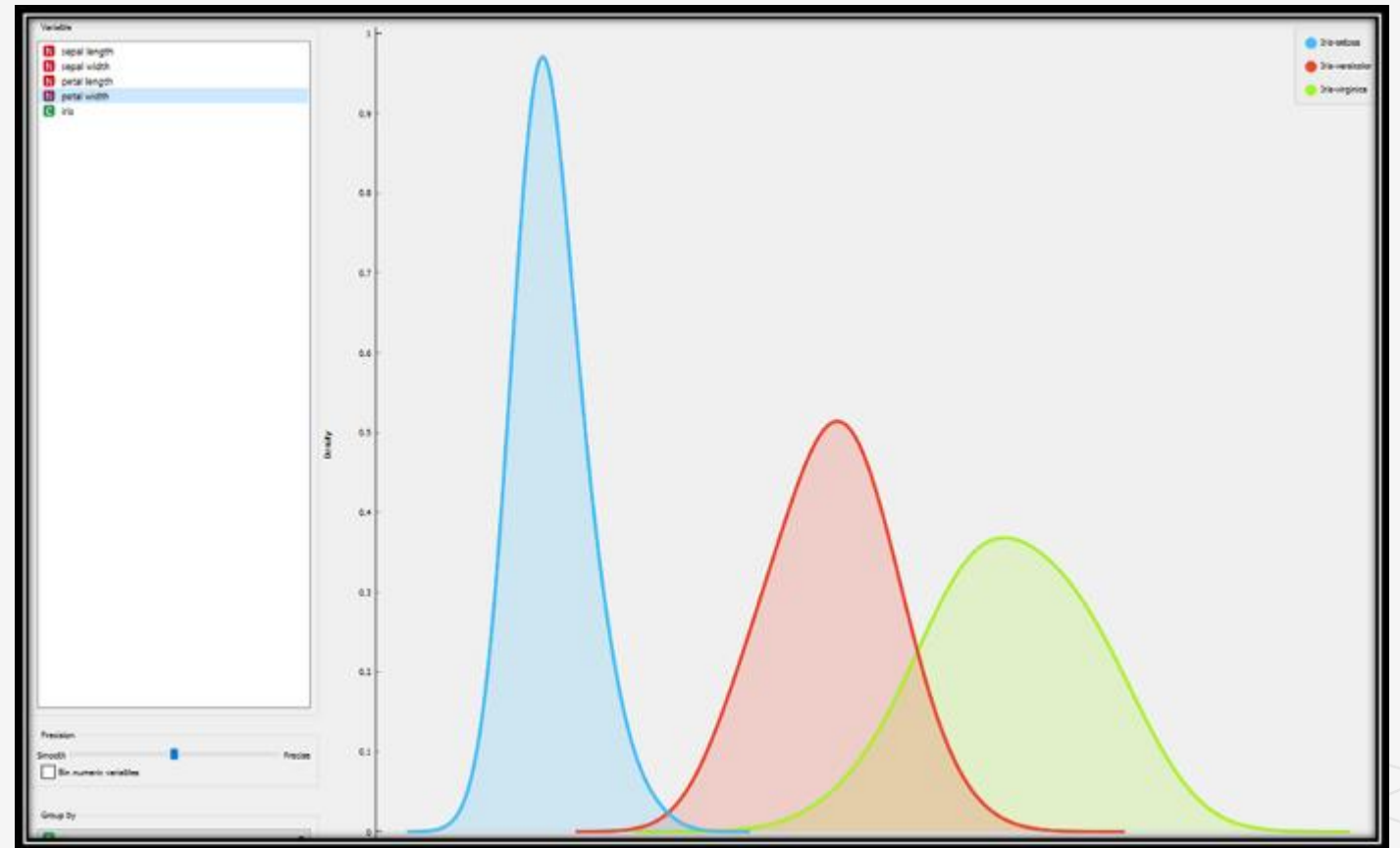

Hamza M. Al-Qatawna

Distribution Widget (Cont'd)

1 Drag a **Distribution** widget to the canvas.

2 Connect File widget to the **Distribution** widget.

3 Double click on the **Distribution** widget to see the visualization.

4 At the top left, select a different variable and check the distribution results.
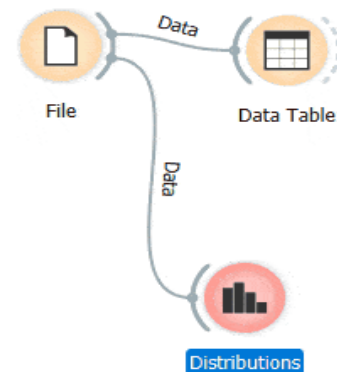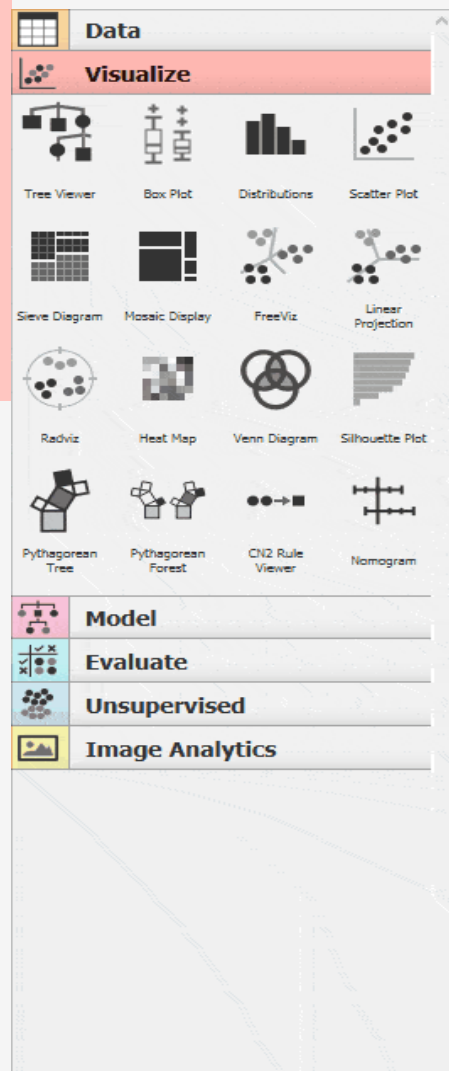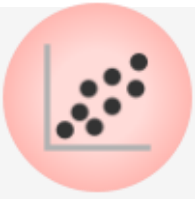

