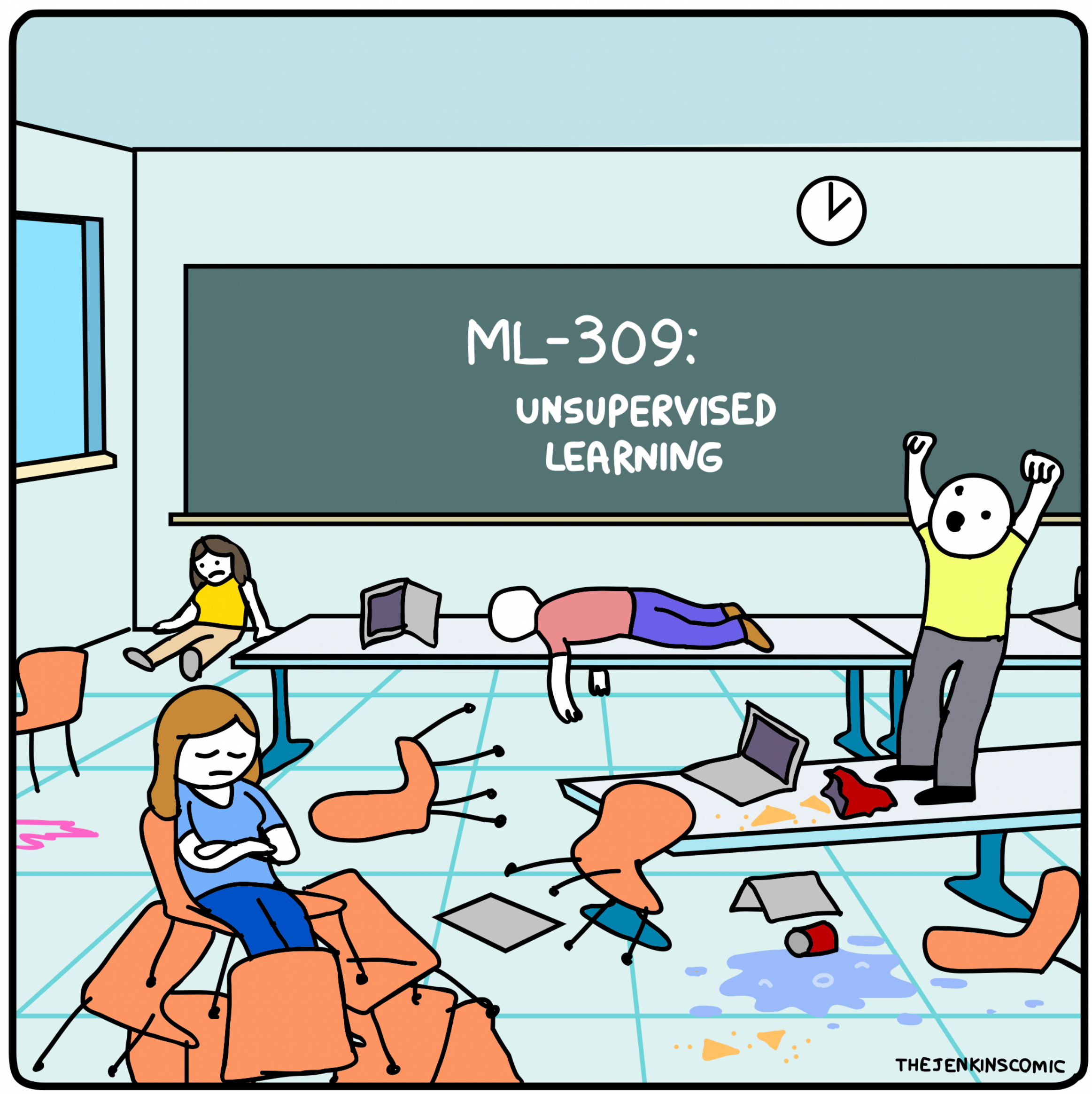
Chapter 10: Unsupervised Learning



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 $X_1, \dots, X_p$ measured on n observations.

We are not interested in prediction because we have no associated response Y

Goal: discover interesting things about the measurements $X_1, \dots, X_p$

- Is there an informative way to plot the data?
- Can we discover subgroups among the variables or the observations?

