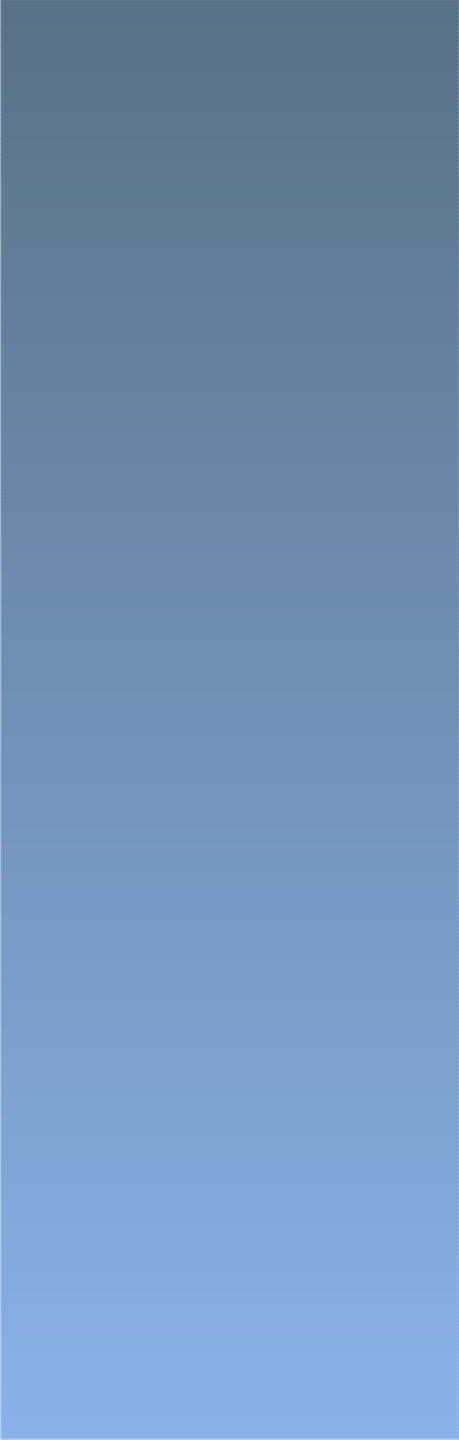
Data Analysis 2021 Spring





**Lecture 13: Unsupervised Learning**

June 2 & 7, 2021

**Taesoo Kwon**

[tskwon@seoultech.ac.kr](mailto:tskwon@seoultech.ac.kr)

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

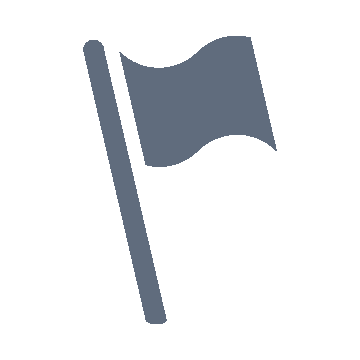
# Course Schedule (Tentative)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Topics** | **Note** | **Date (W)** | **Date (M)** |
| 1 | Orientation, Statistical Learning (Ch2) | Online | 03/03 | 03/08 |
| 2 | Statistical Learning (Ch2), Python Programming | Online | 03/10 | 03/15 |
| 3 | Probability & Statistics | Online | 03/17 | 03/22 |
| 4 | Probability & Statistics | Online | 03/24 | 03/29 |
| 5 | Linear Regression (Ch3) | Online | 03/31 | 04/05 |
| 6 | Linear Regression (Ch3) | Online | 04/07 | 04/12 |
| 7 | Classification (Ch4) | Online | 04/14 | 04/19 |
| 8 | **Midterm exam** | **Class hours (W1-W7)** | **04/21** | **04/26** |
| 9 | Resampling Methods (Ch5) | Online | 04/28 | 05/03 |
| 10 | Linear Model Selection and Regularization (Ch6) | Online | 05/05 | 05/10 |
| 11 | Moving Beyond Linearity (Ch7) | Online | 05/12 | 05/17 |
| 12 | Tree-Based Methods (Ch8) | Online | 05/19 | 05/24 |
| 13 | Support Vector Machines (Ch9) | Online | 05/26 | 05/31 |
| **14** | Unsupervised Learning (Ch10) | Online | 06/02 | 06/07 |
| 15 | **Final exam** | **6pm (W9-W14)** | **06/10Th** | **06/10Th** |