Hamza M. Al-Qatawna

Scatter Plot Widget

Scatter plot is another visualization widget that plot both features together to identify the projection between them.
Let's test it out!

1 Drag a **Scatter Plot** widget into the canvas.

2 Connect the **File** widget to the **Scatter**

3 **Plot** widget. The step is similar to how it was for the **Distribution** widget.
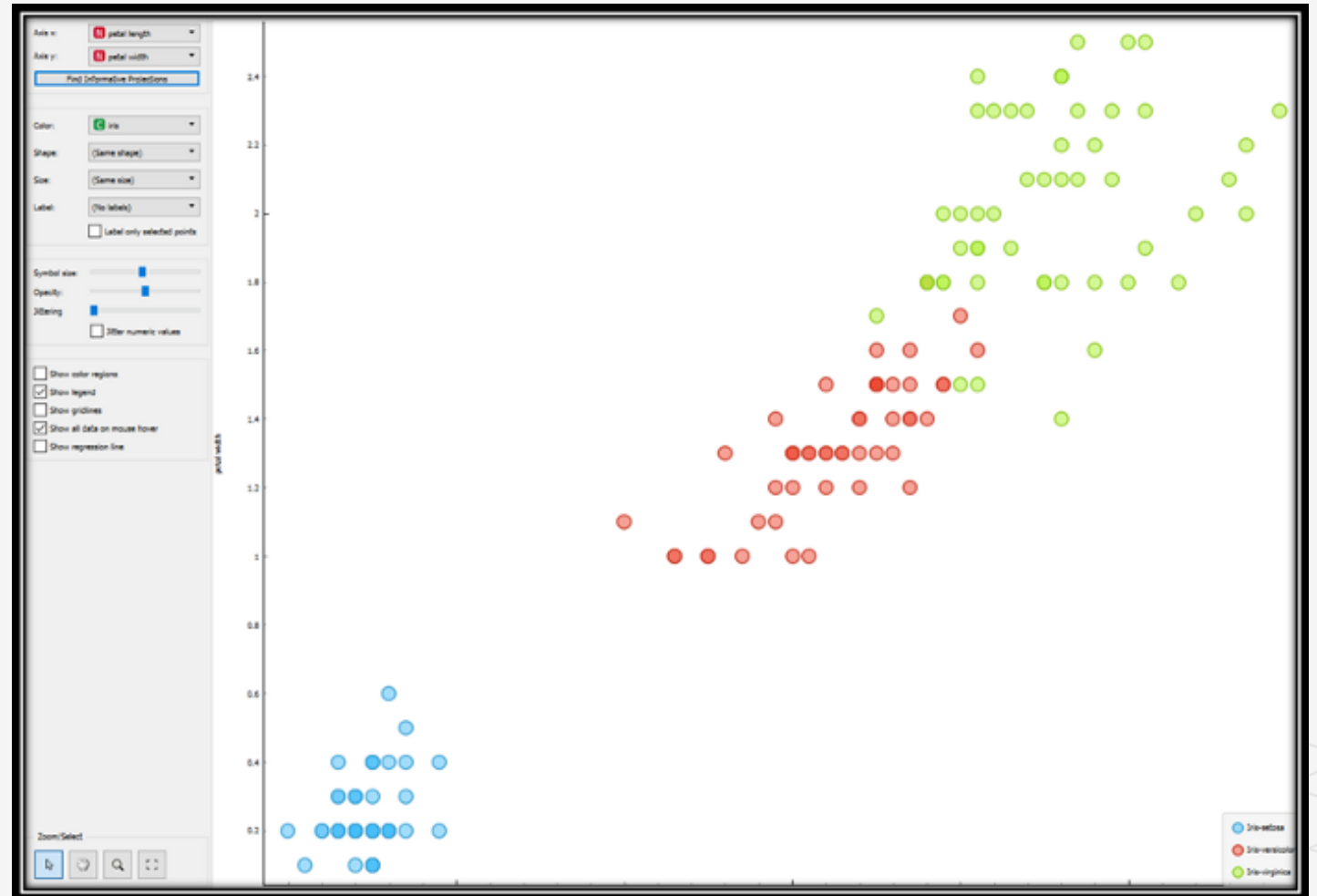

Hamza M. Al-Qatawna

Scatter Plot Widget (Cont'd)

