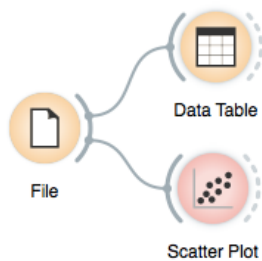
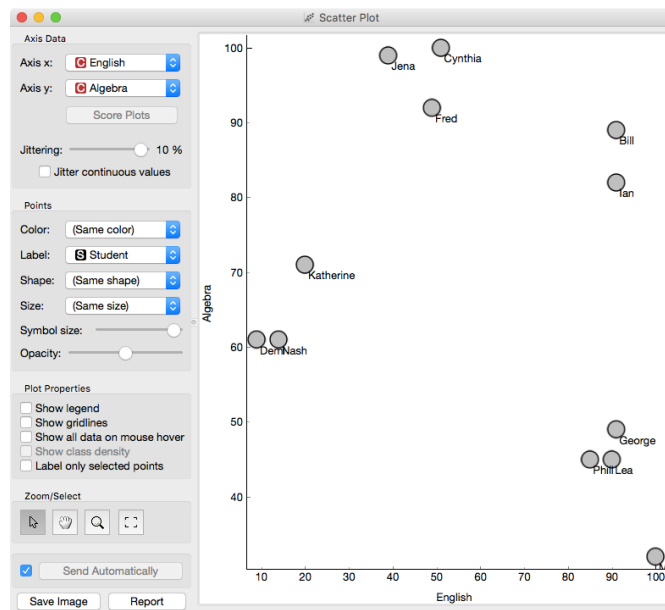


In the class, we will introduce clustering using a simple data set on students and their grades in English and Algebra. Load the data set from <http://file.biolab.si/files/grades2.tab>.



Student	English	Algebra
1 Bill	91.000	89.000
2 Cynthia	51.000	100.000
3 Demi	9.000	61.000
4 Fred	49.000	92.000
5 George	91.000	49.000
6 Ian	91.000	82.000
7 Jena	39.000	99.000
8 Katherine	20.000	71.000
9 Lea	90.000	45.000
10 Maya	100.000	32.000
11 Nash	14.000	61.000
12 Phill	85.000	45.000



How do we measure the similarity between clusters if we only know the similarities between points? By default, Orange computes the average distance between all their pairs of data points; this is called average linkage. We could instead take the distance between the two closest points in each cluster (single linkage), or the two points that are furthest away (complete linkage).

Lesson 23: Hierarchical Clustering

Say that we are interested in finding clusters in the data. That is, we would like to identify groups of data instances that are close together, similar to each other. Consider a simple, two-featured data set (see the side note) and plot it in the Scatter Plot. How many clusters do we have? What defines a cluster? Which data instances belong to the same cluster? What would a procedure for discovering clusters look like?

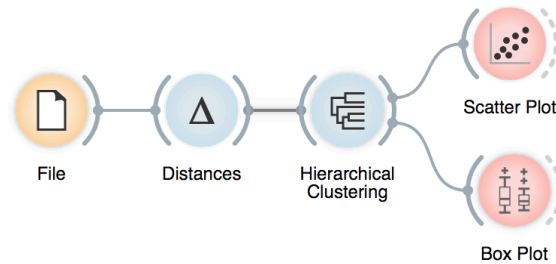
We need to start with a definition of “similar”. One simple measure of similarity for such data is the Euclidean distance: square the differences across every dimension, sum them and take the square root, just like in Pythagorean theorem. So, we would like to group data instances with small Euclidean distances.

Now we need to define a clustering algorithm. We will start with each data instance being in its own cluster. Next, we merge the clusters that are closest together - like the closest two points - into one cluster. Repeat. And repeat. And repeat. And repeat until you end up with a single cluster containing all points.

