Data Science with Orange Toolbox:

Data Science for Everyone

Blaž Zupan
University of Ljubljana, Slovenia &
Baylor College of Medicine, Houston

Data Science

Linear algebra

Probability

Statistics

Mathematical optimization

Data & model representation

Machine learning

High performance computing

Data visualization

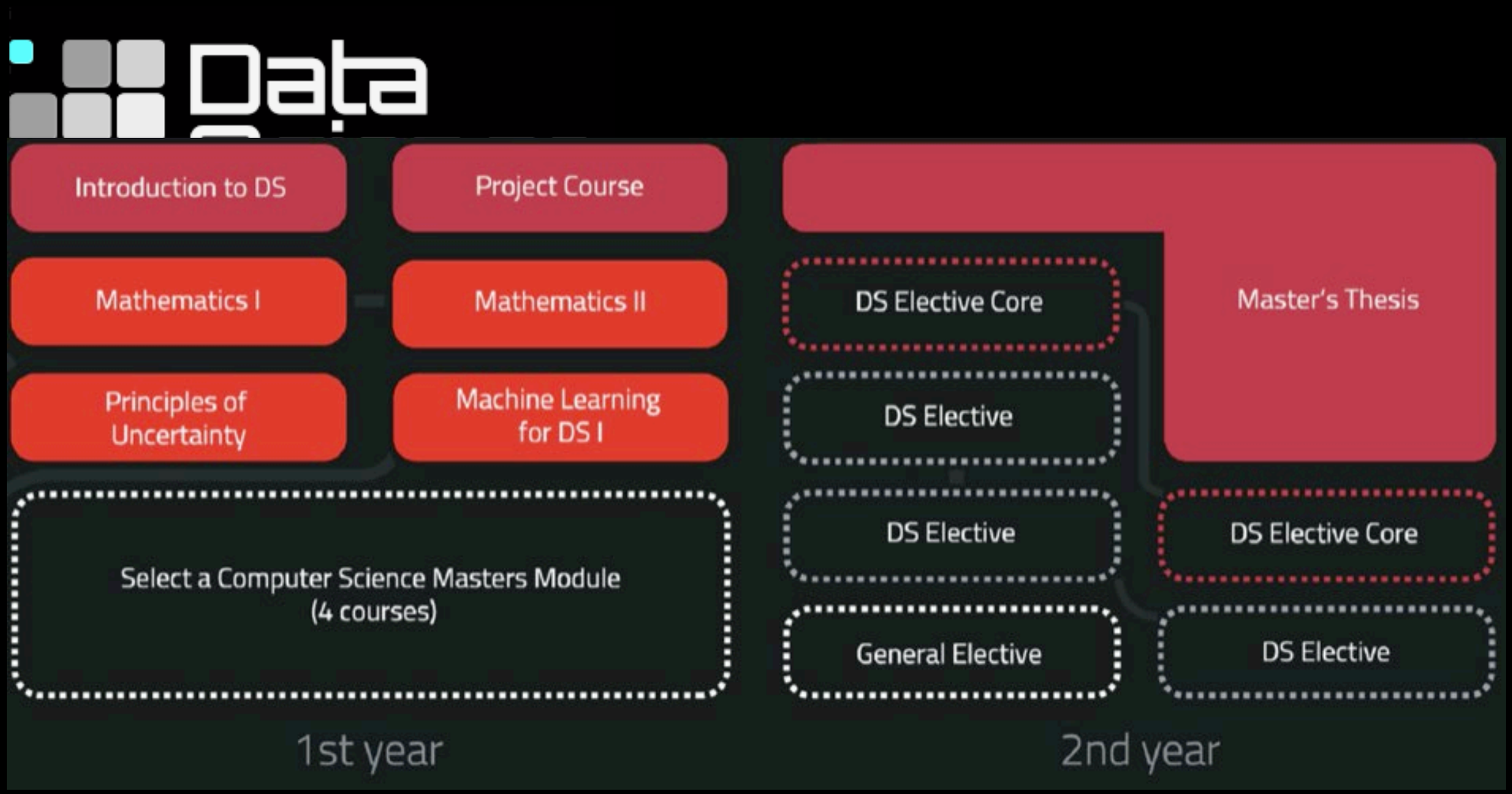
Scripting languages

Data Science

Data Science @UL-FRI

2-5 hours to train concepts

Data Science



Data Science

Data Science @UL-FRI

2-5 hours to train concepts

data preparation, feature engineering, clustering, modeling, evaluation, data understanding

WÄRTSILÄ

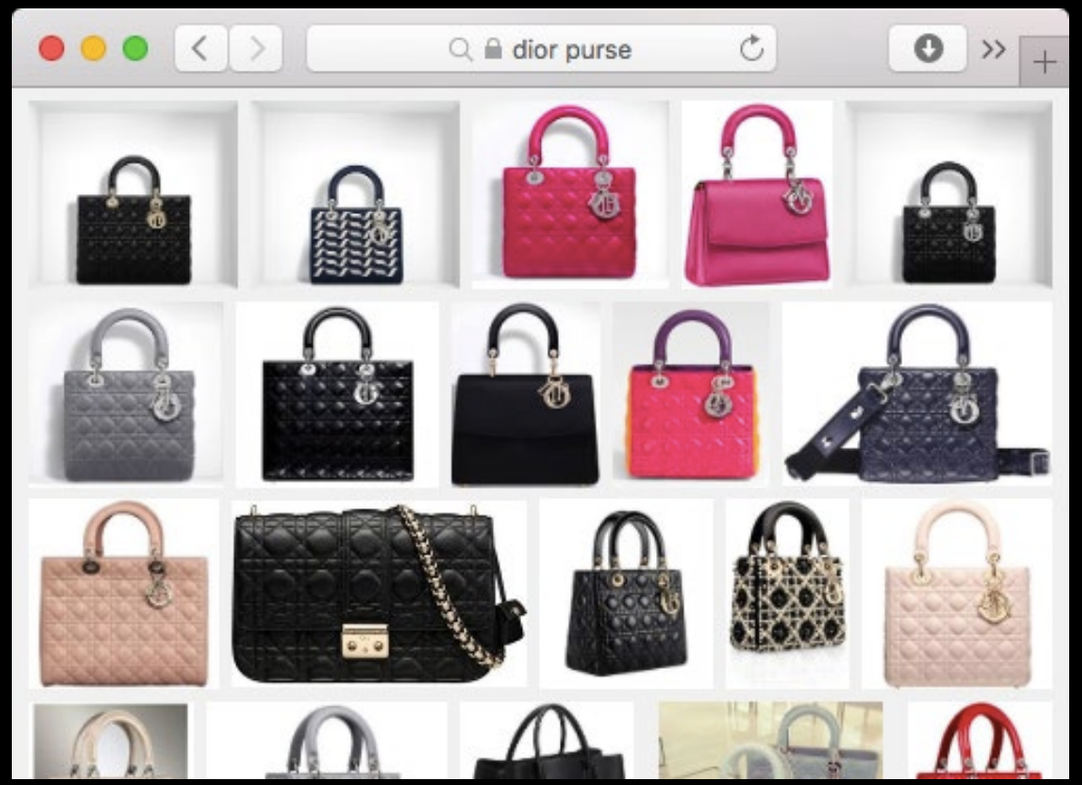
lek

član skupine Sandoz

```
def edges_from_osmdb(osmdb, vertex_namespace, slogs, profiledb=None):
    """generates (vertex1_label, vertex2_label, edgepayload) from osmdb"""

    street_id_counter = 0
    street_names = {}

    # for each edge in the osmdb
    for i, (id, parent_id, node1, node2, distance, geom, tags) in enumerate( (

        # Find rise/fall of edge, if profiledb is given
        rise=0
        fall=0
        if profiledb:
            profile = profiledb.get( id )
            if profile:
                rise, fall = get_rise_and_fall( profile )

        # insert end vertices of edge to graph
        vertex1_label = "%s-%s"%(vertex_namespace,node1)
        vertex2_label = "%s-%s"%(vertex_namespace,node2)

        # create ID for the way's street
        street_name = tags.get("name")
        if street_name is None:
            street_id_counter += 1
            street_id = street_id_counter
        else:
            if street_name not in street_names:
                street_id_counter += 1
                street_names[street_name] = street_id_counter
            street_id = street_names[street_name]

        # Create edges to be inserted into graph
        s1 = Street( id, distance, rise, fall )
        s2 = Street( id, distance, fall, rise, reverse_of_source=True )
        s1.way = street_id
        s2.way = street_id

        # See if the way's highway tag is penalized with a 'slog' value; if s(
        slog = slogs.get( tags.get("highway") )
        if slog:
            s1.slog = s2.slog = slog

        # Add the forward edge and the return edge if the edge is not oneway
    yield vertex1_label, vertex2_label, s1

        oneway = tags.get("oneway")
        if oneway != "true" and oneway != "yes":
        yield vertex2_label, vertex1_label, s2
```

