Data exploration

Fundamentals of Computing and Data Display

Christoph Kern[1] Ruben Bach

(c.kern, r.bach)@uni-mannheim.de

[1] Thanks to Sebastian Sternberg!

Outline

Introduction

Unsupervised Learning

- Tools for finding patterns in (unlabeled) data
- In the unsupervised learning setting, we only observe features X_1, X_2, X_p; no outcome
- The goal is to discover interesting things about the measurements
  - Is there an informative way to visualize the data?
  - Are there subgroups among the variables and/or the observations?
- Unsupervised learning is also called **exploratory data analysis** or knowledge discovery
- Common objective is dimensionality reduction

Introduction

- **Clustering**
  - Clustering refers to a very broad set of techniques for finding subgroups (clusters) in a data set
  - A cluster can be defined as a group of *similar* objects (cases, members, customers, locations etc.)
- **Factor Analysis**
  - Summarize many (correlated) features into a few, uncorrelated dimensions
  - Here we focus on Principal Component Analysis (PCA), which is – strictly speaking – only related to factor analysis

PCA aims at clustering ***features***, whereas **cluster methods** aim at clustering ***objects***

Clustering

Clustering

Cluster methods

- In **K-Means clustering**, we seek to partition the observations into a pre-specified number of clusters
- In **hierarchical clustering**, we do not know in advance how many clusters we want
  - We end up with a tree-like visual representation of the observations that allows us to view at once the clusterings obtained for each possible number of clusters, from 1 to n
- Expectation-maximization clustering, Mean shift clustering, Spectral clustering, ...

K-Means

- K-Means is the simplest and the most common cluster algorithm
- K-Means algorithm is fast and easy to use; it is thus a good solution in the Big Data context
- General idea: Iteratively re-assign observations to the nearest cluster center
- May be used as a preprocessing step for other algorithms
- K-Means also has some drawbacks which should be kept in mind

K-Means

Objective: Find cluster solution with smallest within-cluster variation $W(C_k)$:

$$\underset{C_1,\ldots,C_K}{\text{minimize}} \left\{ \sum_{k=1}^{K} W(C_k) \right\}$$

Within-cluster variation may be defined as the sum of pairwise squared **Euclidean distances** between the observations in a cluster (over all p features):

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

K-Means

Algorithm 1: K-Means Clustering

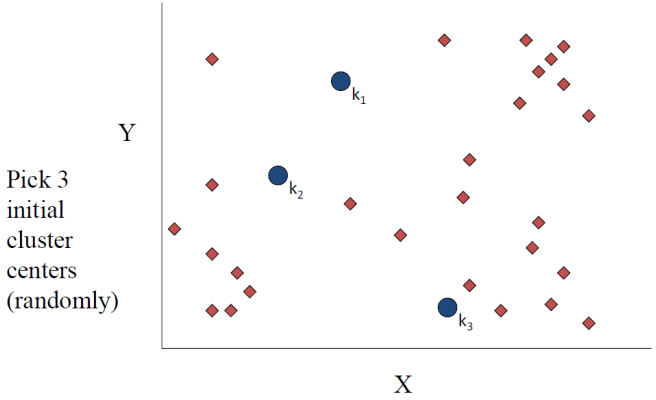
Parameter : Number of cluster K

Initialization: Randomly choose K data points (seeds) to be the initial centroids, cluster centers

1 **repeat**
2 Assign each data point to the closest centroid;
3 Re-compute the cluster centroids (vector of p feature means) using the current cluster memberships;
4 **until** *no reclassification is necessary*;

K-Means example, Step 1

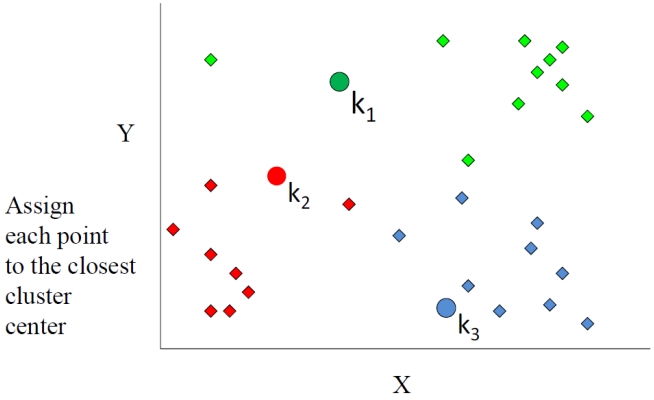
Figure: K-Means: Step 1[2]