This procedure constructs a hierarchy of clusters, which explains why we call it hierarchical clustering. After it is done, we can

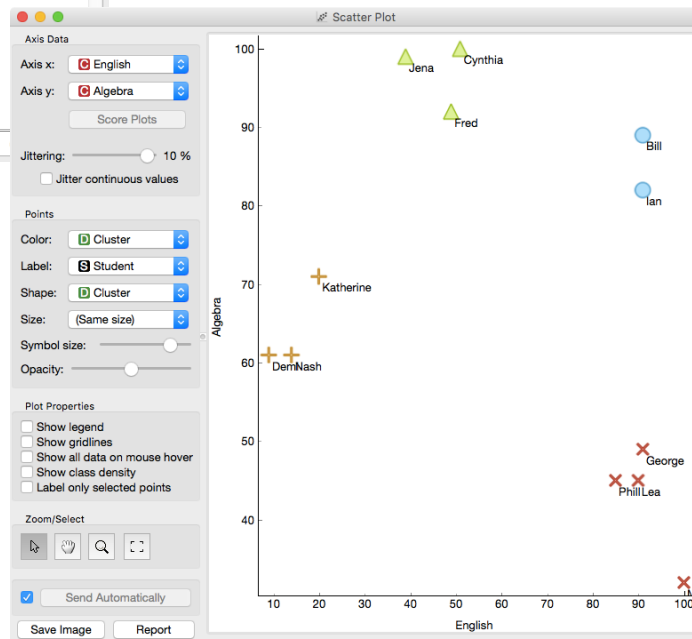
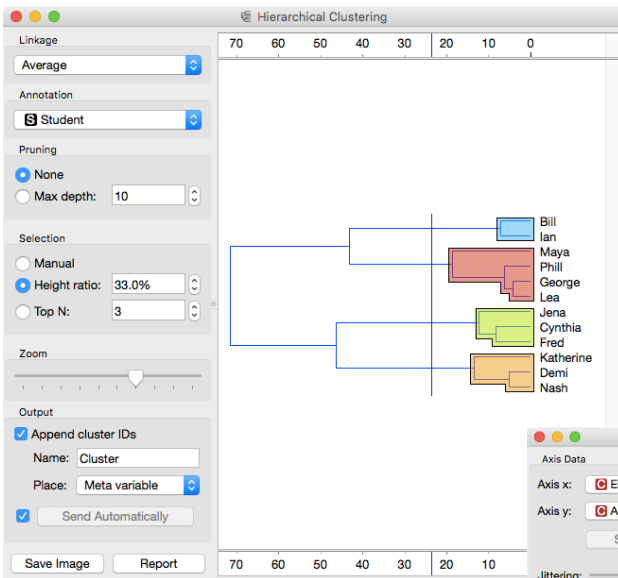
observe the entire hierarchy and decide which would be a good point to stop. With this we decide the actual number of clusters.

One possible way to observe the results of clustering on our small data set with grades is through the following workflow:



Let us see how this works. Load the data, compute the distances and cluster the data. In the Hierarchical clustering widget, cut hierarchy at a certain distance score and observe the corresponding clusters in the Scatter plot.

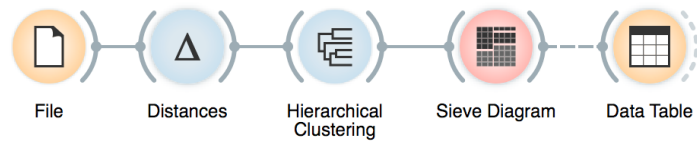
You can also observe the properties of the clusters - that is, the average grades in Algebra and English - in the box plot.