4 Double click on the **Scatter Plot** widget to see the visualization.

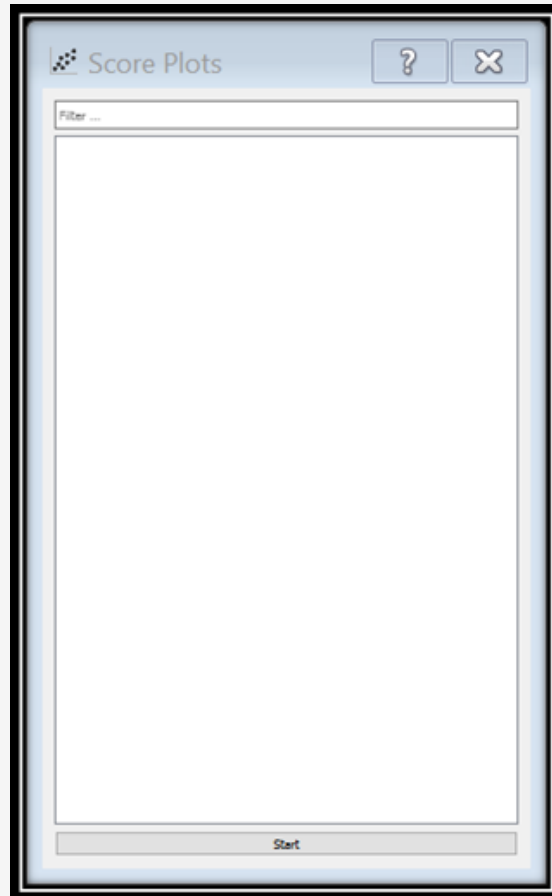**5** You can change the x-axis and y-axis based on the features available.
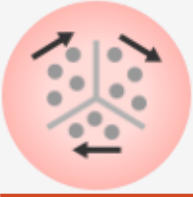
Hamza M. Al-Qatawna

Scatter Plot Widget (Cont'd)

If you are unsure which features to select, click on the "Find Informative Projections" and you will see the following interface.

Click on "Start" and select any item from the list. The scatter plot will be modified based on your selection.

Hamza M. Al-Qatawna

FreeViz widget

FreeViz widget uses a paradigm borrowed from particle physics. Although the points cannot be moved but the anchors are movable.
This allows us to visualize features that attracted each other and features that repel against each other.

Before you proceed, let's clear some of the widgets to keep it nice and tidy. There are two ways to delete widgets:

1 Right click on it and a menu will appear. Select the **Remove** option. You can also rename the widget this way or via the F2 shortcut.

2 Select it via left click and hit the "**Delete**" button on your keyboard. It is possible to multi select widgets and delete all of them together.
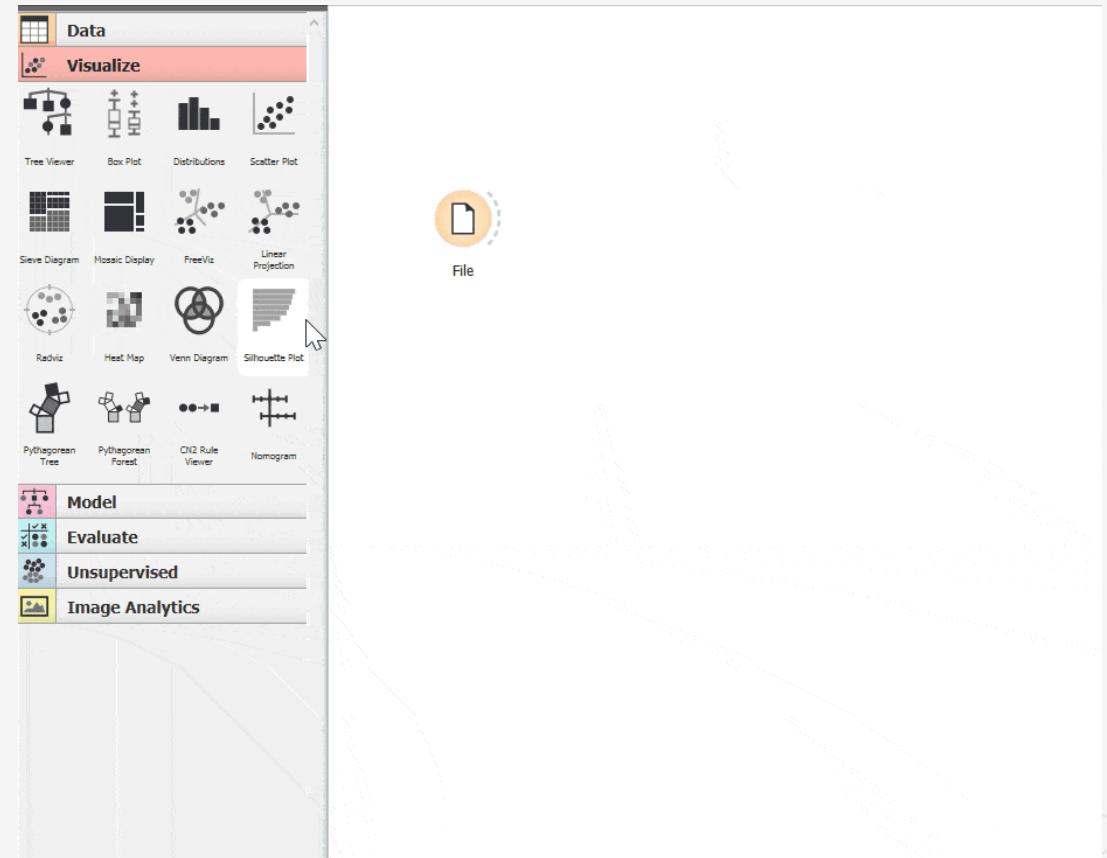

