Orange: Data Mining Fruitful and Fun

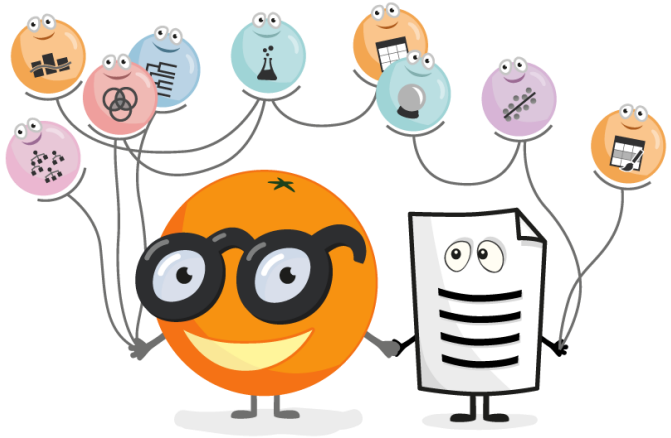
Lorenzo Stacchio

PhD student in Computer Science

Department for Life Quality Studies

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

What is Orange?



- Orange is a free software created to perform Data Mining, Machine learning and data visualization without code!
- It allows you to create data analysis workflows visually, with tons of useful tools.
- Orange is the perfect software for Machine Learning and statistical operations without code!
- We will see the power of Orange applied to different domains in fashion such as text, images and time series!

Why Orange?

- Teachers and students love it;
- Interactive data visualization;
- Visual programming;
- It provides a lot of features in addition to the basic ones;

Teachers and students love it

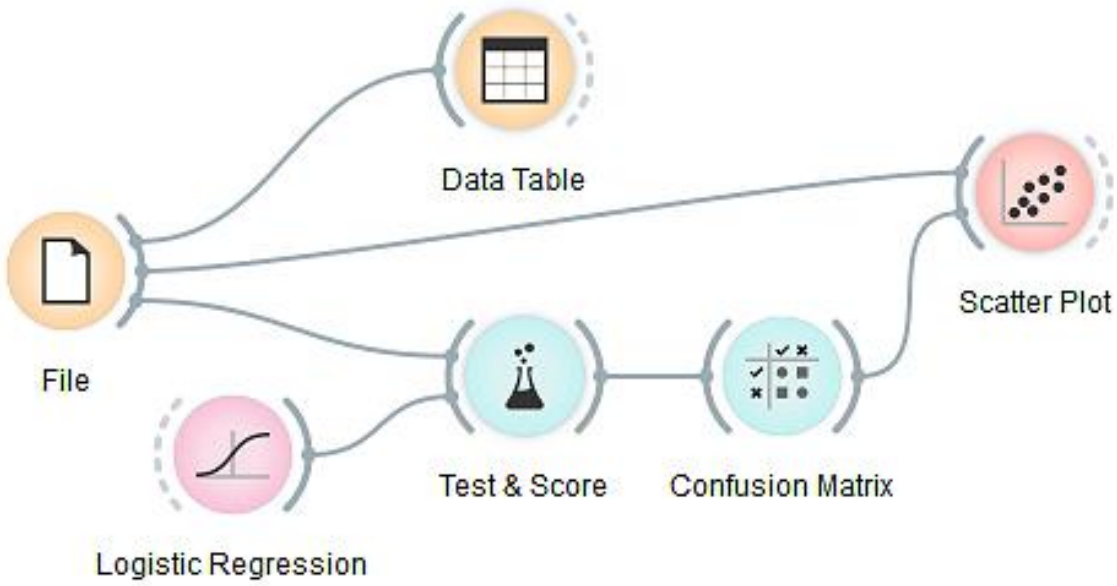
- Orange is the perfect tool for practical training;
- Teachers appreciate the clear program design and visual explorations of data and models;
- Students benefit from the tool's flexibility and the ability to invent new combinations of data mining methods;
- Orange's educational strength comes from the combination of visual programming and interactive visualizations. We have also designed some educational widgets that have been explicitly created to support teaching.
- Here are some sample workflows we recently used in data mining training (yes, we not only develop Orange, we also teach with it).

Interactive data visualization

- Orange is focused on data visualizations that help uncover patterns of data patterns, provide insights behind data analysis procedures, or support communication between data scientists and domain experts;
- **View widgets** include a variety of viewing tools: scatter plot, box plot and histogram, and specific views of some ML models;
- Why interactive? Points can be selected from a scatter plot, a node in a decision tree and many other interactions;
- Any of these interactions will cause a specific change in the various widgets.

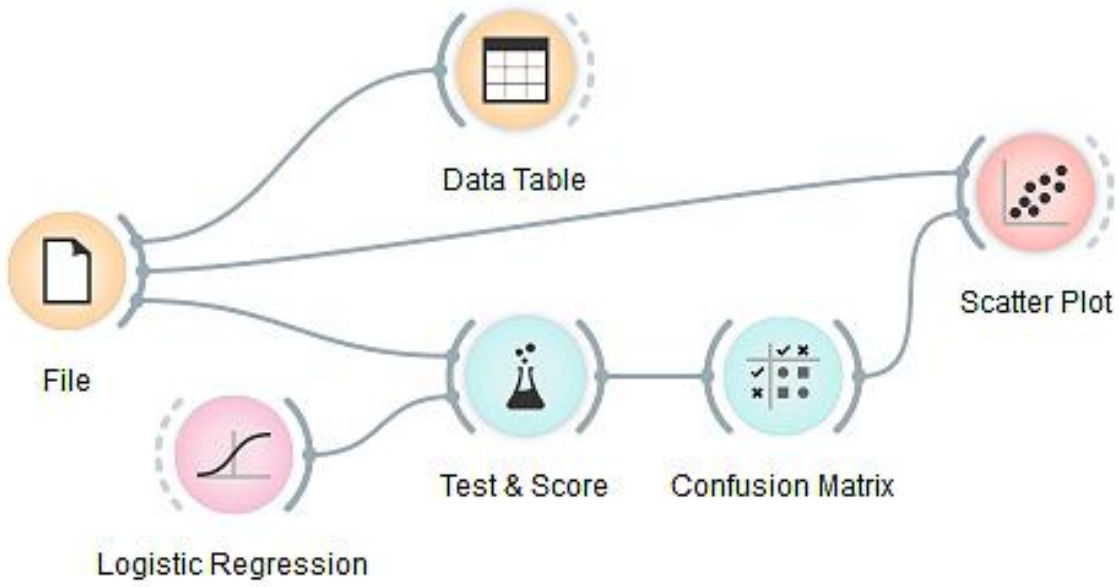
Visual programming

- Orange is a great data mining tool for both beginners and experienced data scientists;
- Thanks to the simple user interface, users can focus on data analysis rather than laborious coding, making it easy to build complex data analysis pipelines;



Visual programming – Components

- With Orange, data analysis is performed by stacking components in workflows;
- Each component also called a widget, incorporates some data collection and processing, visualization, modeling, or evaluation activities;
- Each widget possesses:
  - a color, which indicates its class;
  - A bonding structure (pre/post linking);
  - Your own responsibility;
- Combining different widgets in a workflow allows you to create data analysis patterns piece by piece;



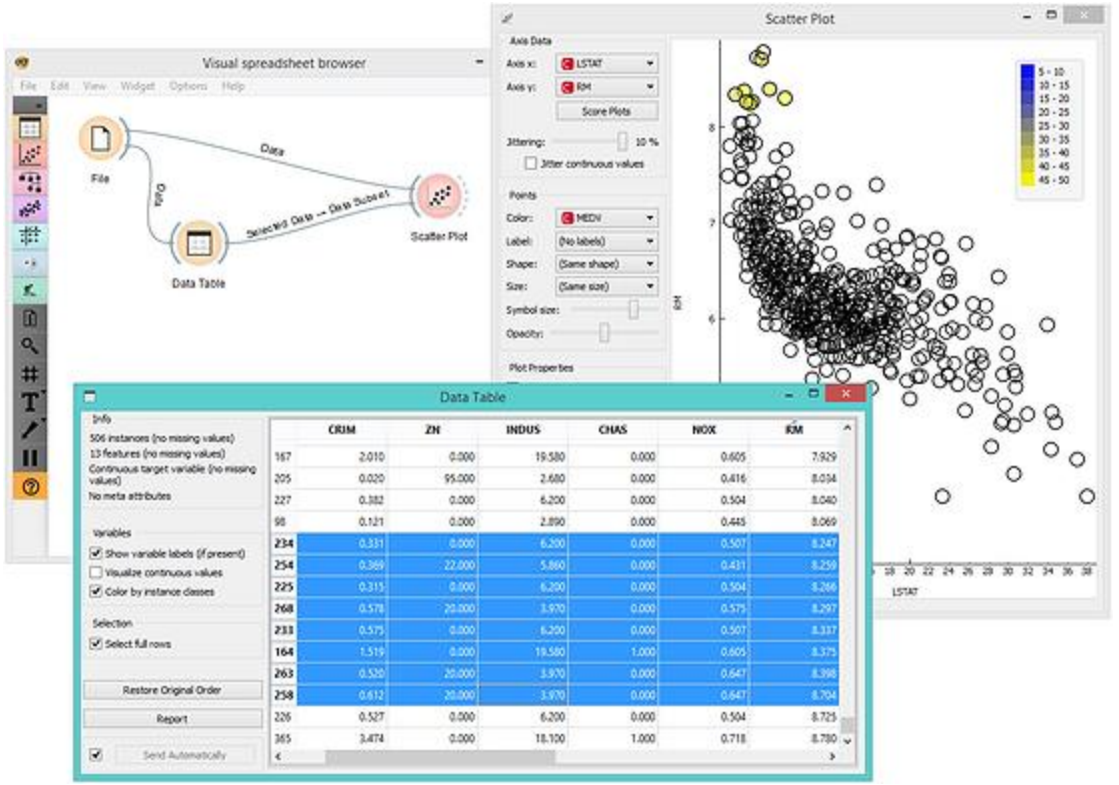
Visual programming – Components and Workflow

- The orange widgets communicate with each other;
- Through linking, they receive input data and send data, templates or whatever the widget produces;
- For example, we could start with a File widget that reads the data and links its output to another widget, such as a machine learning model that learns from that data, finally followed by an evaluation widget for that algorithm!
- After any modification in a widget, the changes are instantly propagated through the workflow to all other connected widgets;
- E.g., editing a data file in the File widget will trigger the response in all other widgets.
- There is therefore a form of workflow reuse!

Visual programming – Components and interactive visualization

- This is especially fun when widgets are open and when you can immediately see the results of any changes to that data, method parameters or selections in interactive views;

- For example, in a simple workflow below, where the data selection in the spreadsheet propagates to a scatter chart, which marks the selected data instances.



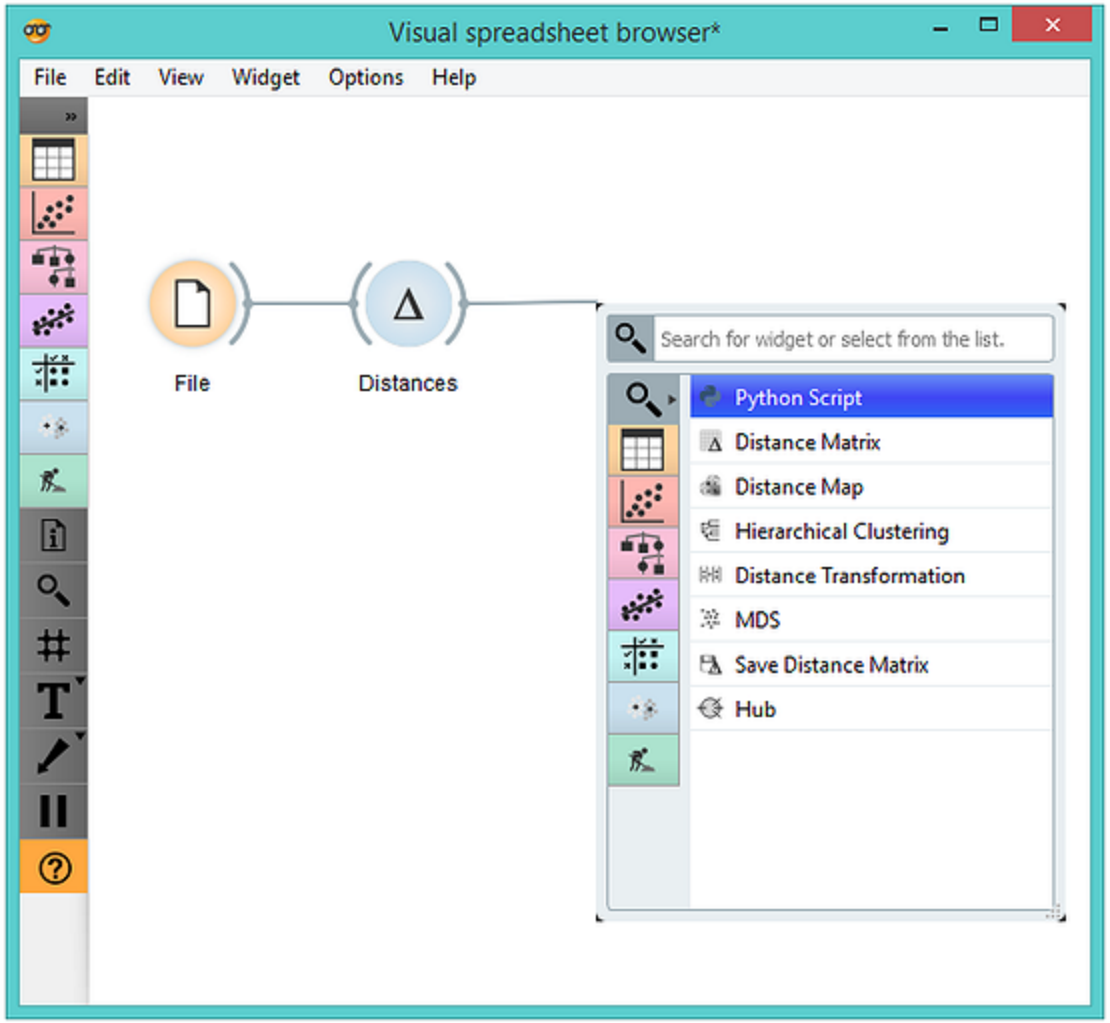
Many add-ons

- Use the various add-ons available in Orange to extract data from external data sources, perform natural language processing and text extraction, analyze time series etc…

- In addition, among others, there is an add-on that allows you to analyze images and build on them Machine learning algorithms! We will field this add-on with a very special dataset.

Why Orange? (extra)

- **Orange is fully coded in Python!**
- What does this mean? You can write python code to completely customize your Orange experience or you can build new widgets to help the community!



