



# Getting started with Orange

# Visual programming

- Open-source tool for data mining
- Built on Python, but designed for non-programmers
- Create analysis workflows with drag-and-drop widgets
- Ideal for teaching, learning, and rapid prototyping
- Supports interactive data exploration and real-time results

# Iris dataset

- One of the most famous datasets in data science and ML
- Contains measurements of 150 iris flowers, across 3 species
- Features:
  - Sepal length
  - Sepal width
  - Petal length
  - Petal width
- Goal: Classify species based on flower measurements
- Great for learning classification, visualization, and model evaluation



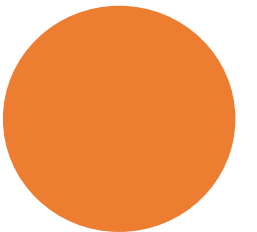
**Iris Versicolor**



**Iris Setosa**



**Iris Virginica**



# Example datasets

- **Iris**

- Collected by Edgar Anderson and famously analyzed by R.A. Fisher in 1936.
- Teaches basic classification, scatter plots, and model evaluation.
- Ideal for understanding how machine learning separates classes.

- **Titanic**

- Based on real passenger data from the 1912 shipwreck.
- Teaches classification, missing data handling, and feature importance.
- Great for illustrating real-world survival prediction.

- **Heart Disease**

- From the Cleveland Heart Disease dataset at the UCI Machine Learning Repository.
- Useful for binary classification and model comparison using medical data.
- Good case for evaluating ROC curves and precision-recall.

- **Housing**

- Originates from 1970s Boston housing data (U.S. Census and housing authorities).
- Used for regression — predicting median home prices.
- A practical introduction to numerical prediction models.

- **Zoo**

- A synthetic dataset describing animals through binary traits.
- Excellent for clustering, classification, and understanding decision trees.
- Often used in education because of its simplicity and logic-based structure.

- **Brown Corpus**

- A historic text dataset from the 1960s, one of the first balanced English corpora.
- Used for text mining and topic modeling.
- Demonstrates how to explore word frequency and document classification.

# Data format

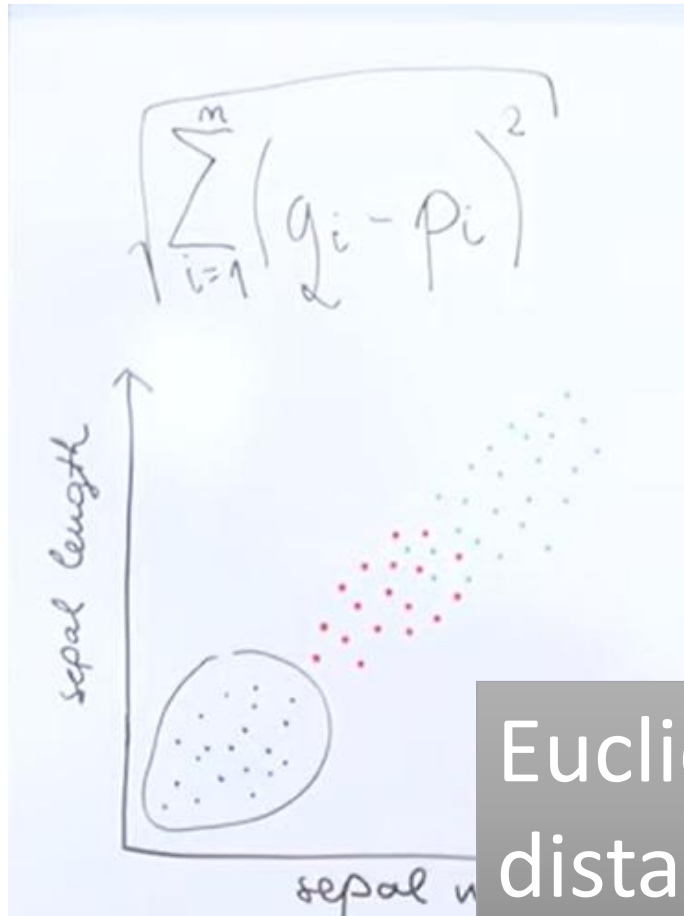
| sepal length | sepal width | petal length | petal width | iris        |
|--------------|-------------|--------------|-------------|-------------|
| c            | c           | c            | c           | d           |
|              |             |              |             | class       |
| 5.1          | 3.5         | 1.4          | 0.2         | Iris-setosa |
| 4.9          | 3.0         | 1.4          | 0.2         | Iris-setosa |
| 4.7          | 3.2         | 1.3          | 0.2         | Iris-setosa |
| 4.6          | 3.1         | 1.5          | 0.2         | Iris-setosa |
| 5.0          | 3.6         | 1.4          | 0.2         | Iris-setosa |
| 5.4          | 3.9         | 1.7          | 0.4         | Iris-setosa |
| 4.6          | 3.4         | 1.4          | 0.3         | Iris-setosa |
| 5.0          | 3.4         | 1.5          | 0.2         | Iris-setosa |
| 4.4          | 2.9         | 1.4          | 0.2         | Iris-setosa |
| 4.9          | 3.1         | 1.5          | 0.1         | Iris-setosa |