1 The Challenge of Unsupervised Learning

Supervised learning is a well-understood area.

You now have a good grasp of supervised learning.

If you were asked to predict a binary response you have many well developed tools at your disposal!

logistic regression, SVM, LDA, QDA, RF, boosting, etc.

and a clear understanding of how to assess the quality of your results:

validation on independent test, CV

In contrast, unsupervised learning is often much more challenging.

More subjective, no simple goal for the analysis, e.g. prediction.

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

1st part of analysis, before fitting models.

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

No universally accepted way to perform validation or CV.

Because there is no way for us to "check our work" with no response.

→ we don't know the true answer.

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

Cancer research: assay gene expression levels in 100 patients and look for subgroups among cancer samples to better understand the disease.

Online shopping: identify similar groups of shoppers and show preferential items that they might be particularly interested in.

My research

Entity Resolution: Many noisy databases without unique identifying attribute → can we find the matches?/links?

2 Principal Components Analysis

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

When faced w/ large set of correlated variables, we used to summarize this set w/ a smaller # of representative variables that collectively explain most of the variability in the data set.

directions in feature space which original data are highly variable

↳ define lines and subspaces that are close as possible to data cloud

PCR = use principal components as predictors in a regression model instead of original variables

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.

Unsupervised approach (involves only features $X_1, \ldots, X_p$) no response Y.

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

Visualize observation, or variables.

2.1 What are Principal Components?

$X_1, \ldots, X_p$

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

We could examine $\binom{p}{2}$ 2D scatterplots for all combinations of variables, e.g. $p=10 \Rightarrow 45$ plots!

- Too many to look at
- Likely no plot will be informative because they contain a small fraction of information in the data.

→ visualization in high dimensions.

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

Then plot observations in low-dimensional space.

PCA provides us a tool to do just this.

It finds low-dim respresentation of a data set that contains as much variation as possible.
variation ⇓ "information"

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

PCA seeks a small # of dimensions that are as interesting as possible.
interesting = amount the observations vary along each dimension.

Each dimension found by PCA is a linear combination of the p features.

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

$$Z_1 = \phi_{11} X_1 + \phi_{21} X_2 + \ldots + \phi_{p1} X_p$$

(normalized) $\sum_{j=1}^{p} \phi_{j1}^2 = 1$

$\phi_{11}, \ldots, \phi_{p1}$ are called "loadings" of the first principal component.

$\phi_1 = (\phi_{11}, \ldots, \phi_{p1})^T$ "loading vector"

that has the largest variance.

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

① Assume each variable has been centered (columns have mean zero)

② Look for linear combination of the form $z_{i1} = \phi_{11} x_{i1} + \phi_{21} x_{i2} + \ldots + \phi_{p1} x_{ip}$ w/ largest variance subject to $\sum_{j=1}^{p} \phi_{j1}^2 = 1$

i.e. solve the following optimization problem:

$$\underset{\phi_{11}, \ldots, \phi_{p1}}{\text{maximize}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 \right\} \text{ subject to } \sum_{j=1}^{p} \phi_{j1}^2 = 1$$

solved via eigen decomposition (outside of scope of course).

can write this way because columns centered $\Rightarrow \frac{1}{n} \sum_{i=1}^{n} x_{ij} = 0 \Rightarrow \frac{1}{n} \sum_{i=1}^{n} z_{i1} = 0.$

$\Rightarrow$ this is the sample variance of $z_{i1}, i = 1, \ldots, n$.

$z_{11}, \ldots, z_{n1}$ are called the "scores" of first PC.

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

The loading vector ϕ_1 defines the direction in the feature space along which the data varies the most.

If we project n data points onto this direction, we get scores $z_{11}, \dots, z_{n1}$.

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 $X_1, \dots, X_p$ that has maximal variance out of all linear combinations that are uncorrelated with Z_1.

The second principal component scores are $z_{i2} = \phi_{12} x_{i1} + \dots + \phi_{p2} x_{ip}$, $\phi_2 =$ 2nd PC loading vector.

Z_2 uncorrelated w/ Z_1

$\iff$

ϕ_2 orthogonal to ϕ_1

$p = 2$

in 2D space, only have one possibility for ϕ_2

But with $p > 2$, there are multiple orthogonal options.

