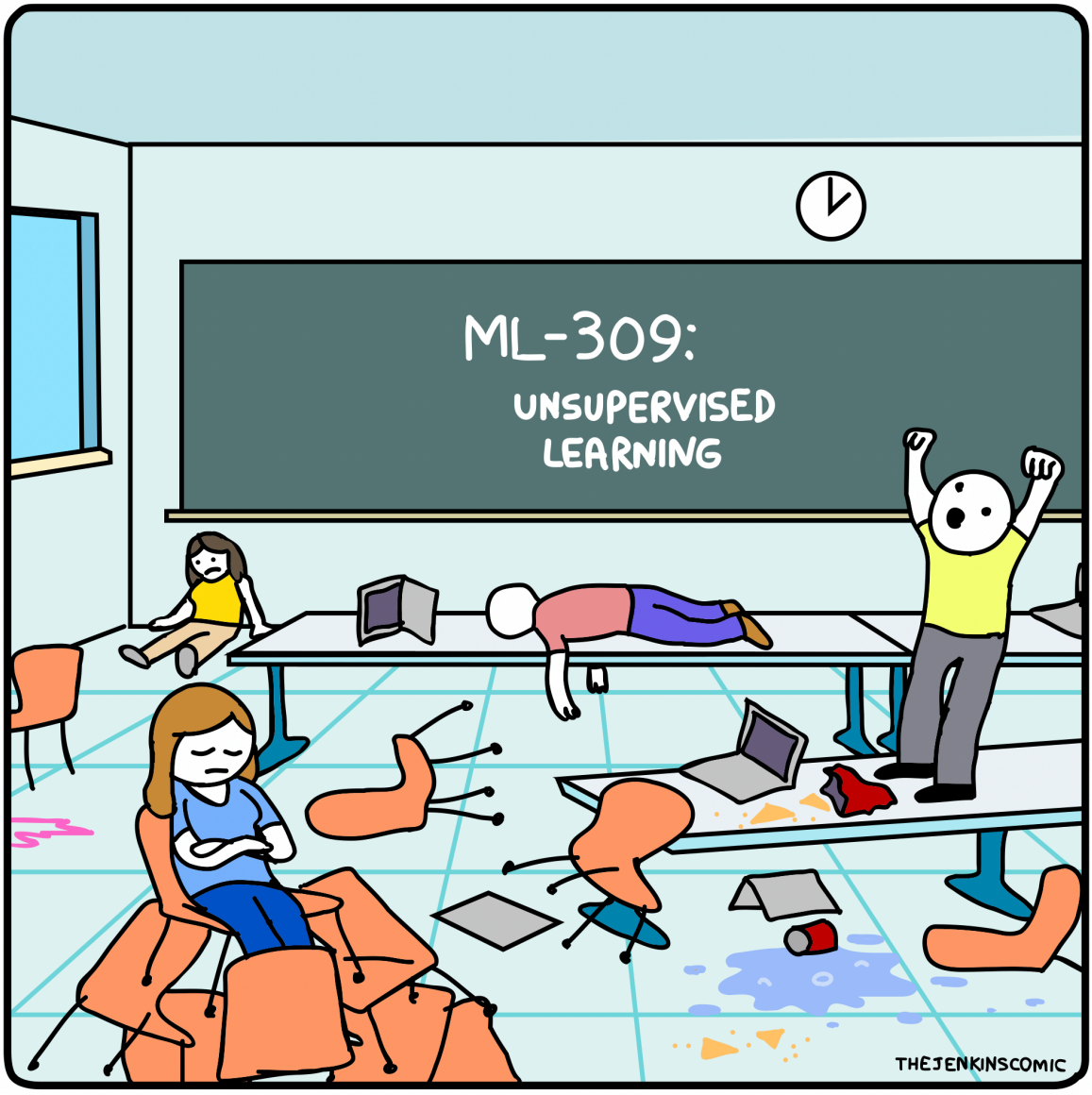
Chapter 10: Unsupervised Learning



Credit: https://thejenkinscomic.net/?id=366

This chapter will focus on methods intended for the setting in which we only have a set of features $X_1, \ldots, X_p$ measured on n observations.