[2] Ghani and Schierholz (2017)

K-Means example, Step 2

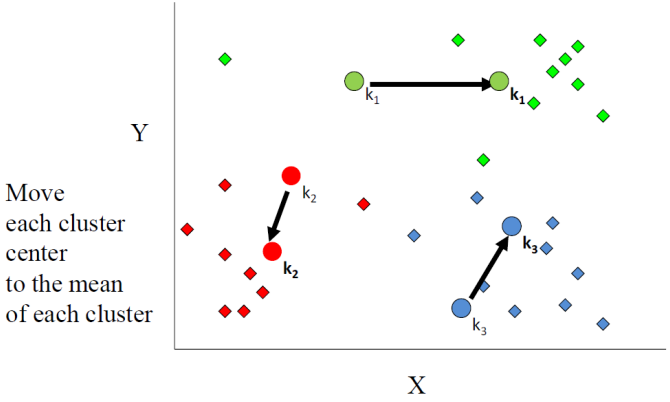
Figure: K-Means: Step 2[3]



[3] Ghani and Schierholz (2017)

K-Means example, Step 3

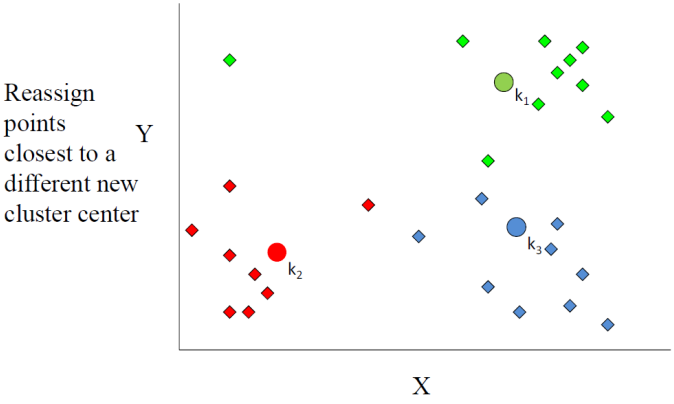
Figure: K-Means: Step 3[4]



[4] Ghani and Schierholz (2017)

K-Means example, Step 4

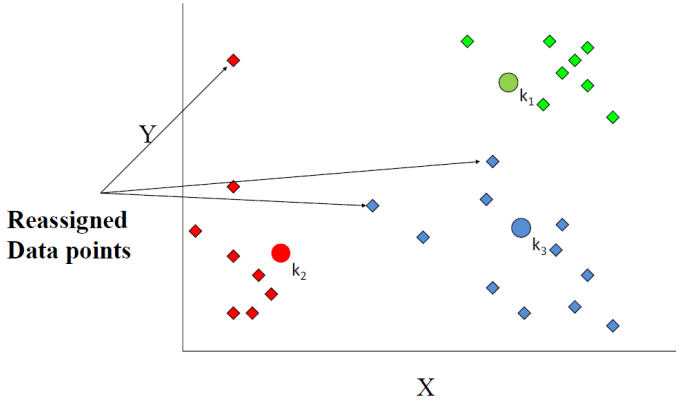
Figure: K-Means: Step 4[5]



[5] Ghani and Schierholz (2017)

K-Means example, Step 4...