Why Orange? (extra)

- Orange is entirely opensource!
- All the code is free and editable to your liking and is located directly on GitHub!
- There is a community that continues to develop free add-ons to increase the number of tools that people who can't code can use!
- See this page: https://github.com/orgs/biolab/repositories

Let's get along with Orange using Iris

Lorenzo Stacchio

PhD student in Computer Science

Department for Life Quality Studies

Iris dataset

• This is perhaps the best-known database found in the pattern recognition literature;

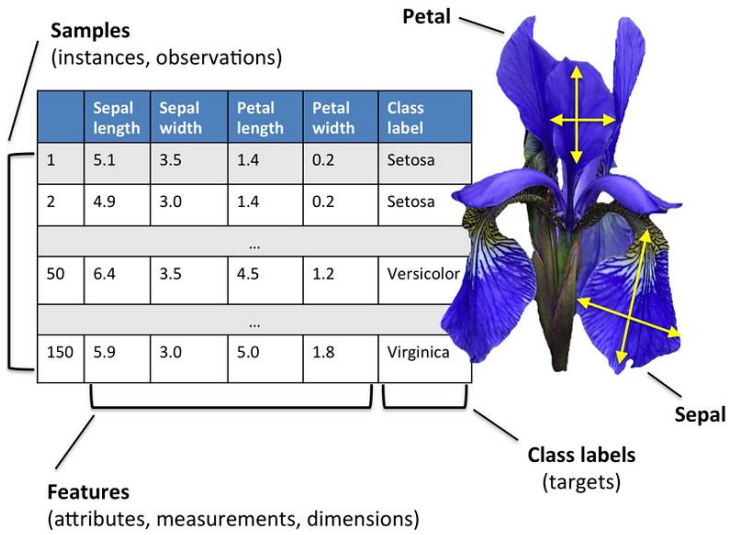
• The dataset contains 3 classes of 50 instances each, where each class refers to a type of iris plant;

• It contains a small challenge from a mathematical point of view;

• Expected attribute: class of the iris plant;

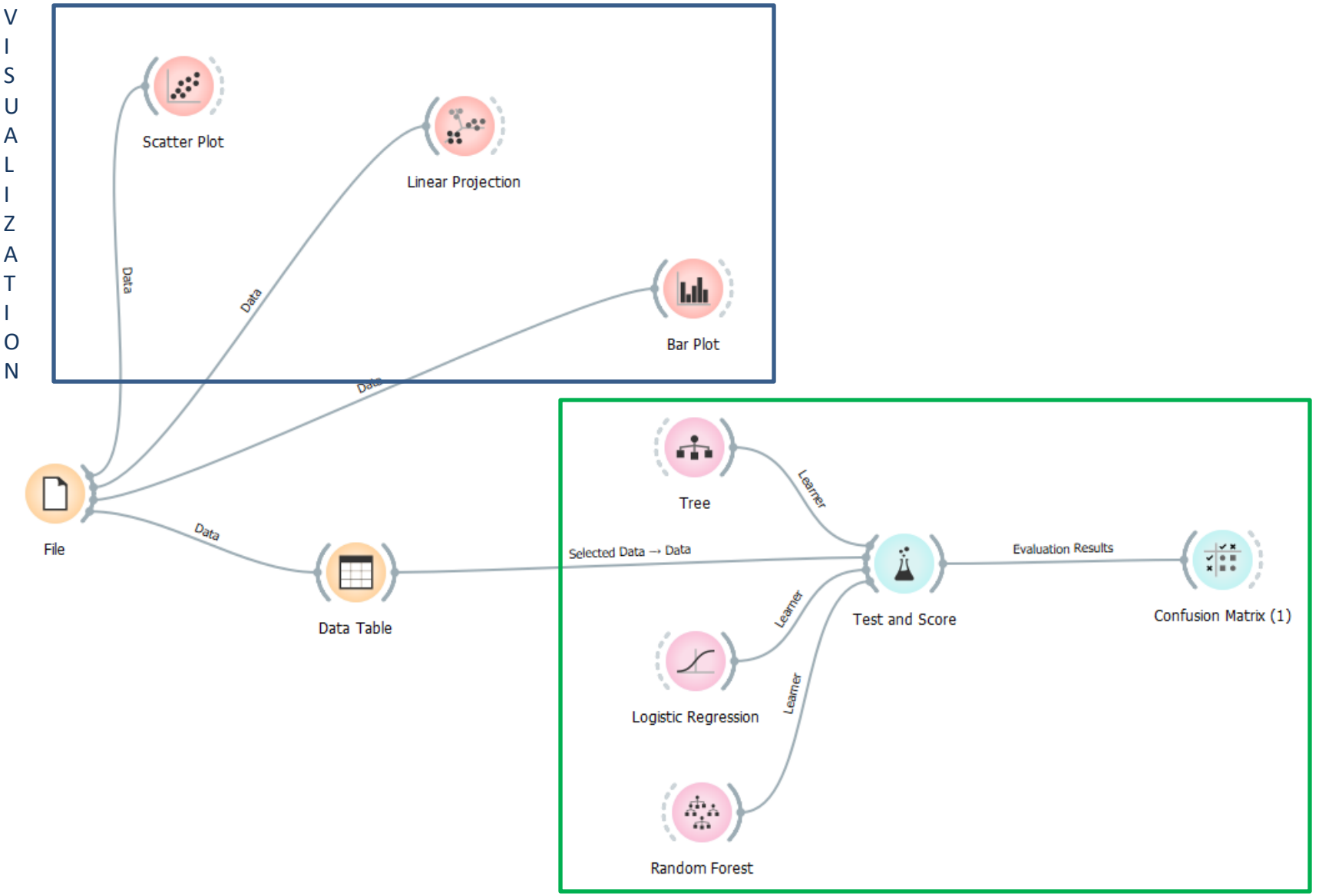
• Characteristics:

  • length of the sepal in cm
  • sepal width in cm
  • petal length in cm
  • petal width in cm
  • classes: Iris Setosa, Iris Versicolor, Iris Virginica



	Sepal length	Sepal width	Petal length	Petal width	Class label
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
...					
50	6.4	3.5	4.5	1.2	Versicolor
...					
150	5.9	3.0	5.0	1.8	Virginica

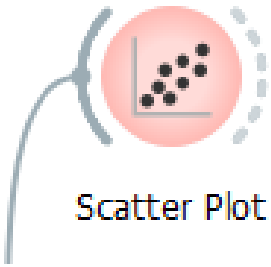
General workflow

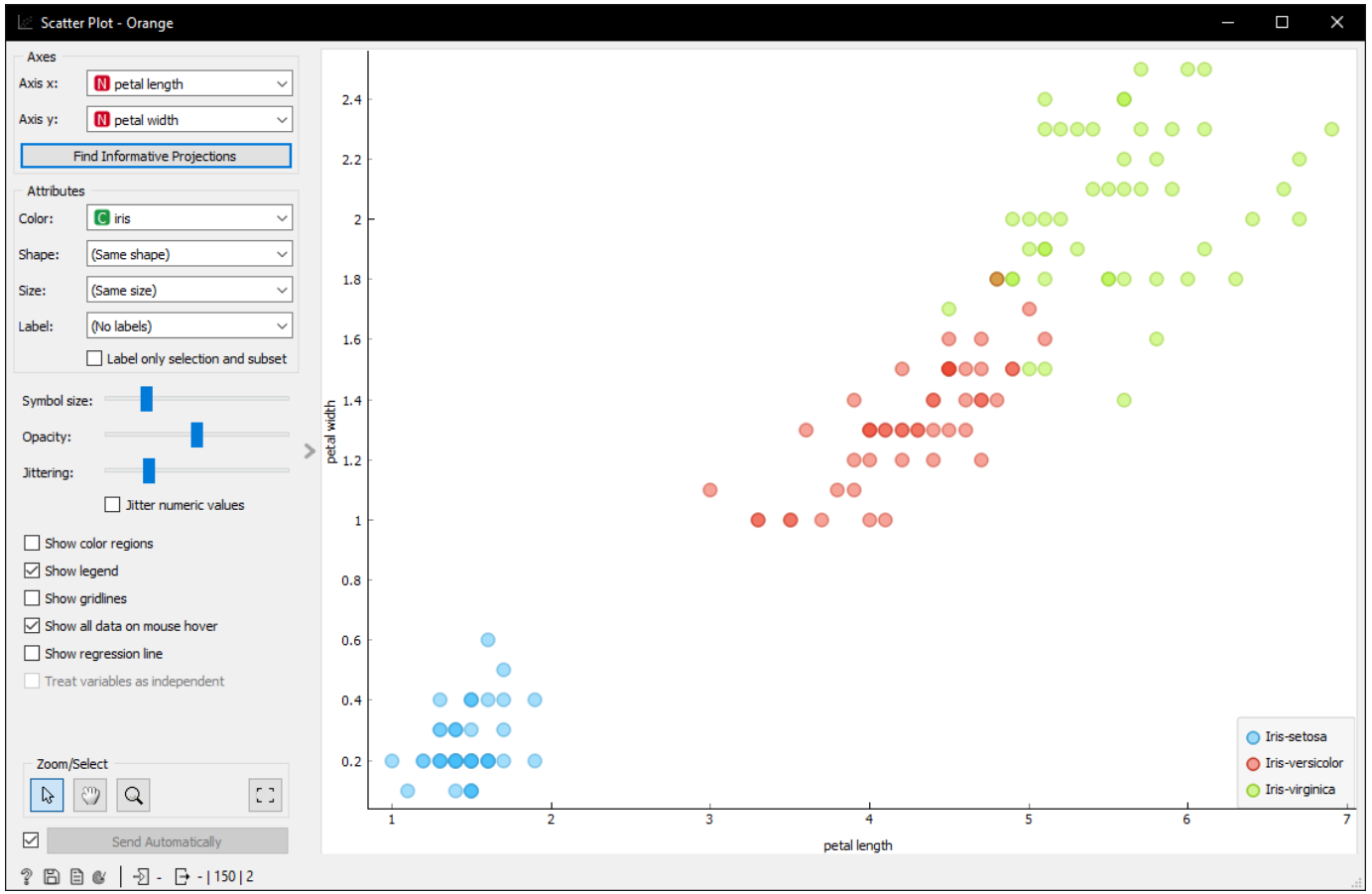


We upload the dataset directly from Orange!

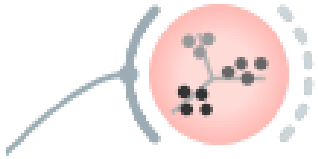
- Orange provides ready-made datasets to get familiar with the software;
- We will start with the now famous iris dataset;
- To load it just use the widget file, click it and choose iris.tab!