1 The Challenge of Unsupervised Learning

Supervised learning is a well-understood area.

In contrast, unsupervised learning is often much more challenging.

Unsupervised learning is often performed as part of an *exploratory data analysis*.

It can be hard to assess the results obtained from unsupervised learning methods.

Techniques for unsupervised learning are of growing importance in a number of fields.

2 Principal Components Analysis

We have already seen principal components as a method for dimension reduction.

Principal Components Analysis (PCA) refers to the process by which principal components are computed and the subsequent use of these components to understand the data.

Apart from producing derived variables forr use in supervised learning, PCA also serves as a tool for data visualization.

2.1 What are Principal Components?

Suppose we wish to visualize n observations with measurements on a set of p features as part of an exploratory data analysis.

Goal: We would like to find a low-dimensional representation of the data that captures as much of the information as possible.

PCA provides us a tool to do just this.

Idea: Each of the n observations lives in p dimensional space, but not all of these dimensions are equally interesting.

The *first principal component* of a set of features $X_1, \ldots, X_p$ is the normalized linear combination of the features

that has the largest variance.

Given a $n \times p$ data set $\boldsymbol{X}$, how do we compute the first principal component?

There is a nice geometric interpretation for the first principal component.

After the first principal component Z_1 of the features has been determined, we can find the second principal component, Z_2. The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance out of all linear combinations that are uncorrelated with Z_1.

Once we have computed the principal components, we can plot them against each other to produce low-dimensional views of the data.

```
str(USArrests)
```

```
## 'data.frame':    50 obs. of  4 variables:
##  $ Murder  : num  13.2 10 8.1 8.8 9 7.9 3.3 5.9 15.4 17.4 ...
##  $ Assault : int  236 263 294 190 276 204 110 238 335 211 ...
##  $ UrbanPop: int  58 48 80 50 91 78 77 72 80 60 ...
##  $ Rape    : num  21.2 44.5 31 19.5 40.6 38.7 11.1 15.8 31.9 25.8
...
```

```
USArrests_pca <- USArrests |>
  prcomp(scale = TRUE, center = TRUE)

summary(USArrests_pca) # summary
```