Silvana

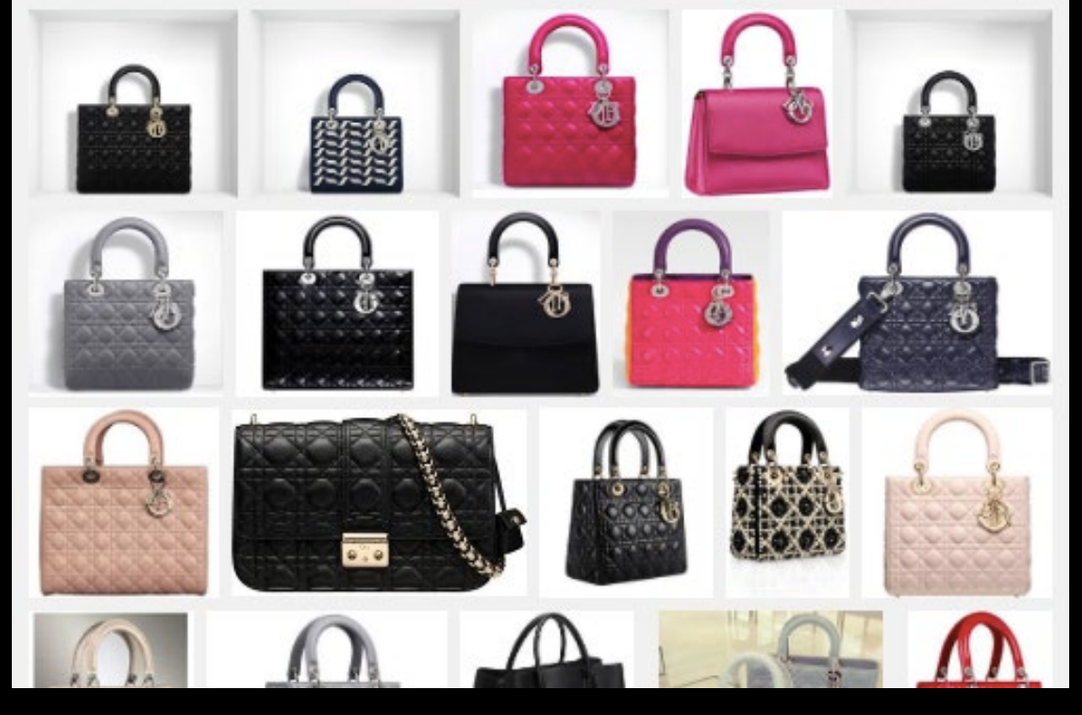

Dior


dior purse

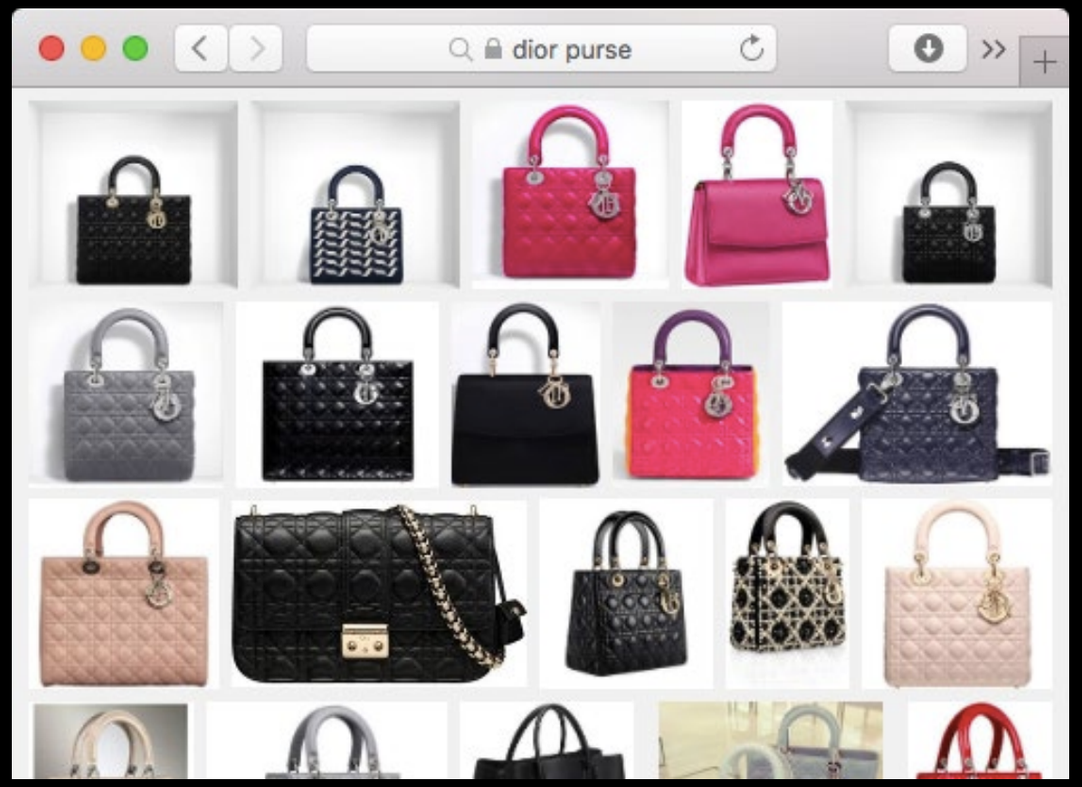

Dior

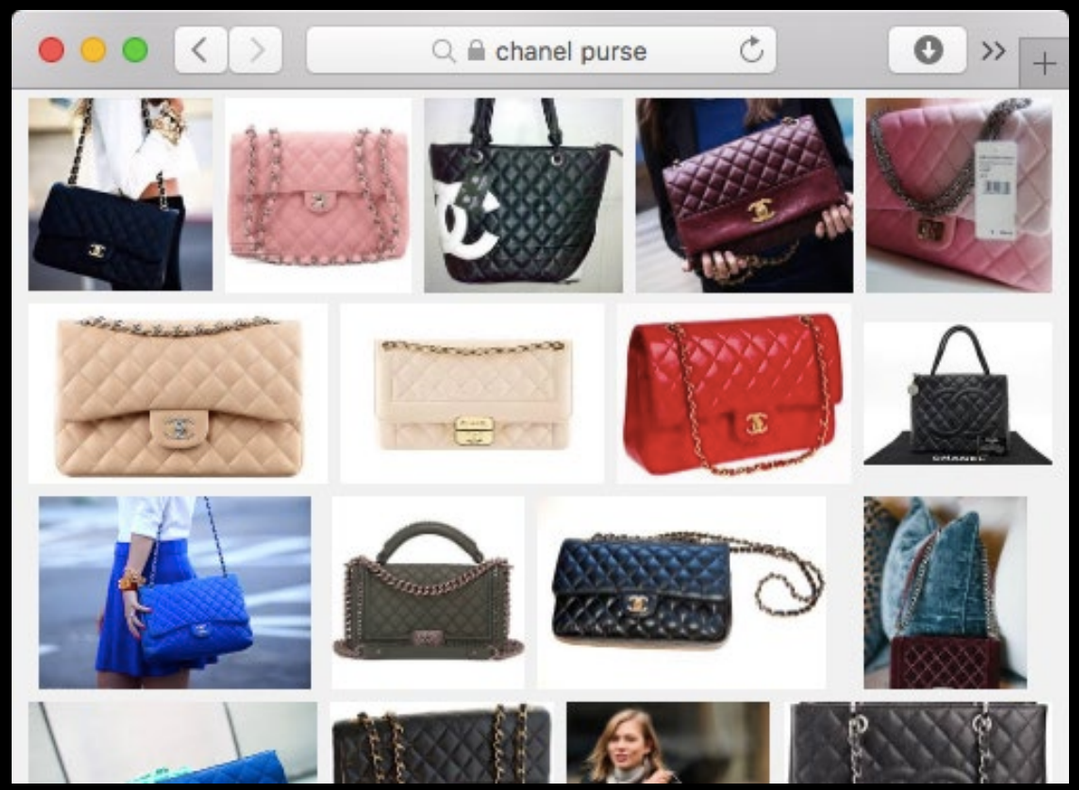
hermes purse

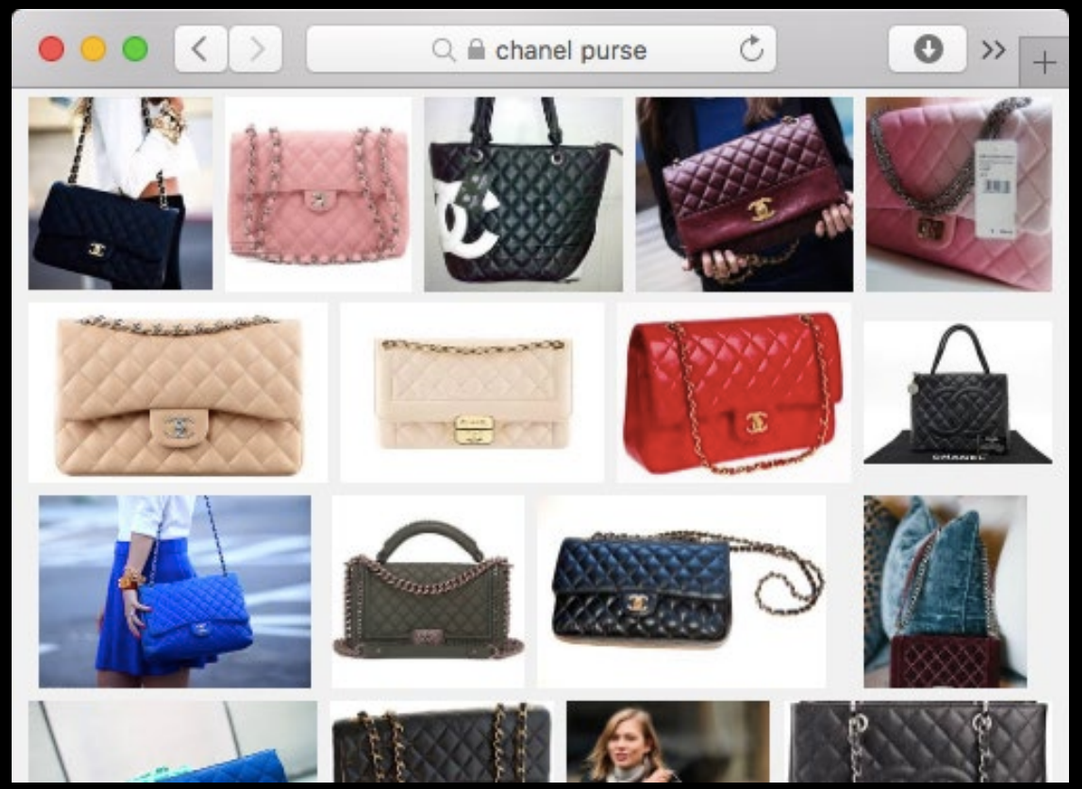
dior purse

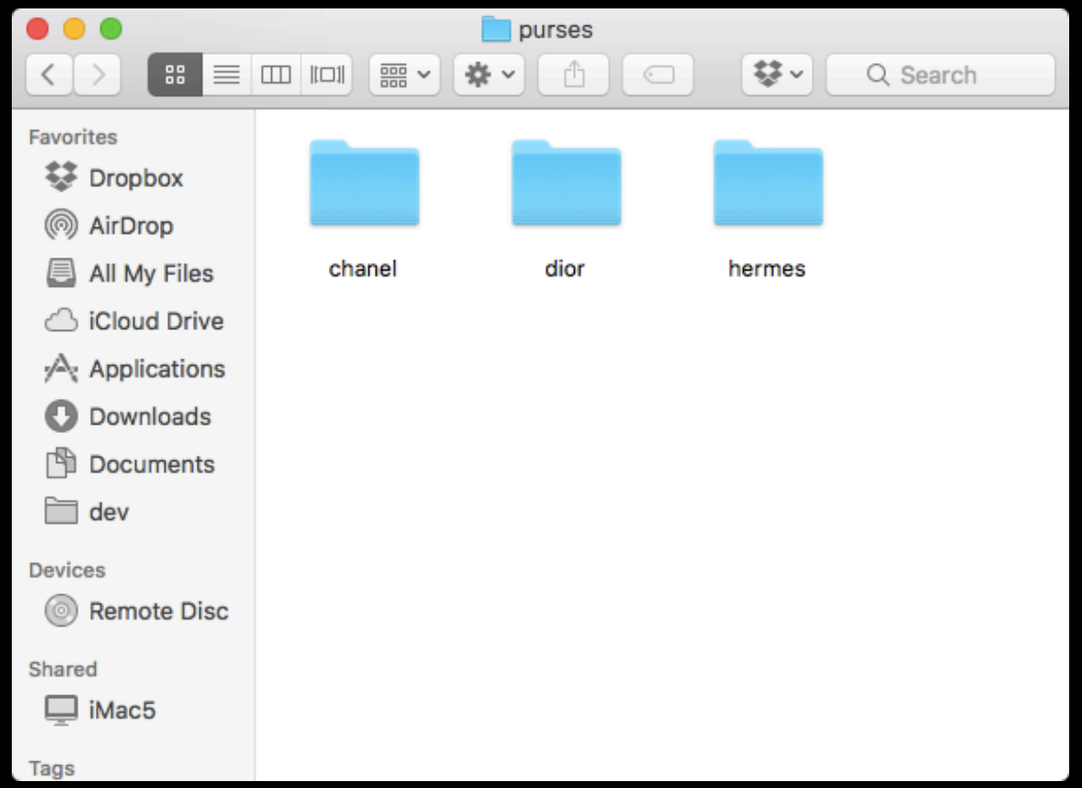


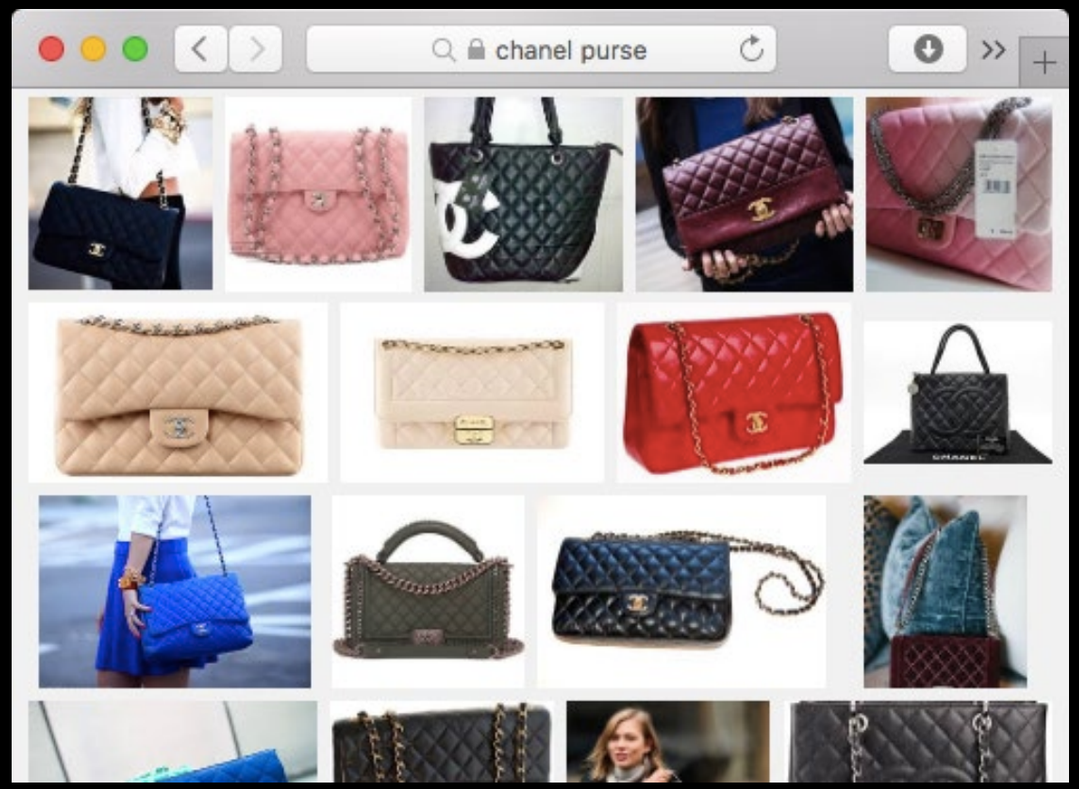

hermes purse
460 × 579 - br...