Data visualization: scatter plot

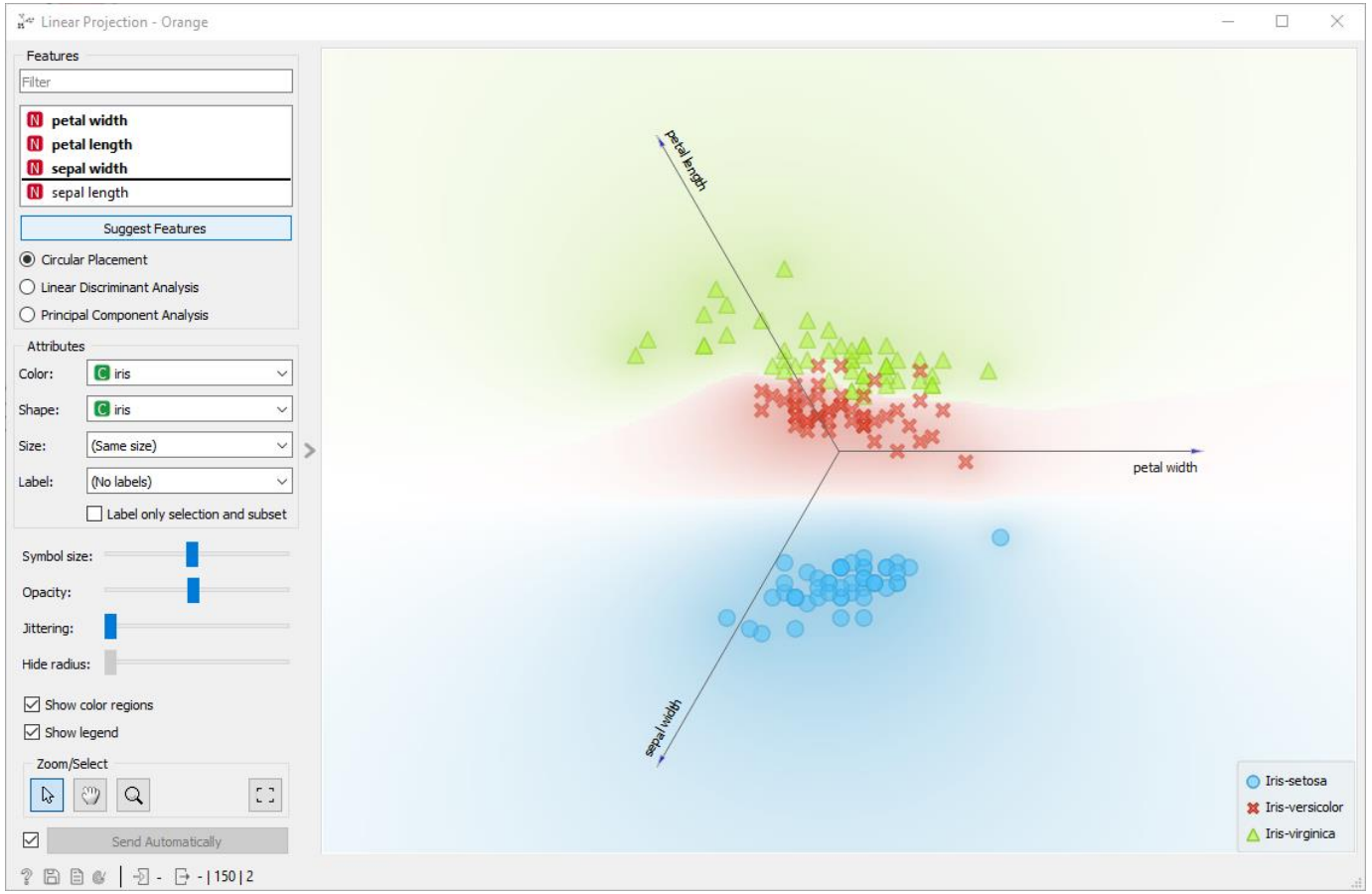




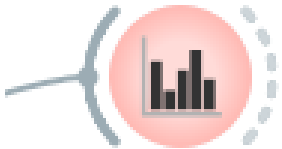
Data display: Linear projection



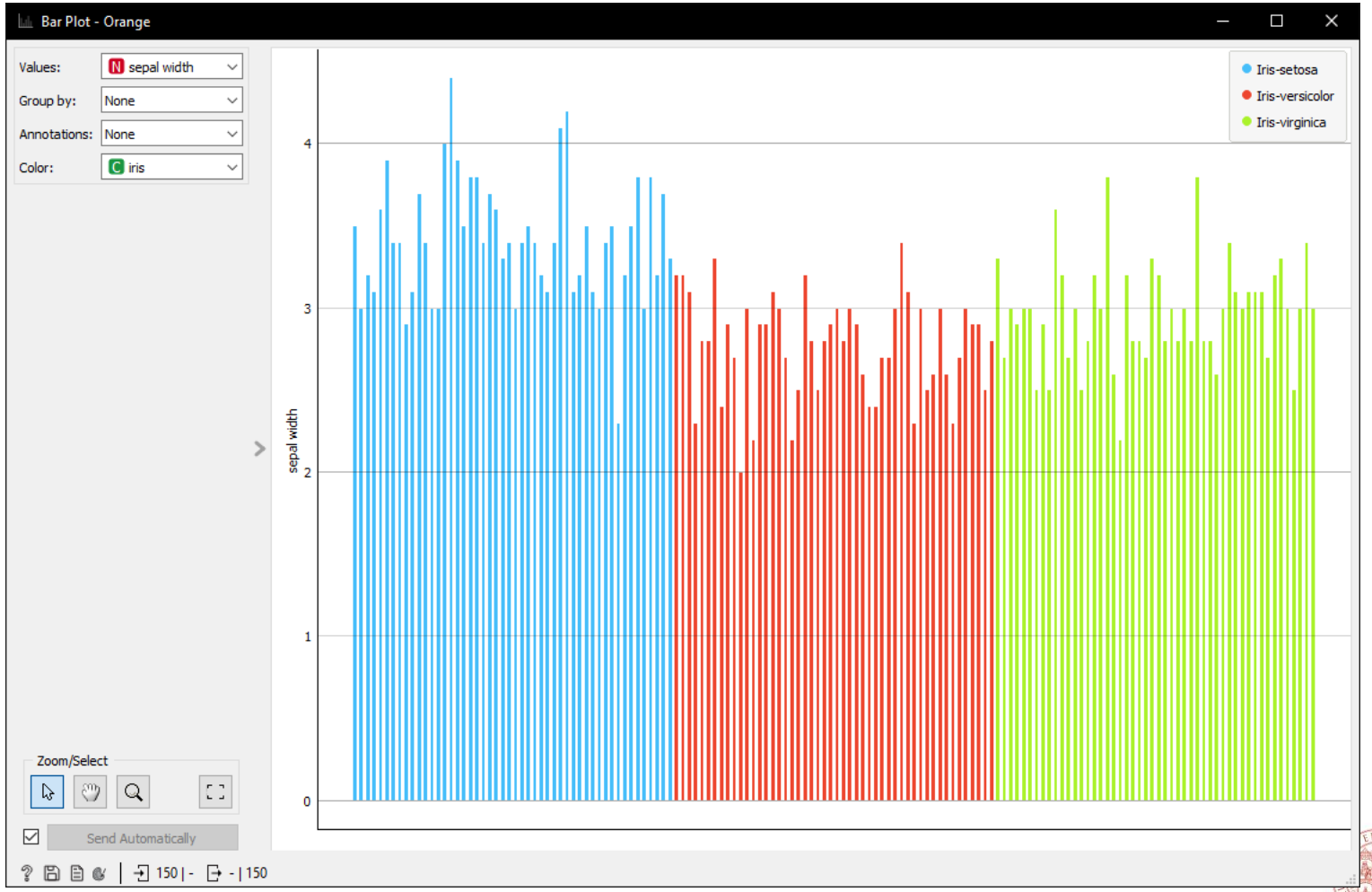
Linear Projection



Data visualization: Bar plot



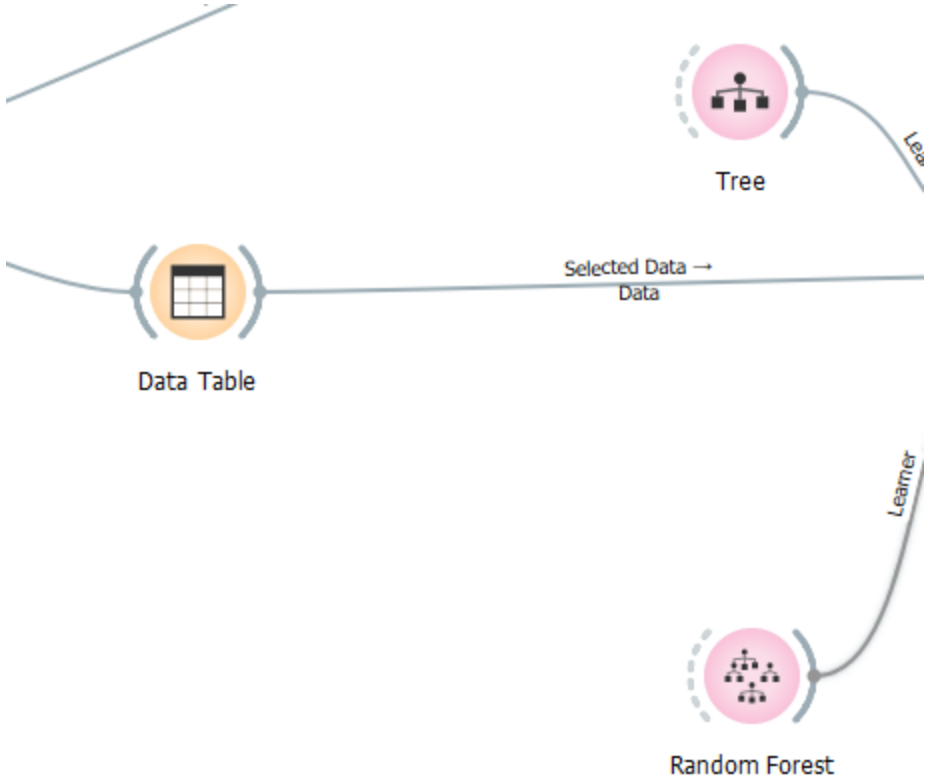
Bar Plot



Let's create a tree and a random forest

• To create two classifiers, just drag and drop the "Tree" and "Random forest" widgets into the scene!

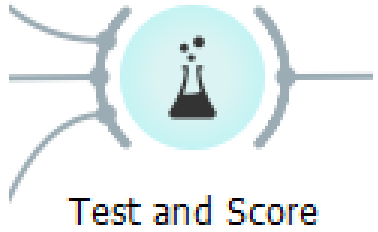
• In the meantime, let's add a "data table" widget and link it to the "file" widget, this will allow orange to treat the file as tabular data!

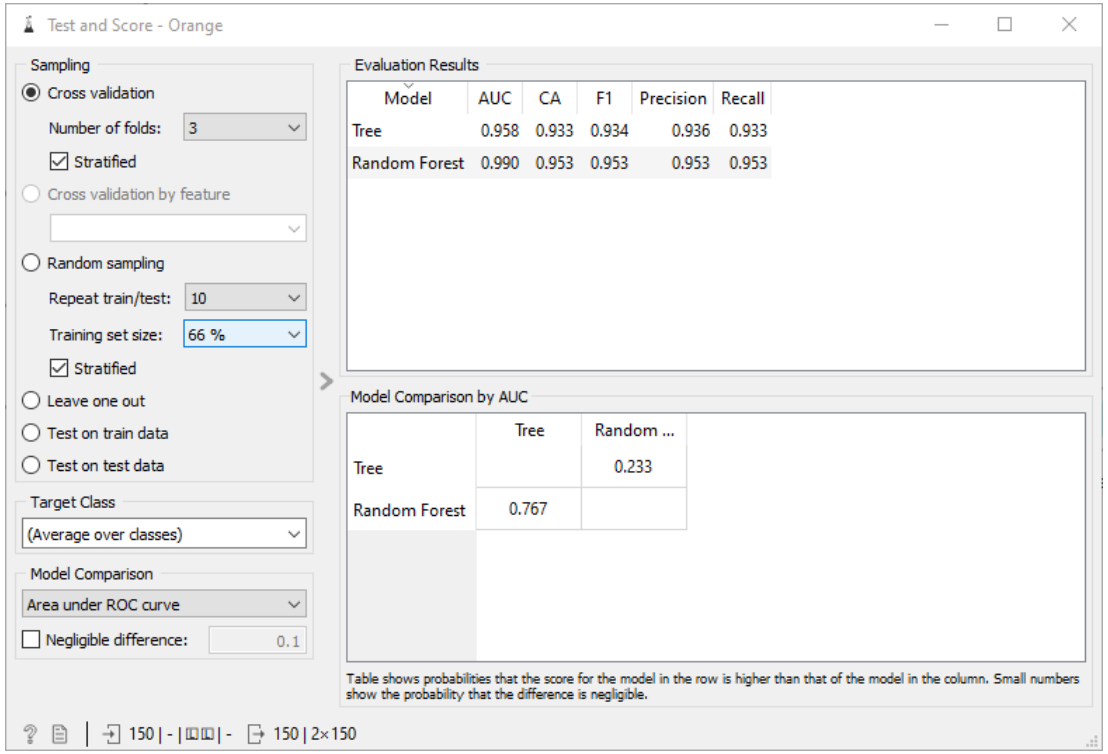


Evaluate two model together?

- Yes: if you connect two different models to the same evaluation module, they will be compared immediately!

- This makes it easy to choose the best model possible (and you can do it with two simple clicks)!





Evaluate two model together?

• Finally, we can print the confusion matrix on the evaluation, to get a complete picture of the performance of our models!

Confusion Matrix (1)

Confusion Matrix (1) - Orange

Learners

Tree

Random Forest

Predicted

Actual	Iris-setosa	Iris-versicolor	Iris-virginica	Σ
Iris-setosa	100.0 %	3.7 %	0.0 %	50
Iris-versicolor	0.0 %	87.0 %	6.2 %	50
Iris-virginica	0.0 %	9.3 %	93.8 %	50
Σ	48	54	48	150

Confusion Matrix (1) - Orange

Learners

Tree

Random Forest

Predicted

Actual	Iris-setosa	Iris-versicolor	Iris-virginica	Σ
Iris-setosa	100.0 %	0.0 %	0.0 %	50
Iris-versicolor	0.0 %	92.2 %	6.1 %	50
Iris-virginica	0.0 %	7.8 %	93.9 %	50
Σ	50	51	49	150

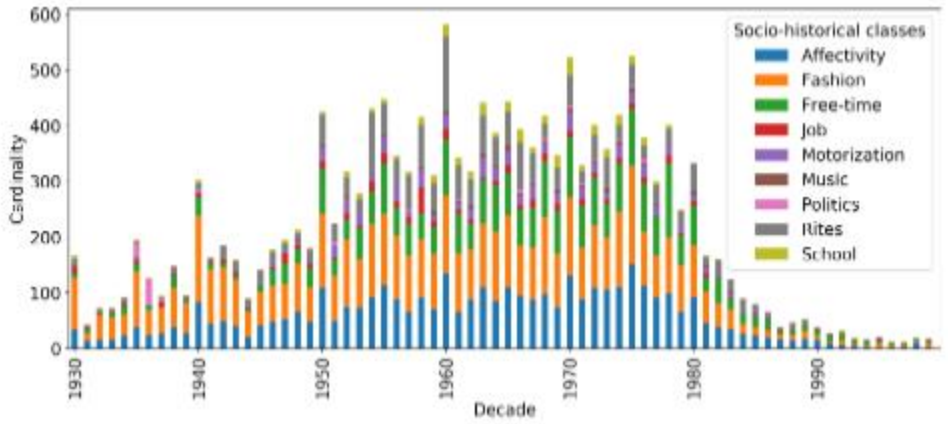
IMAGO analysis con Orange

Lorenzo Stacchio

PhD student in Computer Science

Department for Life Quality Studies

The IMAGO project



(a) Classes distribution

(b) Sample images

- The IMAGO project was started in 2004 by social historians to study the evolution of Social History through the lenses of photographs from family albums;
- This collection today consists of both analogue and digital photos and is collected year by year and kept by the Department of Arts of the University of Bologna in collaboration with Professor Daniela Calanca;
- The collection includes about 80,000 photos, taken between 1845 and 2009, belonging to about 1,500 Italian families;

An important dataset, also from a Machine Learning point of view

- We have published several works with the theme imago, which mainly concern the Machine Learning community;

- Lorenzo Stacchio, Alessia Angeli, Giuseppe Lisanti, Daniela Calanca, and Gustavo Marfia. 2021. Towards a holistic approach to the socio-historical analysis of vernacular photos. ACM Trans. Multimedia Comput. Commun. Appl (December 2021);

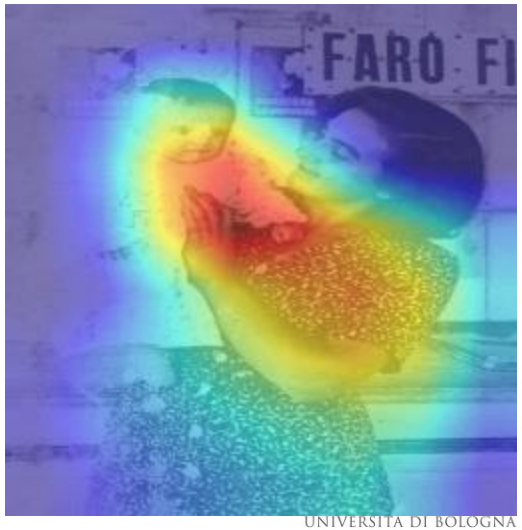
- L. Stacchio, A. Angeli, S. Hajahmadi, G. Marfia: Revive Family Photo Albums through a Collaborative Environment Exploiting the HoloLens 2. In Proceedings of the 2021 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct);

- L. Stacchio, S. Hajahmadi and G. Marfia, "Preserving Family Album Photos with the HoloLens 2," 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), 2021, pp. 643-644, doi: 10.1109/VRW52623.2021.00204.;

Towards a holistic approach to the socio-historical analysis of vernacular photos

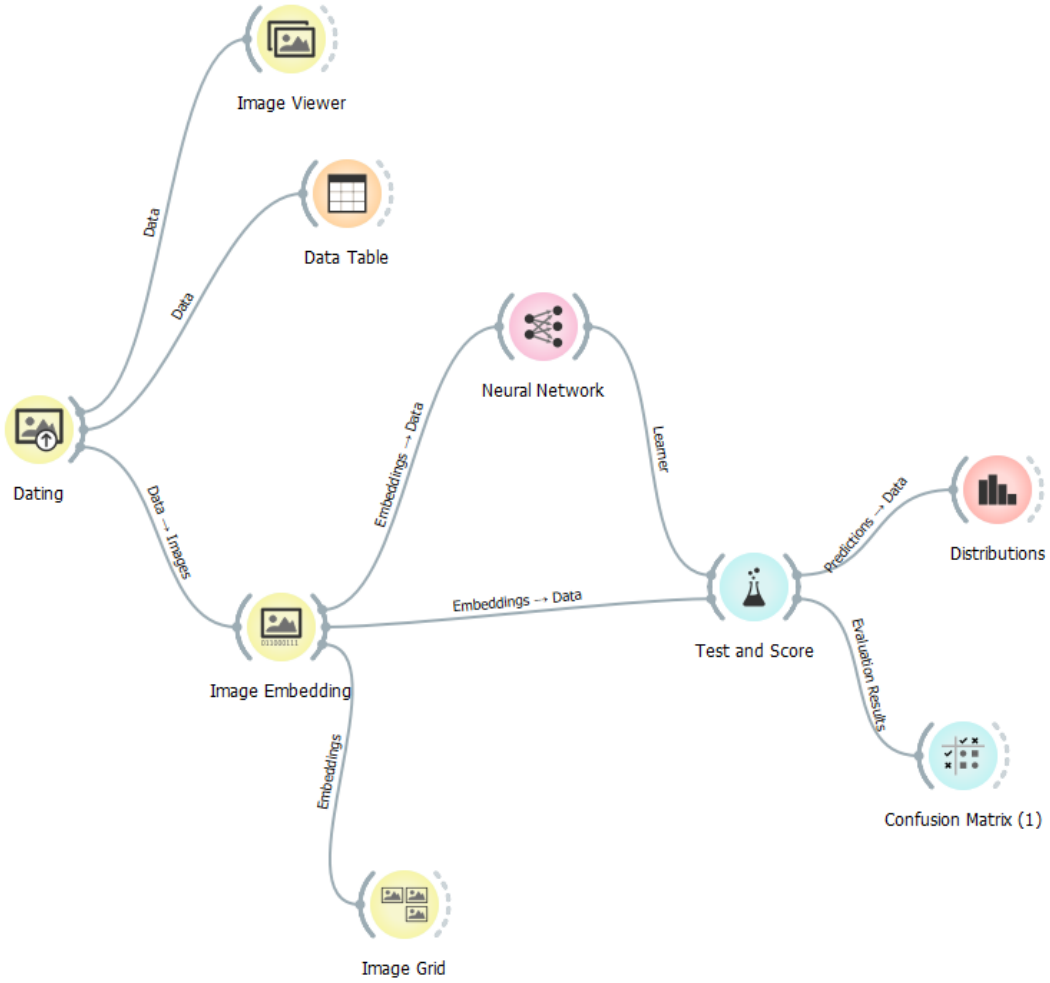
- Develop deep learning models to catalog family album photos based on the shooting date and historical-social context;

- The results of our work showed that deep learning (CNN) models are able to focus on interesting historical-social characteristics when classifying a photo;