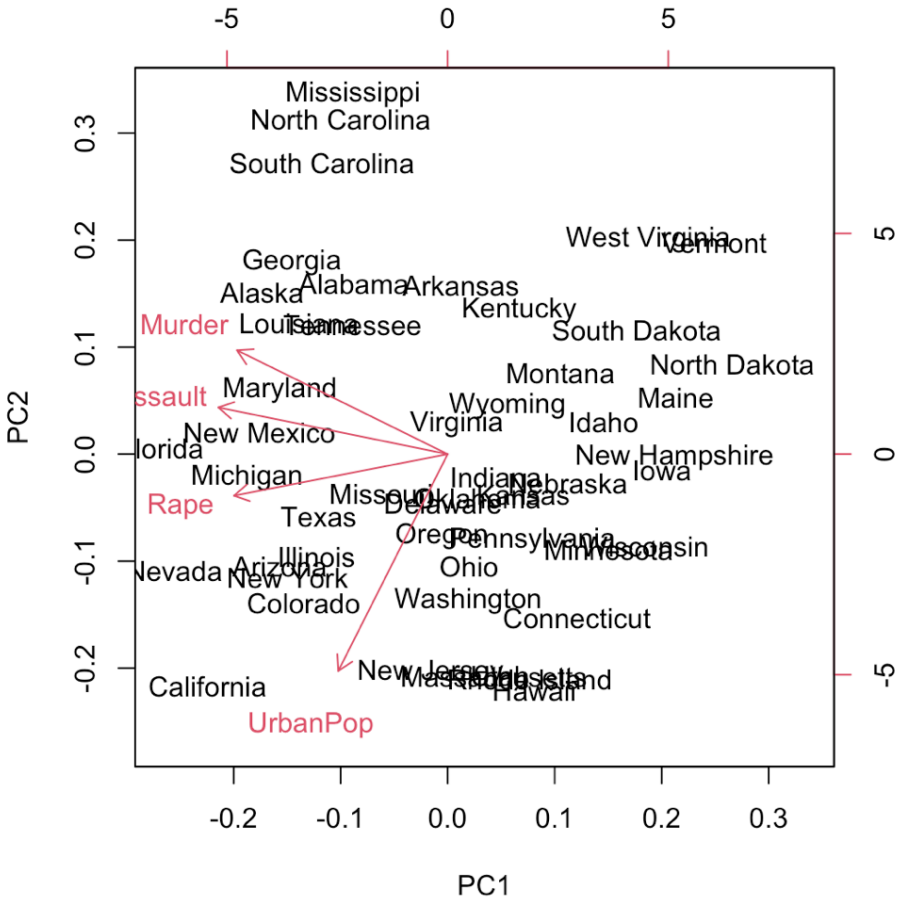
```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.5749 0.9949 0.59713 0.41645
## Proportion of Variance 0.6201 0.2474 0.08914 0.04336
## Cumulative Proportion  0.6201 0.8675 0.95664 1.00000
```

```
tidy(USArrests_pca, matrix = "loadings") |> # principal components
        loading matrix
  pivot_wider(names_from = PC, values_from = value)
```

```
## # A tibble: 4 × 5
##   column      `1`    `2`    `3`     `4`
##   <chr>     <dbl>  <dbl>  <dbl>   <dbl>
## 1 Murder   -0.536  0.418 -0.341  0.649
## 2 Assault  -0.583  0.188 -0.268 -0.743
## 3 UrbanPop -0.278 -0.873 -0.378  0.134
## 4 Rape     -0.543 -0.167  0.818  0.0890
```

```
## plot scores + directions
```

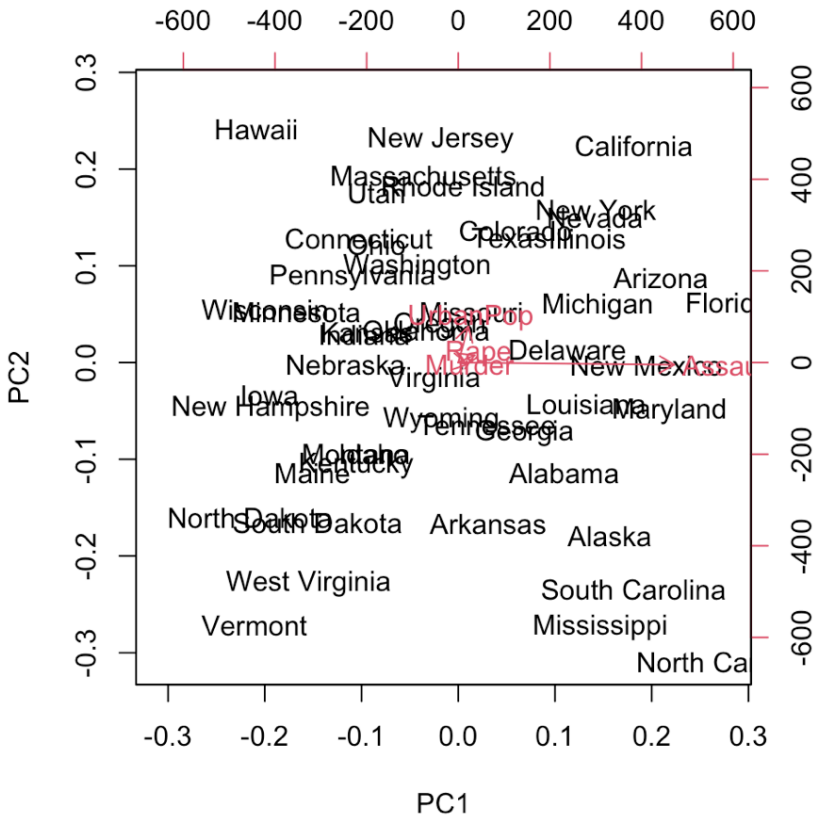
```
biplot(USArrests_pca)
```



2.2 Scaling Variables

We've already talked about how when PCA is performed, the varriables should be centered to have mean zero.