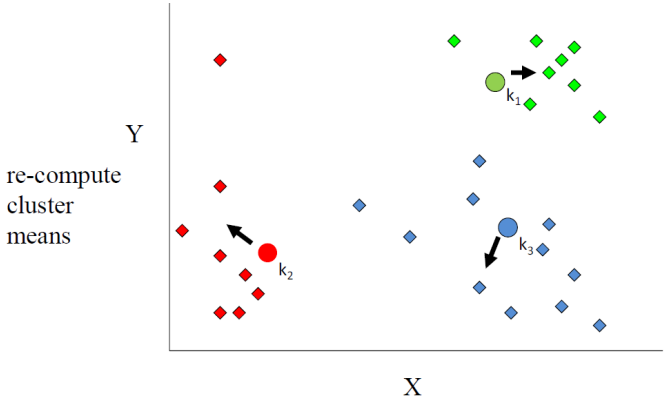
Figure: K-Means: Step 4...[6]



[6] Ghani and Schierholz (2017)

K-Means example, Step 4b

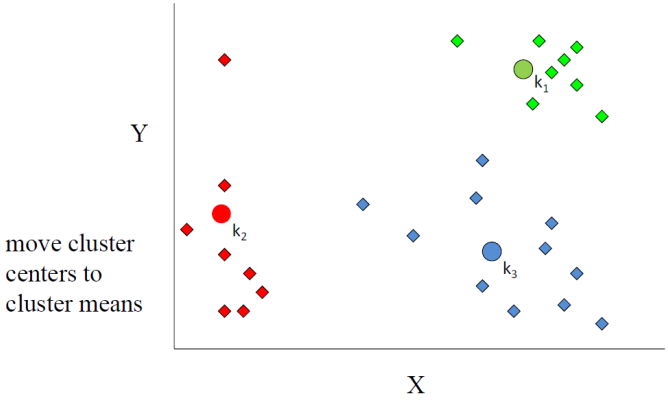
Figure: K-Means: Step 4b[7]



[7] Ghani and Schierholz (2017)

K-Means example, Step 5

Figure: K-Means: Step 5[8]



[8] Ghani and Schierholz (2017)

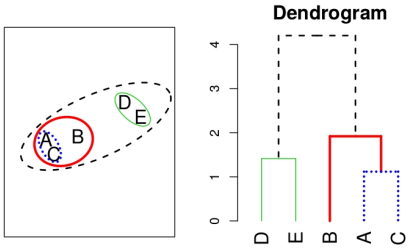
Drawbacks

- Choosing the number of cluster K
  - Run K-Means with different values of K
  - Graphical diagnostic checks such as the elbow-method (plot K against explained variance)
  - Does the same K yield to good solutions for different subsets of the original data?
- Choosing starting values
  - Run K-Means several times with different starting values, and take the best solution

Hierarchical Clustering

Hierarchical Clustering

- K-means clustering requires us to pre-specify the number of clusters K
- Hierarchical clustering does not require that we commit to a particular choice of K
- Here we focus on **bottom-up** or **agglomerative** clustering
- Tree-based representation (dendrogram) is built starting from the leaves and combining clusters up to the trunk



Hierarchical Clustering

Algorithm 2: Hierarchical Clustering

Parameter : Dissimilarity measures w.r.t. observations and clusters

Initialization: Start with each observation as its own cluster

for $i = n, n-1, \ldots, 2$ **do**
    Identify the closest two clusters given all pairwise inter-cluster dissimilarities;
    Merge these two clusters;
    Compute the new pairwise inter-cluster dissimilarities among the remaining clusters;
end

Distances

Computing the dissimilarities between observations; the **distance matrix**

Euclidean Distance: $||\boldsymbol{x}_a - \boldsymbol{x}_b||_2$

$$\sqrt{\sum_{j=1}^{P}(x_{aj} - x_{bj})^2}$$

Minkowski Distance:

$$\left(\sum_{j=1}^{P}|x_{aj} - x_{bj}|^q\right)^{\frac{1}{q}}$$

Manhattan Distance: $||\boldsymbol{x}_a - \boldsymbol{x}_b||_1$

Other metrics available: Mahalanobis, Chebyshev...