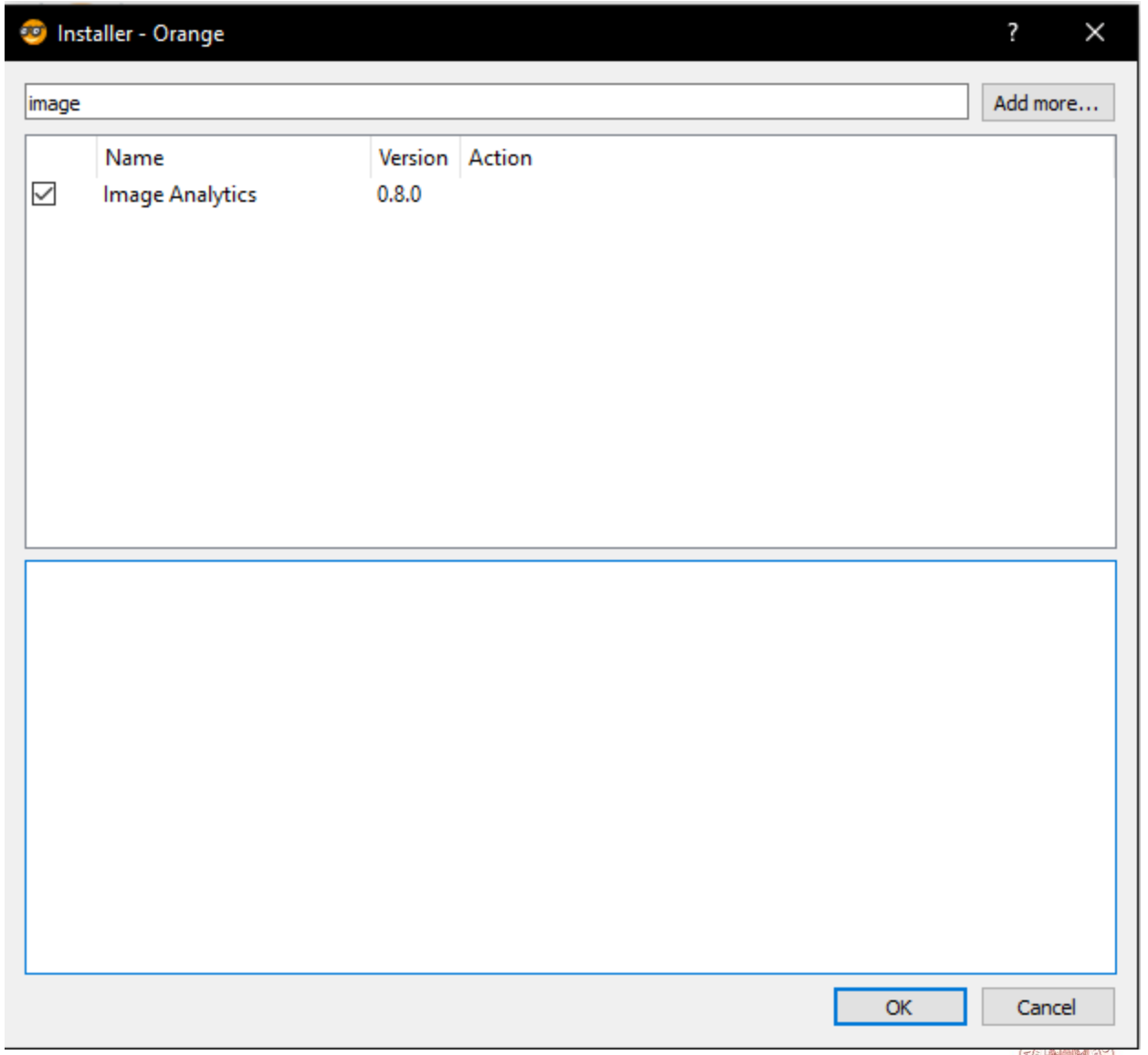
Let's analyze IMAGO with Orange!

- This is the workflow we will create to classify and analyze the IMAGO dataset;
- But first, you will have to download a small subset that I have prepared for you, so that your PCs are also able to do the analyzes!
- The subset of the IMAGO dataset can be downloaded from the [following link](#);



Preliminary step

- Install the add-on for images from:
    - Options;
    - Add-ons;
    - Cercate Image Analytics;
    - Selezionatela;
    - Click ok;

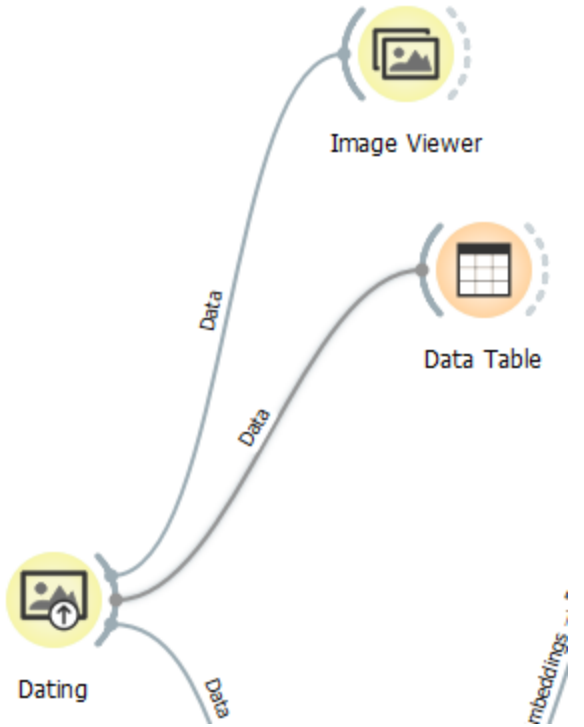


Let's start with dating: importing a folder of images

- Create an “Import Image” widget;
- Click on it and select the "dating" folder, the location depends on your PC (if you have just downloaded it, it is in the Download folder);

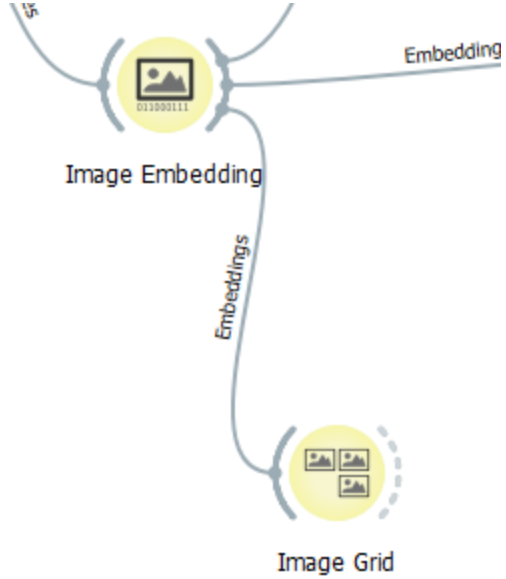
Let's start with dating let's explore the data

- We can create two different widgets, to explore the images from two different points of view:
  - Image viewer will allow you to explore images by viewing them;
- Data Table will convert your images into table records so that you can explore them textually!

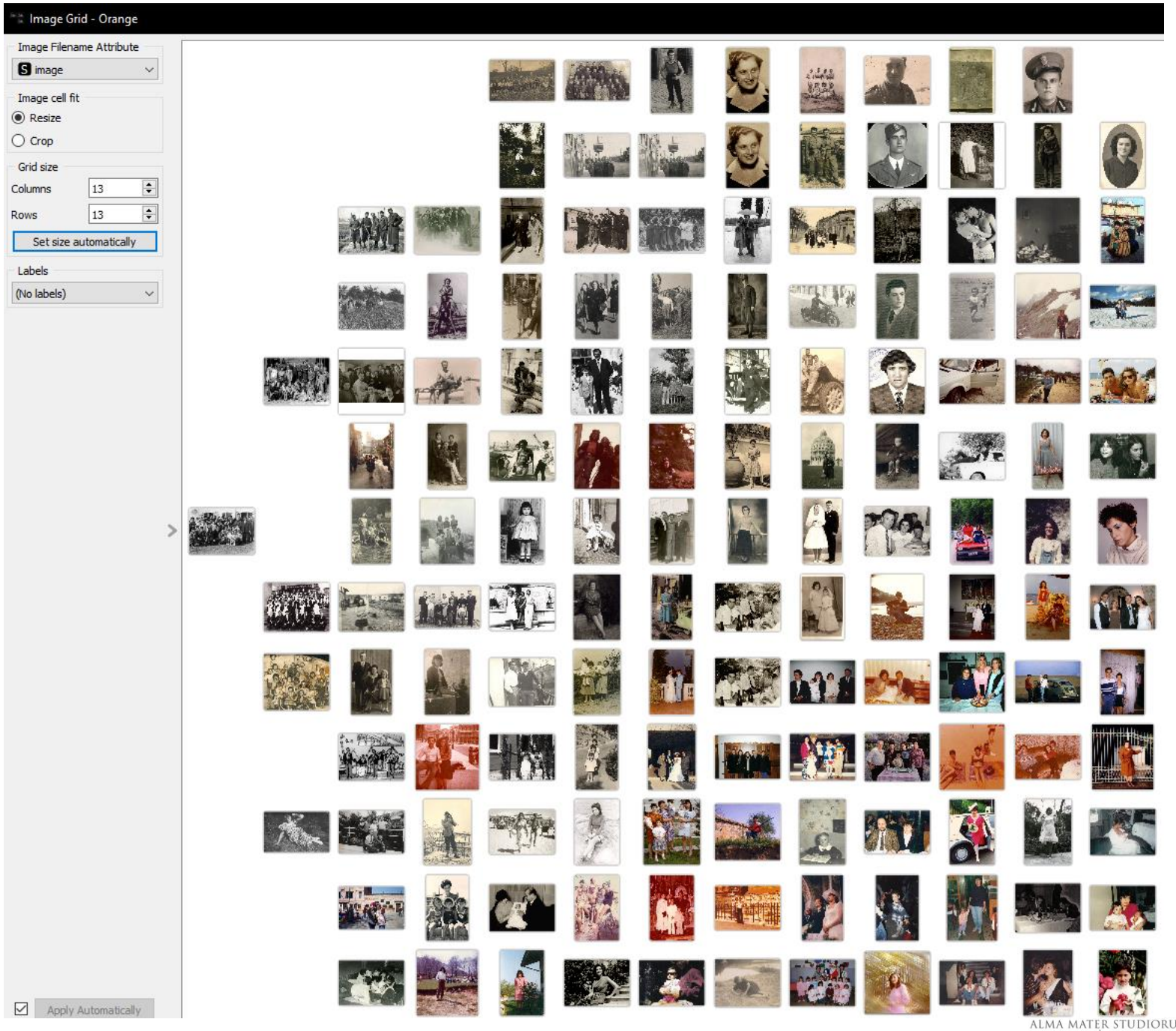


Let's start with dating: image embeddings

- Let's create an “Image Embedding” widget;
- This widget takes advantage of pre-trained CNNs to create good image representations (remember?)!
- With that done, create an “Image Grid” widget and enjoy exploring the magic of deep learning!

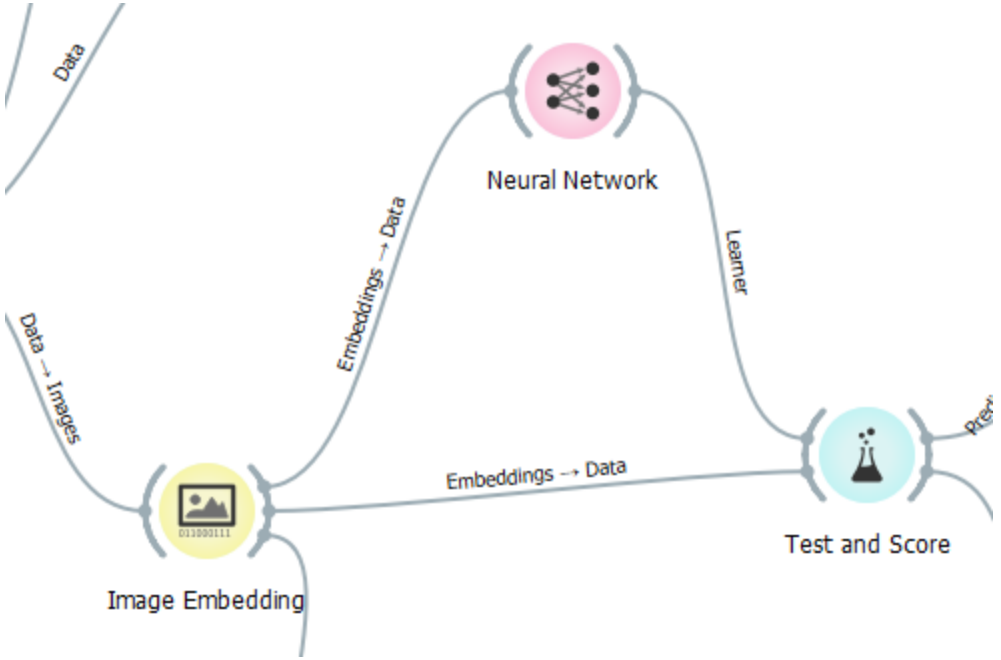


The Image grid widget exploits Deep learning extrapolated features to group similar images!

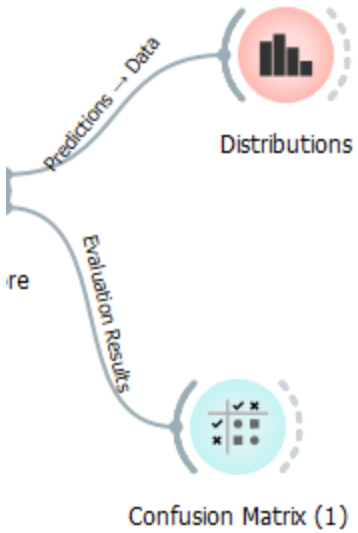


Let's start with dating: train a neural network to classify our images by decade!

- Let's create a “Neural Network” widget;
- This Widget will create a real and simple Multi-feed forward neural network that will learn from Image Embeddings!
- You can then evaluate its performance with test and score widget!