To find Z_2, solve a similar optimization problem, w/ additional constraint:

$$\underset{\phi_{21}, \dots, \phi_{2p}}{\text{maximize}} \left\{ \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{2j} x_{ij} \right)^2 \right\}$$

subject to $\sum_{j=1}^{p} \phi_{2j}^2 = 1$ and ϕ_2 orthogonal to ϕ_1

$$\left(\sum_{j=1}^{p} \phi_{2j} \phi_{1j} = 0 \right)$$

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
```

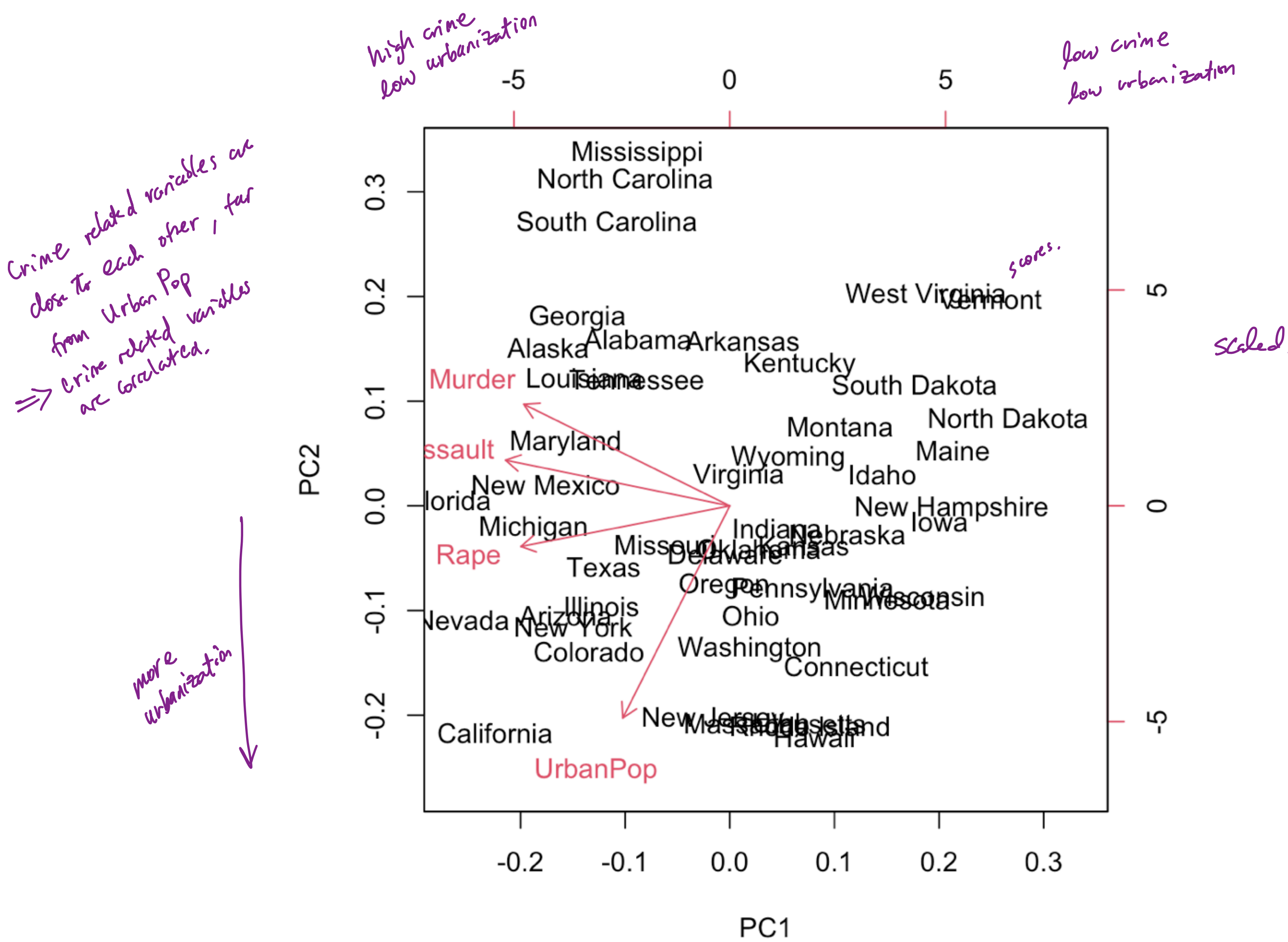
```
## Importance of components:
##                           PC1    PC2     PC3     PC4
## Standard deviation     1.5749 0.9949 0.59713 0.41645
## Proportion of Variance 0.6201 0.2474 0.08914 0.04336
## Cumulative Proportion  0.6201 0.8675 0.95664 1.00000
```