Hamza M. Al-Qatawna
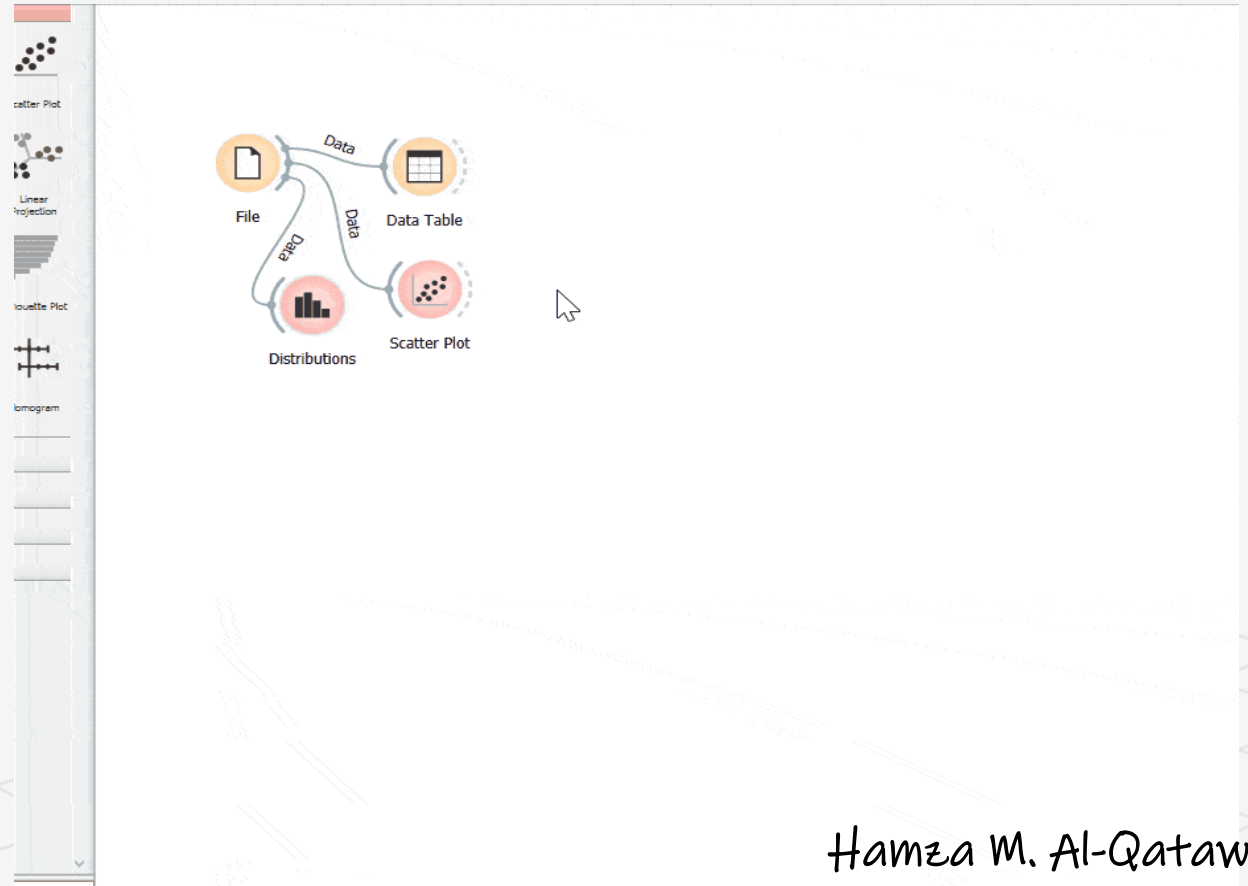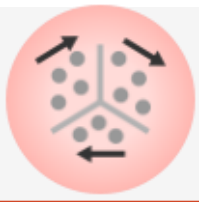
FreeViz widget (Cont'd)

Once you are done, let continue by following the instructions below:

1   Drop a **FreeViz** widget to the canvas.

2   Connect **File** widget to the **FreeViz** widget.

3   Drop a **Data Table** widget to the canvas.

4   Connect **FreeViz** widget to the **Data Table** widget

If you are wondering why we are connecting **Data Table** widget with **FreeViz** widget instead of **File** widget. The reason is that you can select the data points in any of the **Visualize** widget interface and the selected points will be outputted to the **Data Table** widget.
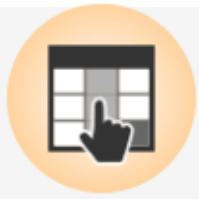
Hamza M. Al-Qatawna
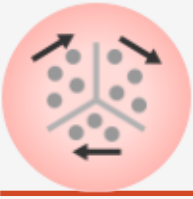
FreeViz widget (Cont'd)

