Unsupervised learning

December 29, 2024

Table of contents

1. The challenge of unsupervised learning

2. Principal components analysis

3. Clustering methods

Unsupervised learning

- Most of this course focuses on *supervised learning* methods such as regression and classification.
- In that setting we observe both a set of features $X_1, X_2, \ldots, X_p$ for each object and a response or outcome variable Y. The goal is to predict Y using $X_1, X_2, \ldots, X_p$.
- Here instead we focus on *unsupervised learning*, where we observe only the features $X_1, X_2, \ldots, X_p$. We are not interested in prediction because we do not have an associated response variable Y.

Unsupervised learning

- Most of this course focuses on *supervised learning* methods such as regression and classification.
- In that setting we observe both a set of features $X_1, X_2, \ldots, X_p$ for each object and a response or outcome variable Y. The goal is to predict Y using $X_1, X_2, \ldots, X_p$.
- Here instead we focus on *unsupervised learning*, where we observe only the features $X_1, X_2, \ldots, X_p$. We are not interested in prediction because we do not have an associated response variable Y.

Unsupervised learning

- Most of this course focuses on *supervised learning* methods such as regression and classification.
- In that setting we observe both a set of features $X_1, X_2, \ldots, X_p$ for each object and a response or outcome variable Y. The goal is to predict Y using $X_1, X_2, \ldots, X_p$.
- Here instead we focus on *unsupervised learning*, where we observe only the features $X_1, X_2, \ldots, X_p$. We are not interested in prediction because we do not have an associated response variable Y.

The goals of unsupervised learning

- Discover interesting things about $X_1, X_2, \ldots, X_p$.
- Find informative ways to visualize the data
- Find sets of observations that are similar

The goals of unsupervised learning

- Discover interesting things about $X_1, X_2, \ldots, X_p$.
- Find informative ways to visualize the data
- Find sets of observations that are similar

The goals of unsupervised learning

- Discover interesting things about $X_1, X_2, \ldots, X_p$.
- Find informative ways to visualize the data
- Find sets of observations that are similar

This course covers two methods in *unsupervised learning*

- *Principal components analysis (PCA)* is a tool for visualization or for pre-processing before applying supervised learning techniques.
- *Clustering* is a class of methods for discovering meaningful subsets of similar observations within the data.

This course covers two methods in *unsupervised learning*

- *Principal components analysis (PCA)* is a tool for visualization or for pre-processing before applying supervised learning techniques.
- *Clustering* is a class of methods for discovering meaningful subsets of similar observations within the data.

The challenge of unsupervised learning

The challenge of unsupervised learning

- More subjective than *supervised* learning
- Lack of response observations means goals are less obvious
- Despite these challenges, there are useful applications that don't require a response variable, including:
  - subsets of breast cancer patient data help distinguish different cancer types based on gene expression measurements
  - shoppers can be clustered by their browsing and purchase histories
  - movies can be clustered by the ratings assigned by movie viewers
  - etc.

The challenge of unsupervised learning

- More subjective than *supervised* learning
- Lack of response observations means goals are less obvious
- Despite these challenges, there are useful applications that don't require a response variable, including:
  - subsets of breast cancer patient data help distinguish different cancer types based on gene expression measurements
  - shoppers can be clustered by their browsing and purchase histories
  - movies can be clustered by the ratings assigned by movie viewers
  - etc.

The challenge of unsupervised learning

- More subjective than *supervised* learning
- Lack of response observations means goals are less obvious
- Despite these challenges, there are useful applications that don't require a response variable, including:
  - subsets of breast cancer patient data help distinguish different cancer types based on gene expression measurements
  - shoppers can be clustered by their browsing and purchase histories
  - movies can be clustered by the ratings assigned by movie viewers
  - etc.

The challenge of unsupervised learning

- More subjective than *supervised* learning
- Lack of response observations means goals are less obvious
- Despite these challenges, there are useful applications that don't require a response variable, including:
  - subsets of breast cancer patient data help distinguish different cancer types based on gene expression measurements
  - shoppers can be clustered by their browsing and purchase histories
  - movies can be clustered by the ratings assigned by movie viewers
  - etc.

The challenge of unsupervised learning

- More subjective than *supervised* learning
- Lack of response observations means goals are less obvious
- Despite these challenges, there are useful applications that don't require a response variable, including:
  - subsets of breast cancer patient data help distinguish different cancer types based on gene expression measurements
  - shoppers can be clustered by their browsing and purchase histories
  - movies can be clustered by the ratings assigned by movie viewers
  - etc.

The challenge of unsupervised learning

- More subjective than *supervised* learning
- Lack of response observations means goals are less obvious
- Despite these challenges, there are useful applications that don't require a response variable, including:
  - subsets of breast cancer patient data help distinguish different cancer types based on gene expression measurements
  - shoppers can be clustered by their browsing and purchase histories
  - movies can be clustered by the ratings assigned by movie viewers
  - etc.

Unlabeled data is easy to obtain

- Getting *unlabeled data* is easier than getting *labeled data*, the latter typically requiring processing by humans.
- For example, is a given movie review favorable or not? Answering this question with yes or not would turn the data into labeled data but may be a non-obvious task. Unsupervised methods can simply use the movie ratings unlabelled.

Unlabeled data is easy to obtain

- Getting *unlabeled data* is easier than getting *labeled data*, the latter typically requiring processing by humans.
- For example, is a given movie review favorable or not? Answering this question with yes or not would turn the data into labeled data but may be a non-obvious task. Unsupervised methods can simply use the movie ratings unlabelled.

Principal components analysis

Principle components analysis reveals structure of data

- Lower-dimensional representation of a dataset via a sequence of linear combinations of the variables that are mutually uncorrelated and explain the largest possible share of total variation.
- Dimension reduction also makes data more amenable to visualization because high-dimensional datasets are hard to visualize.

Principle components analysis reveals structure of data

- Lower-dimensional representation of a dataset via a sequence of linear combinations of the variables that are mutually uncorrelated and explain the largest possible share of total variation.
- Dimension reduction also makes data more amenable to visualization because high-dimensional datasets are hard to visualize.

How does PCA work?

- The *first principal component* of a set of features $X_1, X_2, \ldots, X_p$ is the normalized linear combination of the features

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

  that has the largest variance. *Normalized* means that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.
- We refer to the elements $\phi_{11}, \ldots, \phi_{p1}$ as the *loadings* of the first principal component; together, they make up the *principal component loading vector*, $\phi_1 = (\phi_{11} \phi_{21} \cdots \phi_{p1})^T$.
- Constrain the loadings so their sum of squares is one. Otherwise setting them arbitrarily large would make the variance arbitrarily large.

How does PCA work?