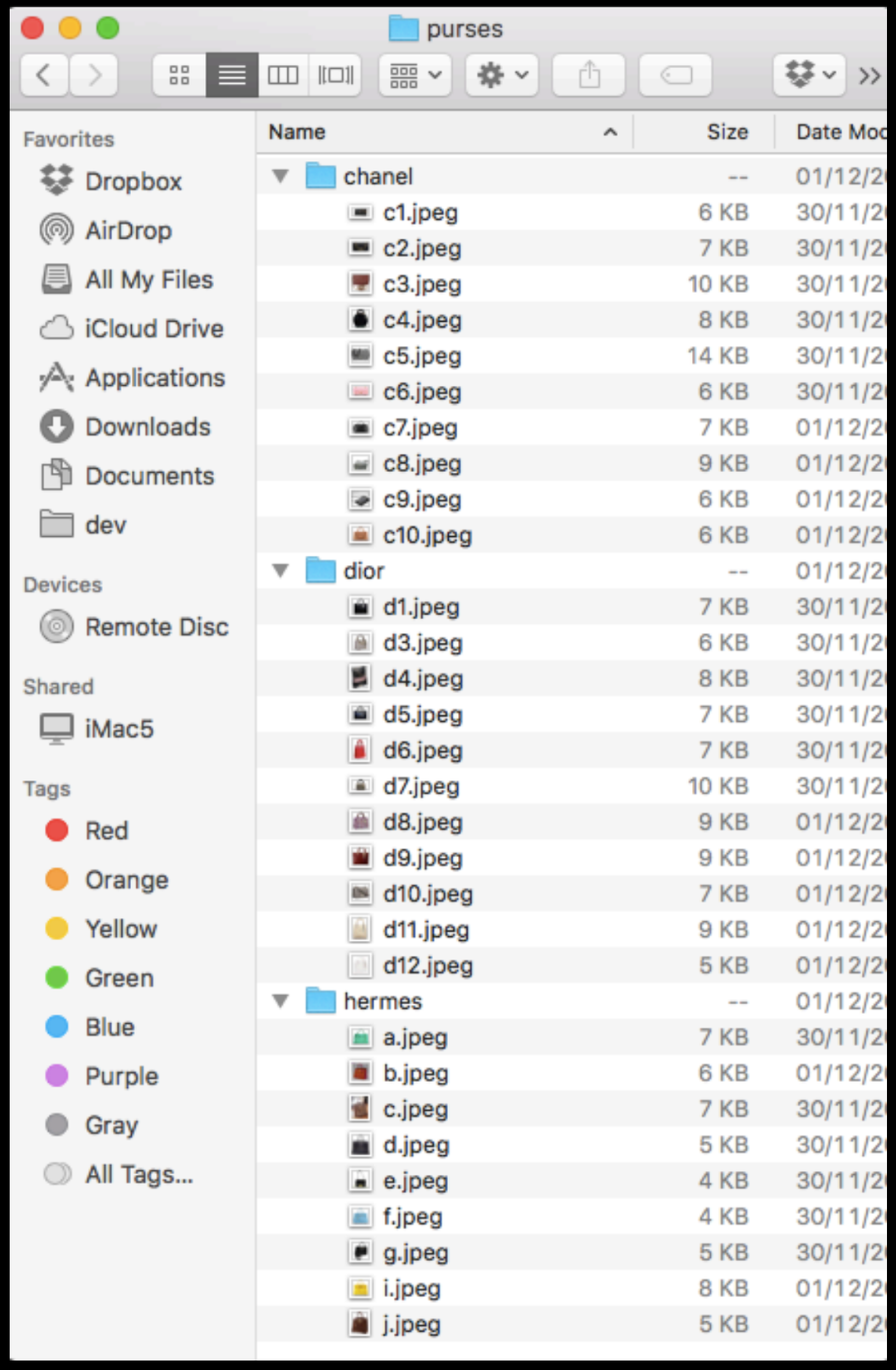

dior purse

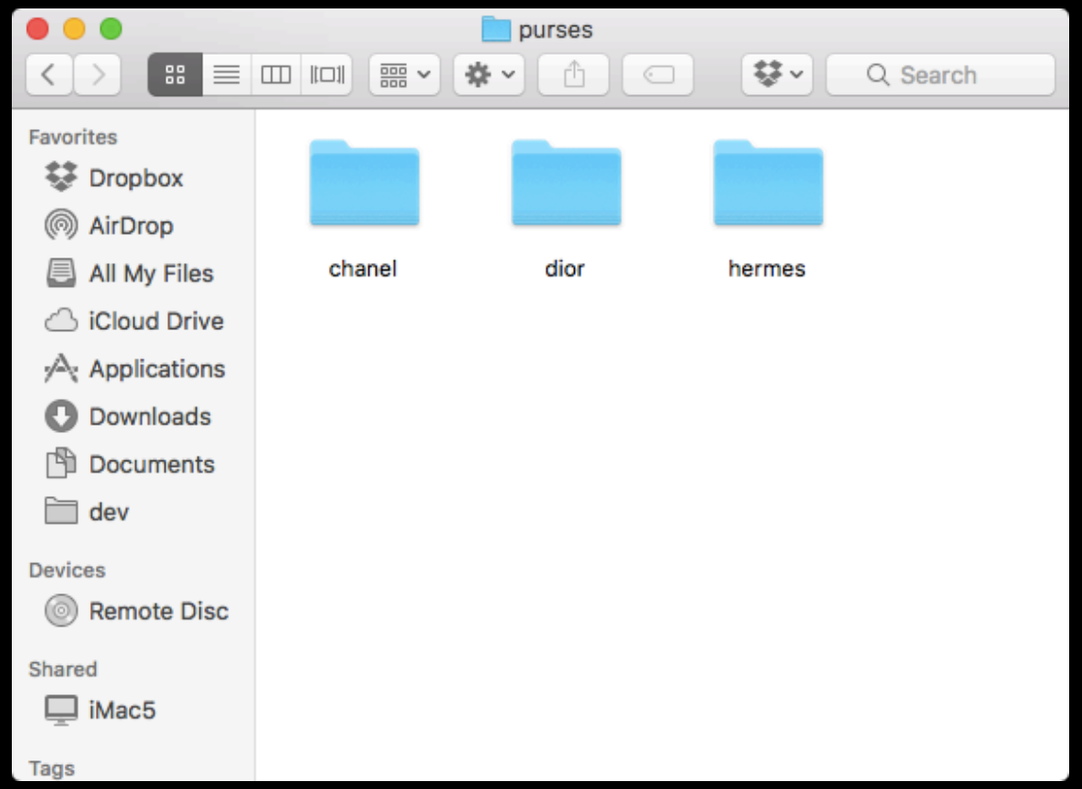

chanel purse

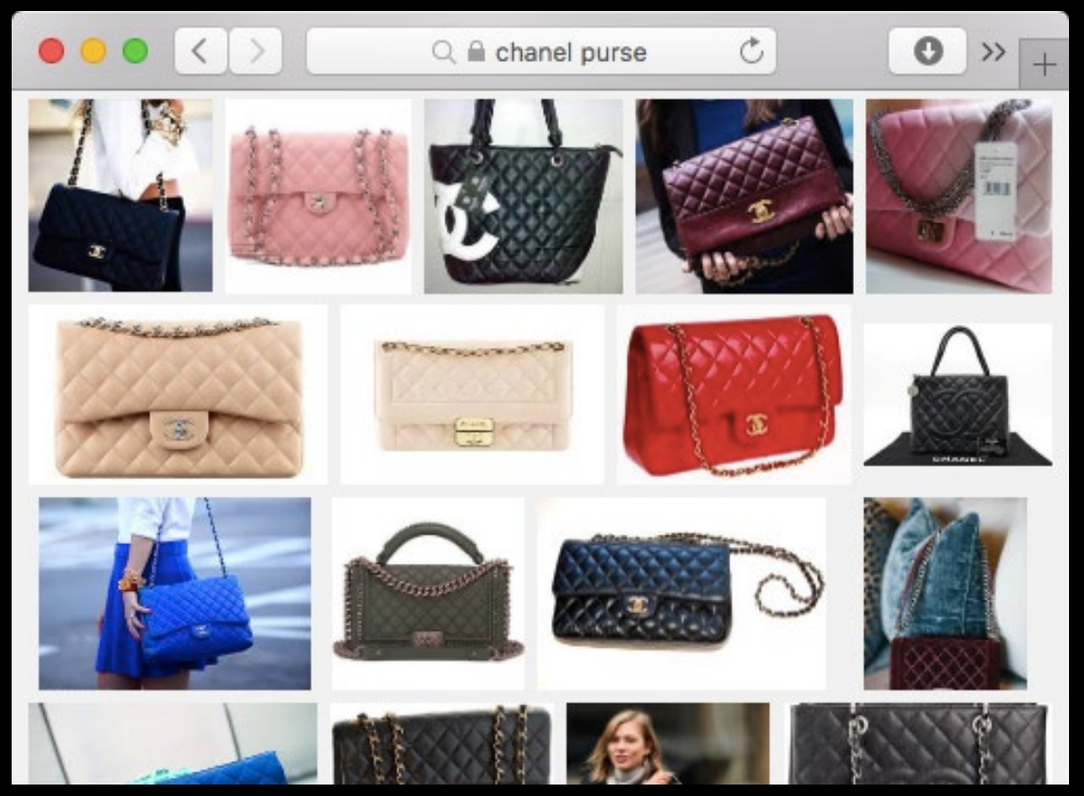

chanel purse


purses
Search
Favorites
Dropbox
AirDrop
All My Files
iCloud Drive
Applications
Downloads
Documents
dev
Devices
Remote Disc
Shared
iMac5
Tags
chanel
dior
hermes


chanel purse


purses
Favorites
Dropbox
AirDrop
All My Files
iCloud Drive
Applications
Downloads
Documents
dev
Devices
Remote Disc
Shared
iMac5
Tags
Red
Orange
Yellow
Green
Blue
Purple
Gray
All Tags...
Name
Size
Date Mod
chanel -- 01/12/2
c1.jpeg 6 KB 30/11/2
c2.jpeg 7 KB 30/11/2
c3.jpeg 10 KB 30/11/2
c4.jpeg 8 KB 30/11/2
c5.jpeg 14 KB 30/11/2
c6.jpeg 6 KB 30/11/2
c7.jpeg 7 KB 01/12/2
c8.jpeg 9 KB 01/12/2
c9.jpeg 6 KB 01/12/2
c10.jpeg 6 KB 01/12/2
dior -- 01/12/2
d1.jpeg 7 KB 30/11/2
d3.jpeg 6 KB 30/11/2
d4.jpeg 8 KB 30/11/2
d5.jpeg 7 KB 30/11/2
d6.jpeg 7 KB 30/11/2
d7.jpeg 10 KB 30/11/2
d8.jpeg 9 KB 01/12/2
d9.jpeg 9 KB 01/12/2
d10.jpeg 7 KB 01/12/2
d11.jpeg 9 KB 01/12/2
d12.jpeg 5 KB 01/12/2
hermes -- 01/12/2
a.jpeg 7 KB 30/11/2
b.jpeg 6 KB 01/12/2
c.jpeg 7 KB 30/11/2
d.jpeg 5 KB 30/11/2
e.jpeg 4 KB 30/11/2
f.jpeg 4 KB 30/11/2
g.jpeg 5 KB 30/11/2
i.jpeg 8 KB 01/12/2
j.jpeg 5 KB 01/12/2