Let's have a look at the following gif to find out more on how to move the anchor point and select data points for FreeViz interface
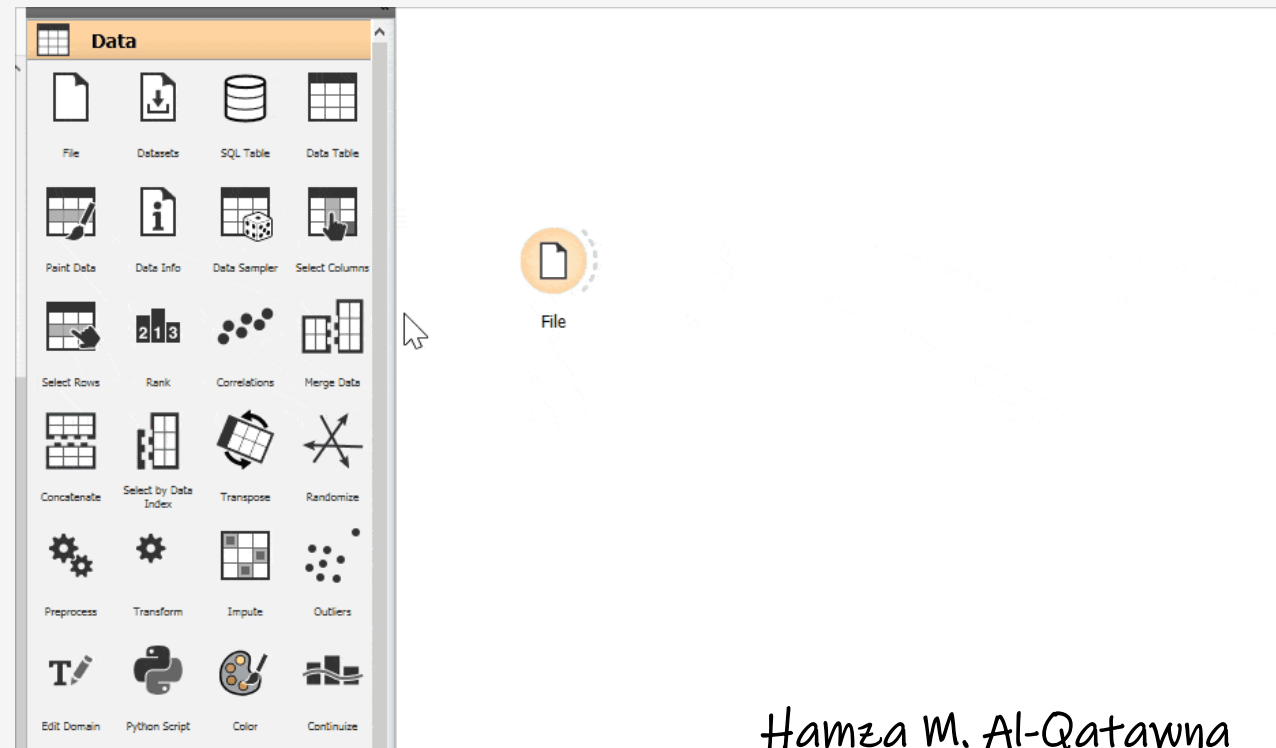
Hamza M. Al-Qatawna
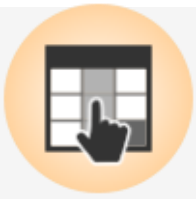
Select Columns Widget

Let's say you have successfully identified a few important features from the dataset and would like to create a new dataset with only those features. You can easily do so with the **Select Columns** widget

Assuming that you have loaded the Iris dataset into the **File** widget via the browse custom documentation

1.Drag a **Select Columns** widget to the canvas.

2.Connect both the **File** Widget and **Select Columns** widget.

3.Double click on the **Select Columns** widget to open up the interface.
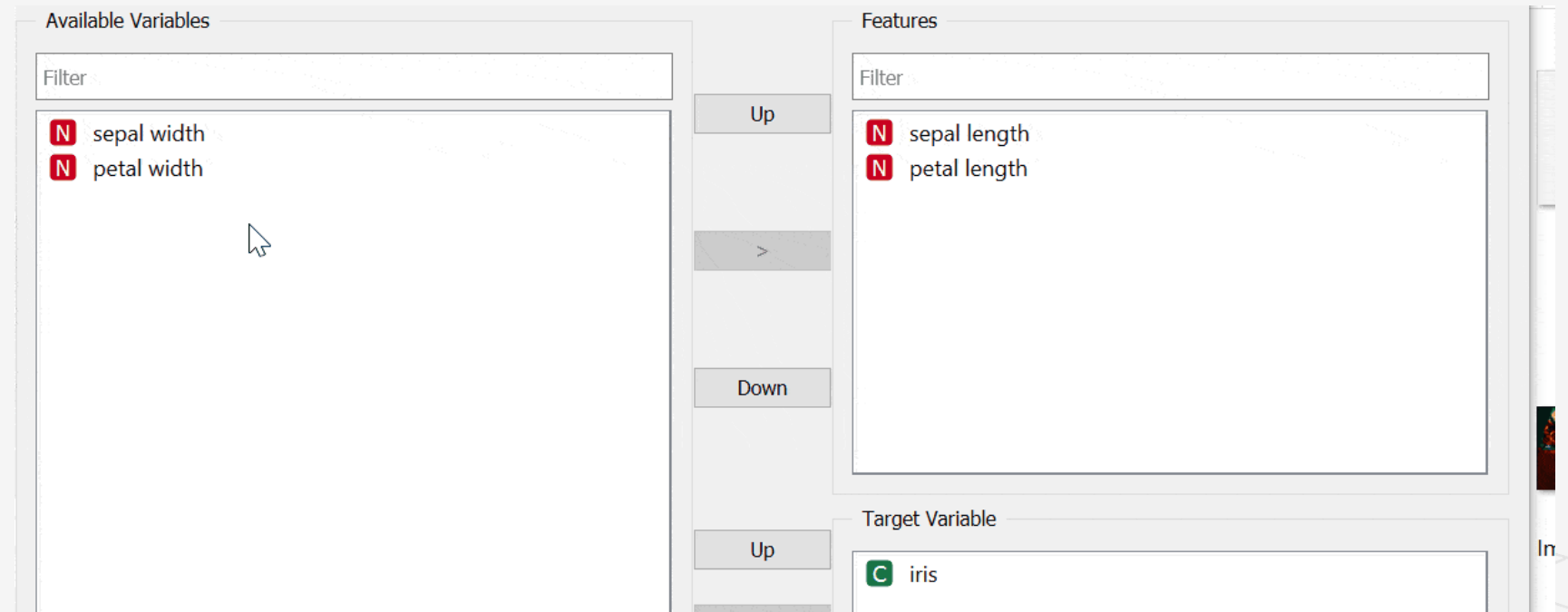
Hamza M. Al-Qatawna

Select Columns Widget (Cont'd)

Once you have opened the **Select Columns** interface, you can move the variables between left and right. Left represents unwanted features while right represents selected features.

Once you have made up your mind, close the interface and double click on the **Data Table** widget.