- The *first principal component* of a set of features $X_1, X_2, \ldots, X_p$ is the normalized linear combination of the features

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

  that has the largest variance. *Normalized* means that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.
- We refer to the elements $\phi_{11}, \ldots, \phi_{p1}$ as the *loadings* of the first principal component; together, they make up the *principal component loading vector*, $\phi_1 = (\phi_{11} \phi_{21} \cdots \phi_{p1})^T$.
- Constrain the loadings so their sum of squares is one. Otherwise setting them arbitrarily large would make the variance arbitrarily large.

How does PCA work?

- The *first principal component* of a set of features $X_1, X_2, \ldots, X_p$ is the normalized linear combination of the features

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

  that has the largest variance. *Normalized* means that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.
- We refer to the elements $\phi_{11}, \ldots, \phi_{p1}$ as the *loadings* of the first principal component; together, they make up the *principal component loading vector*, $\phi_1 = (\phi_{11} \phi_{21} \cdots \phi_{p1})^T$.
- Constrain the loadings so their sum of squares is one. Otherwise setting them arbitrarily large would make the variance arbitrarily large.

Example: Population size and ad spending

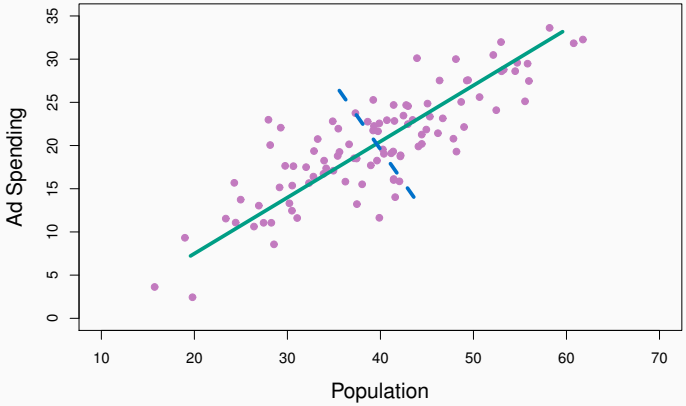


Figure 1: The population size (`pop`) and ad spending (`ad`) for 100 cities shown as purple circles. Green solid line is first principal component direction. Blue dashed line is second principal component direction.

How principal components are calculated

- Suppose we have an $n \times p$ data set $\mathbf{X}$. Since we are only interested in variance, we assume that each of the variables in $\mathbf{X}$ has been centered to have mean zero (that is, the column means of $\mathbf{X}$ are zero).
- We then look for the linear combination of the sample features values of the form

$$z_{i1} = \phi_{11} x_{i1} + \phi_{21} x_{i2} + \cdots + \phi_{p1} x_{ip} \tag{1}$$

  for $i = 1, \ldots, n$ that has largest sample variance, subject to the constraint that $\sum_{j=1}^{p} \phi_{j1}^2 = 1$.
- Since each of the x_{ij} has mean zero, then so does z_{i1} (for any values of ϕ_{j1}). Hence the sample variance of the z_{i1} can be written as $\frac{1}{n} \sum_{i=1}^{n} z_{i1}^2$.

How principal components are calculated (cont'd)

Plugging in Eq. 1 the principal component loading vector solves the optimization problem

$$\phi_{11}, \ldots, \phi_{p1} = \arg\max \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 .$$

This problem can be solved via a *singular-value decomposition*[1] of the matrix $\mathbf{X}$, a standard technique in linear algebra.

We refer to Z_1 as the first principal component, with realized values $z_{11}, \ldots, z_{n1}$.

[1] For this course, it is enough to know that SVD can be used to solve this kind of problem.

How principal components are calculated (cont'd)

Plugging in Eq. 1 the principal component loading vector solves the optimization problem

$$\phi_{11}, \ldots, \phi_{p1} = \arg\max \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 .$$

This problem can be solved via a *singular-value decomposition*[1] of the matrix $\mathbf{X}$, a standard technique in linear algebra.

We refer to Z_1 as the first principal component, with realized values $z_{11}, \ldots, z_{n1}$.

[1] For this course, it is enough to know that SVD can be used to solve this kind of problem.

How principal components are calculated (cont'd)

Plugging in Eq. 1 the principal component loading vector solves the optimization problem

$$\phi_{11}, \ldots, \phi_{p1} = \arg\max \frac{1}{n} \sum_{i=1}^{n} \left(\sum_{j=1}^{p} \phi_{j1} x_{ij} \right)^2 .$$

This problem can be solved via a *singular-value decomposition*[1] of the matrix $\mathbf{X}$, a standard technique in linear algebra.

We refer to Z_1 as the first principal component, with realized values $z_{11}, \ldots, z_{n1}$.

[1] For this course, it is enough to know that SVD can be used to solve this kind of problem.

Geometrically, PCA defines a new set of axes for the data

The first principal component

The loading vector ϕ_1 with elements $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ defines a direction in feature space along which the data vary the most.

If we project the n data points $x_1, \ldots, x_n$ onto this direction, the projected values are the principal component scores $z_{11}, \ldots, z_{n1}$.

Further principal components

The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance among all linear combinations that are *uncorrelated* with Z_1.

The second principal component scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

$$z_{i2} = \phi_{12} x_{i1} + \phi_{22} x_{i2} + \cdots + \phi_{p2} x_{ip},$$

where ϕ_2 is the second principal component loading vector, with elements $\phi_{12}, \phi_{22}, \ldots, \phi_{p2}$.

Geometrically, PCA defines a new set of axes for the data

The first principal component

The loading vector ϕ_1 with elements $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ defines a direction in feature space along which the data vary the most.

If we project the n data points $x_1, \ldots, x_n$ onto this direction, the projected values are the principal component scores $z_{11}, \ldots, z_{n1}$.

Further principal components

The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance among all linear combinations that are *uncorrelated* with Z_1.

The second principal component scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

$$z_{i2} = \phi_{12} x_{i1} + \phi_{22} x_{i2} + \cdots + \phi_{p2} x_{ip},$$

where ϕ_2 is the second principal component loading vector, with elements $\phi_{12}, \phi_{22}, \ldots, \phi_{p2}$.

Geometrically, PCA defines a new set of axes for the data

The first principal component

The loading vector ϕ_1 with elements $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ defines a direction in feature space along which the data vary the most.

If we project the n data points $x_1, \ldots, x_n$ onto this direction, the projected values are the principal component scores $z_{11}, \ldots, z_{n1}$.

Further principal components

The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance among all linear combinations that are *uncorrelated* with Z_1.

The second principal component scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

$$z_{i2} = \phi_{12} x_{i1} + \phi_{22} x_{i2} + \cdots + \phi_{p2} x_{ip},$$

where ϕ_2 is the second principal component loading vector, with elements $\phi_{12}, \phi_{22}, \ldots, \phi_{p2}$.

Geometrically, PCA defines a new set of axes for the data

The first principal component

The loading vector ϕ_1 with elements $\phi_{11}, \phi_{21}, \ldots, \phi_{p1}$ defines a direction in feature space along which the data vary the most.