This is in contrast to other methods we've seen before.



2.3 Uniqueness

Each principal component loading vector is unique, up to a sign flip.

Similarly, the score vectors are unique up to a sign flip.

2.4 Proportion of Variance Explained

We have seen using the `USArrests` data that e can summarize 50 observations in 4 dimensions using just the first two principal component score vectors and the first two principal component vectors.

Question:

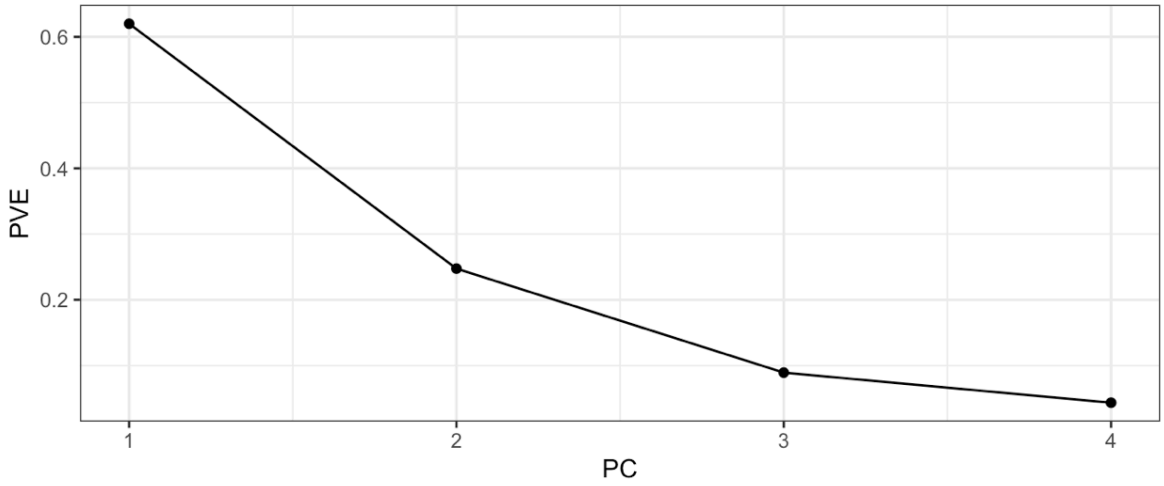
More generally, we are interested in knowing the *proportion of vriance explained (PVE)* by each principal component.

2.5 How Many Principal Components to Use

In general, a $n \times p$ matrix $\boldsymbol{X}$ has $\min(n-1, p)$ distinct principal components.

Rather, we would like to just use the first few principal components in order to visualize or interpret the data.

We typically decide on the number of principal components required by examining a *scree plot*.



2.6 Other Uses for Principal Components

We've seen previously that we can perform regression using the principal component score vectors as features for dimension reduction.

Many statistical techniques can be easily adapted to use the $n \times M$ matrix whose columns are the first $M << p$ principal components.

This can lead to *less noisy* results.

3 Clustering

Clustering refers to a broad set of techniques for finding *subgroups* in a data set.

For instance, suppose we have a set of n observations, each with p features. The n observations could correspond to tissue samples for patients with breast cancer and the p features could correspond to

We may have reason to believe there is heterogeneity among the n observations.

This is *unsupervised* because

Both clustering and PCA seek to simplify the data via a small number of summaries.

- PCA

- Clustering

Since clustering is popular in many fields, there are many ways to cluster.