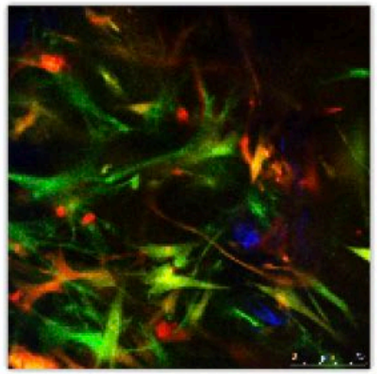

purses
Search
Favorites
Dropbox
AirDrop
All My Files
iCloud Drive
Applications
Downloads
Documents
dev
Devices
Remote Disc
Shared
iMac5
Tags
chanel
dior
hermes


chanel purse

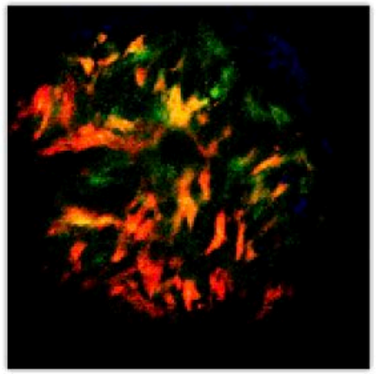
Purse Analytics

(live demo)

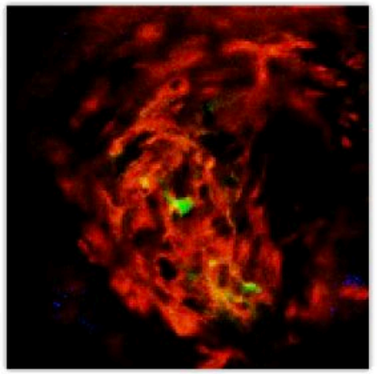
a



Day7

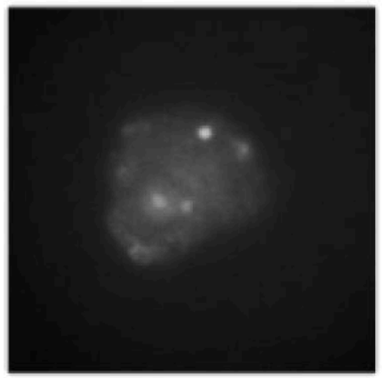


Day7

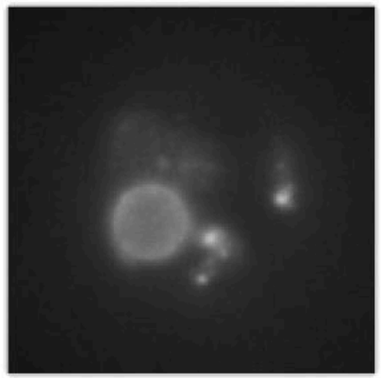


Day14

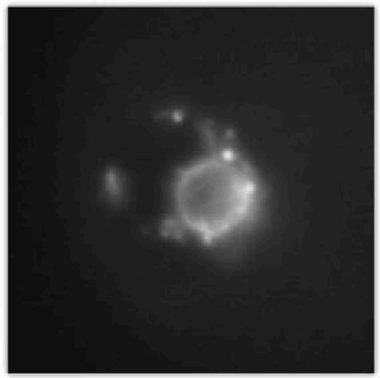
b



NSN

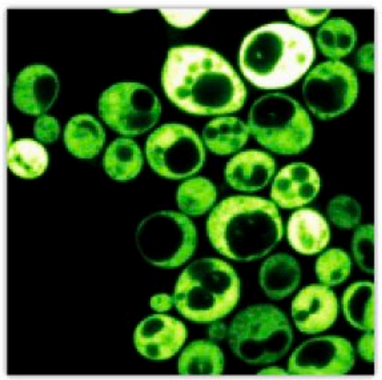


SN

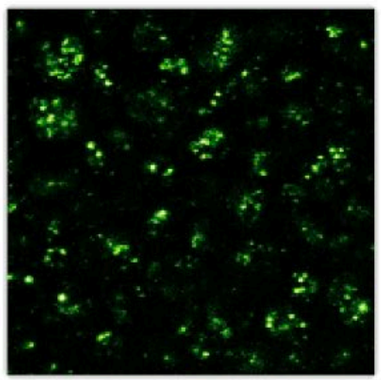


SN

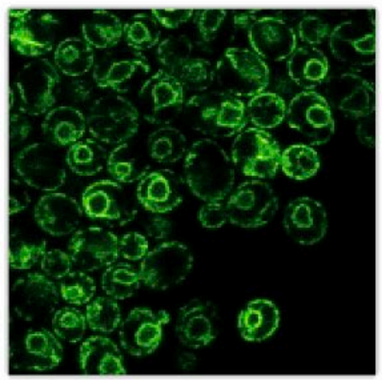
c



cytoplasm



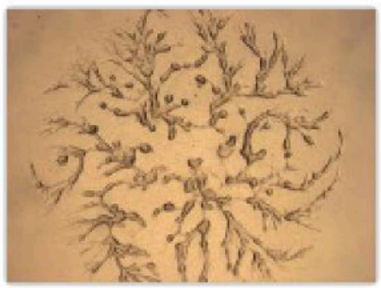
endosome



er

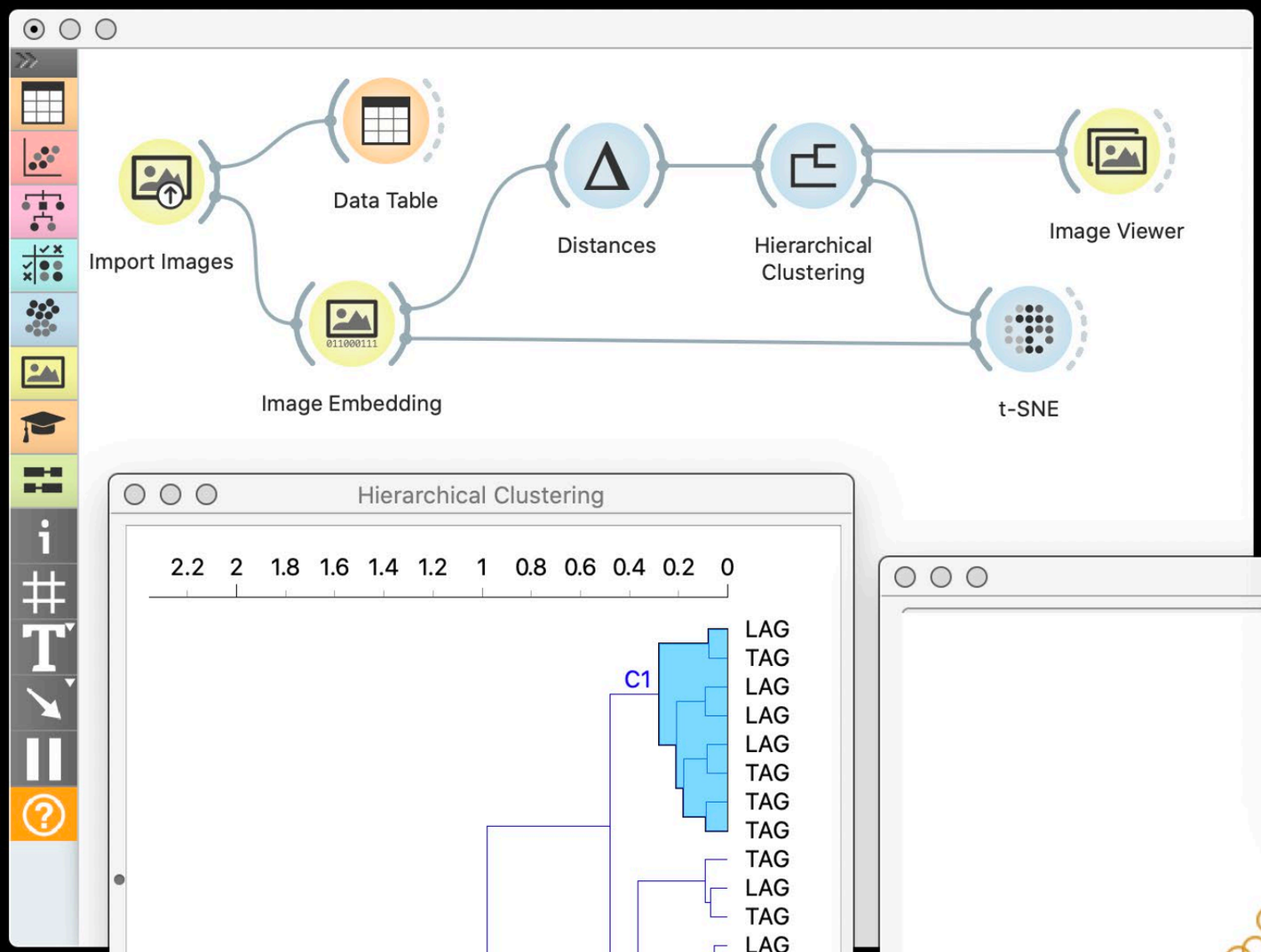
d

LAG



STR

TAG


Import Images
Data Table
Image Embedding
Distances
Hierarchical Clustering
Image Viewer
t-SNE

Hierarchical Clustering
2.2 2 1.8 1.6 1.4 1.2 1 0.8 0.6 0.4 0.2 0
C1
LAG
TAG
LAG
LAG
LAG
TAG
TAG
LAG
TAG
LAG
LAG
TAG
TAG
TAG
TAG
TAG
LAG
2.2 2 1.8 1.6 1.4 1.2 1 0.8 0.6 0.4 0.2 0