If we project the n data points $x_1, \ldots, x_n$ onto this direction, the projected values are the principal component scores $z_{11}, \ldots, z_{n1}$.

Further principal components

The second principal component is the linear combination of $X_1, \ldots, X_p$ that has maximal variance among all linear combinations that are *uncorrelated* with Z_1.

The second principal component scores $z_{12}, z_{22}, \ldots, z_{n2}$ take the form

$$z_{i2} = \phi_{12} x_{i1} + \phi_{22} x_{i2} + \cdots + \phi_{p2} x_{ip},$$

where ϕ_2 is the second principal component loading vector, with elements $\phi_{12}, \phi_{22}, \ldots, \phi_{p2}$.

Correlation and orthogonality of principal components

Constraining Z_2 to be uncorrelated with Z_1 (and so on for Z_3, etc.) is equivalent to constraining the direction ϕ_2 to be *orthogonal* (*perpendicular*) to the direction ϕ_1.

Correlation and orthogonality of principal components

Constraining Z_2 to be uncorrelated with Z_1 (and so on for Z_3, etc.) is equivalent to constraining the direction ϕ_2 to be *orthogonal* (*perpendicular*) to the direction ϕ_1.

Example: PCA of `USArrests` data

Data set contains number of arrests per 100,000 residents for each of the 50 U.S. states for the crimes of `Assault`, `Murder`, and `Rape`; along with `UrbanPop`, the proportion of the population living in urban areas for each state.

The principal component score vectors have length $n = 50$, and the principal component loading vectors have length $p = 4$.

Data were standardized to mean zero and unit standard deviation before PCA.

Example: PCA of `USArrests` data

Data set contains number of arrests per 100,000 residents for each of the 50 U.S. states for the crimes of `Assault`, `Murder`, and `Rape`; along with `UrbanPop`, the proportion of the population living in urban areas for each state.

The principal component score vectors have length $n = 50$, and the principal component loading vectors have length $p = 4$.

Data were standardized to mean zero and unit standard deviation before PCA.

Example: PCA of `USArrests` data

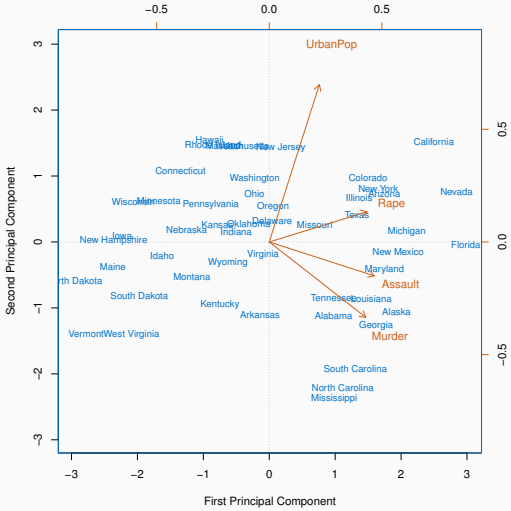
Data set contains number of arrests per 100,000 residents for each of the 50 U.S. states for the crimes of `Assault`, `Murder`, and `Rape`; along with `UrbanPop`, the proportion of the population living in urban areas for each state.

The principal component score vectors have length $n = 50$, and the principal component loading vectors have length $p = 4$.

Data were standardized to mean zero and unit standard deviation before PCA.

Example: PCA of `USArrests` data, biplot

Figure 2: Two first principal components for `USArrests` data. Blue state names are scores for first two principal components. Orange arrows indicate first two principal component loading vectors (with axes on top and right). The loading for `Rape` on the first component is 0.54, and its loading on the second principal component 0.17. This figure is known as a *biplot* because it displays both principal component scores and principal component loadings.



Example: PCA of `USArrests` data, loadings

	Z_1	Z_2
Murder	0.5358995	-0.4181809
Assault	0.5831836	-0.1879856
UrbanPop	0.2781989	0.8728062
Rape	0.5434321	0.1673186

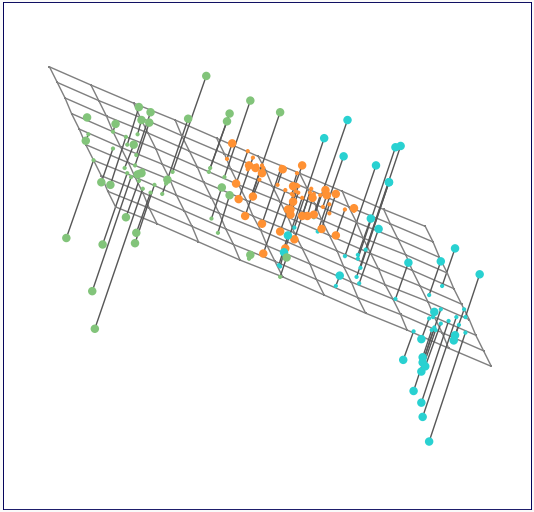
Table 1: The principal component loading vectors, ϕ_1 and ϕ_2, for the `USArrests` data.

Principal components analysis

Another interpretation of principal components

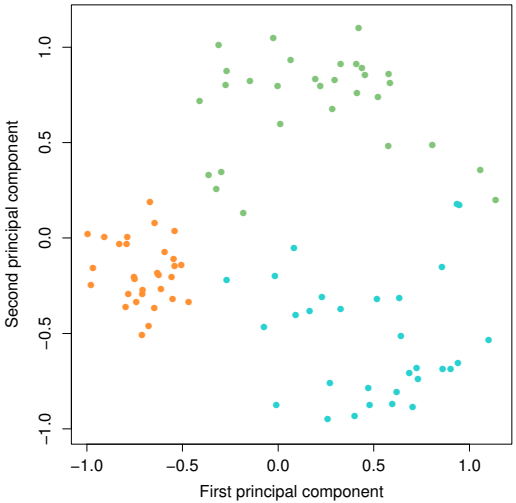
PCA finds the (hyper)plane closest to the observations

Figure 3: The first principal component loading vector defines the line in p-dimensional space that is *closest* to the n observations, measured by average squared Euclidean distance. This notion extends beyond the first principal component. The first two principal components then span the plane that is closest to the n observations. Colors are for readability only.



PCA finds the (hyper)plane closest to the observations

Figure 4: The first two principal component score vectors give the coordinates of the projection of the 90 observations onto the plane. Colors are for readability only.



