**Visualizing hierarchies**

A huge part of your work as a data scientist will be the communication of your insights to other people.

**Visualizations communicate insight**

Visualizations are an excellent way to share your findings, particularly with a non-technical audience. In this chapter, you'll learn about two unsupervised learning techniques for visualization: t-SNE and hierarchical clustering. t-SNE, which we'll consider later, creates a 2d map of any dataset, and conveys useful information about the proximity of the samples to one another. First up, however, let's learn about hierarchical clustering.

**A hierarchy of groups**

You've already seen many hierarchical clusterings in the real world. For example, living things can be organized into small narrow groups, like humans, apes, snakes and lizards, or into larger, broader groups like mammals and reptiles, or even broader groups like animals and plants. These groups are contained in one another, and form a hierarchy. Analogously, hierarchical clustering arranges samples into a hierarchy of clusters.

**Eurovision scoring dataset**

Hierarchical clustering can organize any sort of data into a hierarchy, not just samples of plants and animals. Let's consider a new type of dataset, describing how countries scored performances at the Eurovision 2016 song contest. The data is arranged in a rectangular array, where the rows of the array show how many points a country gave to each song. The "samples" in this case are the countries.

1. 1 https://www.eurovision.tv/page/results

**Hierarchical clustering of voting countries**

The result of applying hierarchical clustering to the Eurovision scores can be visualized as a tree-like diagram called a "dendrogram". This single picture reveals a great deal of information about the voting behavior of countries at the Eurovision. The dendrogram groups the countries into larger and larger clusters, and many of these clusters are immediately recognizable as containing countries that are close to one another geographically, or that have close cultural or political ties, or that belong to single language group. So hierarchical clustering can produce great visualizations. But how does it work?

**Hierarchical clustering**

Hierarchical clustering proceeds in steps. In the beginning, every country is its own cluster - so there are as many clusters as there are countries! At each step, the two closest clusters are merged. This decreases the number of clusters, and eventually, there is only one cluster left, and it contains all the countries. This process is actually a particular type of hierarchical clustering called "agglomerative clustering" - there is also "divisive clustering", which works the other way around. We haven't defined yet what it means for two clusters to be close, but we'll revisit that later on.

**The dendrogram of a hierarchical clustering**

The entire process of the hierarchical clustering is encoded in the dendrogram. At the bottom, each country is in a cluster of its own. The clustering then proceeds from the bottom up. Clusters are represented as vertical lines, and a joining of vertical lines indicates a merging of clusters. To understand better, let's zoom in and look at just one part of this dendrogram.

**Dendrograms, step-by-step**

In the beginning, there are six clusters, each containing only one country.

The first merging is here, where the clusters containing Cyprus and Greece are merged together in a single cluster.

Later on, this new cluster is merged with the cluster containing Bulgaria.

**Dendrograms, step-by-step**

Shortly after that, the clusters containing Moldova and Russia are merged,

which later is in turn merged with the cluster containing Armenia. Later still, the two big composite clusters are merged together. This process continues until there is only one cluster left, and it contains all the countries.

**Hierarchical clustering with SciPy**

We'll use functions from scipy to perform a hierarchical clustering on the array of scores. For the dendrogram, we'll also need a list of country names. Firstly, import the linkage and dendrogram functions. Then, apply the linkage function to the sample array. Its the linkage function that performs the hierarchical clustering. Notice there is an extra method parameter - we'll cover that in the next video. Now pass the output of linkage to the dendrogram function, specifying the list of country names as the labels parameter. In the next video, you'll learn how to extract information from a hierarchical clustering,

**Cluster labels in hierarchical clustering**

In the previous video, we employed hierarchical clustering to create a great visualization of the voting behavior at the Eurovision. But hierarchical clustering is not only a visualization tool. In this video, you'll learn how to extract the clusters from intermediate stages of a hierarchical clustering. The cluster labels for these intermediate clusterings can then be used in further computations, such as cross tabulations, just like the cluster labels from k-means.

**3. Intermediate clusterings & height on dendrogram**

An intermediate stage in the hierarchical clustering is specified by choosing a height on the dendrogram. For example, choosing a height of 15 defines a clustering in which Bulgaria, Cyprus and Greece are in one cluster, Russia and Moldova are in another, and Armenia is in a cluster on its own. But what is the meaning of the height?

**4. Dendrograms show cluster distances**

The y-axis of the dendrogram encodes the distance between merging clusters. For example, the distance between the cluster containing Cyprus and the cluster containing Greece was approximately 6 when they were merged into a single cluster.

**5. Dendrograms show cluster distances**

When this new cluster was merged with the cluster containing Bulgaria, the distance between them was 12.

**6. Intermediate clusterings & height on dendrogram**

So the height that specifies an intermediate clustering corresponds to a distance. This specifies that the hierarchical clustering should stop merging clusters when all clusters are at least this far apart.

**7. Distance between clusters**

The distance between two clusters is measured using a "linkage method". In our example, we used "complete" linkage, where the distance between two clusters is the maximum of the distances between their samples. This was specified via the "method" parameter. There are many other linkage methods, and you'll see in the exercises that different linkage methods give different hierarchical clusterings!

**8. Extracting cluster labels**

The cluster labels for any intermediate stage of the hierarchical clustering can be extracted using the fcluster function. Let's try it out, specifying the height of 15.

**9. Extracting cluster labels using fcluster**

After performing the hierarchical clustering of the Eurovision data, import the fcluster function. Then pass the result of the linkage function to the fcluster function, specifying the height as the second argument. This returns a numpy array containing the cluster labels for all the countries.

**10. Aligning cluster labels with country names**

To inspect cluster labels, let's use a DataFrame to align the labels with the country names. Firstly, import pandas, then create the data frame, and then sort by cluster label, printing the result. As expected, the cluster labels group Bulgaria, Greece and Cyprus in the same cluster. But do note that the scipy cluster labels start at 1, not at 0 like they do in scikit-learn.