©copyright CIEL 2015 **2**



* Unsupervised learning



**OUTLINES**

* + Introduction to unsupervised learning
  + Principle component analysis
  + K-means clustering
  + Hierarchical clustering
* Summary & Next class

©copyright CIEL 2015



**Unsupervised Learning**



**: Ch 10**

## Unsupervised learning

* Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Unsupervised vs Supervised 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 𝑋1, 𝑋2, ⋯ , 𝑋𝑝 for each object, as well as a

response or outcome variable 𝑌

* + The goal is then to predict 𝑌 using 𝑋1, 𝑋2, ⋯ , 𝑋𝑝
* Here we instead focus on unsupervised learning, we where observe only the features

𝑋1, 𝑋2, ⋯ , 𝑋𝑝

* + We are not interested in prediction, because we do not have an associated response variable 𝑌

©copyright CIEL 2015 **5**

# Goals of Unsupervised Learning

* The goal is to discover interesting things about the measurements
  + Is there an informative way to visualize the data?
  + Can we discover subgroups among the variables or among the observations?
* We discuss two methods
  + Principal components analysis
    - A tool used for data visualization or data pre-processing before supervised techniques are applied
  + Clustering
    - A broad class of methods for discovering unknown subgroups in data.

©copyright CIEL 2015 **6**

# Challenge of Unsupervised Learning

* Unsupervised learning is more subjective than supervised learning, as there is no simple goal for the analysis, such as prediction of a response
* But techniques for unsupervised learning are of growing importance in a number of fields:
  + Subgroups of breast cancer patients grouped by their gene expression measurements
  + Groups of shoppers characterized by their browsing and purchase histories
  + Movies grouped by the ratings assigned by movie viewers

©copyright CIEL 2015 **7**

# Another Advantage

* It is often easier to obtain unlabeled data (from a lab instrument or a computer) than labeled data, which can require human intervention
* For example it is difficult to automatically assess the overall sentiment of a movie review
  + Is it favorable or not?

©copyright CIEL 2015 **8**

# Principal Components Analysis (PCA)

* PCA produces a low-dimensional representation of a dataset
* It finds a sequence of linear combinations of the variables that have maximal variance, and are

mutually uncorrelated

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

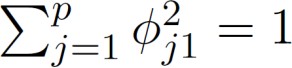
©copyright CIEL 2015 **9**

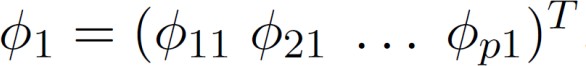
# PCA: Details

* The first principal component of a set of features 𝑋1, 𝑋2, ⋯ , 𝑋𝑝 is the normalized linear combination of the features