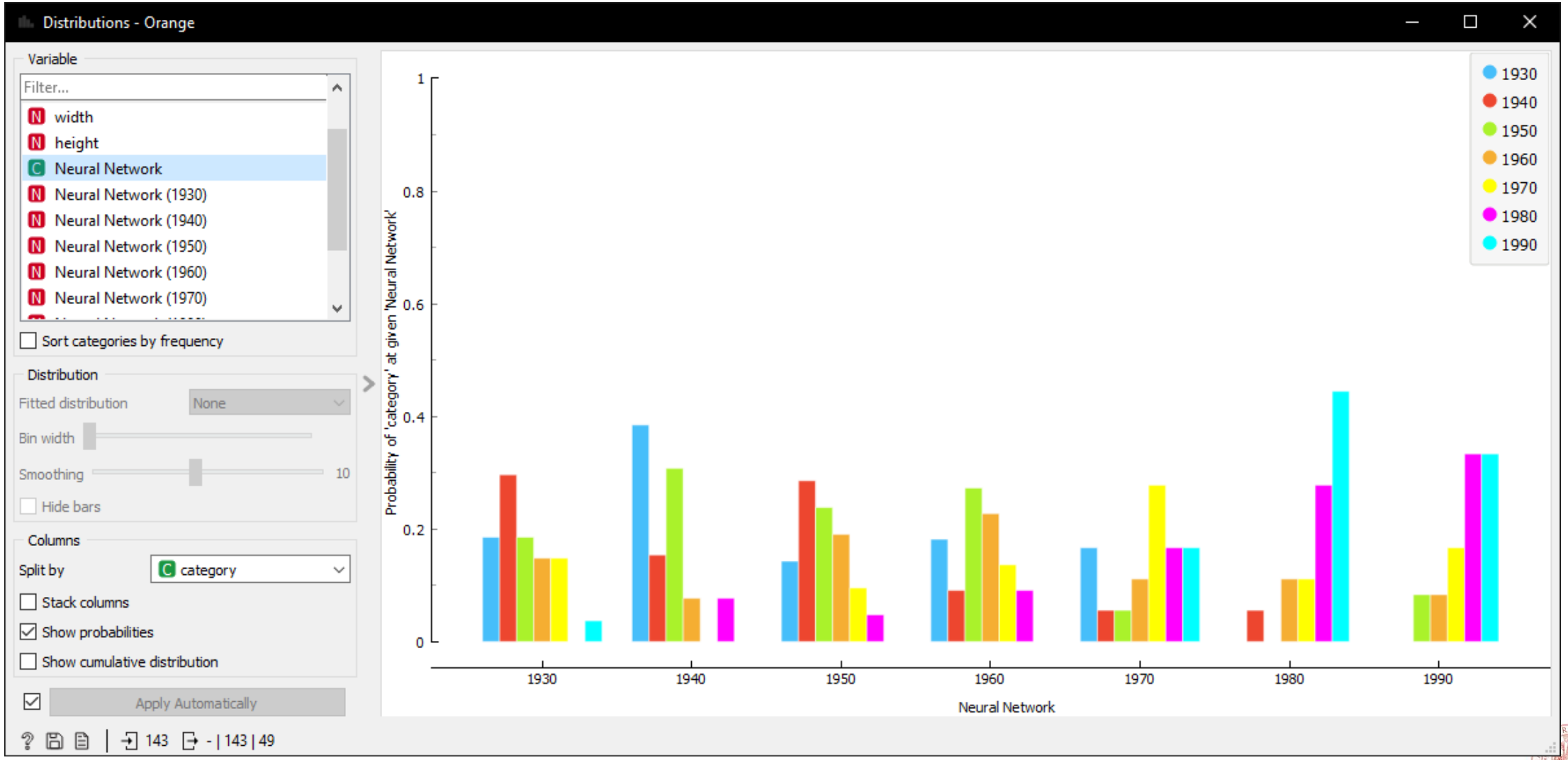


Let's check the results also with the distribution widget

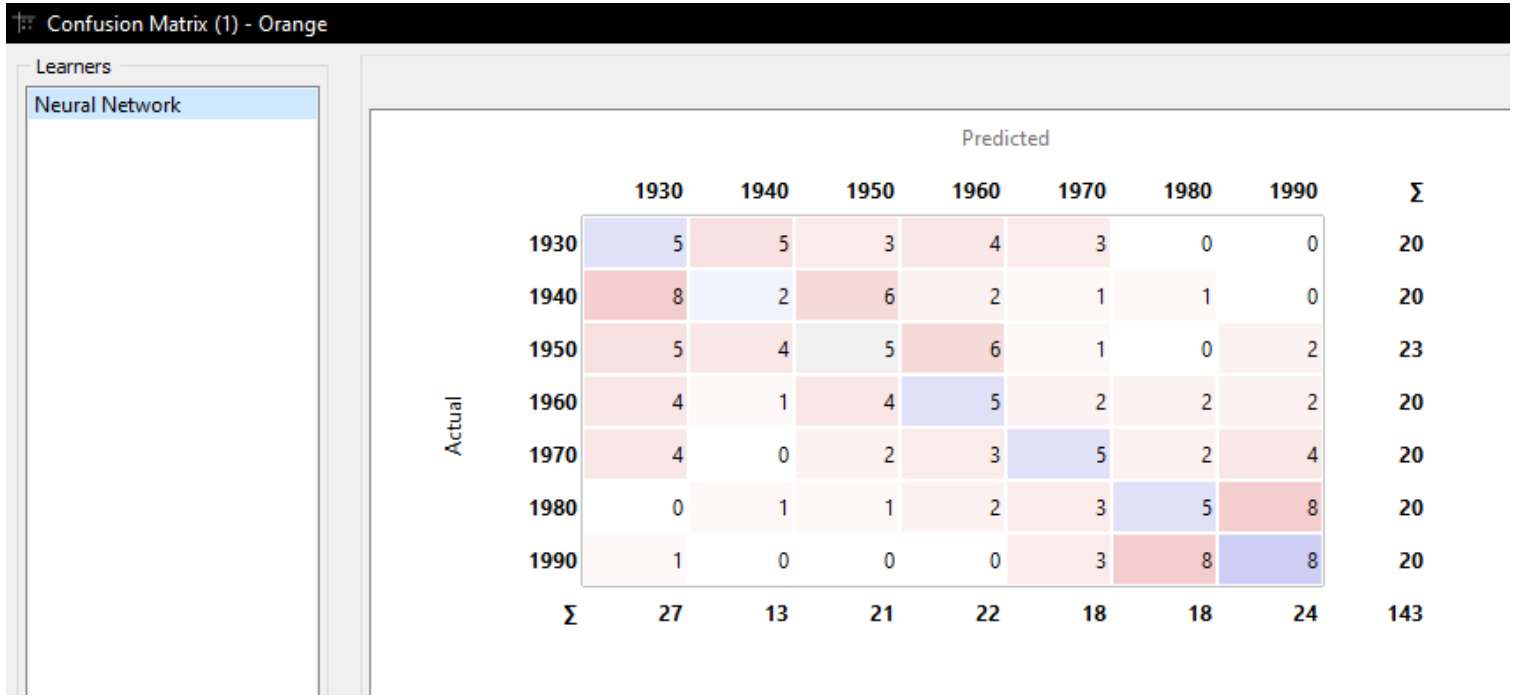


- The distribution widget allows to visualize the distribution of a certain variable of your data;
- It could be used after a Test and score widget, to visualize the distributions of the models predictions!

The distribution widget



Confusion matrix



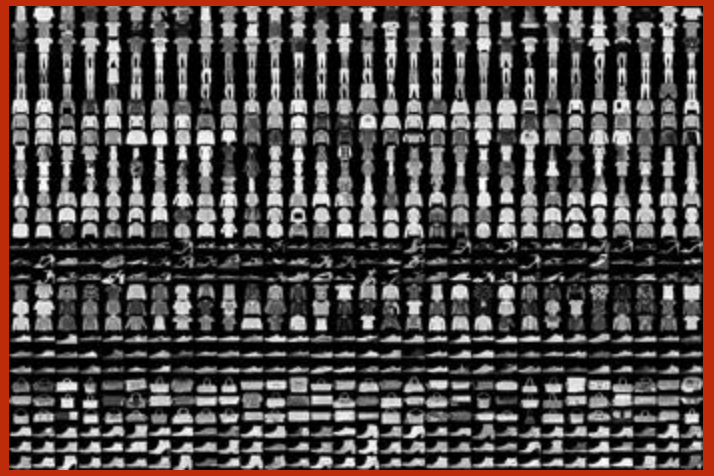
Actual \ Predicted	1930	1940	1950	1960	1970	1980	1990	Σ
1930	5	5	3	4	3	0	0	20
1940	8	2	6	2	1	1	0	20
1950	5	4	5	6	1	0	2	23
1960	4	1	4	5	2	2	2	20
1970	4	0	2	3	5	2	4	20
1980	0	1	1	2	3	5	8	20
1990	1	0	0	0	3	8	8	20
Σ	27	13	21	22	18	18	24	143

We can use the same workflow for the socio-historical dataset!

- Reusing Orange widget schema is a powerful thing;
- By clicking on the "Import Images" widget and changing the folder, selecting the "socio-historical context" and you already have done all you need to analyze the socio-historical dataset, with the same schema of the dating one!
- Although the task has changed, we had everything ready, and we didn't have to do anything!

How long would it take us to code all this in python?

- At least two hours of coding;
    - Upload the dataset;
    - Instantiate a pre-trained Deep Learning model;
    - Create a function to generate the embeddings from it;
    - Define a neural network model;
    - Train it;
    - Create cross-validation metrics;
    - Create a confusion Matrix;
    - Calculate the metrics;
    - Calculate a histogram of the classifications …
- Thanks to this visual approach, not only did we do everything quickly, but you will remember all the workflow, the widgets, and how to compose them!
- The visual approach is essential in learning!



Orange per il fashion: Fashion MNIST

Lorenzo Stacchio

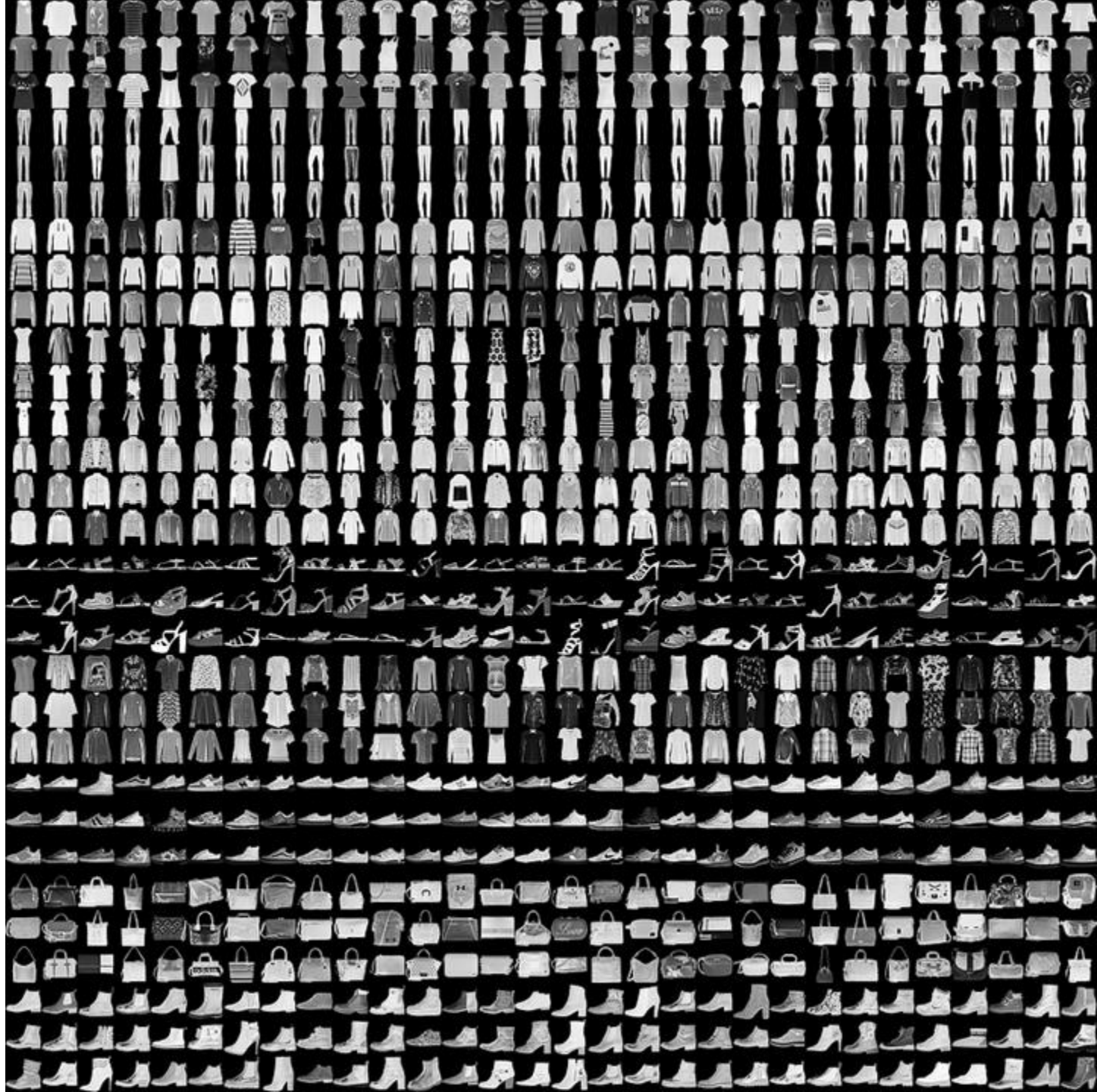
PhD student in Computer Science

Department for Life Quality Studies

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Fashion MNIST

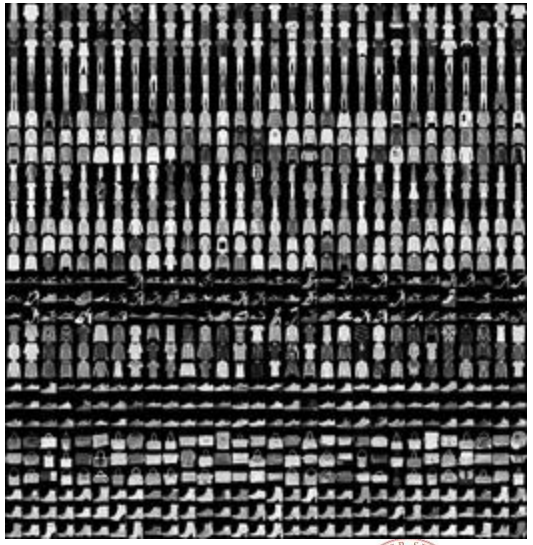
- Fashion-MNIST is a dataset of Zalando's article images consisting of a training set of 60,000 samples and a test set of 10,000 samples.
- Each example is a 28×28 grayscale image, associated with a label from 10 different classes.



Why is this topic of interest for the scientific community?

The original MNIST dataset contains images of handwritten digits. Researchers from Data Science community use this dataset as a benchmark to validate their learning algorithms. In fact, MNIST is often the first dataset they would try on. "If it doesn't work on MNIST, it won't work at all", they said. "Well, if it does work on MNIST, it may still fail on others."

Fashion MNIST is intended to serve as a direct drop-in replacement for the original MNIST dataset to benchmark machine and deep learning algorithms, as it shares the same image size and the structure of training and testing splits.



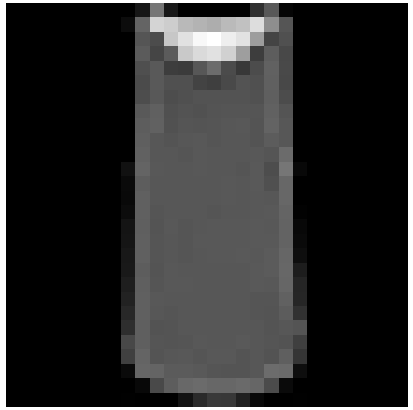
Researchers are talking about replacing MNIST. The following are some good reasons:

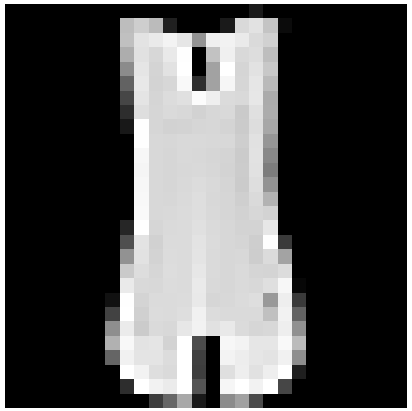
- MNIST is too easy;
- MNIST is overused ;
- MNIST can not represent modern CV tasks ;

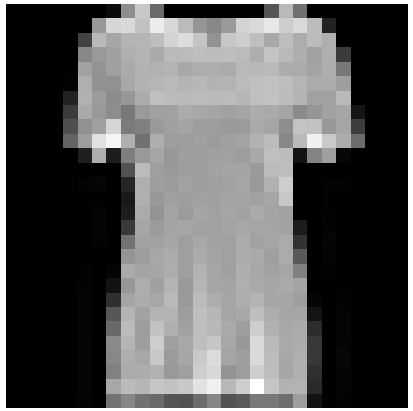
The 10 different classes
of Fashion MNIST dataset

0: T-shirt/Top
1: Trouser
2: Pullover
3: Dress
4: Coat
5: Sandal
6: Shirt
7: Sneaker
8: Bag
9: Ankle Boot