Scaling of the variables matters

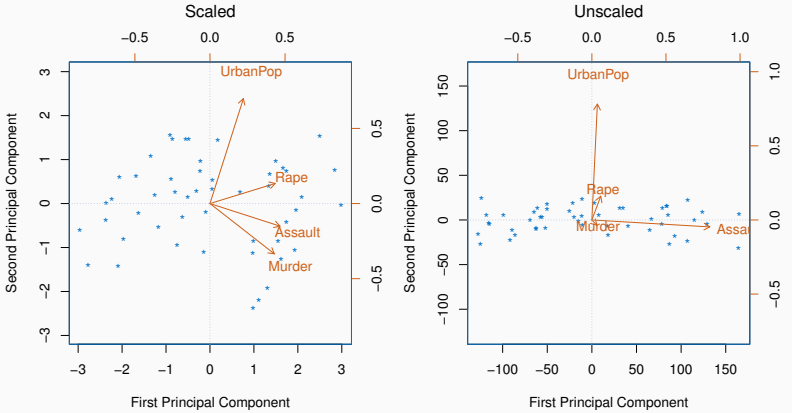


Figure 5: If variables are in different units, scaling each to have unit standard deviation is recommended. If they are in the same units, you might or might not scale the variables.

Principal components analysis

The proportion of variance explained

Proportion of variance explained measures PC strength

Total variance in a data set centered to mean zero is

$$\sum_{j=1}^{p} \mathrm{Var}(X_j) = \sum_{j=1}^{p} \frac{1}{n} \sum_{i=1}^{n} x_{ij}^2,$$

and variance explained by mth principal component is

$$\mathrm{Var}(Z_m) = \frac{1}{n} \sum_{i=1}^{n} z_{im}^2.$$

One can show that $\sum_{j=1}^{p} \mathrm{Var}(X_j) = \sum_{m=1}^{M} \mathrm{Var}(Z_m)$, with $M = \min(n-1, p)$; that is, all PCs jointly explain all of the variance.

Proportion of variance explained measures PC strength

Total variance in a data set centered to mean zero is

$$\sum_{j=1}^{p} \mathrm{Var}(X_j) = \sum_{j=1}^{p} \frac{1}{n} \sum_{i=1}^{n} x_{ij}^2,$$

and variance explained by mth principal component is

$$\mathrm{Var}(Z_m) = \frac{1}{n} \sum_{i=1}^{n} z_{im}^2.$$

One can show that $\sum_{j=1}^{p} \mathrm{Var}(X_j) = \sum_{m=1}^{M} \mathrm{Var}(Z_m)$, with $M = \min(n-1, p)$; that is, all PCs jointly explain all of the variance.

How to calculate *proportion of variance explained* (PVE)

PVE of mth principal component is given by $\frac{\sum_{i=1}^{n} z_{im}^2}{\sum_{j=1}^{p}\sum_{i=1}^{n} x_{ij}^2} \in [0,1]$.

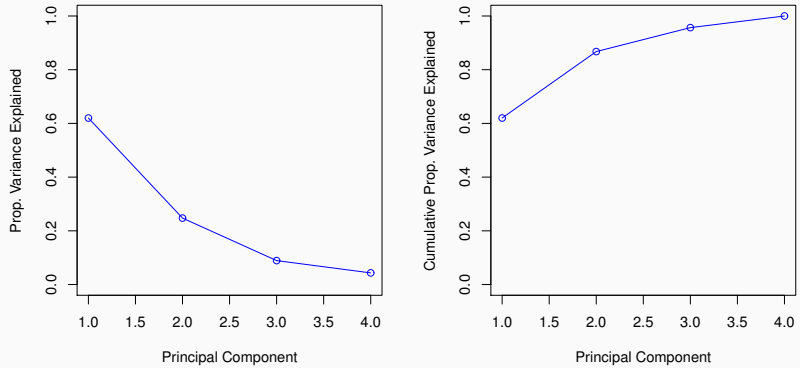


Figure 6: Left: Scree plot depicting proportion of variance explained by each of the four principle components in the `USArrests` data. Right: Cumulative PVE.

Principal components analysis

More on PCA

Looking for “elbows” to decide how many PCs to use

When using PCA as summary of data, how many PCs should we retain?

Cross-validation is not available to answer this question because there is no response data.

A scree plot can provide an indication. Look for an “elbow,” a point at which the additional variation explained by adding PCs decreases significantly.

Looking for “elbows” to decide how many PCs to use

When using PCA as summary of data, how many PCs should we retain?

Cross-validation is not available to answer this question because there is no response data.

A scree plot can provide an indication. Look for an “elbow,” a point at which the additional variation explained by adding PCs decreases significantly.

Looking for “elbows” to decide how many PCs to use

When using PCA as summary of data, how many PCs should we retain?

Cross-validation is not available to answer this question because there is no response data.

A scree plot can provide an indication. Look for an “elbow,” a point at which the additional variation explained by adding PCs decreases significantly.

Fill in the blanks

1. The correlation between the variables generated via PCA is ____.
2. PCA can explain at most _____ of the total variation in the dataset, if all principal components are retained.

True or false?

1. __ PCA is a supervised learning method.
2. __ PCA can be used as a pre-processing step and for visualizing high-dimensional data sets.
3. __ PCA provides a lower-dimensional representation of the data set without losing the variable interpretations.
4. __ In many applications, a small number of the variables explain a large amount of their total variation, which PCA can reveal by computing proportion of variance explained.

Fill in the blanks

1. The correlation between the variables generated via PCA is zero.
2. PCA can explain at most 100 % of the total variation in the dataset, if all principal components are retained.

True or false?

1. F PCA is a supervised learning method.
2. T PCA can be used as a pre-processing step and for visualizing high-dimensional data sets.
3. F PCA provides a lower-dimensional representation of the data set without losing the variable interpretations.
4. T In many applications, a small number of the variables explain a large amount of their total variation, which PCA can reveal by computing proportion of variance explained.

Clustering methods

Clustering is a set of methods for finding similar subsets

- *Clusters* are *subsets* of the data that are similar in some meaningful sense.
- Partition of data set into distinct sets, such that elements in each set are similar to one another.
- What do we mean by *similar*?
- Often specific to domain of application, but we will see some examples.

Clustering is a set of methods for finding similar subsets

- *Clusters* are *subsets* of the data that are similar in some meaningful sense.
- Partition of data set into distinct sets, such that elements in each set are similar to one another.
- What do we mean by *similar*?
- Often specific to domain of application, but we will see some examples.

Clustering is a set of methods for finding similar subsets

- *Clusters* are *subsets* of the data that are similar in some meaningful sense.
- Partition of data set into distinct sets, such that elements in each set are similar to one another.
- What do we mean by *similar*?
- Often specific to domain of application, but we will see some examples.

Clustering is a set of methods for finding similar subsets

- *Clusters* are *subsets* of the data that are similar in some meaningful sense.
- Partition of data set into distinct sets, such that elements in each set are similar to one another.
- What do we mean by *similar*?
- Often specific to domain of application, but we will see some examples.

PCA explains variation while clustering finds similarity

PCA finds low-dimensional representation of the data set that *explains a good fraction of total variance.*

Clustering *finds homogenous subsets* of observations; i.e., subsets whose elements are *similar* to one another.