that has the largest variance

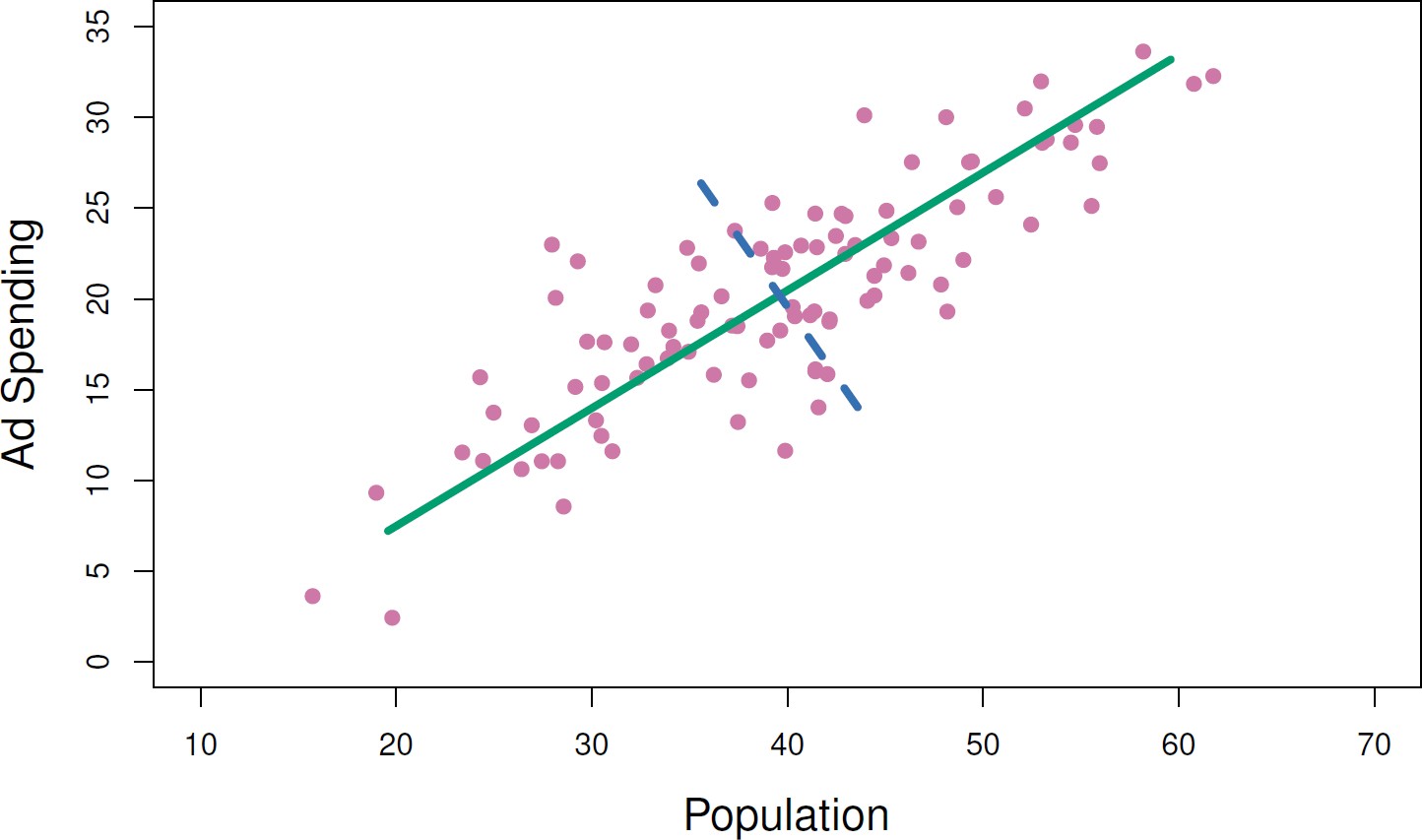
* + By normalized, we mean that 
* We refer to the elements 𝜙11, ⋯,𝜙𝑝1 as the loadings of the first principal component; together, the loadings make up the principal component loading vector



* We constrain the loadings so that their sum of squares is equal to one, since otherwise setting these elements to be arbitrarily large in absolute value could result in an arbitrarily large variance

©copyright CIEL 2015 **10**

# PCA: Example



* The population size (pop) and ad spending (ad) for 100 different cities are shown as purple circles
* The green solid line indicates the first principal component direction, and the blue dashed line indicates the second principal component direction

©copyright CIEL 2015 **11**

# [Review] Principal Components Regression

* Here we apply principal components analysis (PCA) (discussed in Chapter 10 of the text) to define the linear combinations of the predictors, for use in our regression
* The first principal component is that (normalized) linear combination of the variables with the