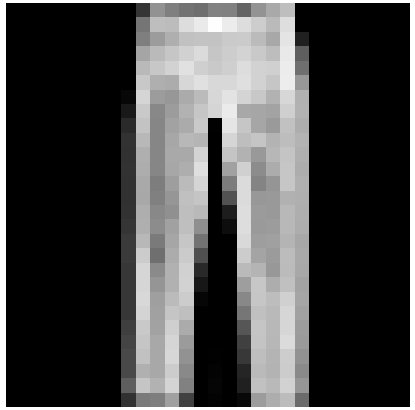
0: T-shirt/Top

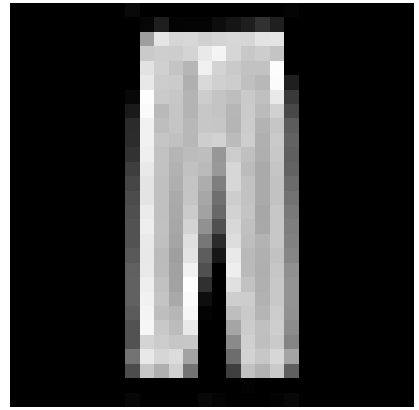


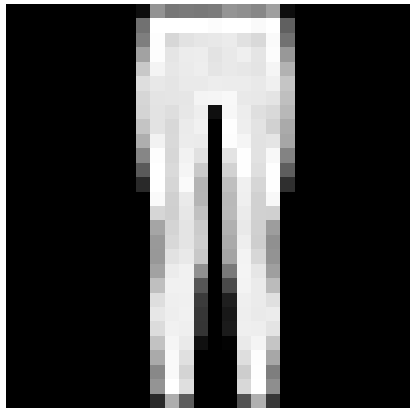


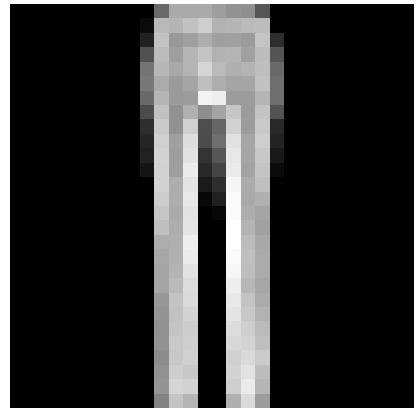


1: Trouser

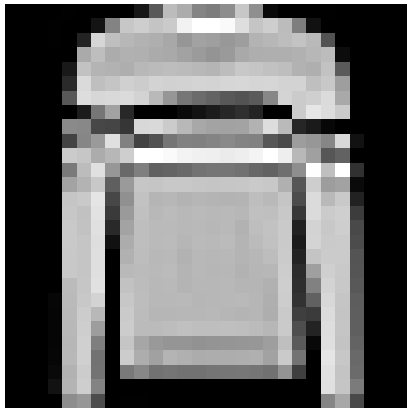
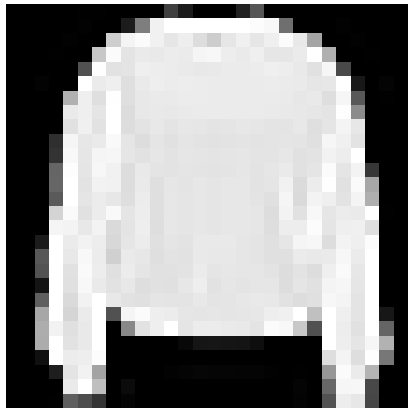
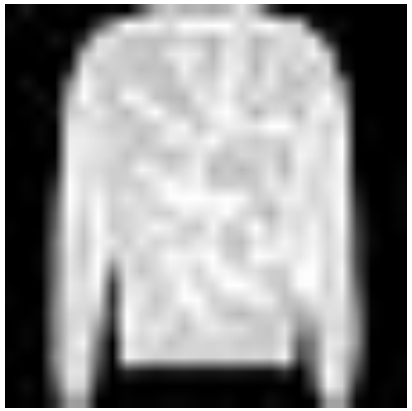
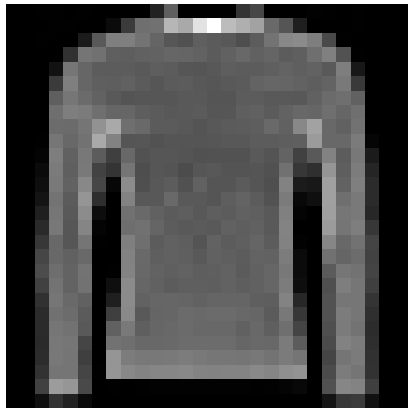




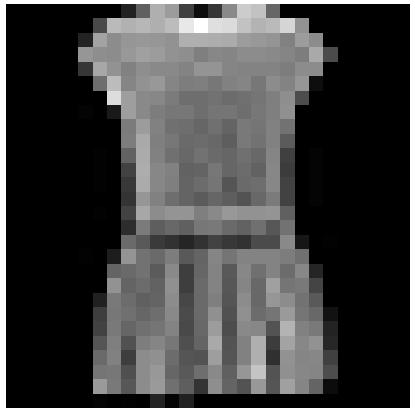
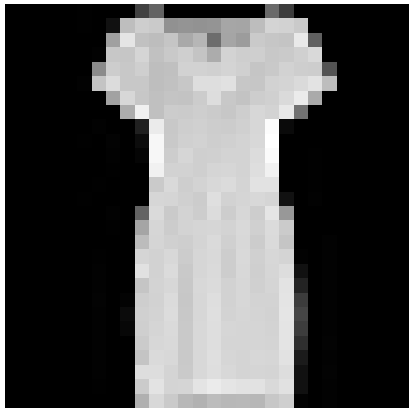
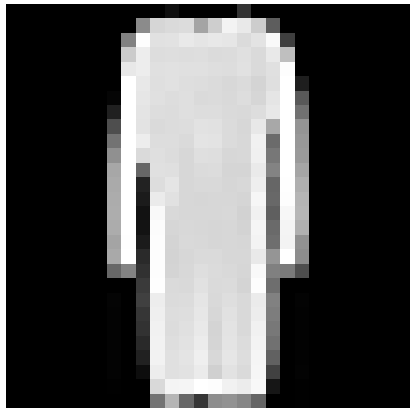
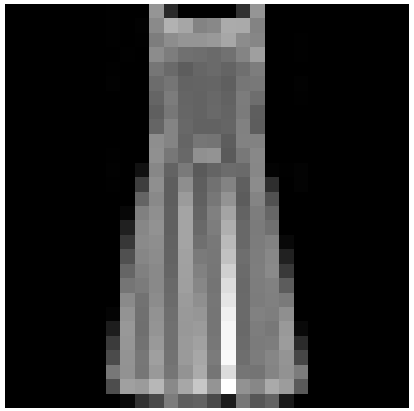




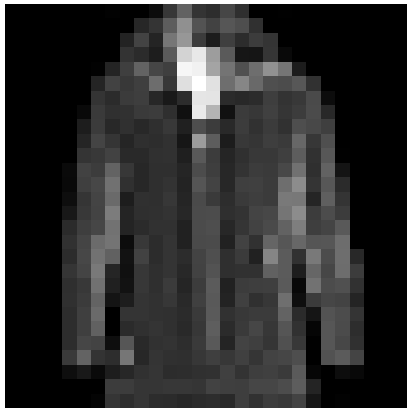
2: Pullover

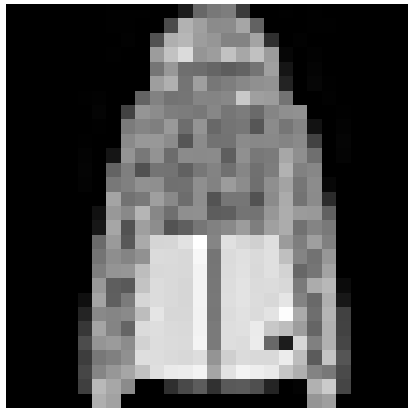





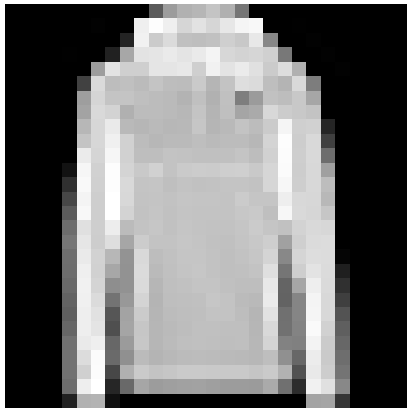
3: Dress

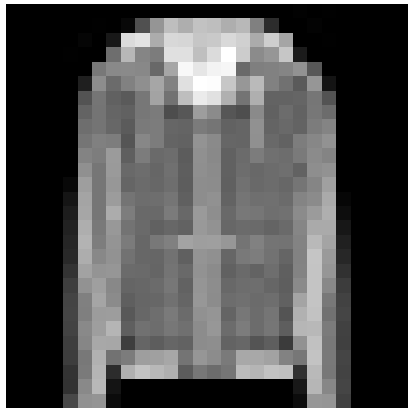





4: Coat

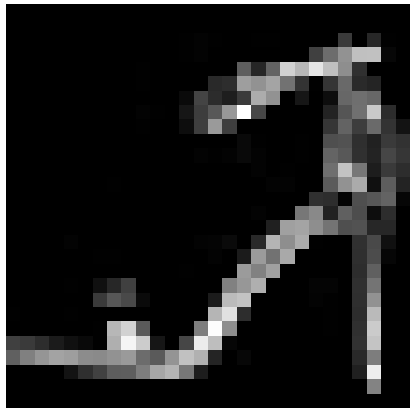


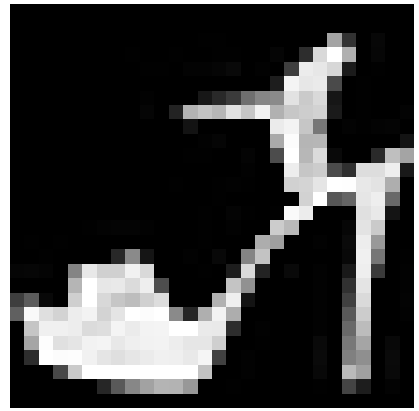


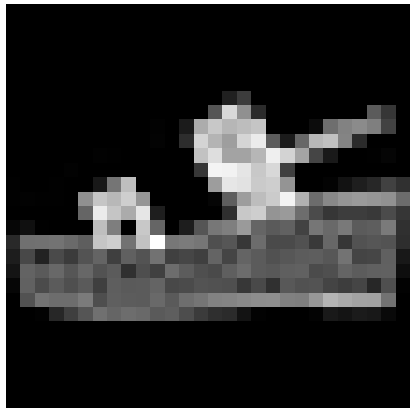


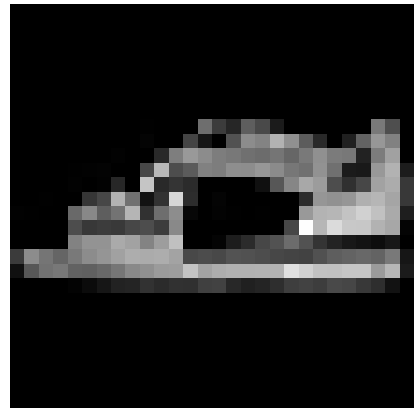


5: Sandal

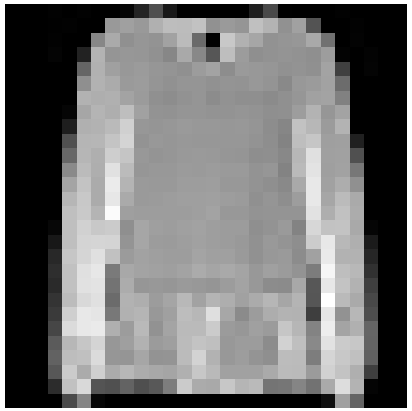
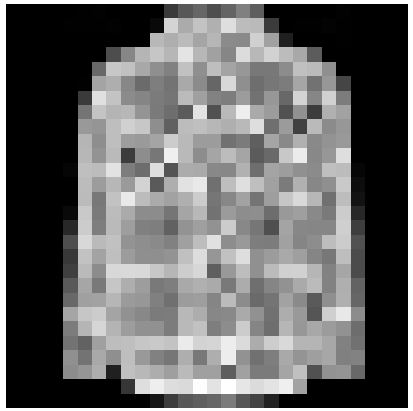
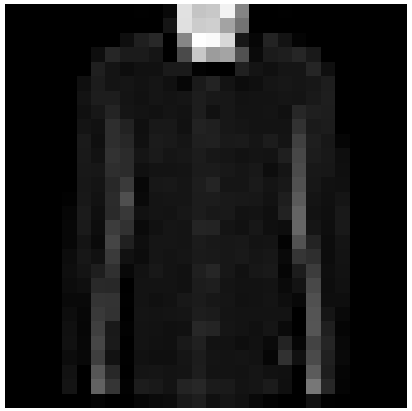
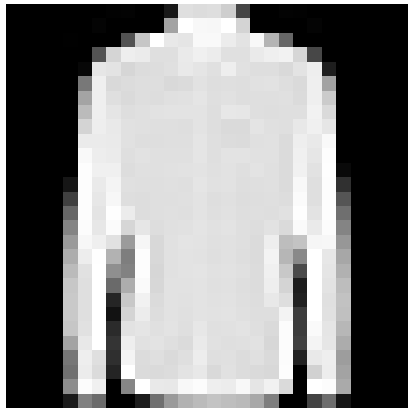




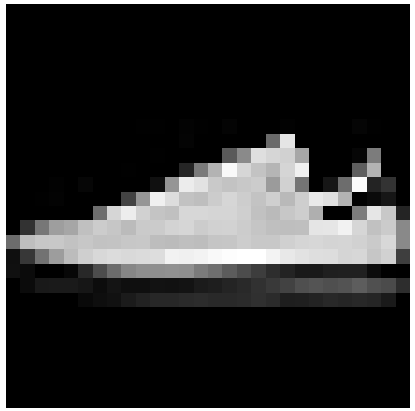
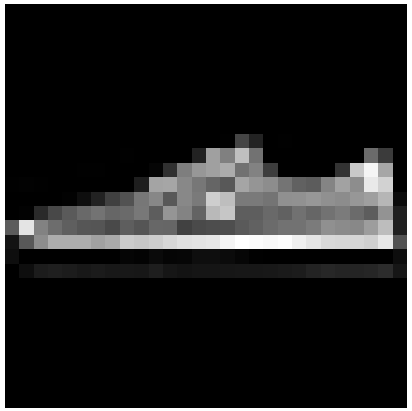
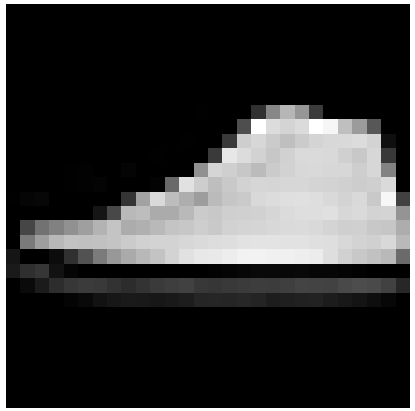
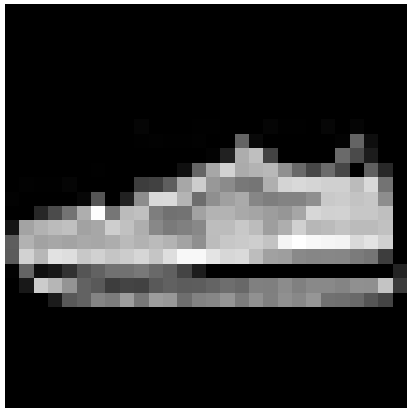




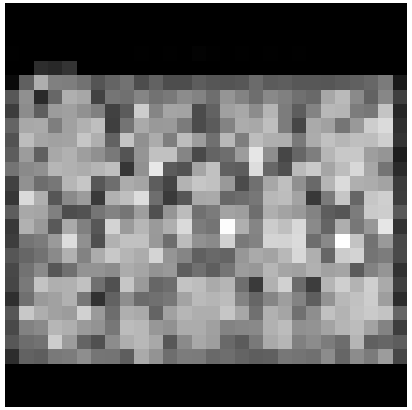
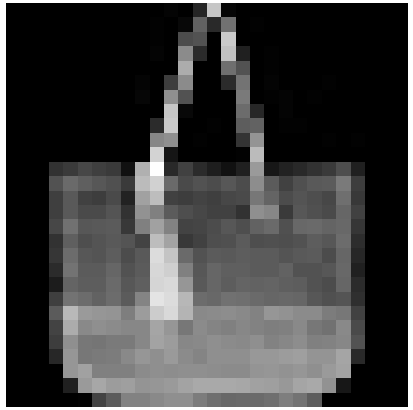
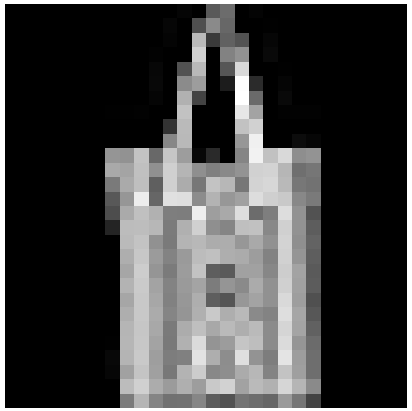
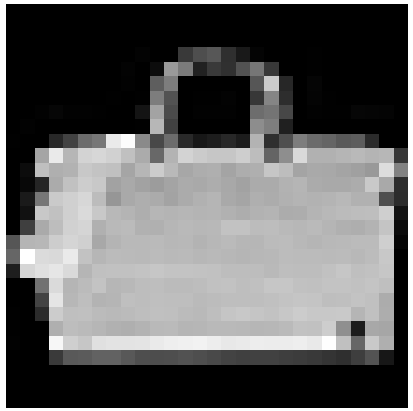
6: Shirt