You should be able to see the features that you have selected.
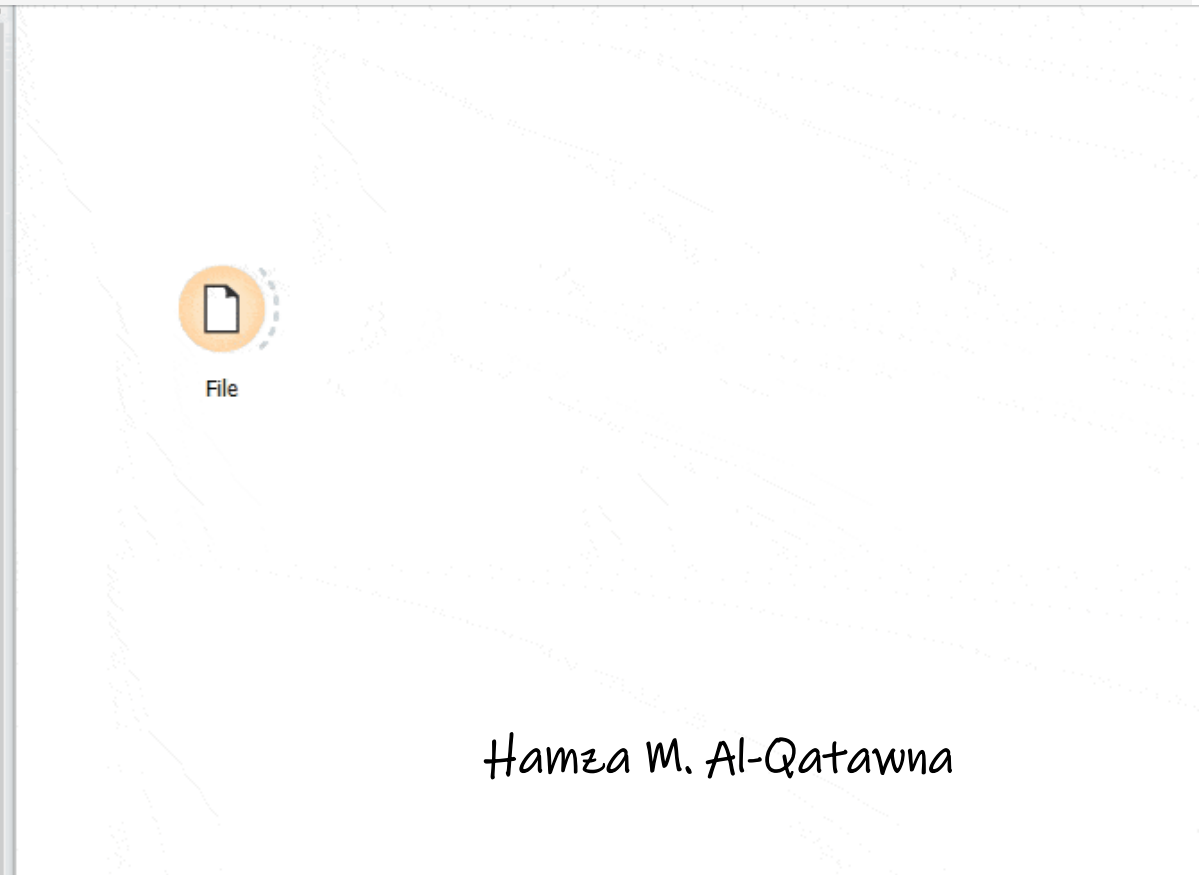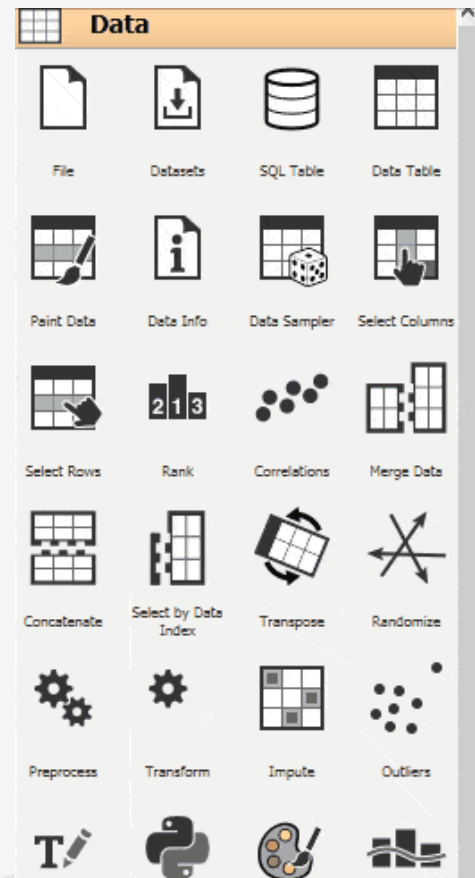
Select Rows Widget

Apart from being able to just select features, you can also filter out the data into something that you desired.

Let's explore a little more on the **Select Rows** widget.

1.Drag a **Select Rows** widget to the canvas.

2.Connect both the **File** Widget and **Select Rows** widget.