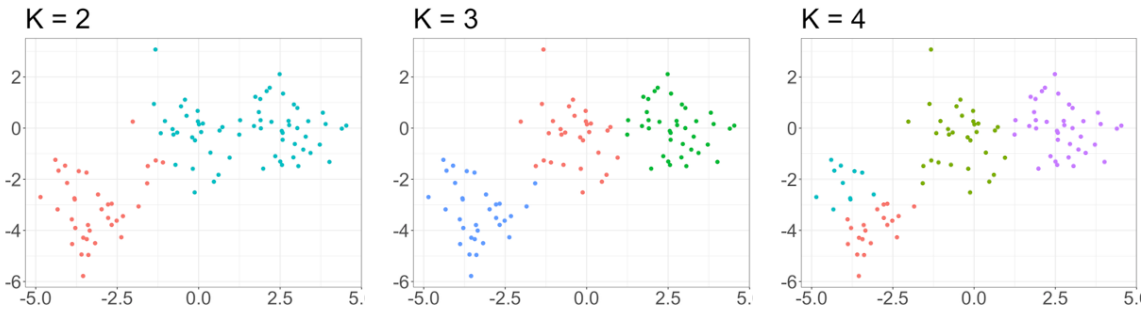
- K-means clustering

- Hierarchical clustering

In general, we can cluster observations on the basis of features or we can cluster features on the basis of observations.

3.1 K-Means Clustering

Simple and elegant approach to parition a data set into K distinct, non-overlapping clusters.



The K-means clustering procedure results from a simple and intuitive mathematical problem. Let $C_1, \ldots, C_K$ denote sets containing the indices of observations in each cluster. These satisfy two properties:

1.

2.

Idea:

The *within-cluster variation* for cluster C_k is a measure of the amount by which the observations within a cluster differ from each other.

To solve this, we need to define within-cluster variation.

This results in the following optimization problem that defines K-means clustering:

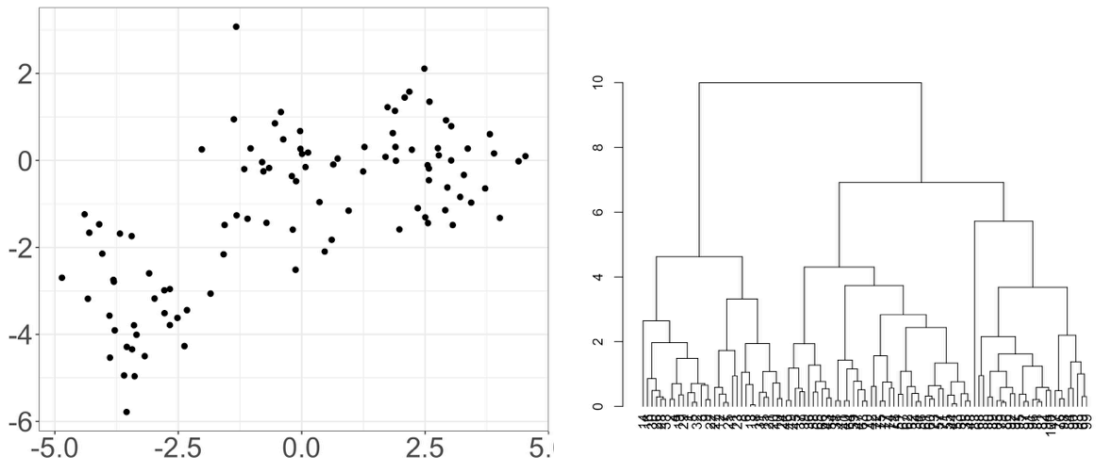
A very simple algorithm has been shown to find a local optimum to this problem:

3.2 Hierarchical Clustering

One potential disadvantage of K-means clustering is that it requires us to specify the number of clusters K. *Hierarchical clustering* is an alternative that does not require we commit to a particular K.

We will discuss *bottom-up* or *agglomerative* clustering.