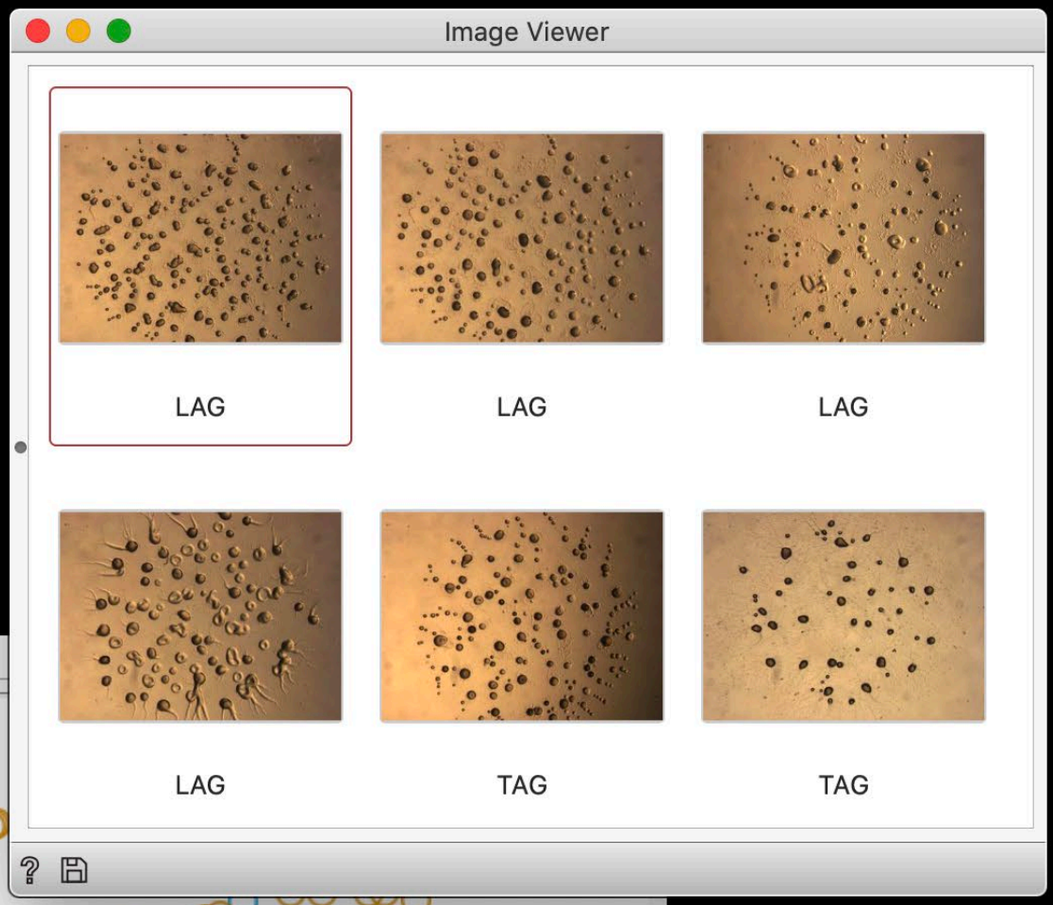
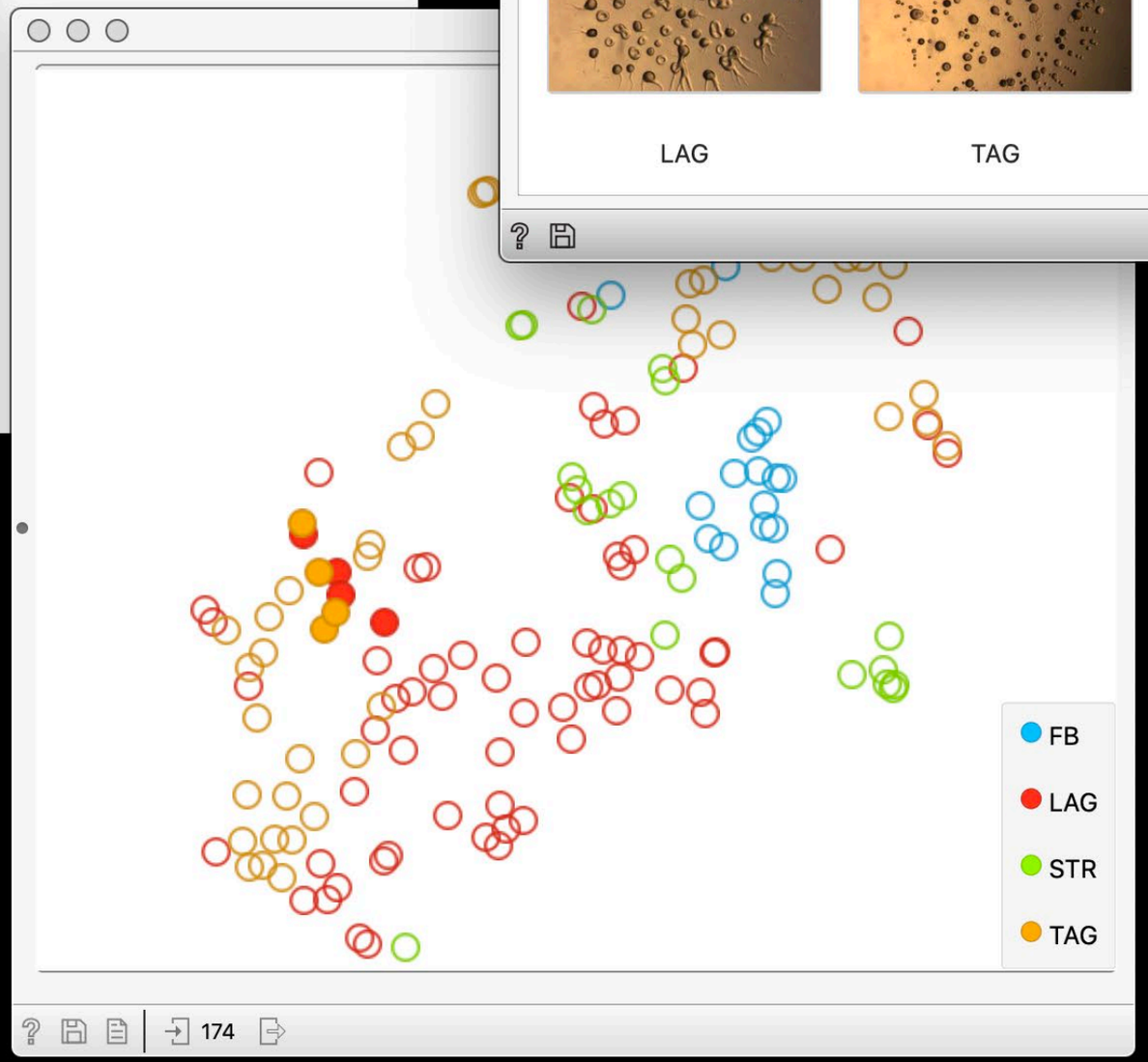


Image Viewer
LAG
LAG
LAG
LAG
TAG
TAG


FB
LAG
STR
TAG
174

nature communications

Article | Open Access | Published: 07 October 2019

Democratized image analytics by visual programming through integration of deep models and small-scale machine learning

Primož Godec, Matjaž Pančur, Nejc Ilenič, Andrej Čopar, Martin Stražar, Aleš Erjavec, Ajda Pretnar, Janez Demšar, Anže Starič, Marko Toplak, Lan Žagar, Jan Hartman, Hamilton Wang, Riccardo Bellazzi, Uroš Petrovič, Silvia Garagna, Maurizio Zuccotti, Dongsu Park, Gad Shaulsky & Blaž Zupan ✉

Nature Communications **10**, Article number: 4551 (2019) | Cite this article

Abstract

Analysis of biomedical images requires computational expertize that are uncommon among biomedical scientists. Deep learning approaches for image analysis provide an opportunity to develop user-friendly tools for exploratory data analysis. Here, we use the visual programming toolbox Orange (http://orange.biolab.si) to

Download PDF

Sections | Figures | References

Abstract
Introduction
Results
Discussion
Methods
Data availability
Code availability
References
Acknowledgements
Author information
Ethics declarations
Additional information