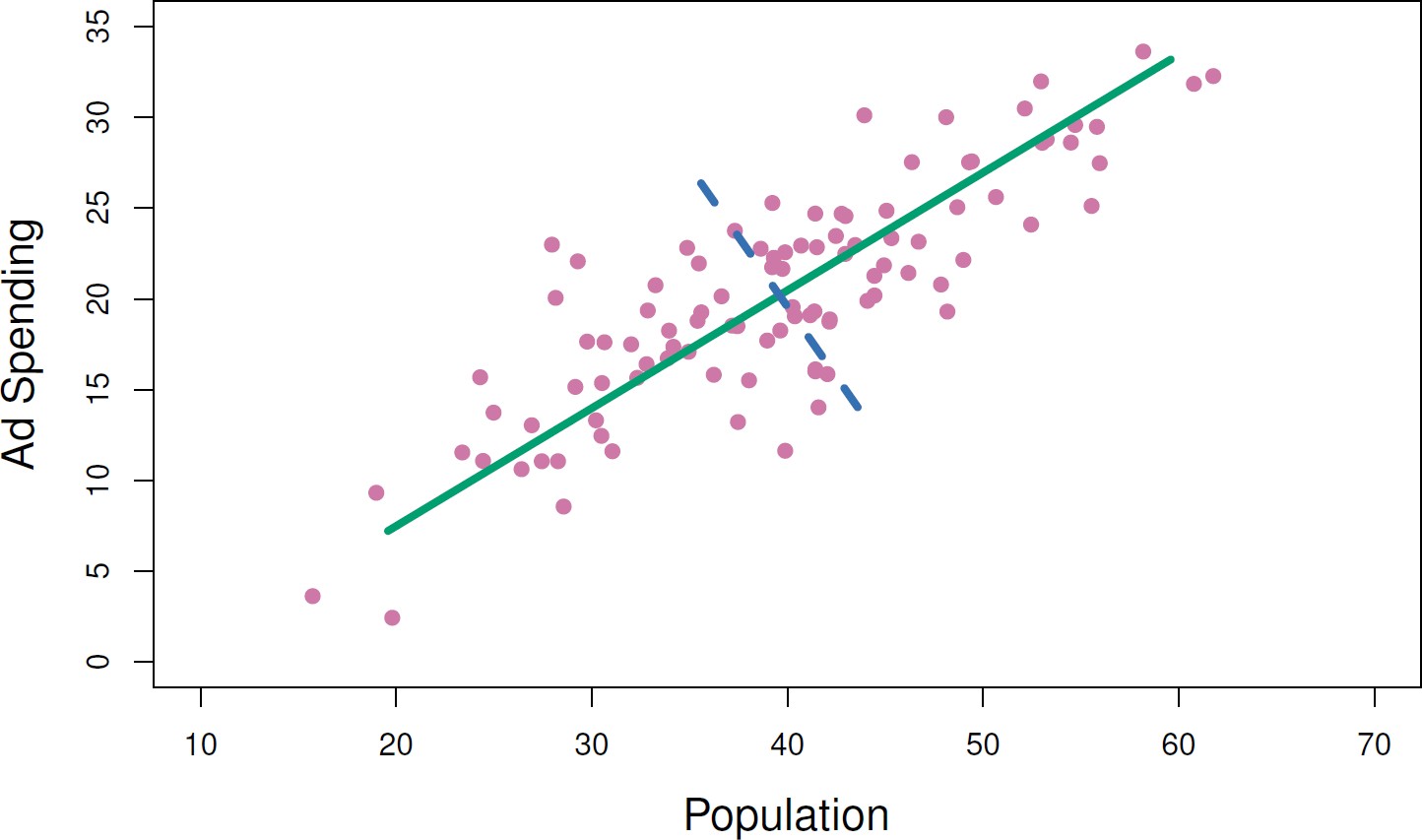
largest variance

* The second principal component has largest variance, subject to being uncorrelated with the first
* And so on
* Hence with many correlated original variables, we replace them with a small set of principal