PCA explains variation while clustering finds similarity

PCA finds low-dimensional representation of the data set that *explains a good fraction of total variance.*

Clustering *finds homogenous subsets* of observations; i.e., subsets whose elements are *similar* to one another.

One can use clustering for *market segmentation*

- Consider measurements for *median household income*, *occupation*, *distance from nearest urban area*, etc., for a large number of people.
- Goal: *Market segmentation* to identify subsets of people particularly receptive to some forms of advertising or more likely to buy a particular product.
- This is a clustering problem.

One can use clustering for *market segmentation*

- Consider measurements for *median household income*, *occupation*, *distance from nearest urban area*, etc., for a large number of people.
- Goal: *Market segmentation* to identify subsets of people particularly receptive to some forms of advertising or more likely to buy a particular product.
- This is a clustering problem.

One can use clustering for *market segmentation*

- Consider measurements for *median household income*, *occupation*, *distance from nearest urban area*, etc., for a large number of people.
- Goal: *Market segmentation* to identify subsets of people particularly receptive to some forms of advertising or more likely to buy a particular product.
- This is a clustering problem.

K-means clustering vs. *hierarchical clustering*

K-means clustering

Partition observations into a pre-specified number K of clusters

Hierarchical clustering

Explore clusters arising from all possible numbers of clusters between 1 and n, typically using a *dendrogram*[2]

[2]From Greek δένδρον meaning *tree* and γράμμα meaning *drawing* or *figure*.

Clustering methods

K-means clustering

K-means clustering

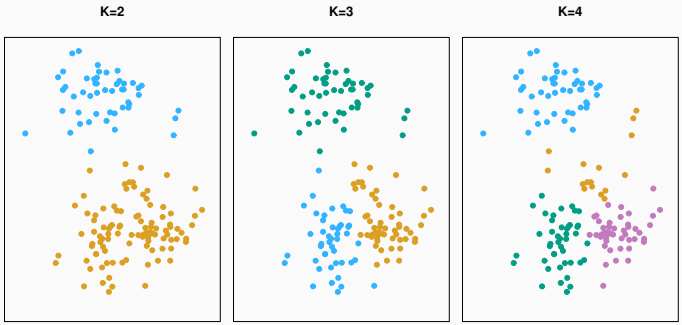


Figure 7: A simulated data set with 150 observations in 2-dimensional space. Panels show the results of applying K-means clustering with different values of K, the number of clusters. The color of each observation indicates the cluster to which it was assigend using the K means clustering algorithm. Note that there is no ordering of the clusters, so the cluster coloring is arbitrary. These cluster labels were not used in clustering; instead, they are the outputs of the clustering procedure.

Details of K means clustering

Let $C_1, \ldots, C_K$ denote sets containing the indices of the observations in each cluster. If the ith observation is in the kth cluster, then $i \in C_k$. These sets satisfy two properties:[3]

1. $C_1 \cup C_2 \cup \cdots \cup C_K = \{1, \ldots, n\}$. In other words, each observation belongs to at least one of the K clusters.
2. $C_k \cup C_{k'} = \emptyset$ for all $k \neq k'$. In other words, the clusters are non-overlapping: no observation belongs to more than one cluster.

[3]This is actually the definition of the *partition* of a set. The same concept was used in tree-based methods for segmentation of the predictor space.

K-means clustering minimizes within-cluster variation

- A good clustering is one for which the *within-cluster variation* is as small as possible. This is what we meant above by saying that clustering found subsets of the data whose observations were similar.
- For a cluster C_k, within-cluster variation $\text{WCV}(C_k)$ measures how different observations within the cluster are.
- K-means clustering solves the problem

$$\{C_1, \ldots, C_K\} = \arg\min \left\{ \sum_{k=1}^{K} \text{WCV}(C_k) \right\}. \tag{2}$$

- In words, we partition the observations into K clusters such that total within-cluster variation summed over all K clusters is as small as possible.

K-means clustering minimizes within-cluster variation

- A good clustering is one for which the *within-cluster variation* is as small as possible. This is what we meant above by saying that clustering found subsets of the data whose observations were similar.
- For a cluster C_k, within-cluster variation $\text{WCV}(C_k)$ measures how different observations within the cluster are.
- K-means clustering solves the problem

$$\{C_1, \ldots, C_K\} = \arg\min \left\{ \sum_{k=1}^{K} \text{WCV}(C_k) \right\}. \tag{2}$$

- In words, we partition the observations into K clusters such that total within-cluster variation summed over all K clusters is as small as possible.

Euclidean distance can measure within-cluster variation

$$\mathrm{WCV}(C_k) = \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2, \tag{3}$$

where $|C_k|$ is number of observations in kth cluster.

Combining 2 and 3 gives the optimization problem that defines K-means clustering,

$$\{C_1, \ldots, C_K\} = \arg\min \left\{ \sum_{k=1}^{K} \frac{1}{|C_k|} \sum_{i,i' \in C_k} \sum_{j=1}^{p} (x_{ij} - x_{i'j})^2 \right\}. \tag{4}$$

K-means clustering algorithm

Algorithm

1. Randomly assign a number, from 1 to K to each of the observations. These serve as initial cluster assignments for the observations.
2. Iterate until the cluster assignments stop changing:
   - 2.1 For each of the K clusters, compute the cluster *centroid*. The kth cluster centroid is the vector of the p feature means for the observations in the kth cluster.
   - 2.2 Assign each observation to the cluster whose centroid is closest (where *closest* is defined using Euclidean distance).

K-means algorithm decreases WCV at each iteration

Note that

$$\frac{1}{|C_k|}\sum_{i,i'\in C_k}\sum_{j=1}^{p}(x_{ij}-x_{i'j})^2 = 2\sum_{i\in C_k}\sum_{j=1}^{p}(x_{ij}-\bar{x}_{kj})^2.$$

However, this is not guaranteed to give the global minimum.

Example: K-means algorithm with $K = 3$

Figure 8: Top left: Observations. Top center: Assign each observation randomly to a cluster; Top right: Compute cluster centroids (colored disks); for random initial cluster assignment, centroids overlap; Bottom left: Each observation is assigned to the nearest centroid; Bottom center: cluster centroid assignment is repeated, leading to new cluster centroids; Bottom right: Result obtained after 10 iterations.

Example: K-means algorithm with $K = 3$

Figure 8: Top left: Observations; Top center: Assign each observation randomly to a cluster; Top right: Compute cluster centroids (colored disks); for random initial cluster assignment, centroids overlap; Bottom left: Each observation is assigned to the nearest centroid; Bottom center: cluster centroid assignment is repeated, leading to new cluster centroids; Bottom right: Result obtained after 10 iterations.