3.Double click on the **Select Rows** widget to open up the interface.
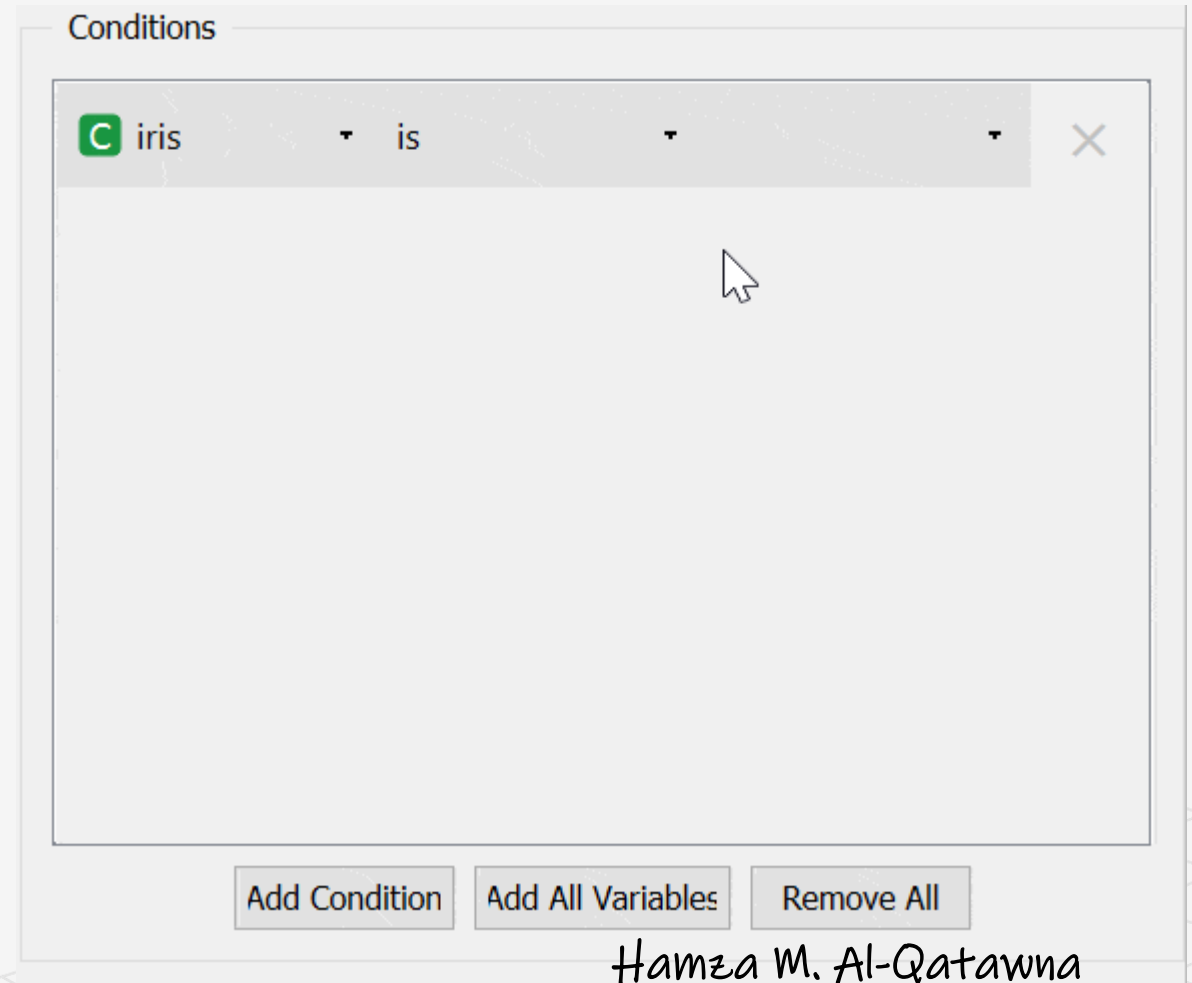

Hamza M. Al-Qatawna

Select Rows (Cont'd)

You can set your conditions using the **Select Rows** interface to filter out the desired data.

Assuming that you are using the Iris dataset, we can filter out data that are of specific class.

Likewise, we can filter out data that are not from specific class. In addition, we can also set condition for the features such as the sepal length must be greater than certain figure.

You can add as many condition as you like. Double click on the **Data Table** widget, you should be able to see the filtered data.

Conditions

| C iris | ▾ | is | ▾ | ▾ | ✕ |

Add Condition   Add All Variables   Remove All
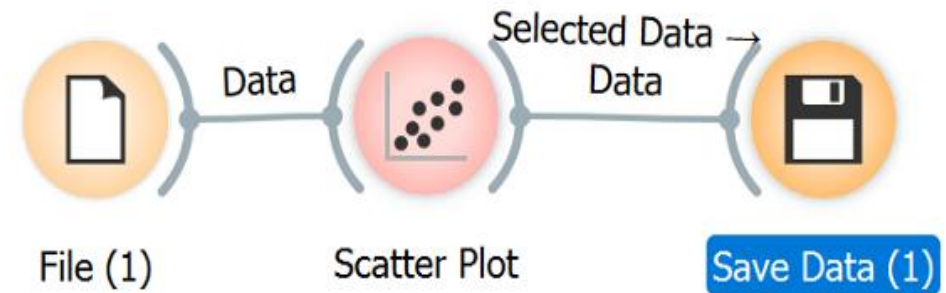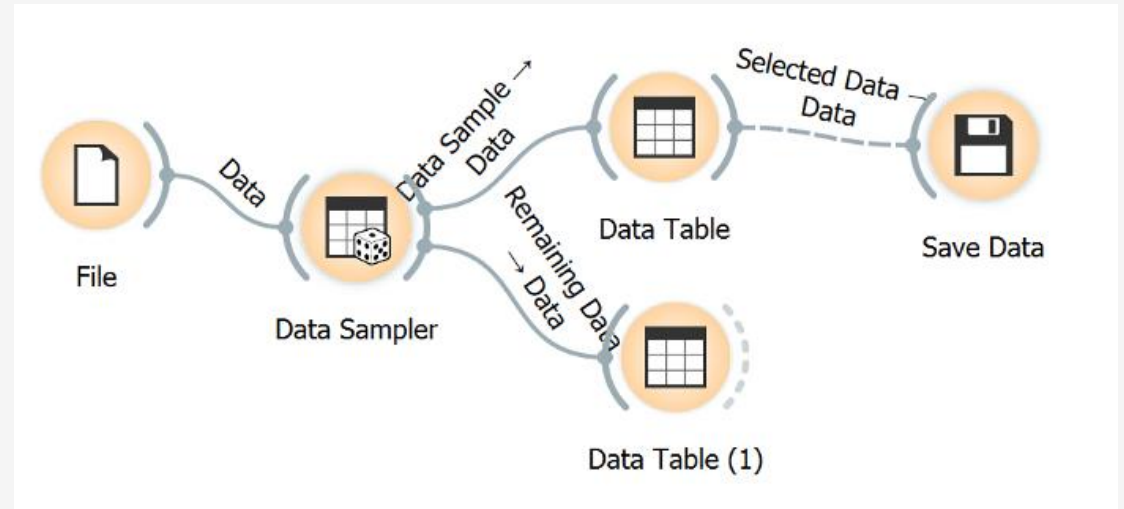
Hamza M. Al-Qatawna