components that capture their joint variation

©copyright CIEL 2015 **12**

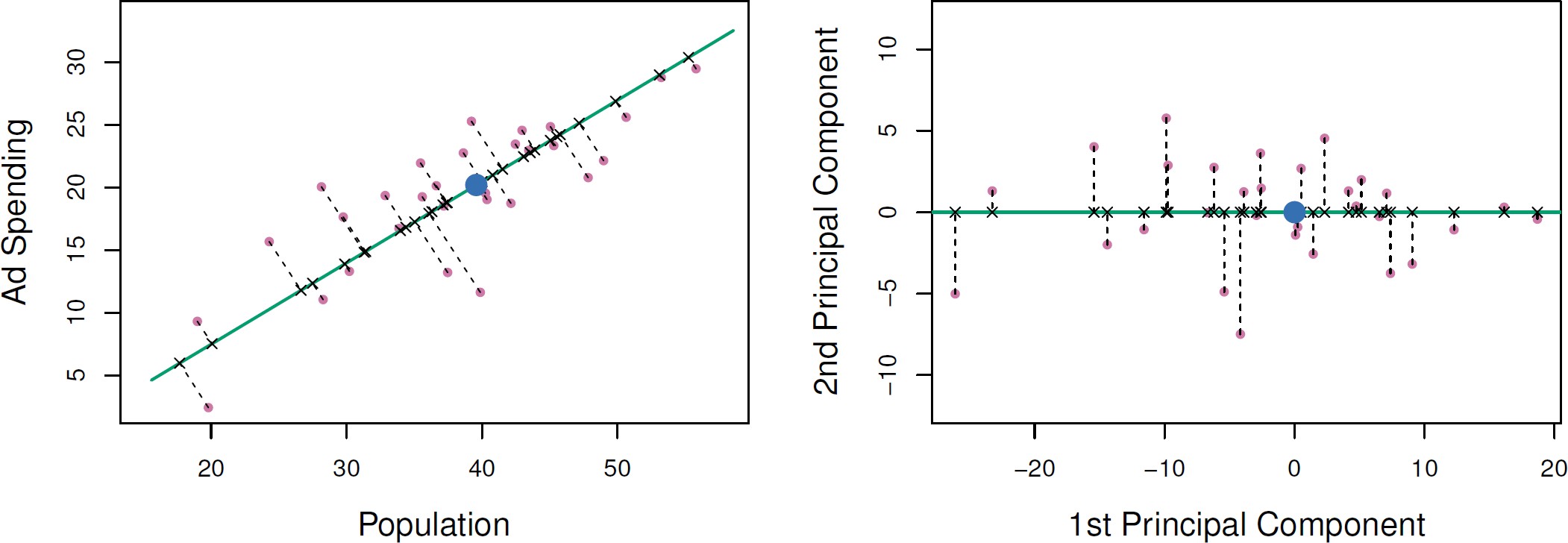
# [Review] Pictures of PCA



* The population size (pop) and ad spending (ad) for 100 different cities are shown as purple circles.
* The green solid line indicates the first principal component, and the blue dashed line indicates the second principal component

©copyright CIEL 2015 **13**

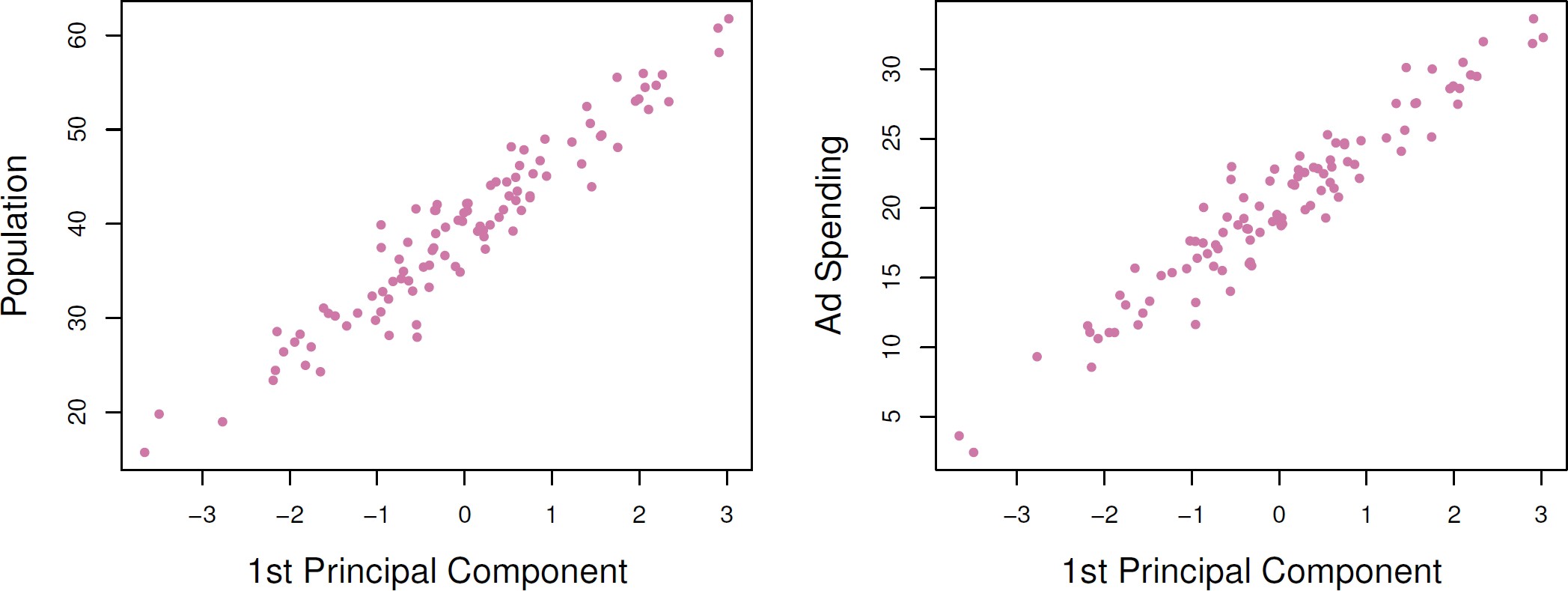
# [Review] Pictures of PCA [cont.]