Example: K-means algorithm with $K = 3$

Figure 8: Top left: Observations; Top center: Assign each observation randomly to a cluster; Top right: Compute cluster centroids (colored disks); for random initial cluster assignment, centroids overlap. Bottom left: Each observation is assigned to the nearest centroid; Bottom center: cluster centroid assignment is repeated, leading to new cluster centroids; Bottom right: Result obtained after 10 iterations.

Example: K-means algorithm with $K = 3$

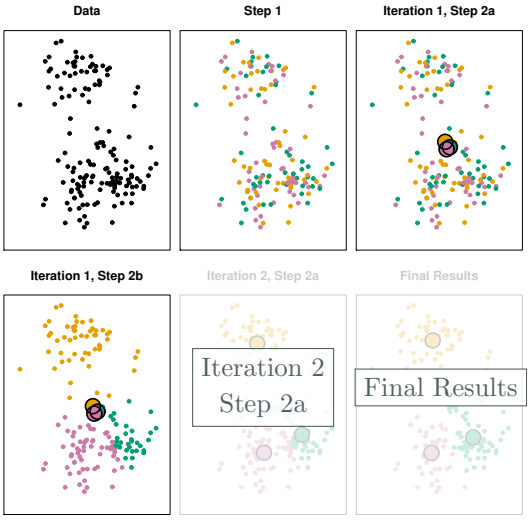


Figure 8: Top left: Observations; Top center: Assign each observation randomly to a cluster; Top right: Compute cluster centroids (colored disks); for random initial cluster assignment, centroids overlap; Bottom left: Each observation is assigned to the nearest centroid. Bottom center: cluster centroid assignment is repeated, leading to new cluster centroids; Bottom right: Result obtained after 10 iterations.

Example: K-means algorithm with $K = 3$

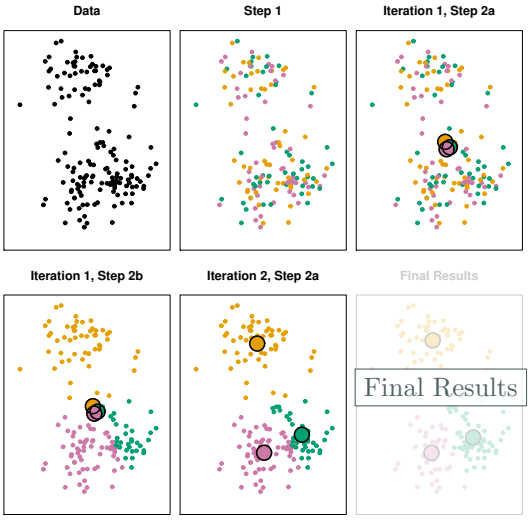


Figure 8: Top left: Observations; Top center: Assign each observation randomly to a cluster; Top right: Compute cluster centroids (colored disks); for random initial cluster assignment, centroids overlap; Bottom left: Each observation is assigned to the nearest centroid; Bottom center: cluster centroid assignment is repeated, leading to new cluster centroids; Bottom right: Result obtained after 10 iterations.

Example: K-means algorithm with $K = 3$

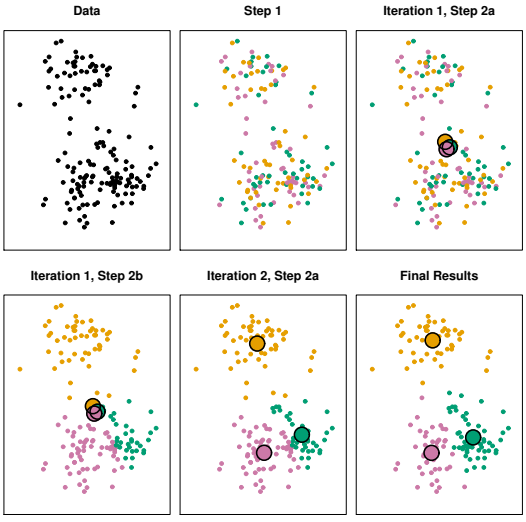


Figure 8: Top left: Observations; Top center: Assign each observation randomly to a cluster; Top right: Compute cluster centroids (colored disks); for random initial cluster assignment, centroids overlap; Bottom left: Each observation is assigned to the nearest centroid; Bottom center: cluster centroid assignment is repeated, leading to new cluster centroids; Bottom right: Result obtained after 10 iterations.

Example: K-means algorithm with $K = 3$, starting values

Figure 9: K-means clustering performed six times on data from previous figure with $K = 3$, with different random assignment of observations in Step 1 of the algorithm. Above each plot is the value of the objective (Eq. 4). Three local optima were obtained, one of which resulted in a smaller value of the objective and provides better cluster separation. Those labeled in red all achieved the same best solution.

Clustering methods

Hierarchical clustering

Hierarchical clustering allows variable number of clusters

- In K-means clustering, we specify number of clusters K in advance.
- *Hierarchical clustering* does not require this.
- This section describes *bottom-up* or *agglomerative* clustering, the most common form of clustering that builds a dendrogram from the leaves up to the trunk ("bottom-up").[4]

[4] Recall that in tree diagrams—unlike in botany—terminal nodes/leaves are at the bottom of the diagram.

Hierarchical clustering allows variable number of clusters

- In K-means clustering, we specify number of clusters K in advance.
- *Hierarchical clustering* does not require this.
- This section describes *bottom-up* or *agglomerative* clustering, the most common form of clustering that builds a dendrogram from the leaves up to the trunk ("bottom-up").[4]

[4] Recall that in tree diagrams—unlike in botany—terminal nodes/leaves are at the bottom of the diagram.

Hierarchical clustering allows variable number of clusters

- In K-means clustering, we specify number of clusters K in advance.
- *Hierarchical clustering* does not require this.
- This section describes *bottom-up* or *agglomerative* clustering, the most common form of clustering that builds a dendrogram from the leaves up to the trunk ("bottom-up").[4]

[4] Recall that in tree diagrams—unlike in botany—terminal nodes/leaves are at the bottom of the diagram.

Hierarchical clustering on simulated data

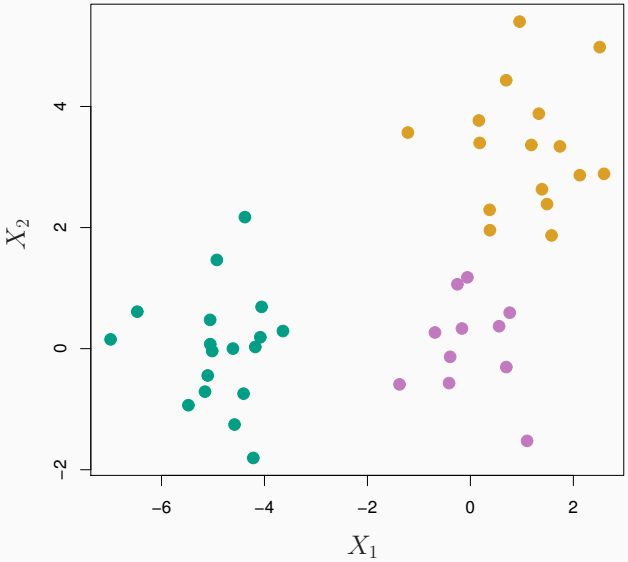


Figure 10: 45 observations generated in 2-dimensional space. In reality there are three distinct classes, shown in separate colors. However, we will treat these class labels as unknown and will seek to cluster the observations in order to discover the classes from the data.