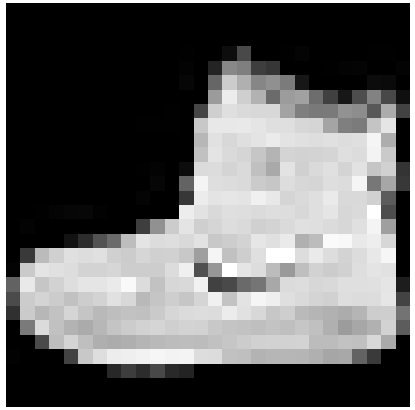
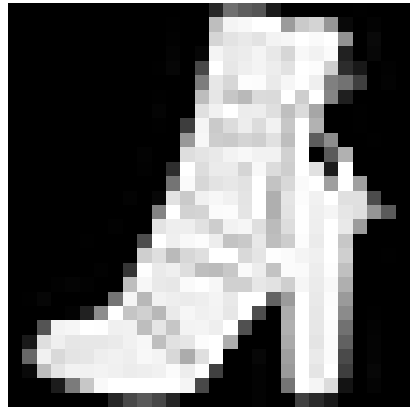
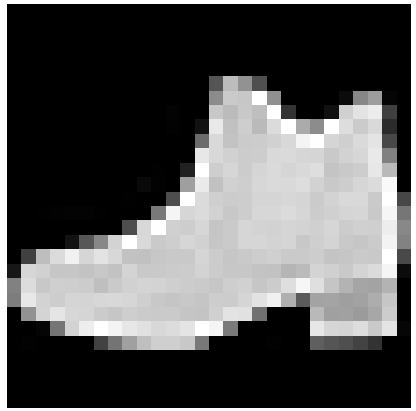
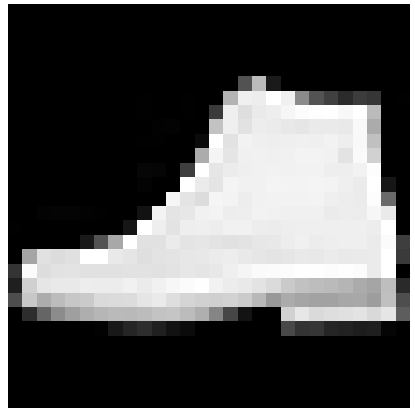
7: Sneaker

8: Bag

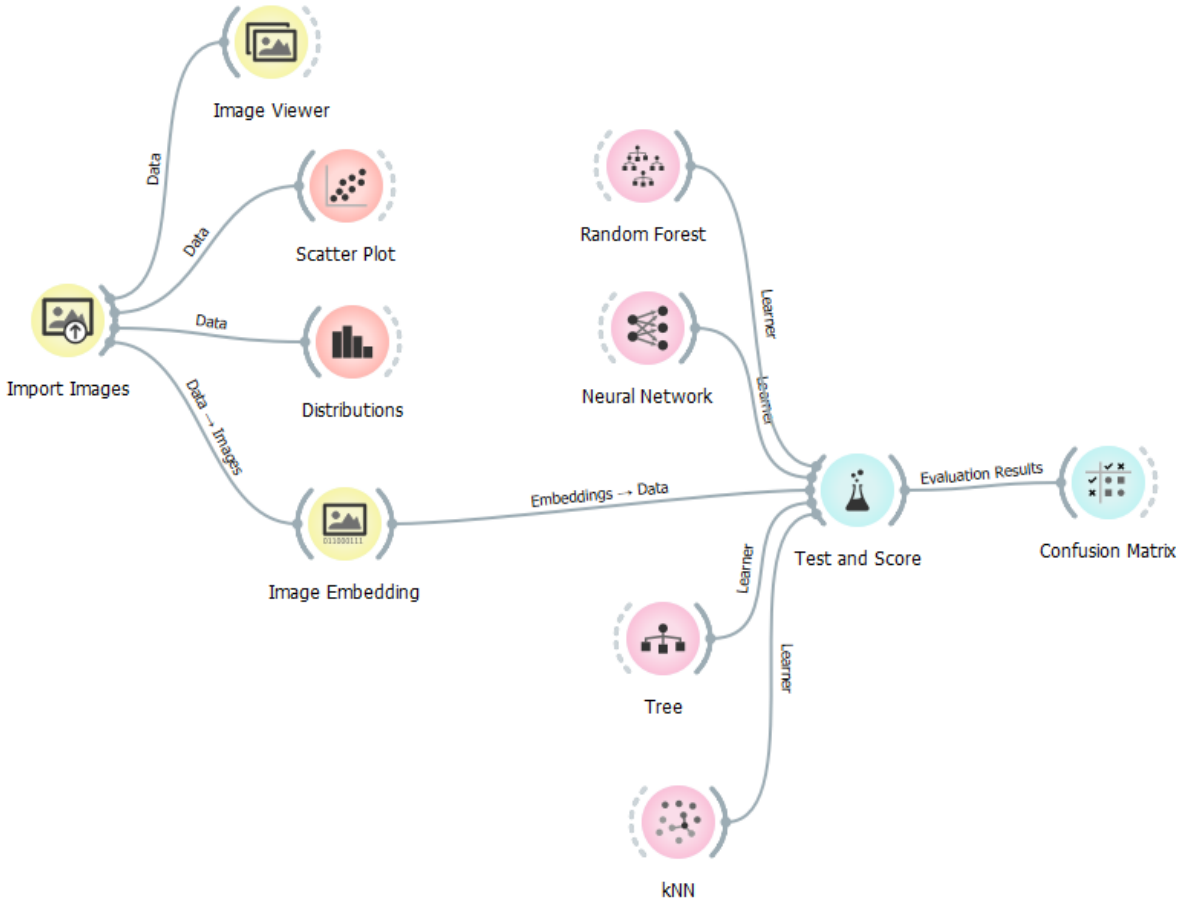





9: Ankle Boot

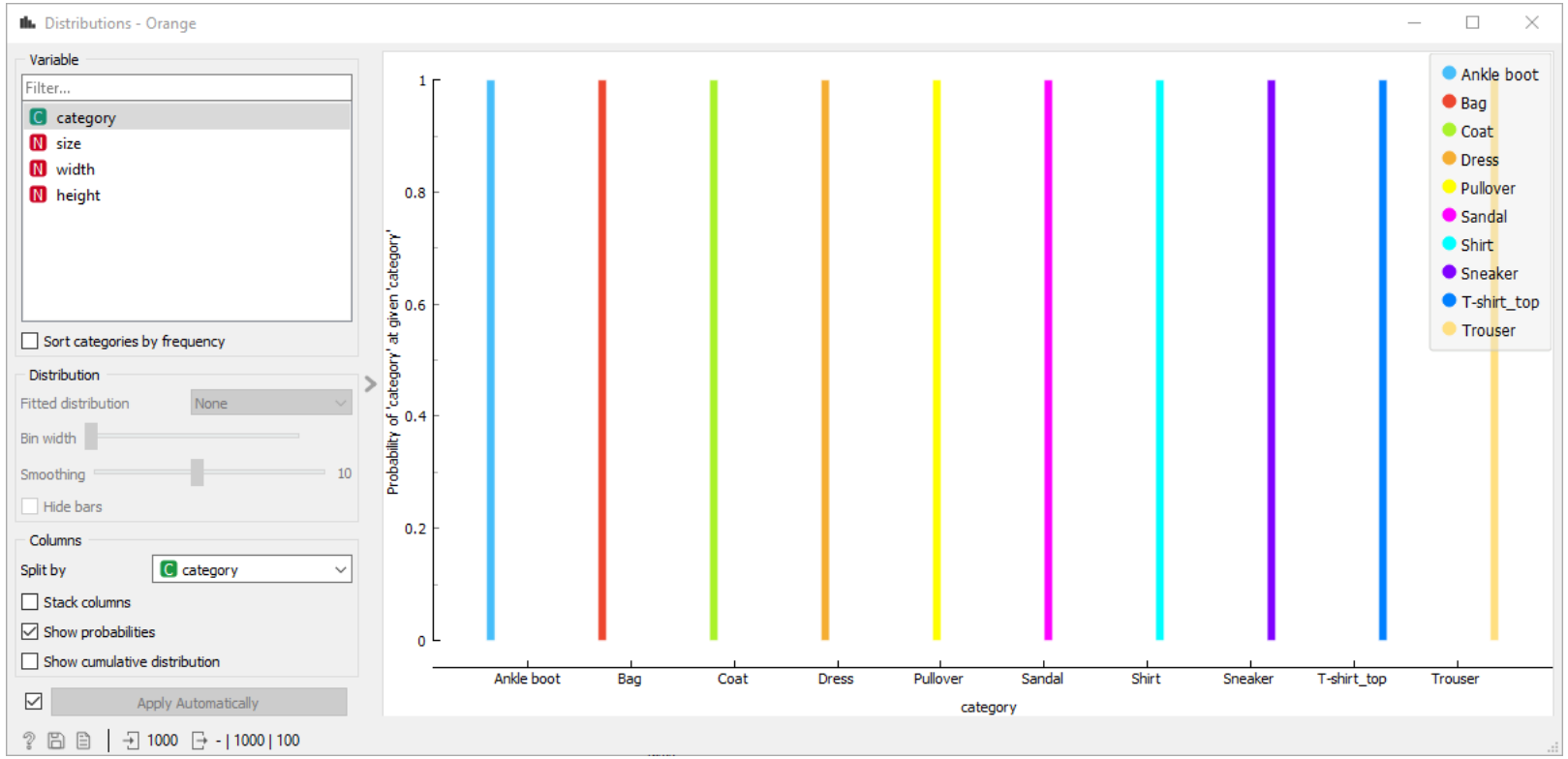





Let's analyze FashionMINST with Orange!

- This is the workflow we will create to classify and analyze the FashionMNIST dataset;
- But first, you will have to download a small subset that I have prepared for you, so that your PCs are also able to do the analyzes!
- The subset of the FashionMNIST dataset can be downloaded from the [following link](#);



Distribution widget

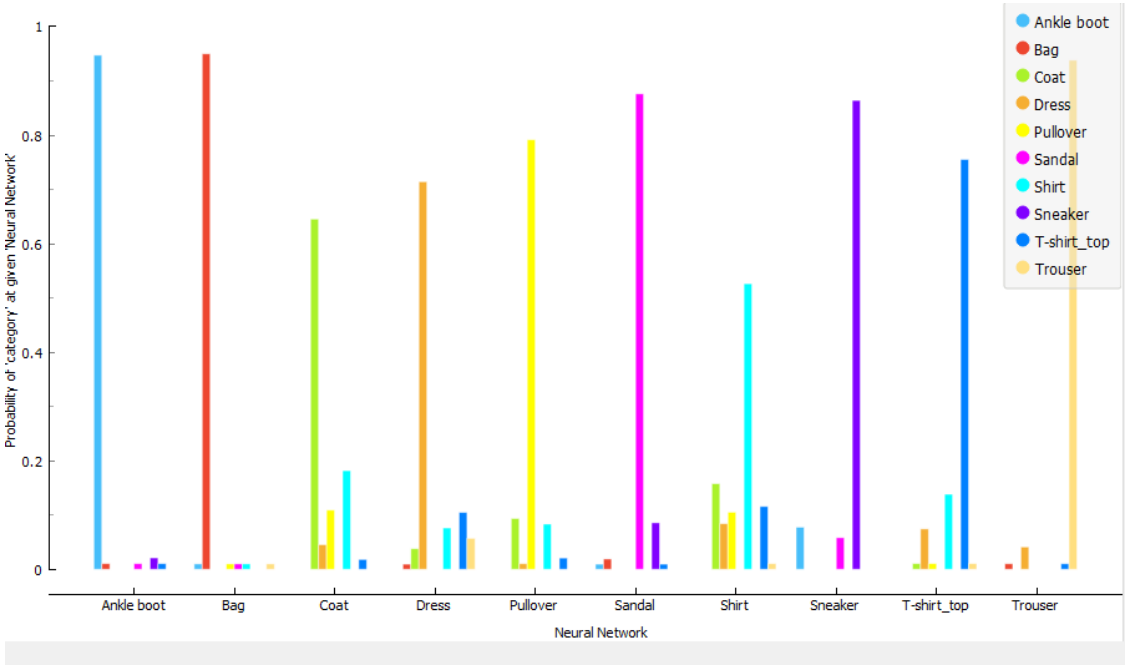


- The distribution widget allows to visualize the distribution of a certain variable of your data;
- We are here considering labelled images, so we can visualize the distribution of the fashion item categories, and the size (memory occupied in bytes), width, height of the images.
- Those values would be all splitted by the target variable: i.e., the category!

What kind of results we have obtained on FashionMNIST?

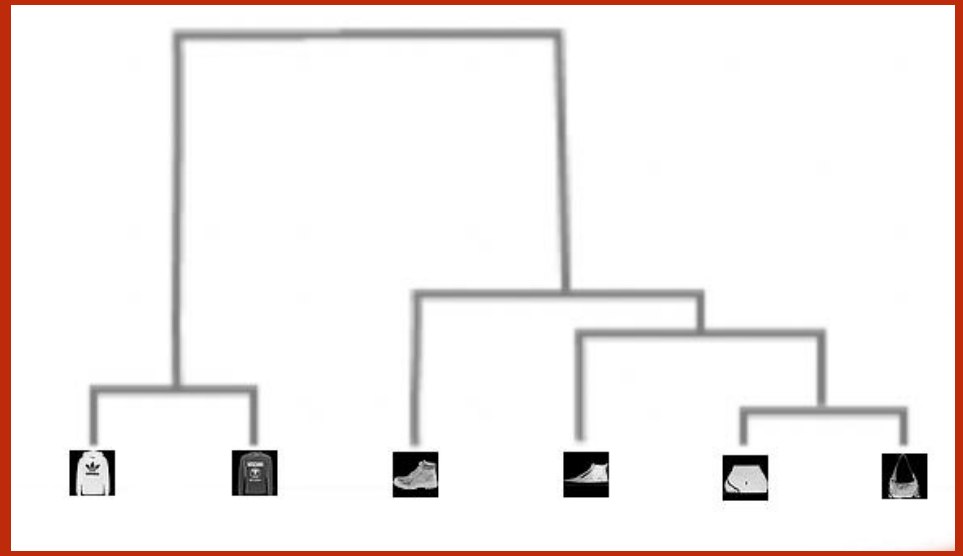
Evaluation Results

Model	AUC	CA	F1	Precision	Recall
kNN	0.940	0.768	0.767	0.777	0.768
Tree	0.776	0.573	0.573	0.574	0.573
Random Forest	0.925	0.688	0.686	0.685	0.688
Neural Network	0.975	0.800	0.800	0.801	0.800