* A subset of the advertising data
* Left: The first principal component, chosen to minimize the sum of the squared perpendicular distances to each point, is shown in green.
  + These distances are represented using the black dashed line segments.
* Right: The left-hand panel has been rotated so that the first principal component lies on the 𝑥-axis

©copyright CIEL 2015 **14**

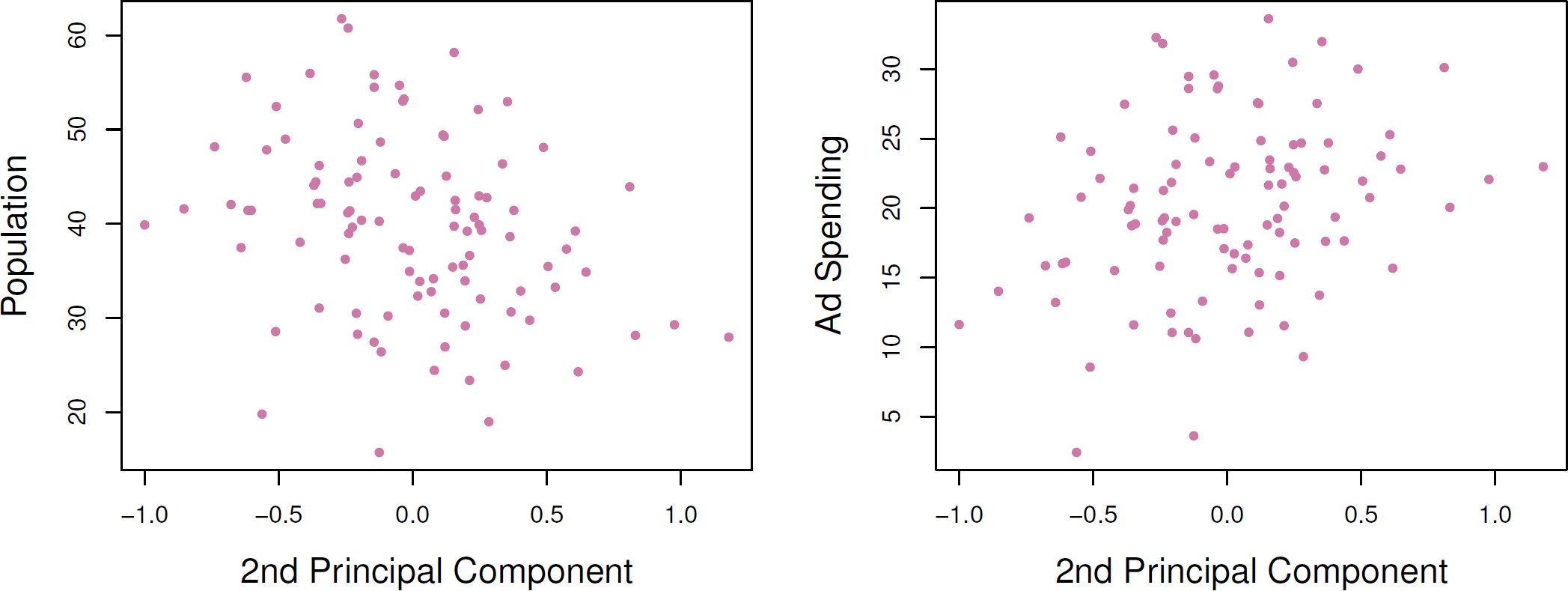
# [Review] Pictures of PCA [cont.]



* Plots of the first principal component scores 𝑧𝑖1 versus pop and ad
* The relationships are strong

©copyright CIEL 2015 **15**

# [Review] Pictures of PCA [cont.]

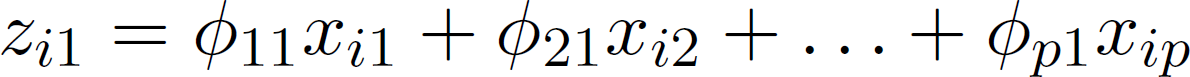


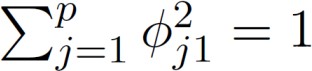
* Plots of the second principal component scores 𝑧𝑖2 versus pop and ad
* The relationships are weak