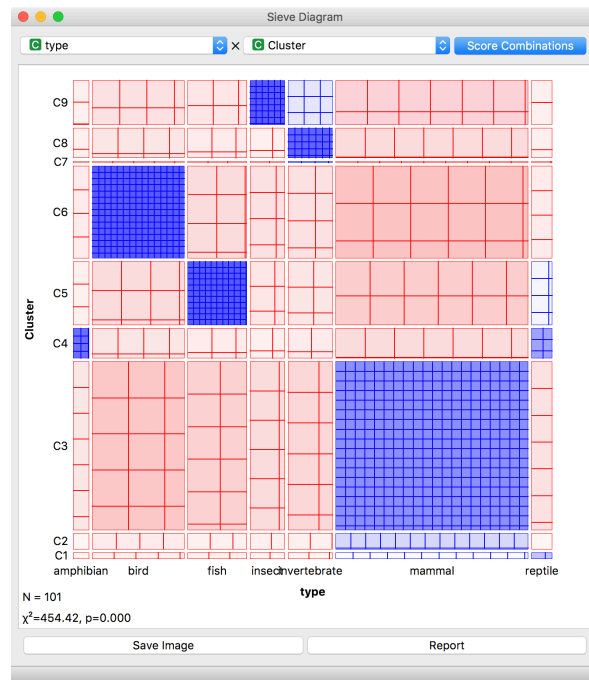
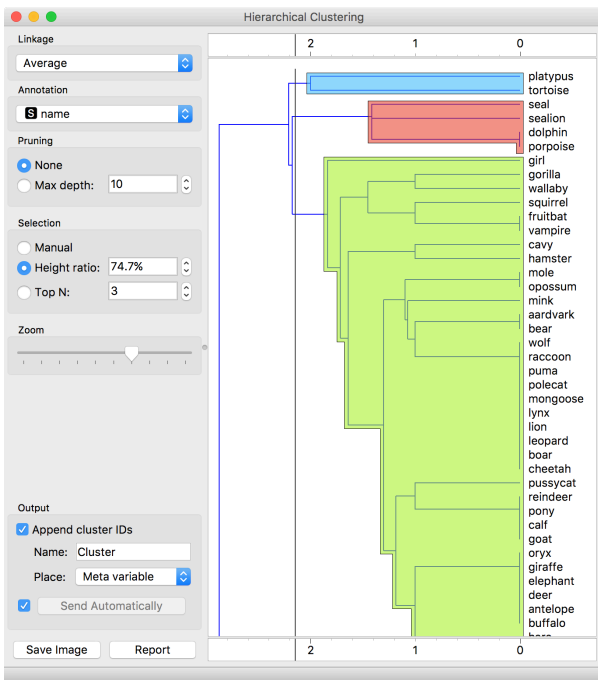
Lesson 24: Animal Kingdom



Your lecturers spent substantial part of their youth admiring a particular Croatian chocolate called Animal Kingdom. Each chocolate bar came with a card — a drawing of some (random) animal, and the associated album made us eat a lot of chocolate. Then our kids came, and the story repeated. Some things stay forever. Funny stuff was we never understood the order in which the cards were laid out in the album. We later learned about taxonomy, but being more inclined to engineering we never mastered learning it in our biology classes. Luckily, there's data mining and the idea that taxonomy simply stems from measuring the distance between species.

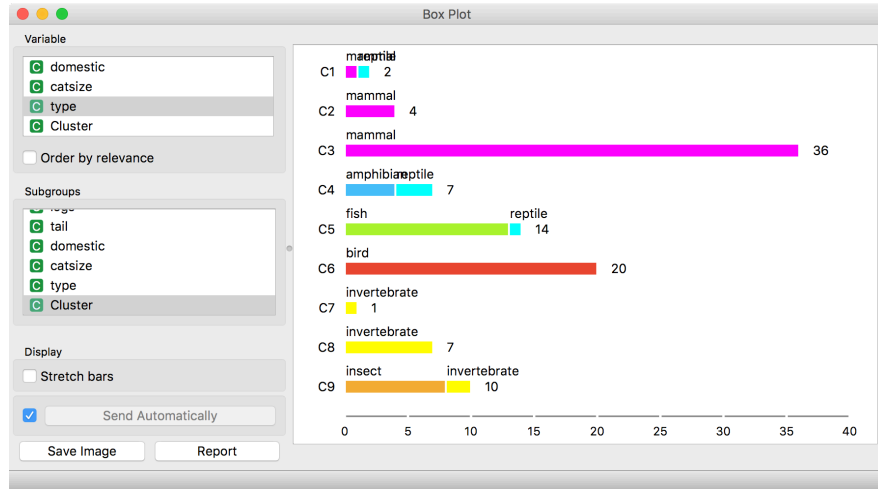


Here we use zoo data (from documentation data sets) with attributes that report on various features of animals (has hair, has feathers, lays eggs). We measure the distance and compute the clustering. Animals in this data set are annotated with type (mammal, insect, bird, and so on). It would be cool to know if the clustering re-discovered these groups of animals. We can do this through marking the clusters in Hierarchical Clustering widget, and then observing the results in the Sieve Diagram.