Save Data Widget

All the data processing conducted so far are are only applicable on Orange. The changes are not reflected on the original dataset. If you wanted to save the processed data, you should use the **Save Data** widget.

You can place it after the **Data Table** to save the train data independently just like the figure above.

Likewise, you can select some data in any **Visualize** widget and save only the selected data.
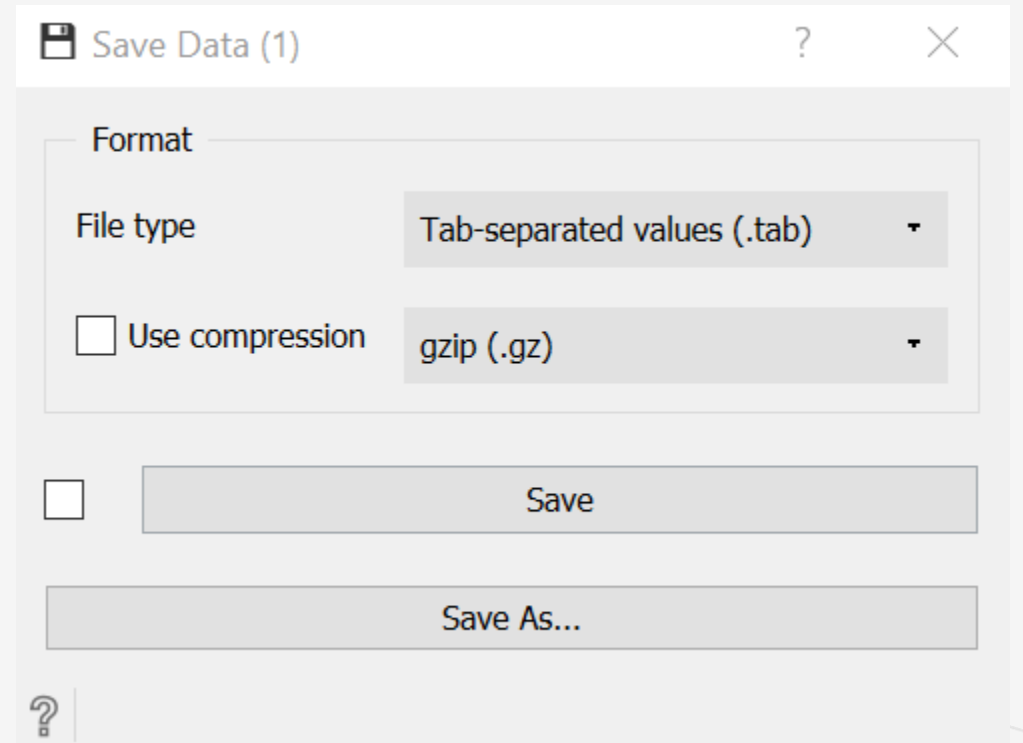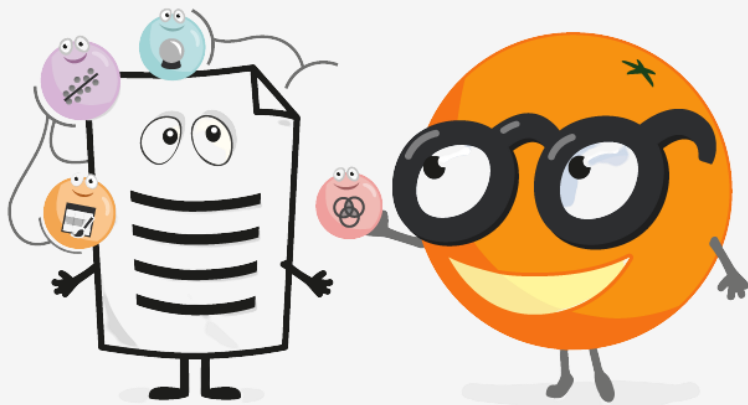
Hamza M. Al-Qatawna

Save Data Widget (Cont'd)

You can determine the File type and check to use compression or not. There are two ways to save:

- **Save**: overwrite the existing files
- **Save As**: create a new file

Save Data (1)

Format

File type: Tab-separated values (.tab)

☐ Use compression   gzip (.gz)

☐   Save

Save As...

Hamza M. Al-Qatawna

Hope you found this tutorial useful!

References:

1. Faculty of Computer and Information Science University, Slovenia; Orange: Data Mining Toolbox in Python; research Paper

2. JANEZ DEMSAR AND BLAZ ZUPAN; From Experimental Machine Learning to Interactive Data Mining, Researchgate

3. Websites:

www.orangedatamining.com

www.towardsdatascience.com

Special thanks to :

Dr. Anas H. Blasi

for this great opportunity and experience.

Hamza M. Algatawna

Mutah University

Hmsqv12@gmail.com