Dendrogram for simulated data set cut at different heights

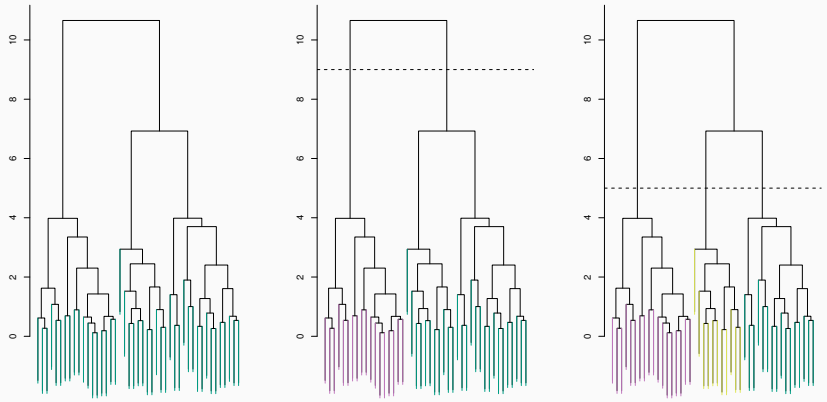


Figure 11: Left: Dendrogram obtained from hierarchically clustering simulated data from previous slide, with *complete* linkage and Euclidean distance; Center: Dendrogram from left panel cut at a height of 9 (dashed line), resulting in two distinct clusters shown in different colors; Right: Same dendrogram cut at a height of 5, resulting in three clusters.

There are different types of linkage in clustering

Linkage	Description
Complete	Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between observations in cluster A and observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities.
Average	Mean inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

There are different types of linkage in clustering

Linkage	Description
Complete	Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between observations in cluster A and observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities.
Average	Mean inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

There are different types of linkage in clustering

Linkage	Description
Complete	Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between observations in cluster A and observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities.
Average	Mean inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

There are different types of linkage in clustering

Linkage	Description
Complete	Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between observations in cluster A and observations in cluster B, and record the *largest* of these dissimilarities.
Single	Minimal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *smallest* of these dissimilarities.
Average	Mean inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the *average* of these dissimilarities.
Centroid	Dissimilarity between the centroid for cluster A (a mean vector of length p) and the centroid for cluster B. Centroid linkage can result in undesirable *inversions*.

Different dissimilarity measures can be used for clustering

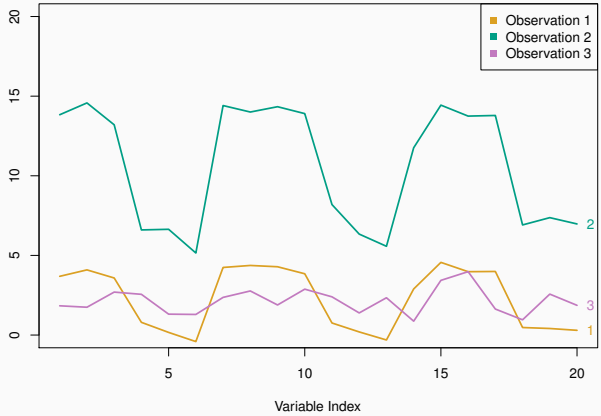


Figure 12: So far we have used *Euclidean distance*. An alternative is *correlation-based distance* which considers two observations to be similar if their features are highly correlated. This is an unusual use of correlation, which is normally computed between variables; here it is computed between observation profiles for each pair of observations.

Hierarchical clustering visualized

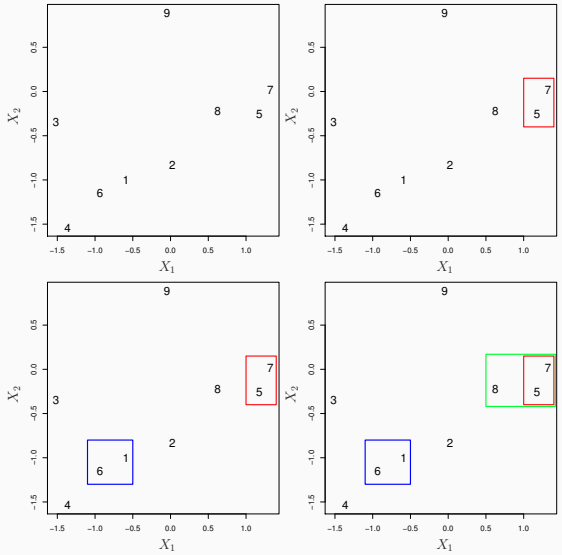


Figure 13: First few steps of the hierarchical clustering algorithm using complete linkage and Euclidean distance. **Top Left**: initially, there are nine distinct clusters, $\{1\}, \{2\}, \ldots, \{9\}$.

Hierarchical clustering visualized

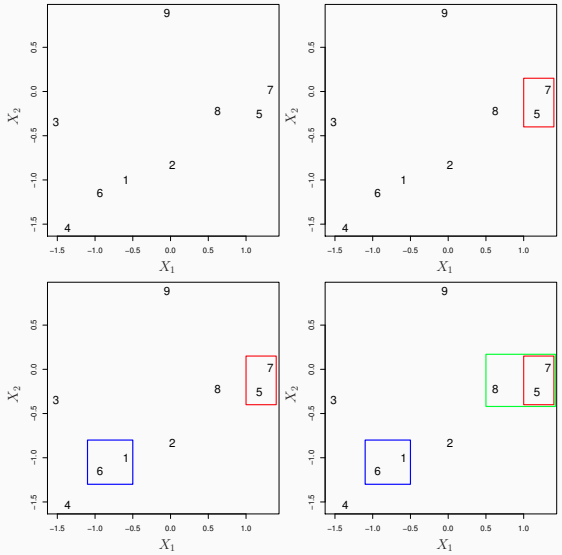


Figure 13: First few steps of the hierarchical clustering algorithm using complete linkage and Euclidean distance. **Top Left**: initially, there are nine distinct clusters, $\{1\}, \{2\}, \ldots, \{9\}$. **Top Right**: the two clusters that are closest together $\{5\}$ and $\{7\}$ are fused into a single cluster.