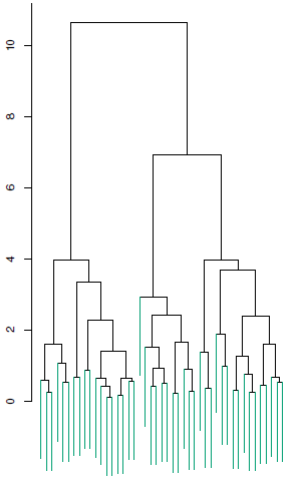
Linkage

Computing the dissimilarities between clusters; types of **linkages**

- Complete linkage
  - Maximal inter-cluster dissimilarity. Record the *largest* of the dissimilarities between observations in cluster A and B
- Average linkage
  - Mean inter-cluster dissimilarity. Record the *average* of the dissimilarities between observations in cluster A and B
- Single linkage
  - Minimal inter-cluster dissimilarity. Record the *smallest* of the dissimilarities between observations in cluster A and B
- Ward linkage
  - Fuse cluster A and B that result in the smallest increase in the *within-cluster variation* of the new cluster

Dendrograms

Figure: Dendrogram example[9]

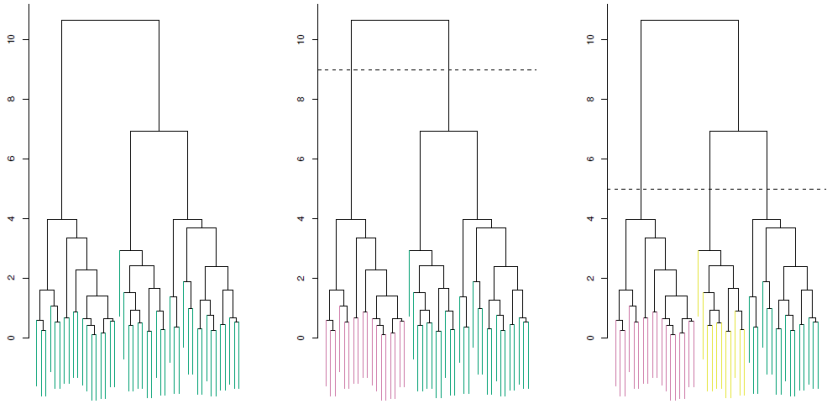


- Each leaf represents one observation
- Moving up the tree, some similar leaves begin to fuse into branches
- As we move higher up the tree, branches themselves fuse
- Observations fusing at the bottom are quite similar to each other, whereas observations fusing close to the top will tend to be quite different (indicated by height)

[9] James et al. (2013)

Dendrograms

Figure: Different cuts result in different clusters[10]



[10] James et al. (2013)

Practical issues

- Should the observations or features first be standardized in some way?
- What dissimilarity measure should be used?
- What type of linkage should be used?
- Did cluster algorithm really found true subgroups, or are the obtained clusters a result of clustering the noise?
- Outliers can heavily distort clusters, because they belong to no group but are forced into one
- Clustering methods are not very robust to perturbations to the data

Clustering should be performed with different choices of parameters, and for different subsets of the data.

Principal Component Analysis

Principal Component Analysis

Given a set of data on n dimensions, PCA aims to find a linear subspace of dimension d lower than n such that the data points lie mainly on this subspace

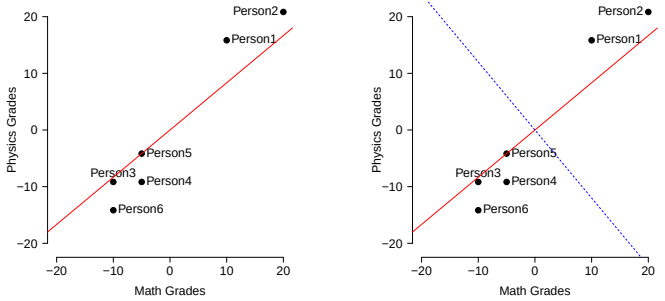
- Combine a correlated group of variables into a new characteristic (component)
- Aim is to find characteristics which strongly differ across different groups of variables, but are good in "reconstructing" the original variables
- We therefore "throw away" information by combining variables to a factor
- The better the components are in explaining the variance of all variables, the better job it did in summarizing the variables

Principal Component Analysis

- First principal component is the best linear approximation for the data in the direction with the **maximum variance**
- This results in a line which minimizes the sum of squared distance between a data point and the line

Figure: PCA Example

(a) First principle component (b) Second principle component



Loadings and scores

- Suppose we have a $n \times p$ data set X where each of the variables in X has been centered to have mean zero
- The first principal component of a set of features $x_1, x_2, \ldots, x_p$ is the normalized ($\sum_{j=1}^{p} \phi_{j1}^2 = 1$) linear combination of the features

$$z_{i1} = \phi_{11} x_{i1} + \phi_{21} x_{i2} + \ldots + \phi_{p1} x_{ip}$$

that has the largest variance.