```
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)
```



First loading places approx. equal weight on crime-related variables, less weight on Urban pop.

=> this component ≈ measure of rate of serious crimes.

Second loading places most weight on Urban pop => ≈ level of urbanization.

2.2 Scaling Variables

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

Also the results depend on whether the variables have been scaled.

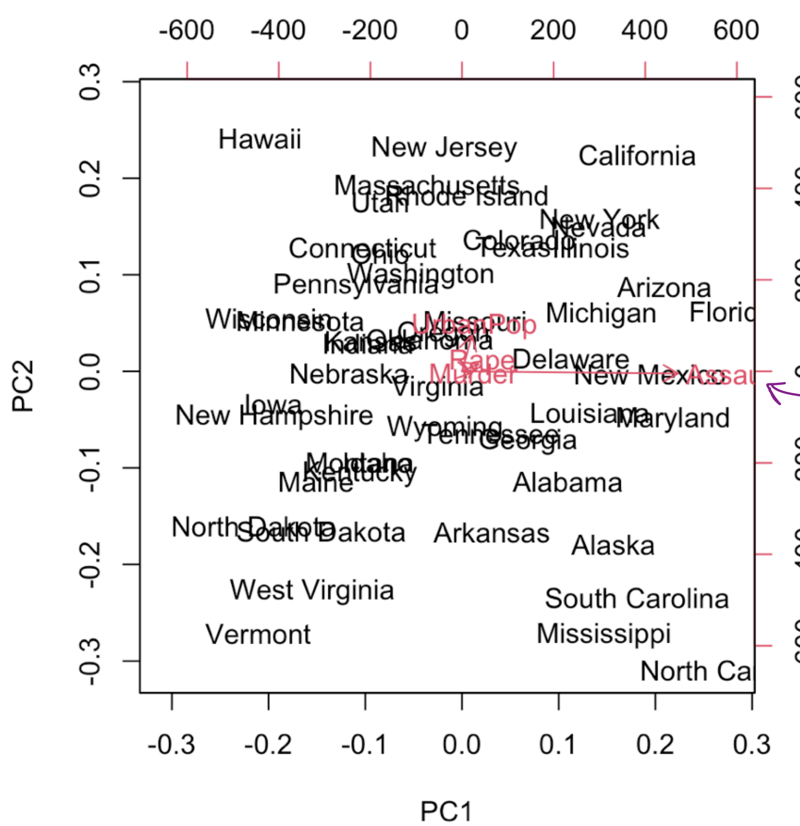
This is in contrast to other methods we've seen before.

same data as before, unscaled.

Variables are measured in different units

Crime: #/100,000 ppl.

UrbanPop: percentage.



largest loading on Assault for 1st PC (large variance based on scale)

Undesireable for PCA to depend on something arbitrary like scale ⇒ scale each variable to have sd = 1.

UNLESS: all variables are measured in same units ⇒ might not want to scale them!

2.3 Uniqueness

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