Key Concepts

Interactive Visualizations

Workflows

“Lego Bricks” for Data Science

Democratization of Data Science

Easy Access to Data

Reproducibility

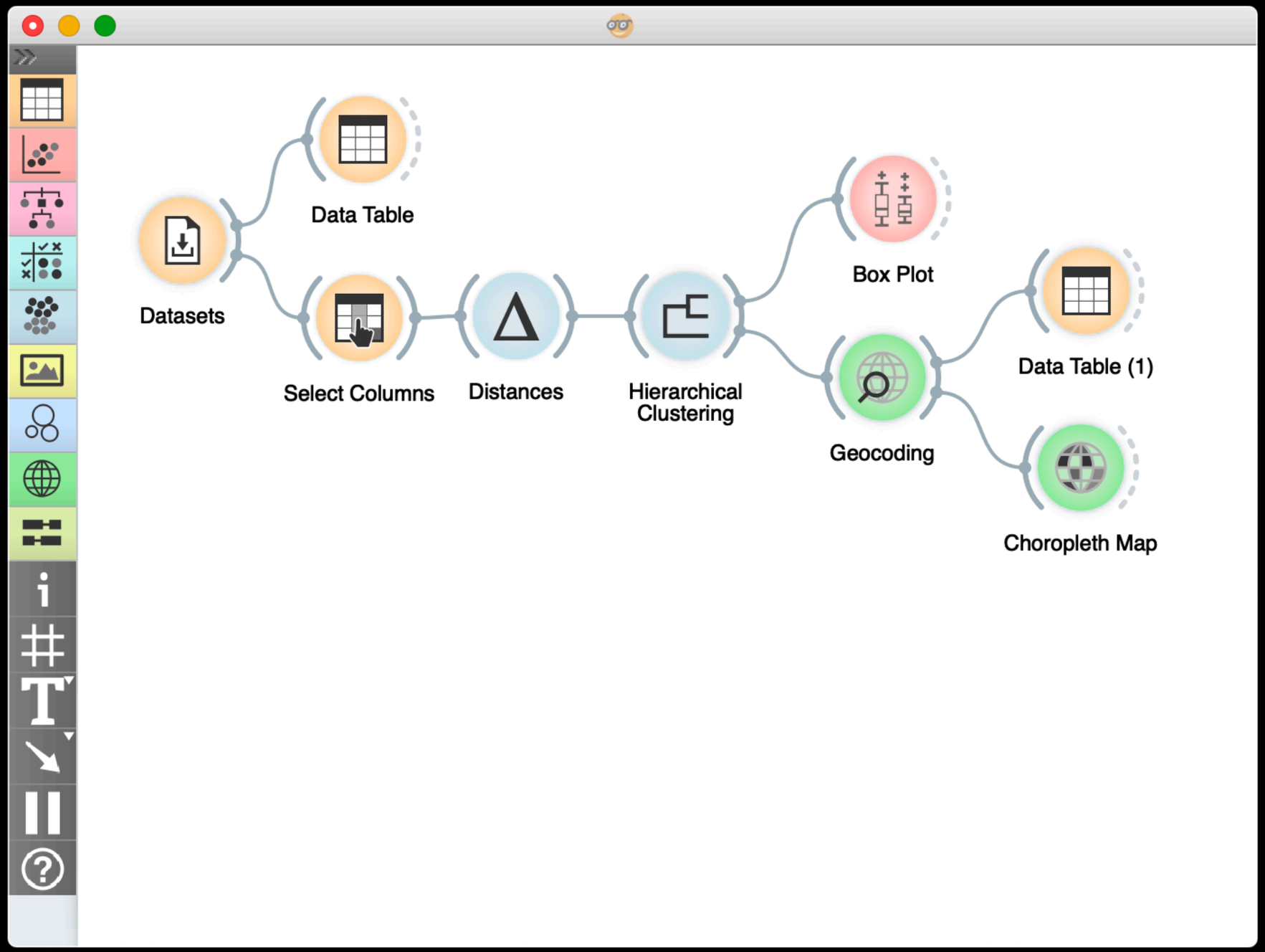
Customization

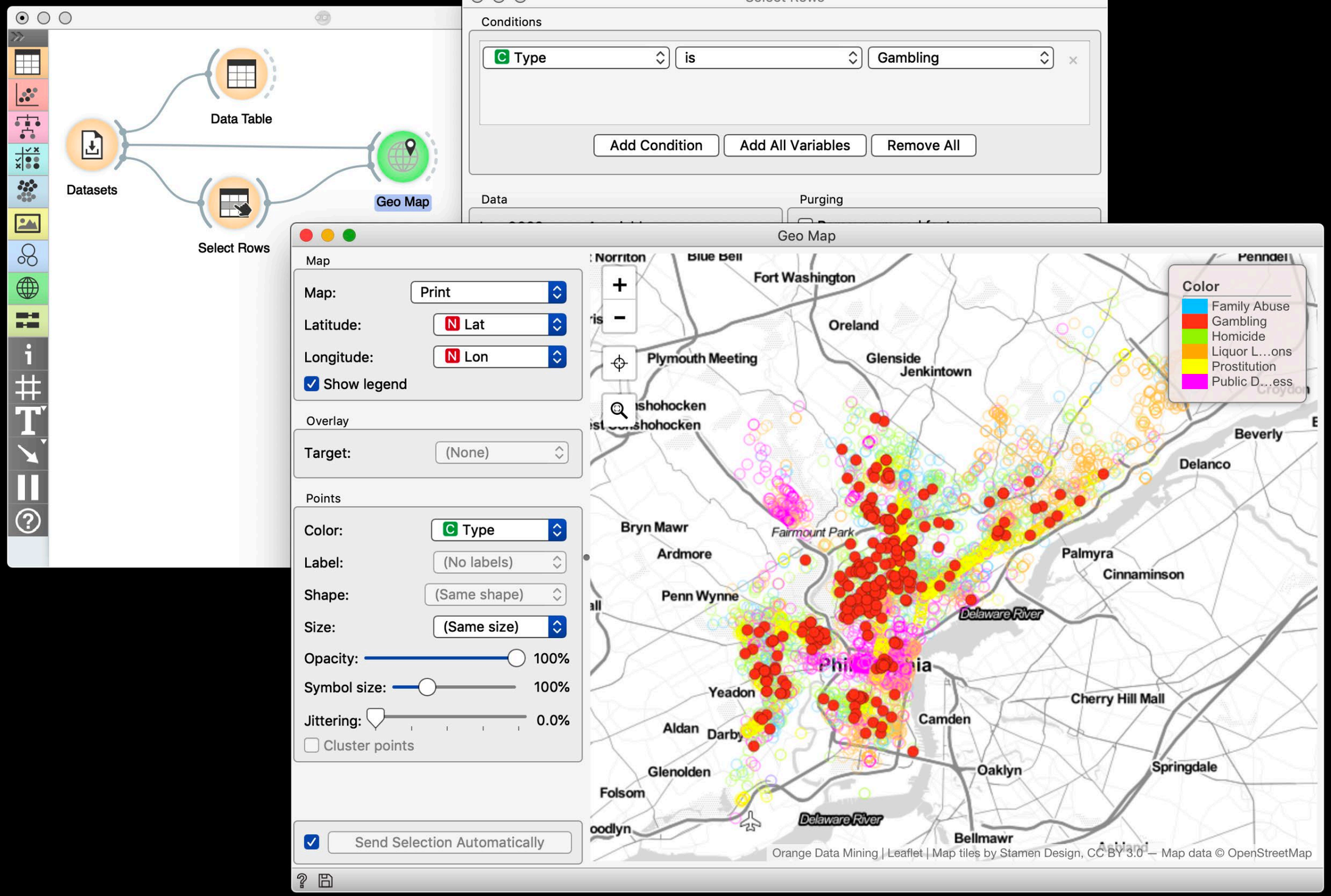
Story Telling

Experimentation

Tools for Training

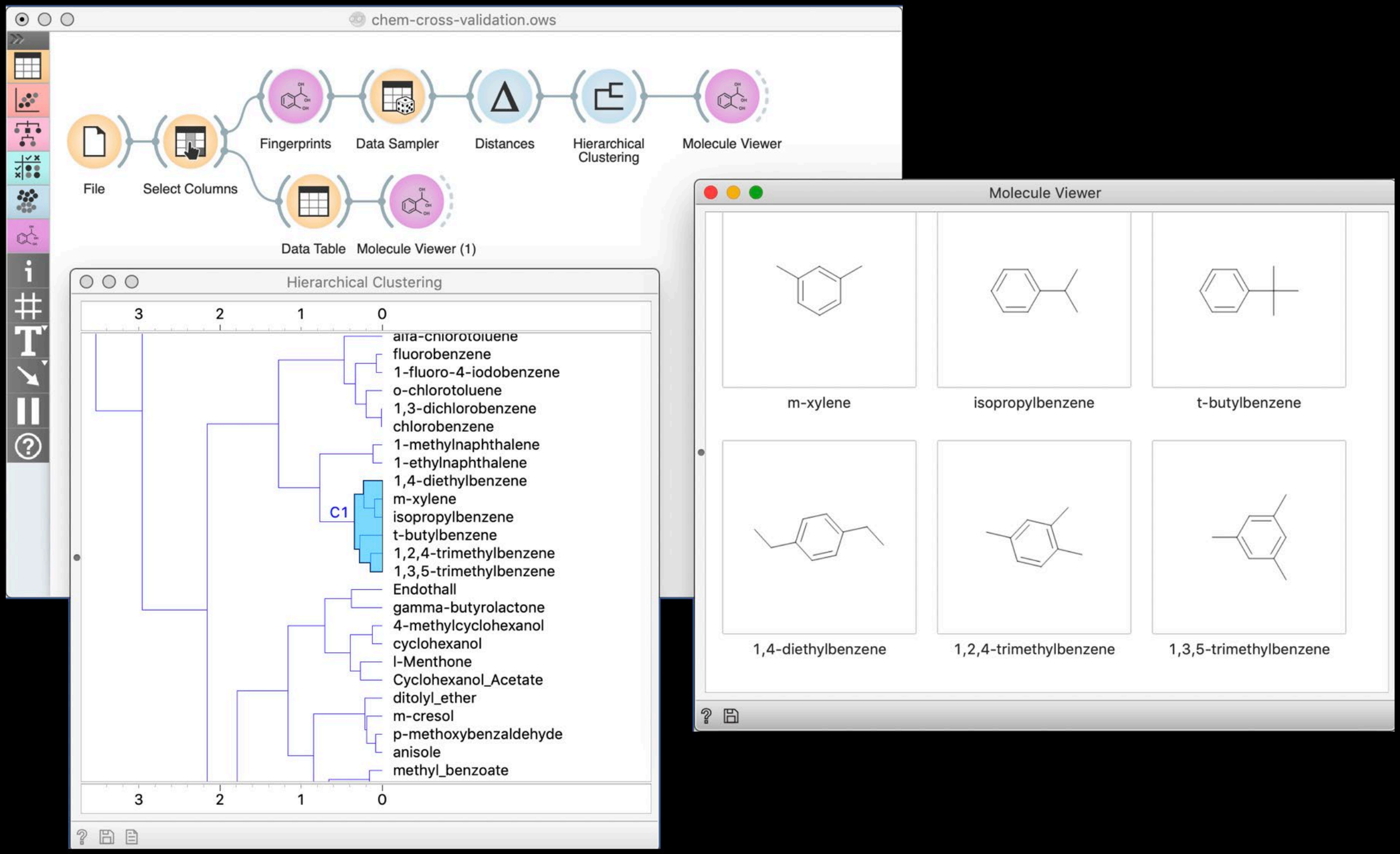
“Lego” Bricks for Data Science

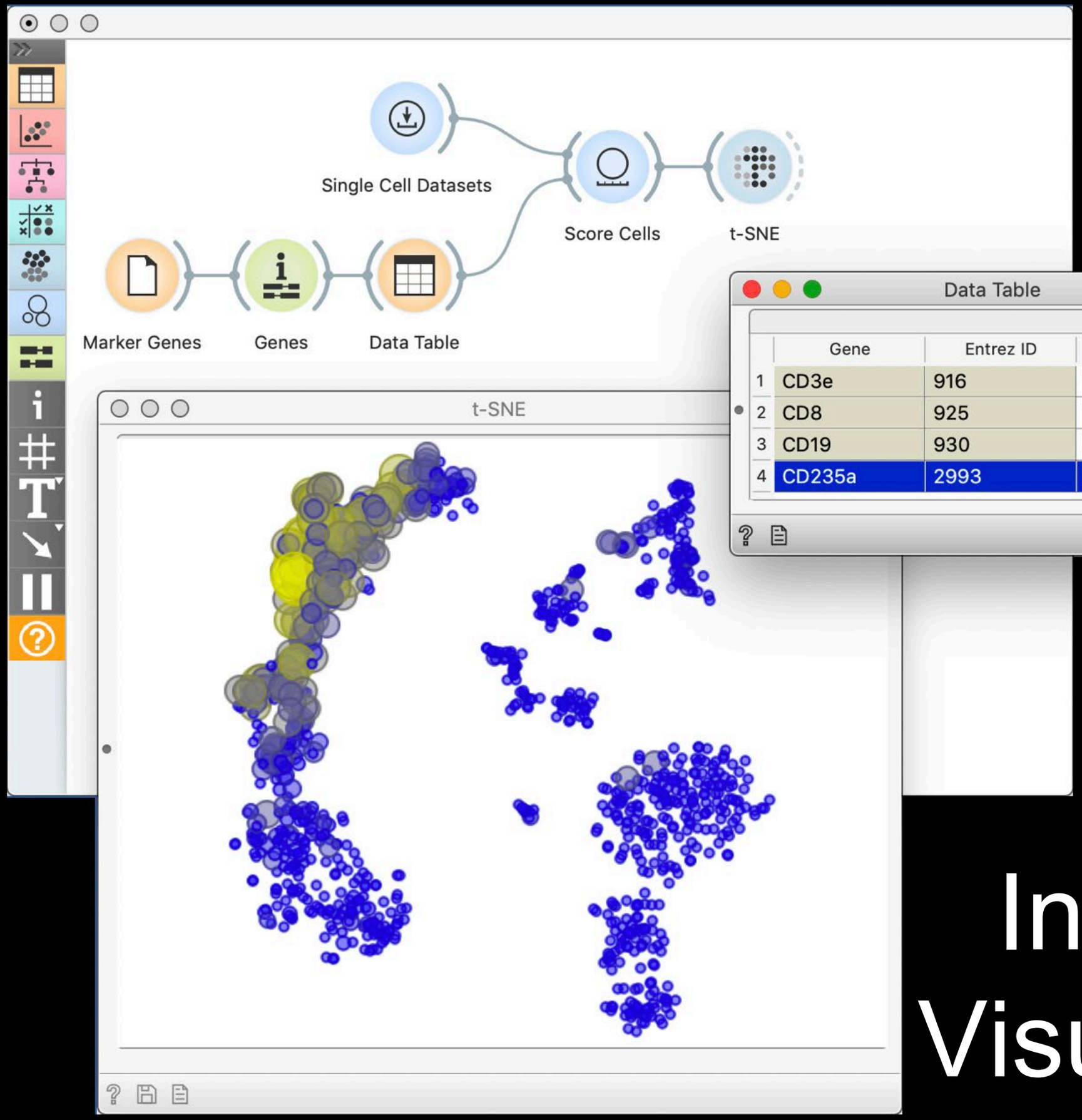




Responsive Workflows

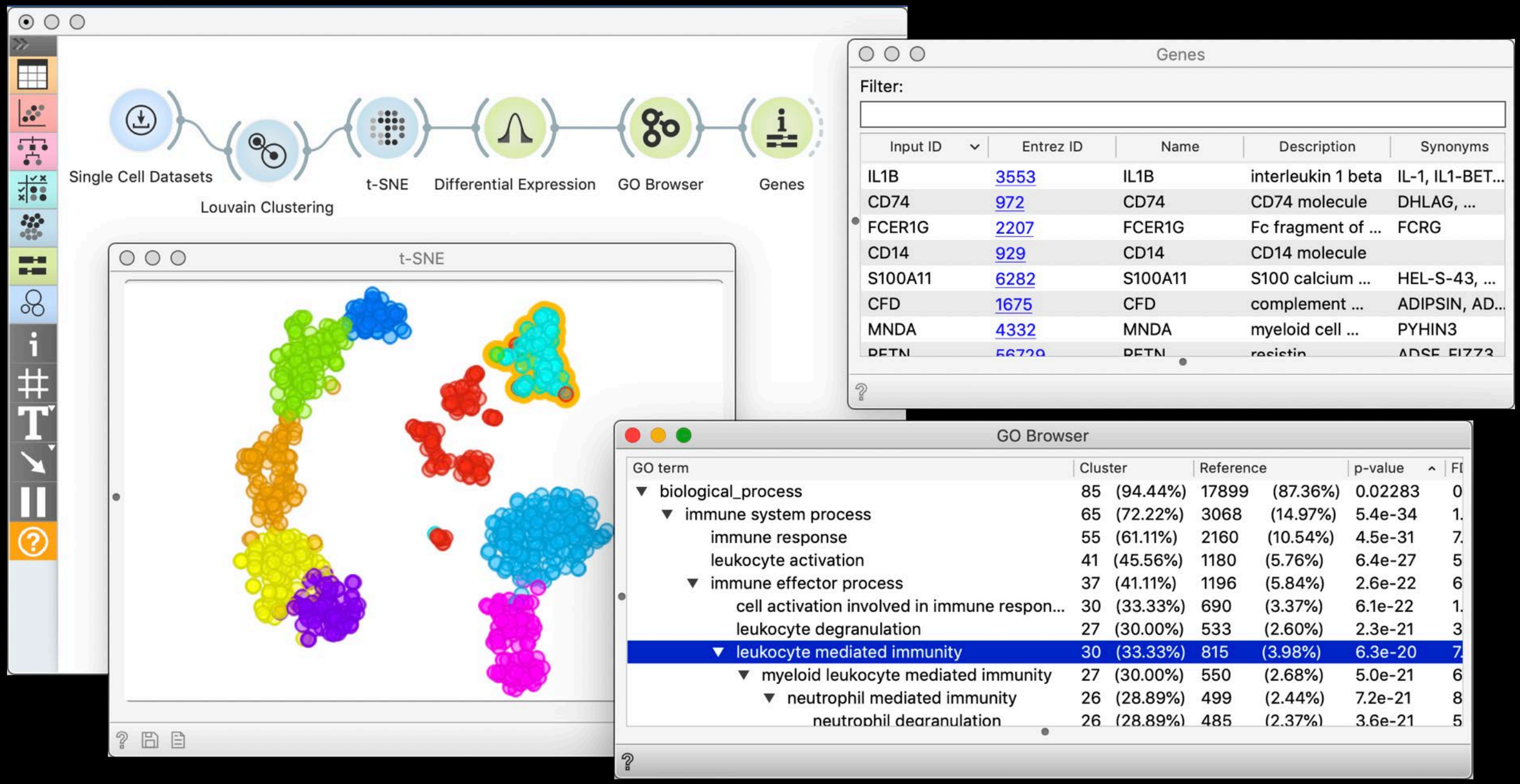
Interactive Visualizations



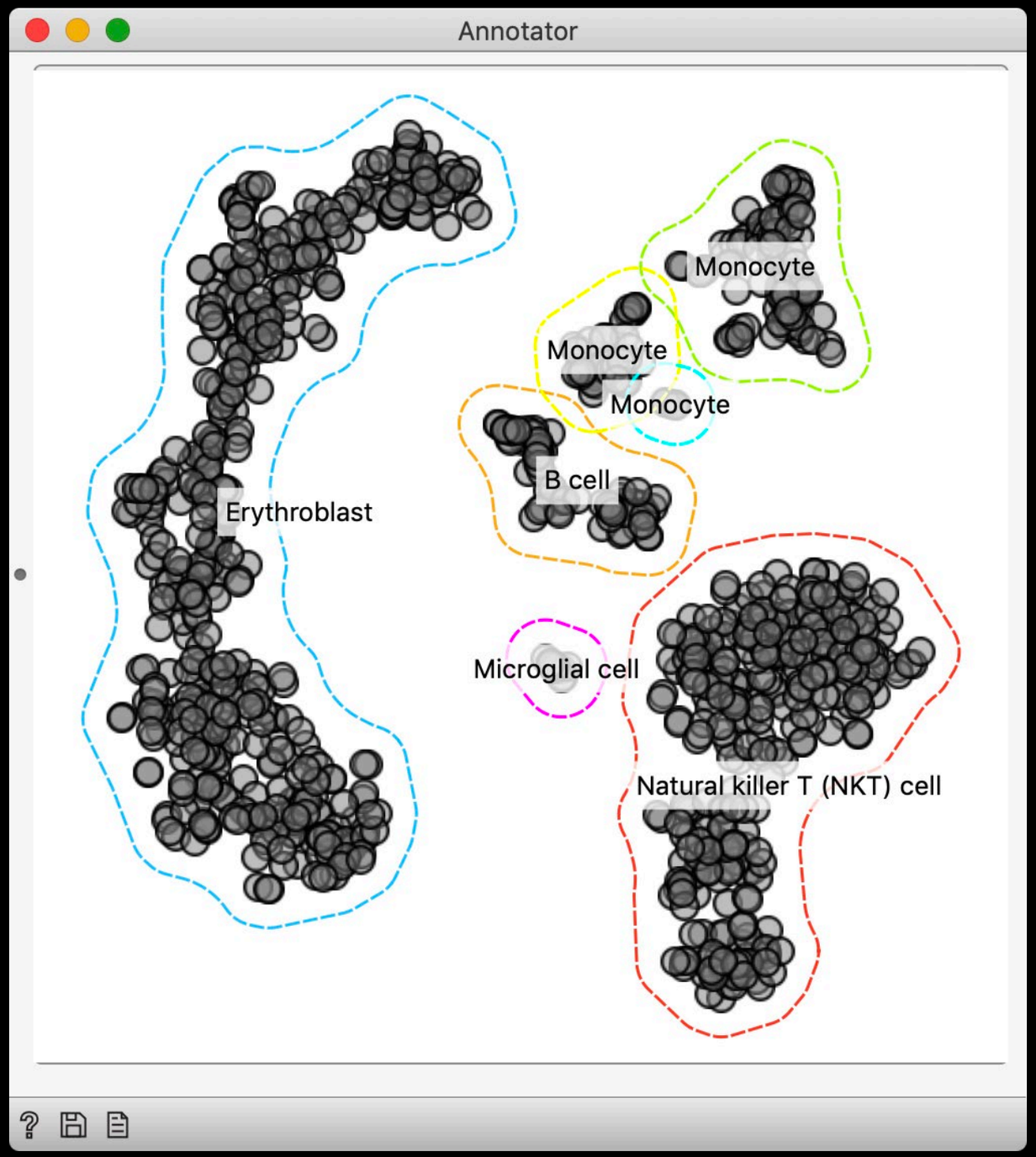


Interactive Visualizations

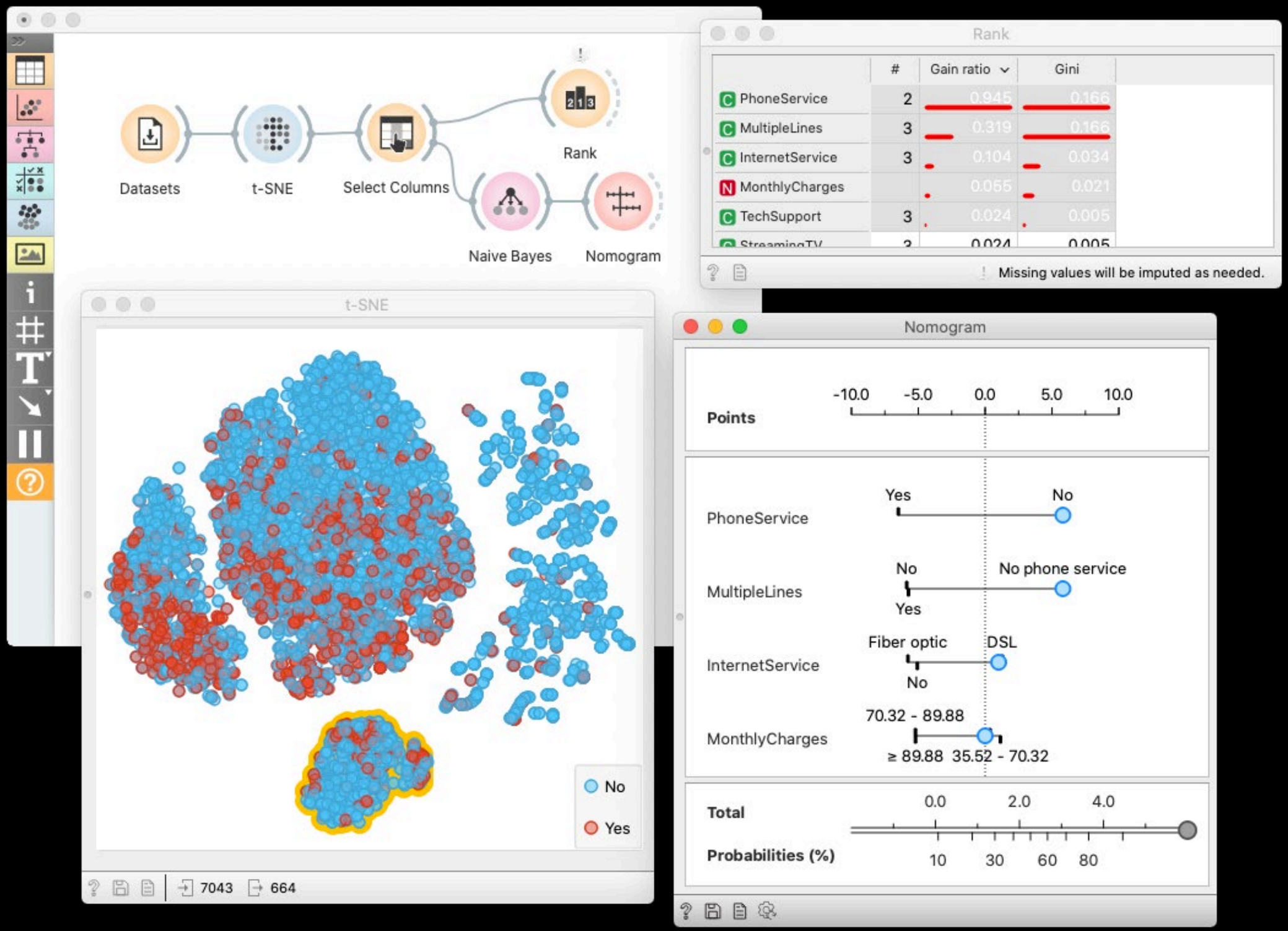
Data Integration

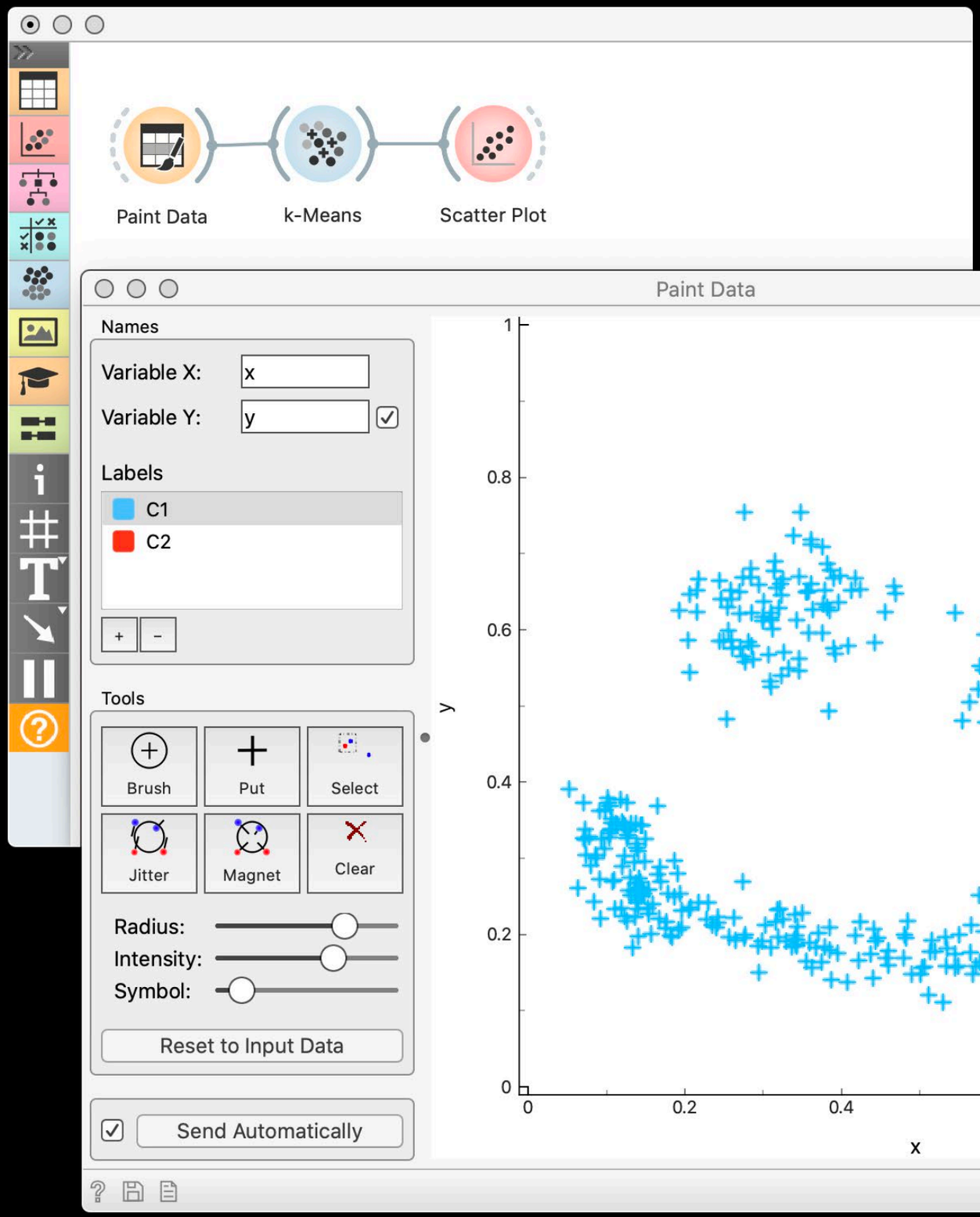


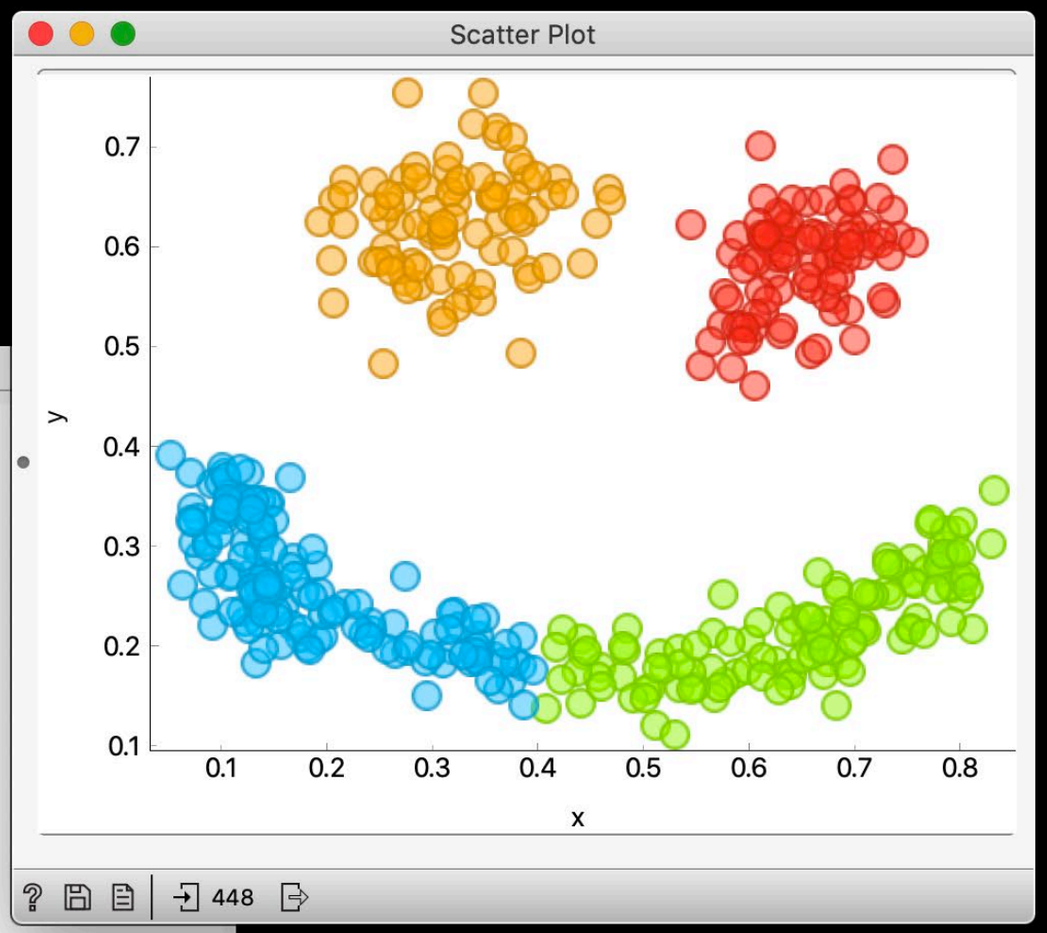
Explanation



Interactive Explanation

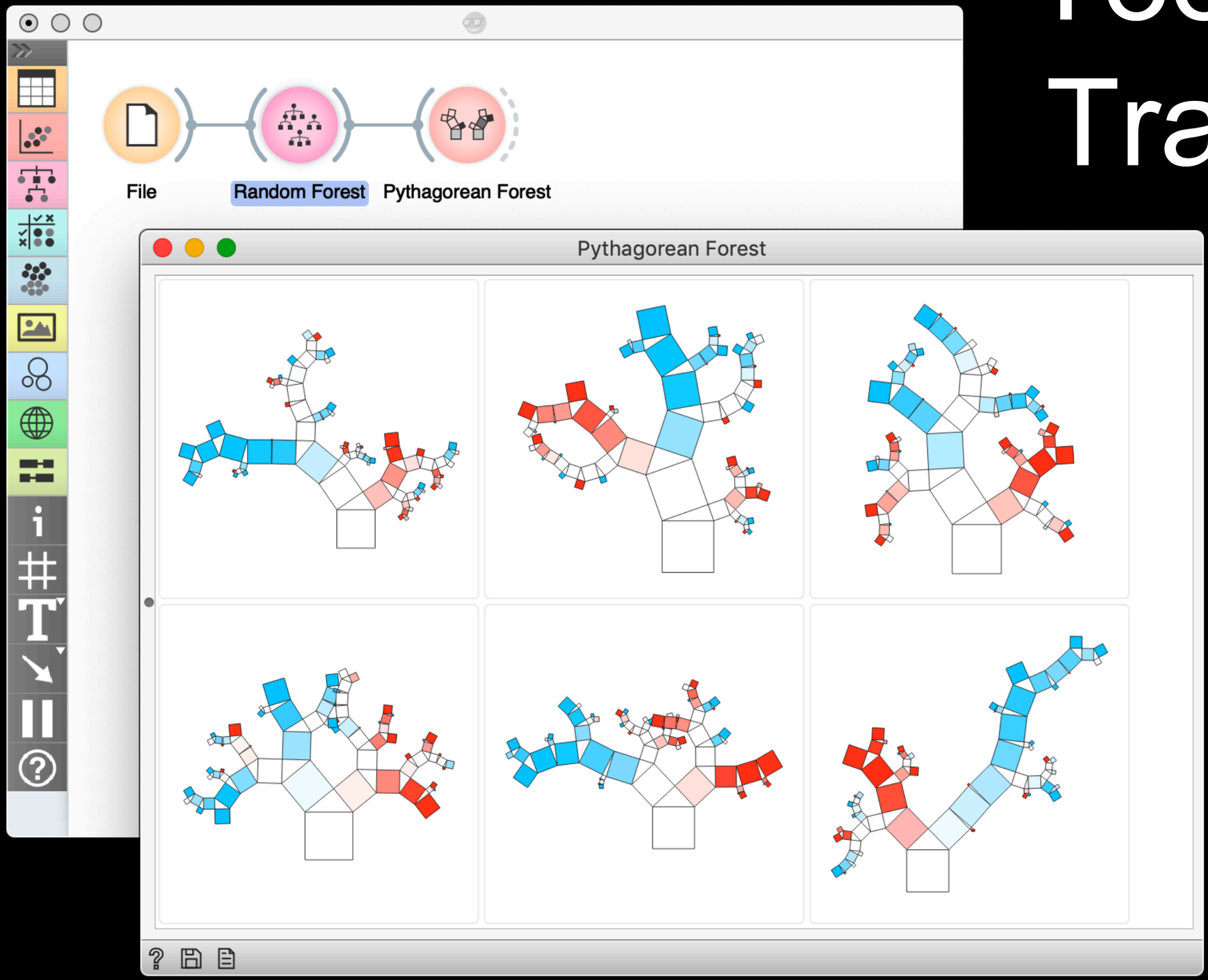