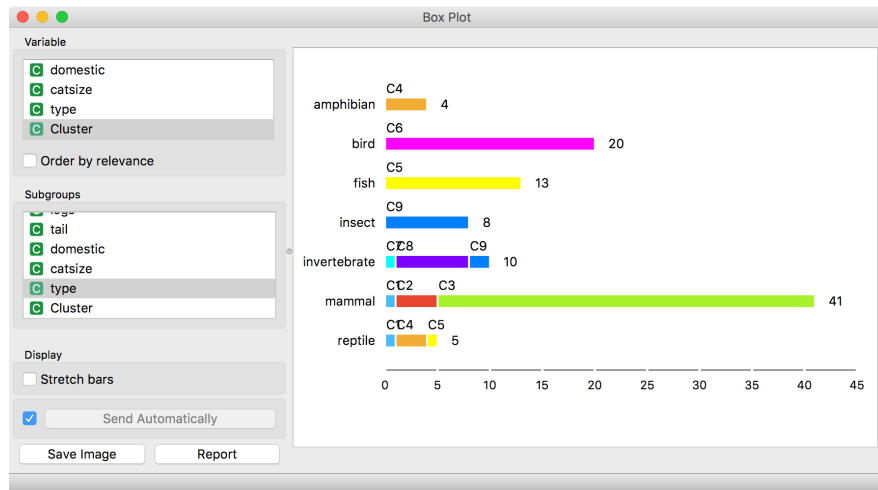


Looks great. Birds, say, are in cluster C6. Cluster C4 consists of amphibians and some reptiles. And so forth.

Checking this in the Box plot is even cooler. We can get a distribution of animal types in each cluster:



Or we can turn it around and see how different types of animals are spread across clusters.



What is wrong with those mammals? Why can't they be in one single cluster? Two reasons. First, they represent 40 % of the data instances. Second, they include some weirdos. Click on the clusters in the box plot and discover who they are.

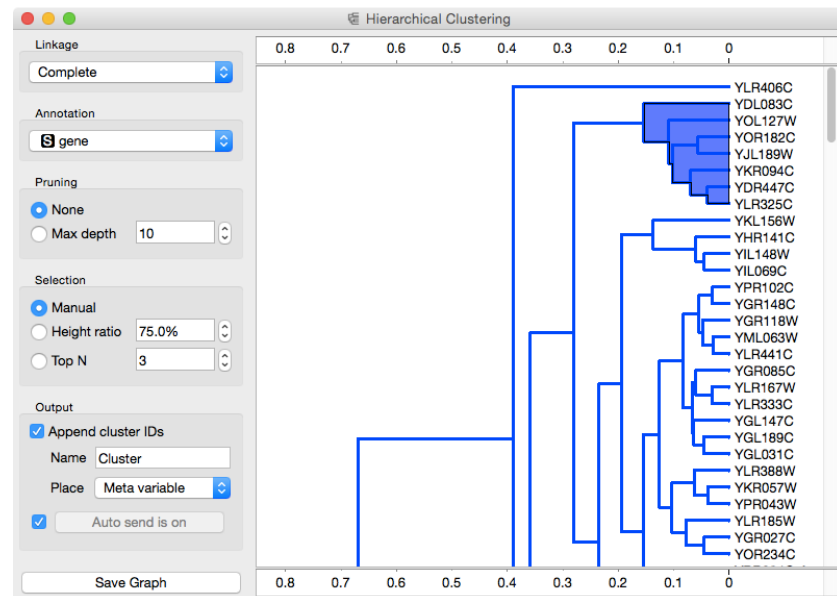
Lesson 25: Discovering clusters

Can we replicate this on some real data? Can clustering indeed be useful for defining meaningful subgroups?