⇒ software will result in same loading vectors, but sign might flip.

sign may differ because each prin. comp. loading specifies a direction in p-space → line extends in both directions.

↳ flipping sign has no effect on this.

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

$Var(Z) = Var(-Z)$

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:

How much of the information (≈ "variance") in a given data set is lost by projecting observations onto the first two principal components?

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

Total variance in data set: $\sum_{j=1}^{p} Var(X_j) = \sum_{j=1}^{p} \frac{1}{n} \sum_{i=1}^{n} x_{ij}^2$

Variance explained by mth principal component: $\frac{1}{n} \sum_{i=1}^{n} z_{im}^2 = \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{jm} x_{ij} \right)^2$

⇒ PVE by mth prin. comp.: $\dfrac{\sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{jm} x_{ij} \right)^2}{\sum_{j=1}^{p} \sum_{i=1}^{n} x_{ij}^2}$ (positive quantity)

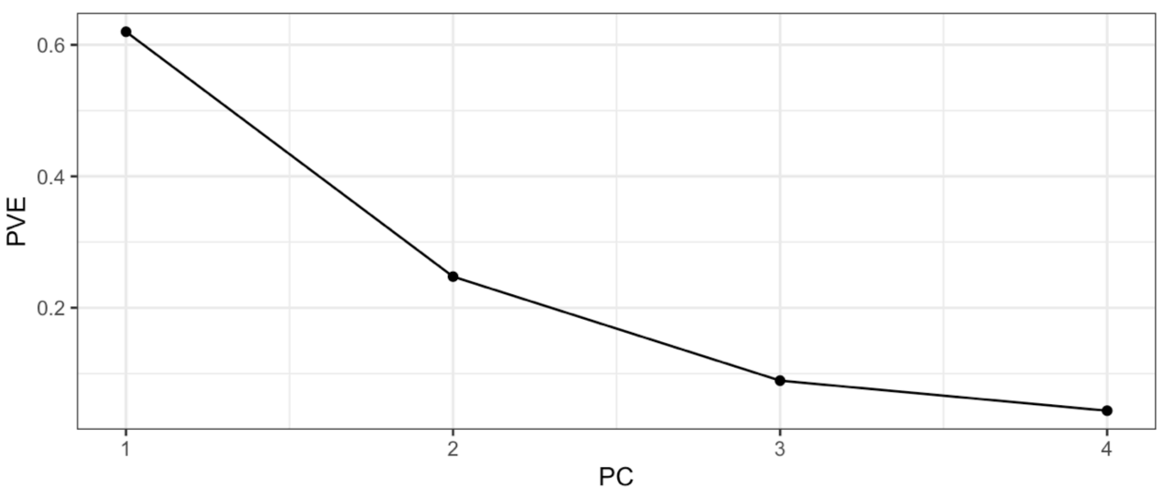