3.2.1 Dendrograms

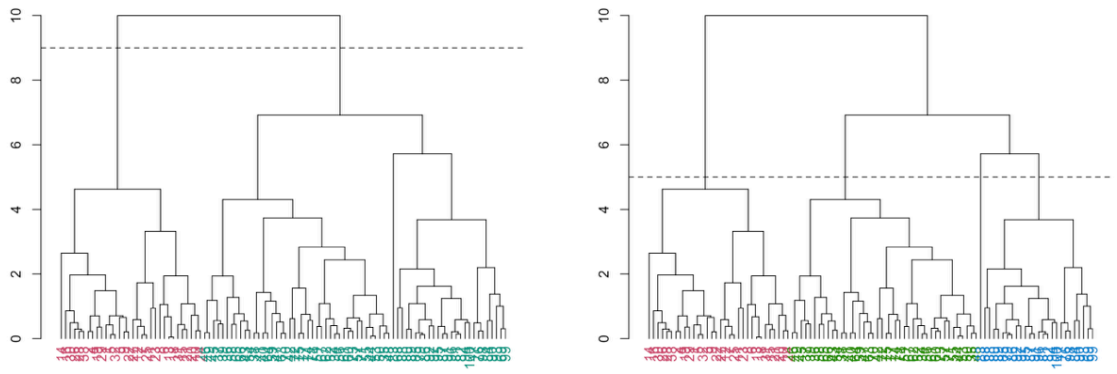


Each *leaf* of the dendrogram represents one of the 100 simulated data points.

As we move up the tree, leaves begin to fuse into branches, which correspond to observations that are similar to each other.

For any two observations, we can look for the point in the tree where branches containing those two observations are first fused.

How do we get clusters from the dendrogram?

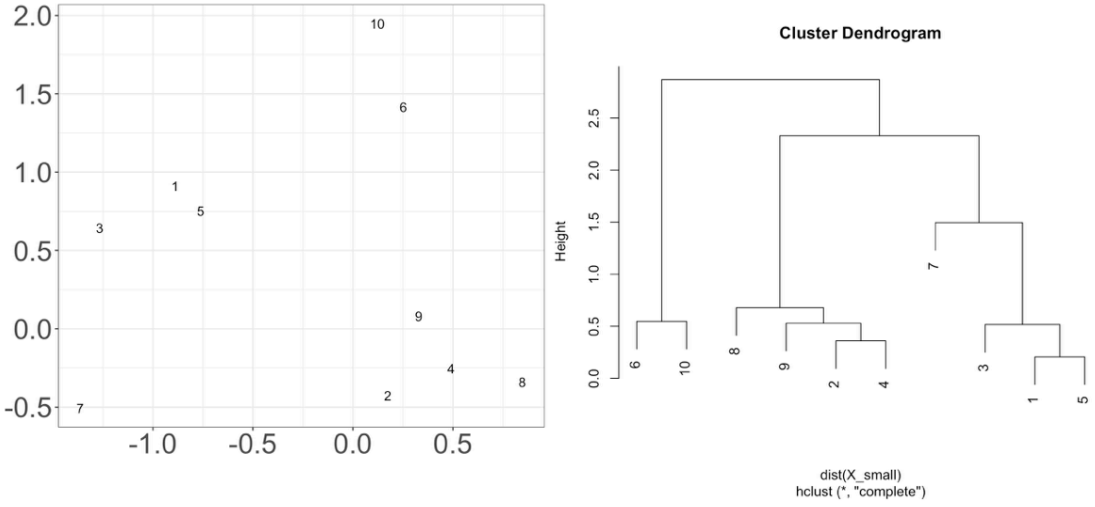


The term *hierarchical* refers to the fact that clusters obtained by cutting the dendrogram at a given height are necessarily nested within the clusters obtained by cutting the dendrogram at a greater height.

3.2.2 Algorithm

First, we need to define some sort of *dissimilarity* metric between pairs of observations.

Then the algorithm proceeds iteratively.



More formally,

One issue has not yet been addressed.

How do we determine the dissimilarity between two clusters if one or both of them contains multiple observations?

1.

2.

3.

4.

3.2.3 Choice of Dissimilarity Metric

3.3 Practical Considerations in Clustering

In order to perform clustering, some decisions should be made.

-
-
-

Each of these decisions can have a strong impact on the results obtained. What to do?