- $\phi_{11}, \ldots, \phi_{p1}$ is referred to as the loadings of the first principle component; they make up the principle component loading vector

$$\phi_1 = (\phi_{11} \ \phi_{21} \ \ldots \ \phi_{p1})^T$$

Loadings and scores

- 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

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

  where ϕ_2 is the second principal component loading vector with elements $\phi_{12}, \ldots, \phi_{p2}$.
- Constraining z_2 to be uncorrelated with z_1 is equivalent to constraining the direction ϕ_2 to be orthogonal to the direction ϕ_1

Computation of Principle Components

Extract Principle Components

1. Standardize the variables
2. Calculate covariance matrix $X'X$
3. Perform eigen-decomposition to find eigenvectors (q_j) and eigenvalues (λ_j) of the covariance matrix

$$X'X = Q\Lambda Q'$$

   - The eigenvector with the highest eigenvalue is the first principal component

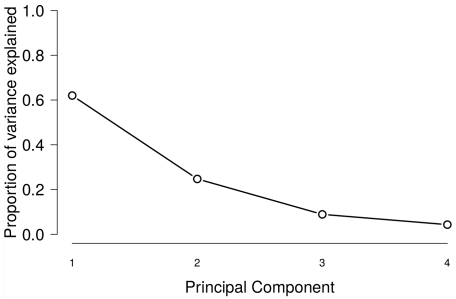
4. Reorient the data by multiplying the original data with the eigenvectors to compute the scores

$$F = XQ$$

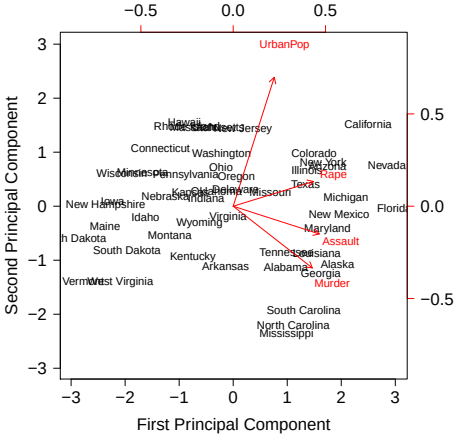
How many components are needed?

- How much of the information is lost by projecting the observations onto the first few principal components?
  - Proportion of variance explained by component m: $\frac{\sum_{i=1}^{n}\left(\sum_{j=1}^{p}\phi_{jm}x_{ij}\right)^2}{\sum_{j=1}^{p}\sum_{i=1}^{n}x_{ij}^2}$
- If we use PCA to summarize our data, how many components are sufficient?
  - Decision may be based on "eyeballing" the scree plot and looking for the "elbow"

Figure: Screeplot example



Biplots



- Biplots are often used to visualize the PCA results using the first two PCs
- Is named biplot because it displays both the PC *scores* and the PC *loadings* in one figure
- The red arrows indicate the first two principle component *loading vectors*
- The black names represent the *scores* for the first two principle components

Summary

PCA is a useful tool in big data context: **dimension reduction** and **data inspection**

Direct usage

- Summarize many (correlated) features into a few, uncorrelated dimensions
- Detecting interesting patterns in data for a deeper investigation

Indirect usage

- PCA as data pre-processing: only use first few principle components instead of all features

References

Ghani, R., Schierholz, M. (2017). Machine Learning. In: Foster, I., Ghani, R., Jarmin, R. S., Kreuter, F., Lane, J. (Eds.). (2017). *Big Data and Social Science: A Practical Guide to Methods and Tools*. Boca Raton, FL: CRC Press Taylor & Francis Group.

James, G., Witten, D., Hastie, T., Tibshirani, R. (2013). *An Introduction to Statistical Learning*. New York, NY: Springer.