Cumulative PVE for 1st M components = sum PVE first M.

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.

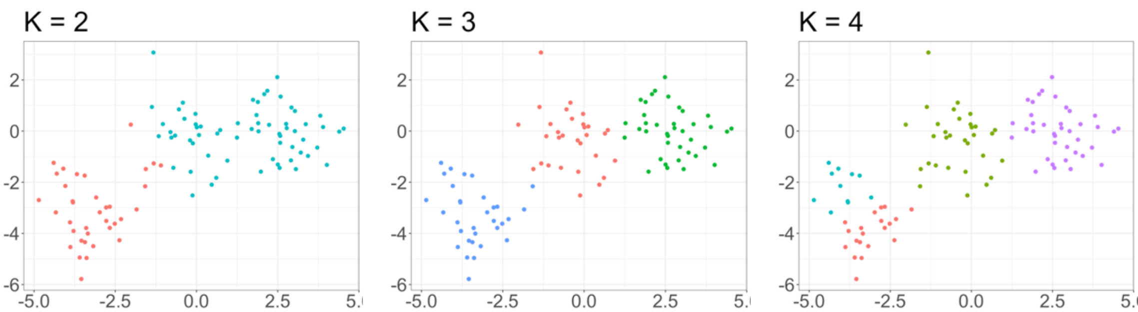
- K-means clustering

- 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 $C_1, \dots, C_K$ 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:

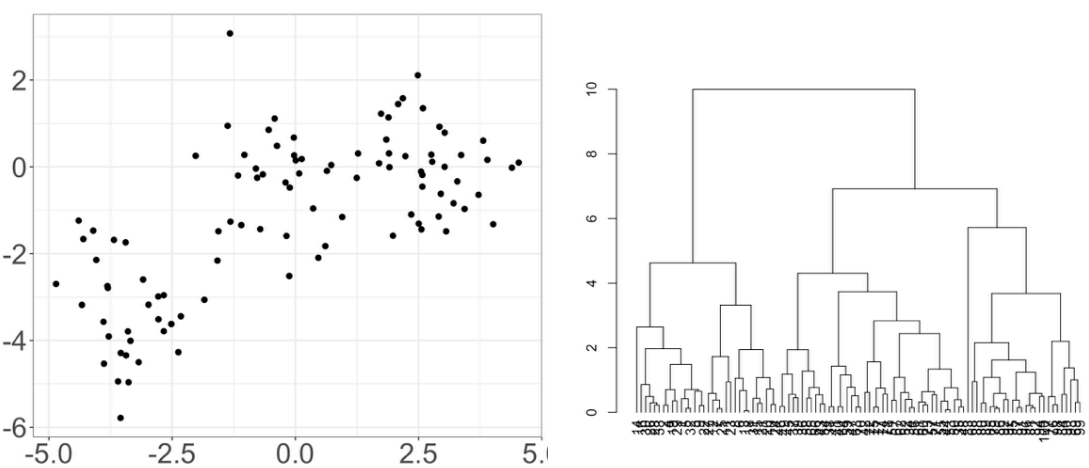
A very simple algorithm has been shown to find a local optimum to this problem:

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

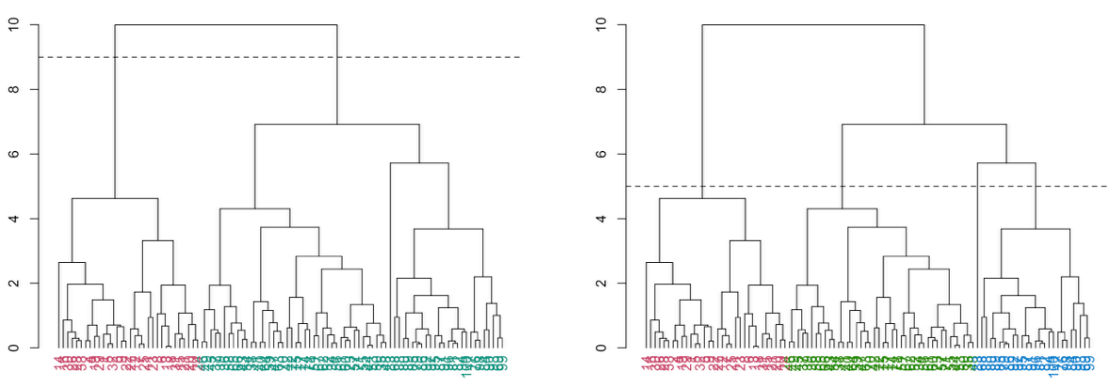


Each *leaf* of the dendrogram represents one of the 100 simulated data points.

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?

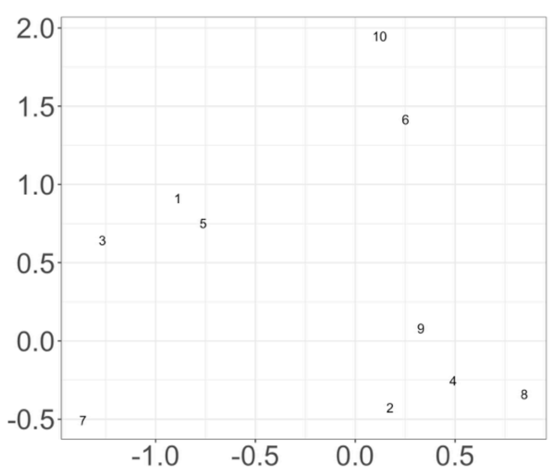


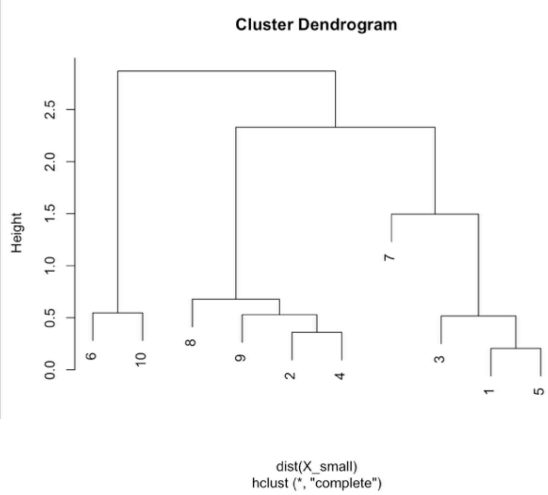
The term *hierarchical* refers to the fact that clusters obtained by cutting the dendrogram at a given height are necessarily nested within the clusters obtained by cutting the dendrogram at a greater height.

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?