Hierarchical clustering visualized

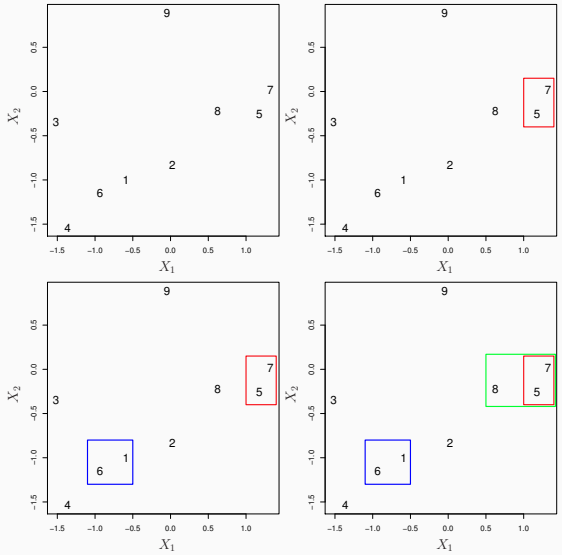


Figure 13: First few steps of the hierarchical clustering algorithm using complete linkage and Euclidean distance. **Top Left**: initially, there are nine distinct clusters, $\{1\}, \{2\}, \ldots, \{9\}$. **Top Right**: the two clusters that are closest together $\{5\}$ and $\{7\}$ are fused into a single cluster. **Bottom Left**: the two clusters that are closest together, $\{6\}$ and $\{1\}$, are fused together into a single cluster.

Hierarchical clustering visualized

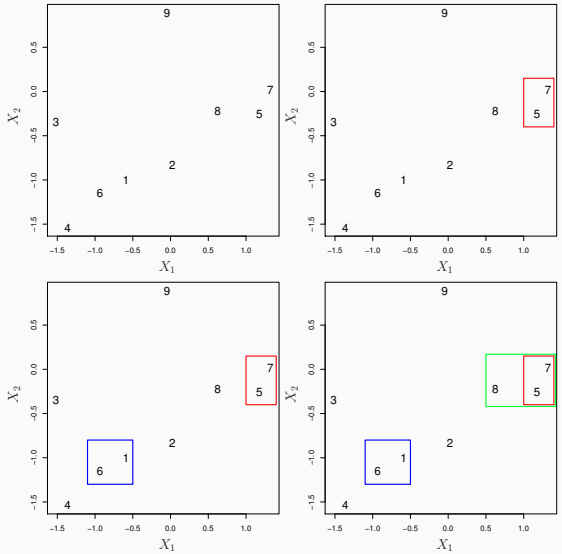


Figure 13: First few steps of the hierarchical clustering algorithm using complete linkage and Euclidean distance. **Top Left**: initially, there are nine distinct clusters, $\{1\}, \{2\}, \ldots, \{9\}$. **Top Right**: the two clusters that are closest together $\{5\}$ and $\{7\}$ are fused into a single cluster. **Bottom Left**: the two clusters that are closest together, $\{6\}$ and $\{1\}$, are fused together into a single cluster. **Bottom Right**: the two clusters that are closest together using complete linkage, $\{8\}$ and the cluster $\{5, 7\}$, are fused into a single cluster.

Clustering methods

Practical issues in clustering

Practical issue in clustering

- *Scaling of the variables matters!* Should the observations of features first be standardized in some way? For instance, maybe the variables should be centered to have mean zero and scaled to have standard deviation one.
- In the case of hierarchical clustering,
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
- How many clusters to choose? (in both K-means or hierarchical clustering). Difficult problem. No agreed-upon method. See *Elements of Statistical Learning*, Chapter 13, for more details.
- Which features should we use to drive the clustering?

Practical issue in clustering

- *Scaling of the variables matters!* Should the observations of features first be standardized in some way? For instance, maybe the variables should be centered to have mean zero and scaled to have standard deviation one.
- In the case of hierarchical clustering,
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
- How many clusters to choose? (in both K-means or hierarchical clustering). Difficult problem. No agreed-upon method. See *Elements of Statistical Learning*, Chapter 13, for more details.
- Which features should we use to drive the clustering?

Practical issue in clustering

- *Scaling of the variables matters!* Should the observations of features first be standardized in some way? For instance, maybe the variables should be centered to have mean zero and scaled to have standard deviation one.
- In the case of hierarchical clustering,
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
- How many clusters to choose? (in both K-means or hierarchical clustering). Difficult problem. No agreed-upon method. See *Elements of Statistical Learning*, Chapter 13, for more details.
- Which features should we use to drive the clustering?

Practical issue in clustering

- *Scaling of the variables matters!* Should the observations of features first be standardized in some way? For instance, maybe the variables should be centered to have mean zero and scaled to have standard deviation one.
- In the case of hierarchical clustering,
  - What dissimilarity measure should be used?
  - What type of linkage should be used?
- How many clusters to choose? (in both K-means or hierarchical clustering). Difficult problem. No agreed-upon method. See *Elements of Statistical Learning*, Chapter 13, for more details.
- Which features should we use to drive the clustering?

Fill in the blanks

1. Hierarchical clustering provides a graphical representation called a _____.

True or false?

1. __ Clusters are subsets of meaningfully similar observations.
2. __ K-means clustering determines the optimal number of clusters K for us.
3. __ Choosing the right number of clusters is inherently subjective.
4. __ Clustering is robust to variable scaling.

Fill in the blanks

1. Hierarchical clustering provides a graphical representation called a dendrogram.

True or false?

1. T Clusters are subsets of meaningfully similar observations.
2. F K-means clustering determines the optimal number of clusters K for us.
3. T Choosing the right number of clusters is inherently subjective.
4. F Clustering is robust to variable scaling.

Summary[5]

Principal components analysis (PCA) is a method for reducing the dimensionality of a data set. It does this by finding a new set of dimensions, called principal components, that capture as much of the variance in the data as possible. These new dimensions are typically fewer in number than the original dimensions, which makes the data easier to visualize and analyze. PCA is a common technique used in data analysis and machine learning.

K-means clustering is a method for grouping a set of data points into clusters. It does this by finding cluster centers (also called means) that are representative of each group, and assigning each data point to the cluster whose center is closest to it. K-means clustering is an iterative process, and the final clusters depend on the initial cluster centers chosen. This method is often used in data analysis and machine learning to find structure in data.

Hierarchical clustering is a method for grouping data points into clusters. It does this by creating a hierarchy of clusters, where each cluster is defined as a subset of the data points. This hierarchy can be represented as a tree, with the clusters at the leaves and the clusters containing those clusters at the higher levels. Hierarchical clustering is often used in data analysis and machine learning to find structure in data. Unlike k-means clustering, hierarchical clustering does not require the user to specify the number of clusters upfront.

[5]The content of this slide was prepared by GPT 3.5. No modifications were made to the model's output.

This material draws extensively on James, G., Witten, D., Hastie, T. & Tibshirani, R. (2021). *An introduction to statistical learning* and the lecture slides available from these authors.