Tools for Training

Tools for Training



Dashboards

Interactive Workflows

Python, R

Simplicity

Flexibility

Training

Data Science Workshop for Public Administration of Slovenia, 2018 & 2019.

Houston

Kolkata

Pavia

youtube.com

YouTube SI

Search

Getting Started with Orange 01: Welcome to Orange

271,544 views • Dec 21, 2015

790 11 SHARE SAVE

Orange Data Mining
12K subscribers

SUBSCRIBED

Open Problems

Tech

Response Time & Interactivity

Servers-Side Compute & Visual Analytics

Data Fusion

Human

Simplicity vs. Flexibility

Gamification

Interpretation vs. Explanation

biolab

Andrej Čopar, GPUs, data fusion

Niko Colnerič, text mining

Tomaž Curk, bioinformatics

Janez Demšar, core development, visualization

Aleš Erjavec, lead developer

Tomaž Hočevar, algorithms, speed-up

Primož Godec, image analytics, text mining

Pavlin Poličar, clustering, embedding, speed-up

Ajda Pretnar, communication, digital humanities

Martin Stražar, bioinformatics

Vesna Tanko, development, testing

Marko Toplak, spectral analysis, bioinformatics

Robert Cvitkovič, bug fixes

Blaž Zupan, enjoys

Lan Žagar, code review, testing

Marinka Žitnik, method expert

external

Nejc Ilenič, deep learning, cloud

Nejc Debevc, web

Jaka Kokošar, bioinformatics

Matjaž Pančur, cloud services

Fabio Ricciato, telecommunications

Veljko Pejović, telecommunications

Anže Starič, software development

support

ARRS Slovenia, NIH USA, MVZT Slovenia

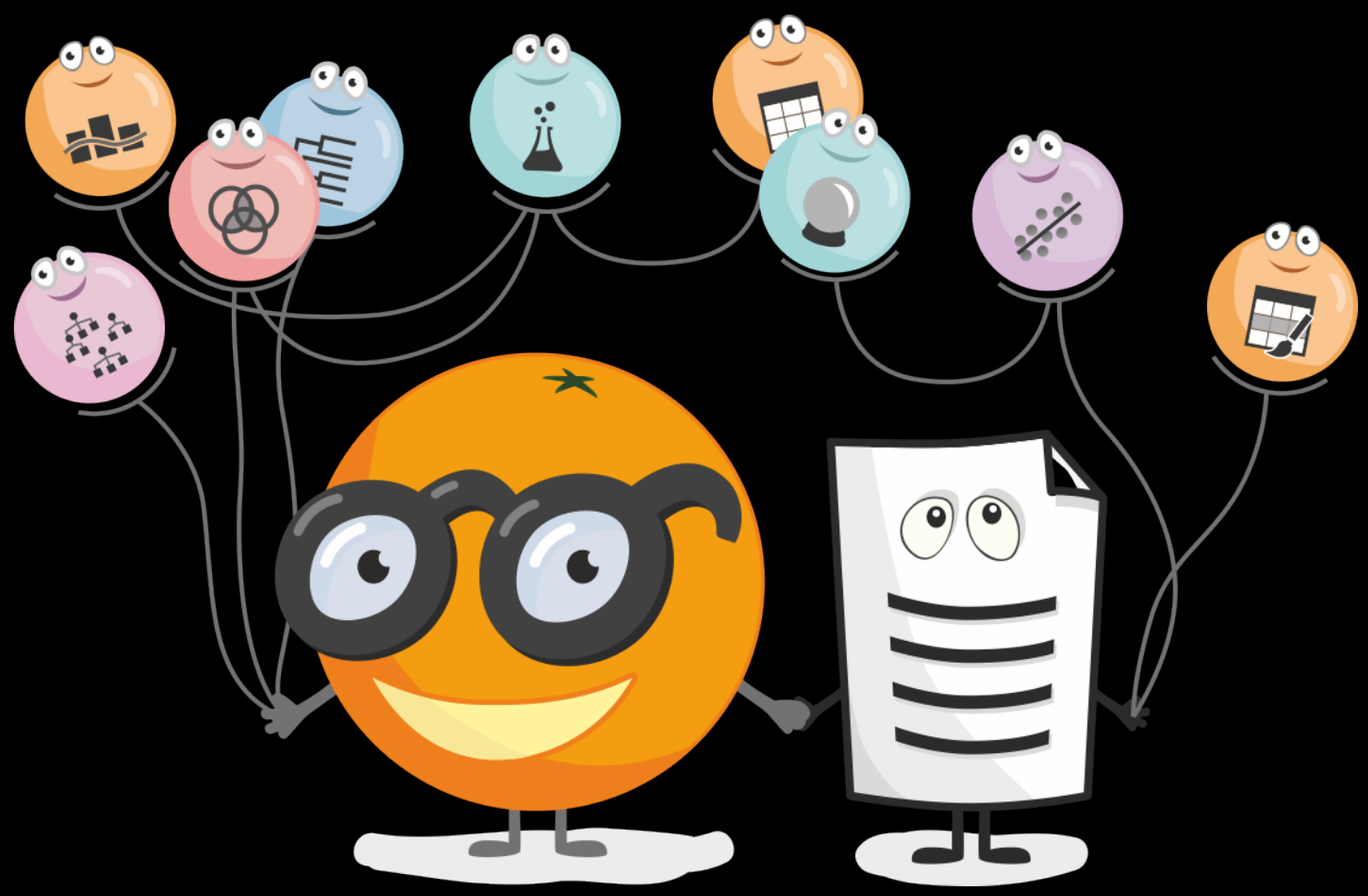
EU FP7 & Horizon 2020

Astra Zeneca, Lek, Elletra, Soleil

Fulbright

Google Summer of Code

NumFOCUS



orange.biolab.si