Take brown-selected (from documentation data sets) connect the hierarchical clustering so the you can see a cluster as a subset in the scatterplot.



So far, we used the dendrogram to set a cut-off point. Now we will click on a branch in a dendrogram to select a subset of the data instances. By combining it with the Scatter Plot widget, we get a great tool for exploring the clusters. Try it with an appropriate pair of features to visualize (use Rank projections).



By using a scatter plot or other widgets, an expert can determine whether the clusters are meaningful.

For this data set, though, we can do something even better. The data already contains some predefined groups. Let us check how

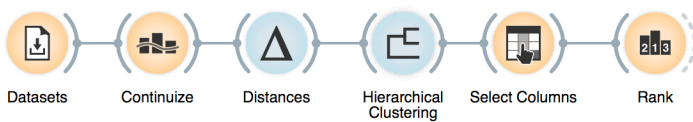
well the clusters match the classes - which we know, but clustering did not.

We will use the dendrogram to set a suitable threshold that splits the data into some three to five clusters. We can plot this data in a new scatter plot; we find a reasonable pair of attributes and then set the color of the points to represent the cluster they belong to. Do the clusters match the actual classes? The result is rather impressive if you keep two things in mind. First, the clustering algorithm did not actually know about the classes, it discovered them by itself. Second, it did not operate on the picture you see in the scatter plot and in which the clusters are quite pronounced, but in a 79-dimensional data space with possibly plenty of redundant features. Yet it identified the three groups of genes almost without mistakes.

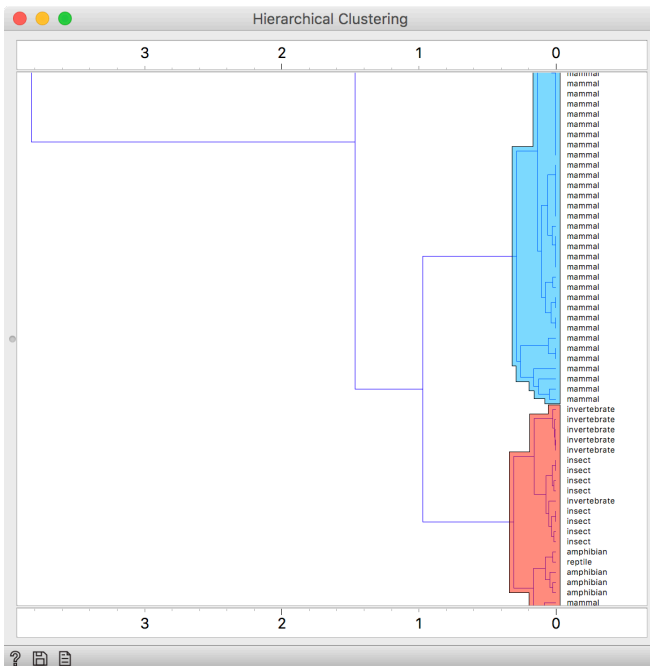
This lesson is not a recipe for what you should be doing in practice. If your data already contains group labels, say gene group annotations, there is no need to discover them (again) by using clustering. In this case you should be interested in predictive models from previous lessons. If you do not have such a grouping but you suspect that the data contains distinct subgroups, run clustering. The sole purpose of this lesson was to demonstrate that clustering can indeed find a meaningful subgroups in the data; we pretend we did not know the groups, use the clustering to discover them, and checked how well they correspond to the actual groups.

Lesson 26: Cluster interpretation

Once we have inferred the clusters, we would like to know what are the distinguishing features. For the zoo data set, we could, for instance, mark two clusters, and then ask for the features that distinguish among these. Having data marked with cluster identifiers takes us back to classification, and we can use any of visualization, model inference, or feature ranking techniques we have introduced there. Here, we will show how to use ranking to infer what features characterize the group of mammals when compared to a close cluster of other species.