- Tab-delimited text file with three header rows:
  - attribute names
  - type (continuous, discrete or string)
  - class, meta, time

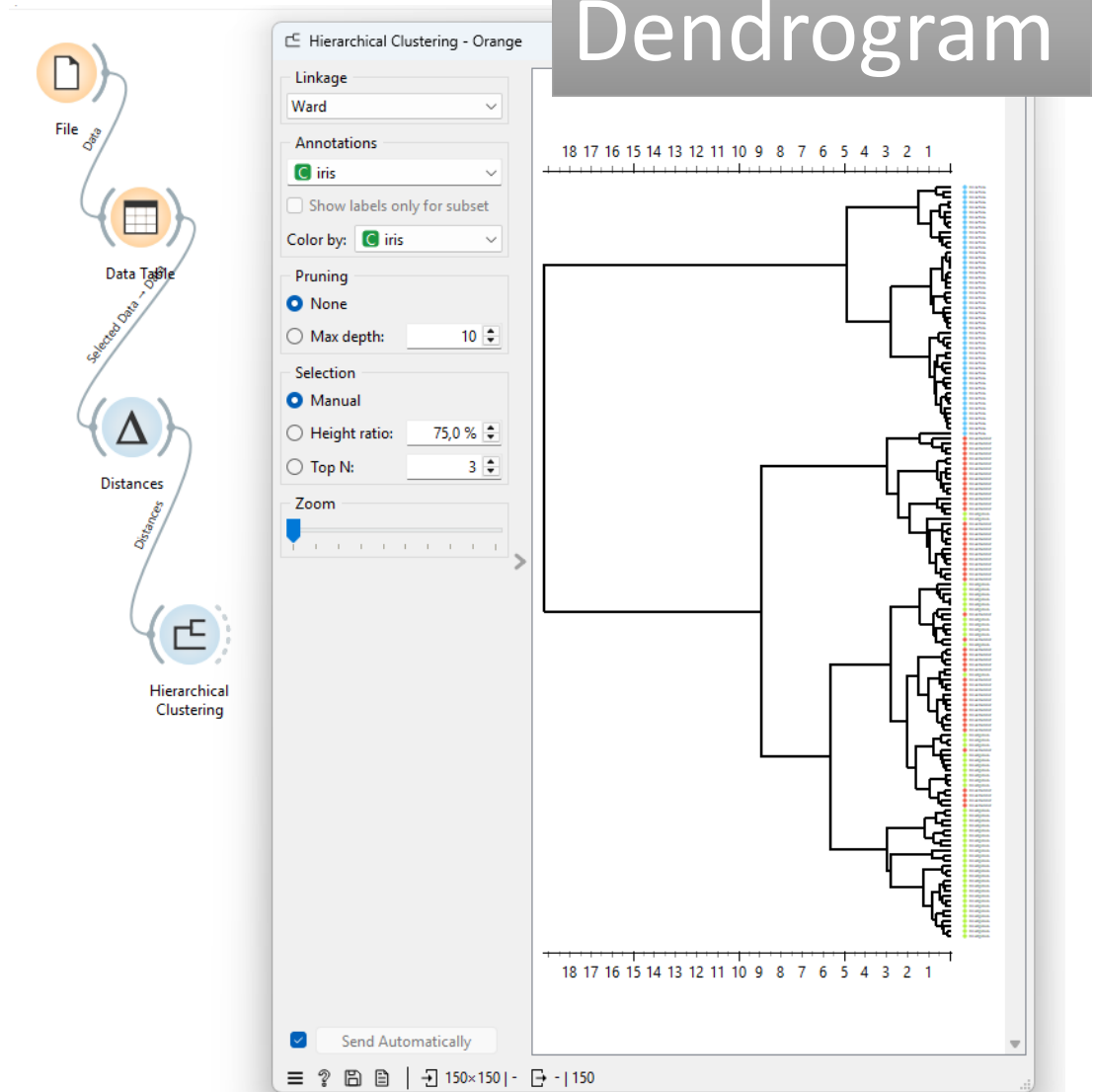
# Building a simple data workflow in Orange



# Hierarchical clustering



Euclidian  
distances



Dendrogram

# Use hierarchical clustering to explore patterns in a dataset of your choice

- **Load a Dataset**

- Load one of the datasets:
  - – iris.tab
  - – zoo.tab
  - – heart\_disease.tab

- **Create a workflow**

- Connect widgets in this order:
  - File → Distance Matrix → Hierarchical Clustering → Data Table
  - Optionally add Scatter Plot or Box Plot to visualize clusters

- **Try these tasks:**

- How many clusters make sense based on the dendrogram?
- Which features seem to drive the clustering the most?
- Can you identify any clear groups or outliers?
- Compare results when changing the distance metric (Euclidean, Manhattan, etc.)



# Classification models