Predicted

Actual	Ankle boot	Bag	Coat	Dress	Pullover	Sandal	Shirt	Sneaker	T-shirt_top	Trouser	Σ
Ankle boot	94.7 %	1.0 %	0.0 %	0.0 %	0.0 %	1.0 %	0.0 %	7.8 %	0.0 %	0.0 %	100
Bag	1.1 %	95.0 %	0.0 %	1.0 %	0.0 %	1.9 %	0.0 %	0.0 %	0.0 %	1.0 %	100
Coat	0.0 %	0.0 %	64.5 %	3.8 %	9.4 %	0.0 %	15.8 %	0.0 %	1.1 %	0.0 %	100
Dress	0.0 %	0.0 %	4.5 %	71.4 %	1.0 %	0.0 %	8.4 %	0.0 %	7.4 %	4.1 %	100
Pullover	0.0 %	1.0 %	10.9 %	0.0 %	79.2 %	0.0 %	10.5 %	0.0 %	1.1 %	0.0 %	100
Sandal	1.1 %	1.0 %	0.0 %	0.0 %	0.0 %	87.6 %	0.0 %	5.8 %	0.0 %	0.0 %	100
Shirt	0.0 %	1.0 %	18.2 %	7.6 %	8.3 %	0.0 %	52.6 %	0.0 %	13.8 %	0.0 %	100
Sneaker	2.1 %	0.0 %	0.0 %	0.0 %	0.0 %	8.6 %	0.0 %	86.4 %	0.0 %	0.0 %	100
T-shirt_top	1.1 %	0.0 %	1.8 %	10.5 %	2.1 %	1.0 %	11.6 %	0.0 %	75.5 %	1.0 %	100
Trouser	0.0 %	1.0 %	0.0 %	5.7 %	0.0 %	0.0 %	1.1 %	0.0 %	1.1 %	93.8 %	100
Σ	95	100	110	105	96	105	95	103	94	97	1000



Fashion MNIST and Hierarchical clustering

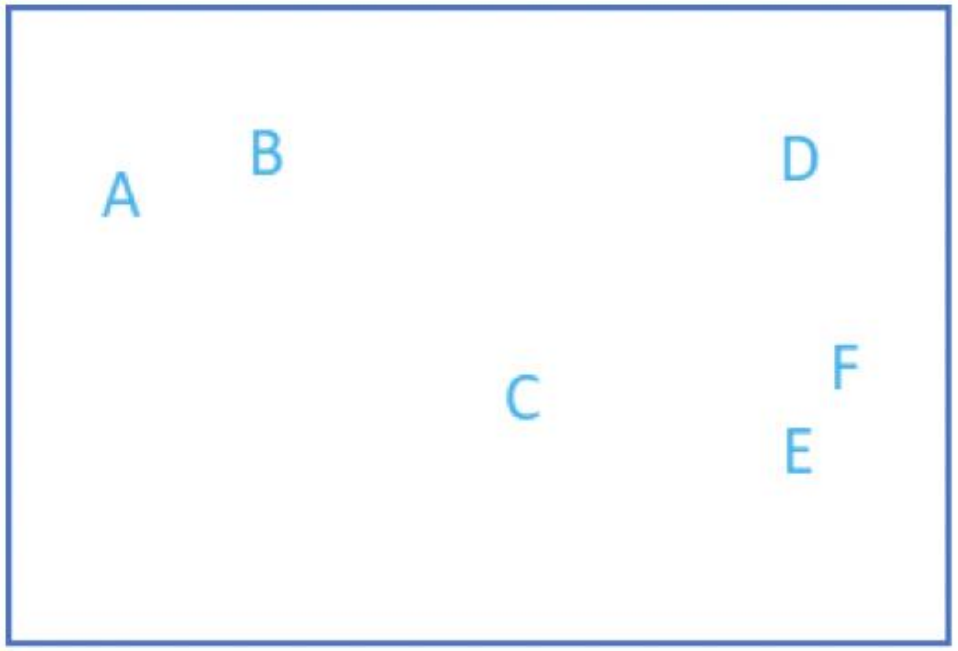
Lorenzo Stacchio

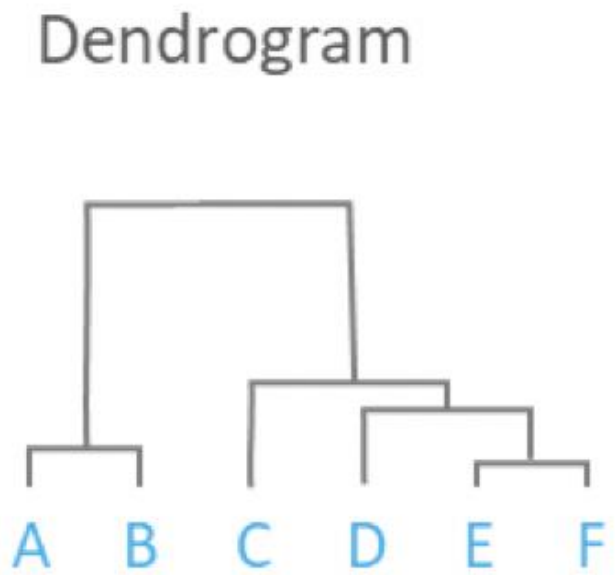
PhD student in Computer Science

Department for Life Quality Studies

ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Unsupervised Machine learning and Hierarchical clustering





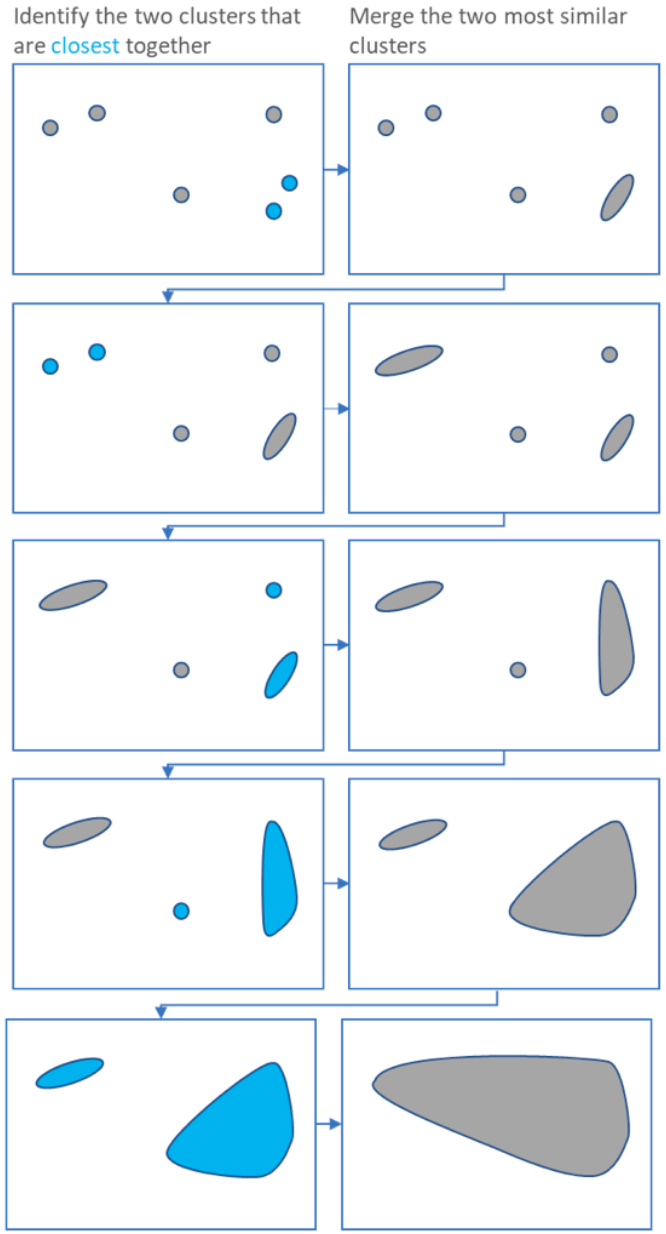
- ***Hierarchical clustering*** is a machine-learning-based algorithm that groups similar objects into groups called *clusters*.

- The endpoint is a set of clusters, where each cluster is distinct from each other cluster, and the objects within each cluster are broadly similar to each other.

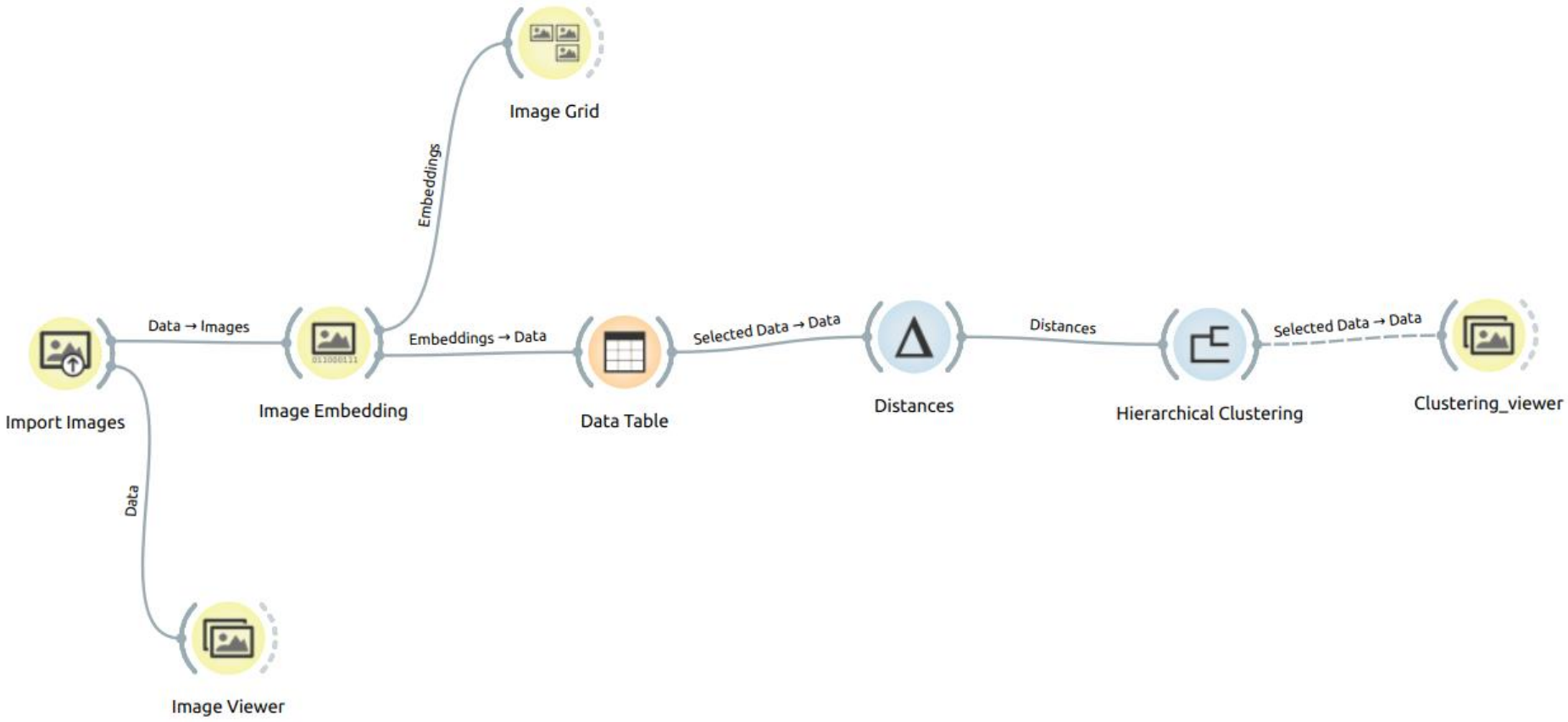
- How similar objects are, can be decided with image features coming from Deep Learning models in Orange and a distance metric!

How Hierarchical clustering works

- Hierarchical clustering starts by treating each observation as a separate cluster.
- Then, it repeatedly executes the following two steps: (1) identify the two clusters that are closest together, and (2) merge the two most similar clusters.
- This iterative process continues until all the clusters are merged together;



Free Sample with Orange and FashionMNIST



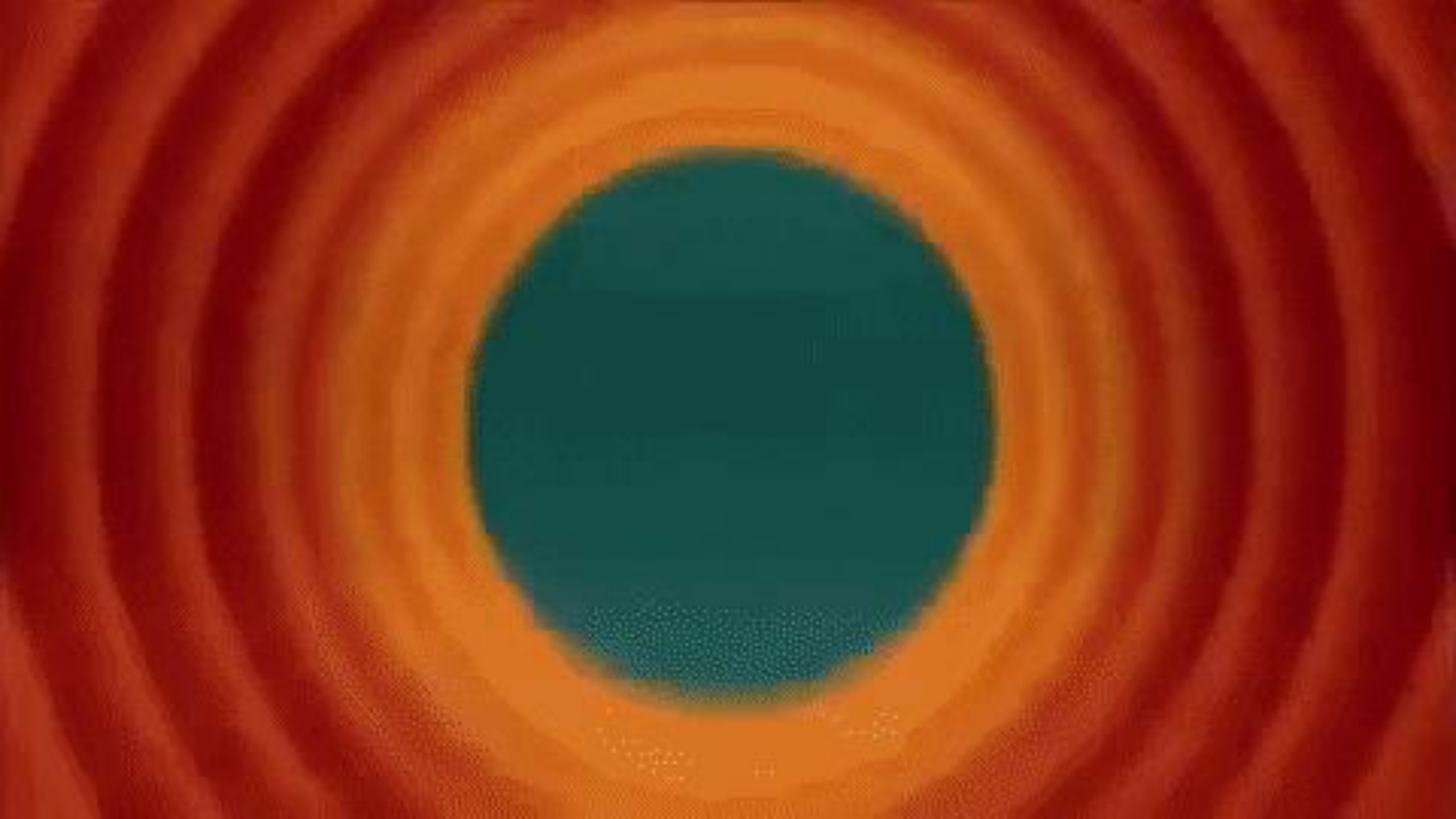
Sources

[1] Lorenzo Stacchio, Alessia Angeli, Giuseppe Lisanti, Daniela Calanca, and Gustavo Marfia. 2021. Towards a holistic approach to the socio-historical analysis of vernacular photos. ACM Trans. Multimedia Comput. Commun. Appl (December 2021);

[2] L. Stacchio, A. Angeli, S. Hajahmadi, G. Marfia: Revive Family Photo Albums through a Collaborative Environment Exploiting the HoloLens 2. In Proceedings of the 2021 IEEE International Symposium on Mixed and Augmented Reality Adjunct (ISMAR-Adjunct);

[3] L. Stacchio, S. Hajahmadi and G. Marfia, "Preserving Family Album Photos with the HoloLens 2," 2021 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW), 2021, pp. 643-644, doi: 10.1109/VRW52623.2021.00204.;

[4] Xiao, H., Rasul, K., & Vollgraf, R. (2017). Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*.



ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

Lorenzo Stacchio

Department for Life Quality Studies

lorenzo.stacchio2@unibo.it