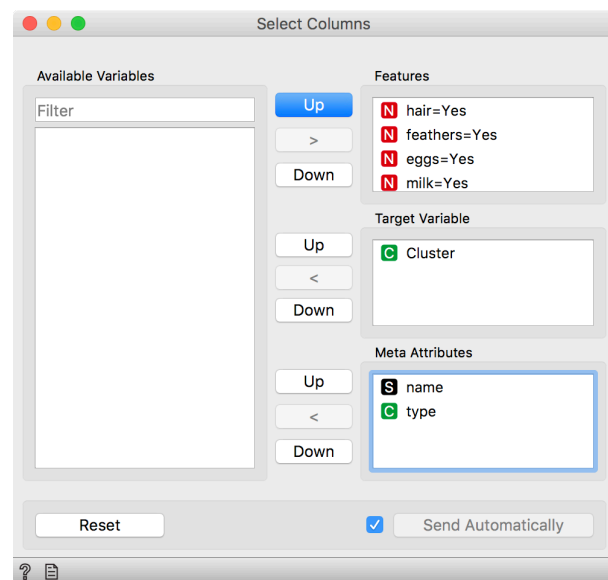


We load zoo data from Datasets, continuize the categorical features (this will be changed in Orange soon, Distances should automatically perform continuization), estimate the distances, and feed everything in Hierarchical Clustering. So far, nothing new.

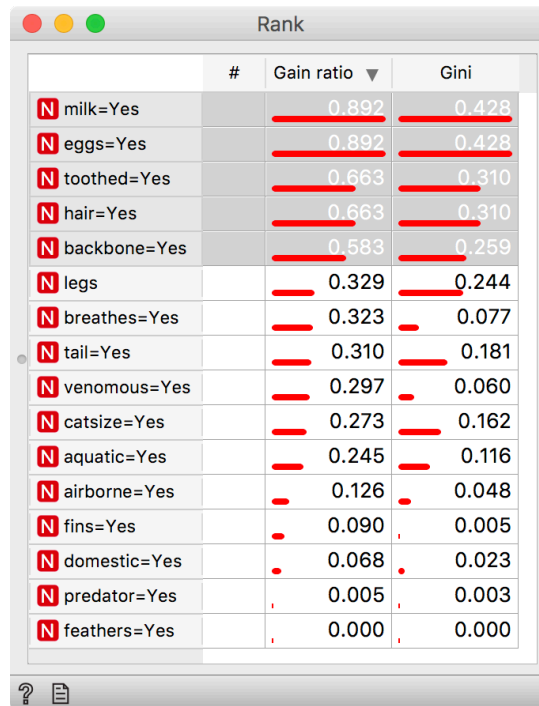


In the hierarchical clustering, we choose two clusters. Note that Hierarchical Clustering adds cluster identifier as a meta feature; to make the data ready for classification-specific tasks, we need to promote cluster identifier into target variable (a class) by reassigning the feature types in Select Columns.

Use modifier keys (command) to select different branches of the dendrogram and mark them as separate clusters. Done correctly, Hierarchical Clustering will mark the branches with different colors.



The data is now ready for classification-based analysis. Here, we used a rank widget.



	#	Gain ratio ▼	Gini
N milk=Yes		0.892	0.428
N eggs=Yes		0.892	0.428
N toothed=Yes		0.663	0.310
N hair=Yes		0.663	0.310
N backbone=Yes		0.583	0.259
N legs		0.329	0.244
N breathes=Yes		0.323	0.077
N tail=Yes		0.310	0.181
N venomous=Yes		0.297	0.060
N catsize=Yes		0.273	0.162
N aquatic=Yes		0.245	0.116
N airborne=Yes		0.126	0.048
N fins=Yes		0.090	0.005
N domestic=Yes		0.068	0.023
N predator=Yes		0.005	0.003
N feathers=Yes		0.000	0.000