©copyright CIEL 2015 **16**

# Computation of Principal Components

* Suppose we have a 𝑛 × 𝑝 data set **X**
  + Since we are only interested in variance, we assume that each of the variables in **X** has been centered to have mean zero (that is, the column means of **X** are zero)
* We then look for the linear combination of the sample feature values of the form

 (1)

for 𝑖 = 1, ⋯ , 𝑛 that has largest sample variance, subject to the constraint that 

* Since each of the 𝑥𝑖𝑗 has mean zero, then so does 𝑧𝑖1 (for any values of 𝜙𝑗1)
  + Hence the sample variance of the 𝑧 can be written as 1 σ𝑛 𝑧2

𝑖1

𝑛 𝑖=1

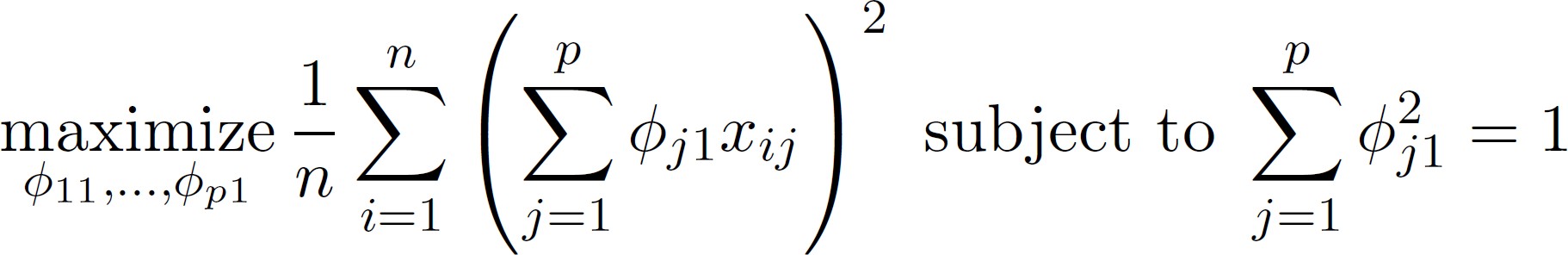
𝑖1



©copyright CIEL 2015 **17**

# Computation of Principal Components [cont.]

* Plugging in (1) the first principal component loading vector solves the optimization problem



* This problem can be solved via a singular-value decomposition (SVD) of the matrix **X**, a standard technique in linear algebra
* We refer to 𝑍1 as the first principal component, with realized values 𝑧11, ⋯ , 𝑧𝑛1

©copyright CIEL 2015 **18**

# Geometry of PCA

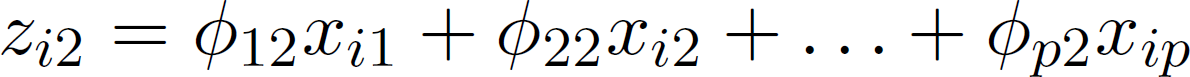
* The loading vector 𝜙1 with elements 𝜙11, 𝜙21, ⋯,𝜙𝑝1 denes a direction in feature space along which the data vary the most
* If we project the 𝑛 data points 𝑥1, ⋯,𝑥𝑛 onto this direction, the projected values are the principal

component scores 𝑧11, ⋯ , 𝑧𝑛1 themselves

©copyright CIEL 2015 **19**

# Further Principal Components

* The second principal component is the linear combination of 𝑋1, 𝑋2, ⋯ , 𝑋𝑝 that has maximal variance among all linear combinations that are uncorrelated with 𝑍1
* The second principal component scores 𝑧11, 𝑧22 ⋯ , 𝑧𝑛2 take the form



* + 𝜙2 is the second principal component loading vector, with elements 𝜙12, 𝜙22, ⋯,𝜙𝑝2

©copyright CIEL 2015 **20**

# Further Principal Components [cont.]

* It turns out that constraining 𝑍2 to be uncorrelated with 𝑍1 is equivalent to constraining the direction 𝜙2 to be orthogonal (perpendicular) to the direction 𝜙1. And so on
* The principal component directions 𝜙1, 𝜙2, 𝜙3, ⋯ are the ordered sequence of right singular

vectors of the matrix **X**, and the variances of the components are 1

𝑛

times the squares of the

singular values

* + There are at most min(𝑛 − 1, 𝑝) principle components

©copyright CIEL 2015 **21**

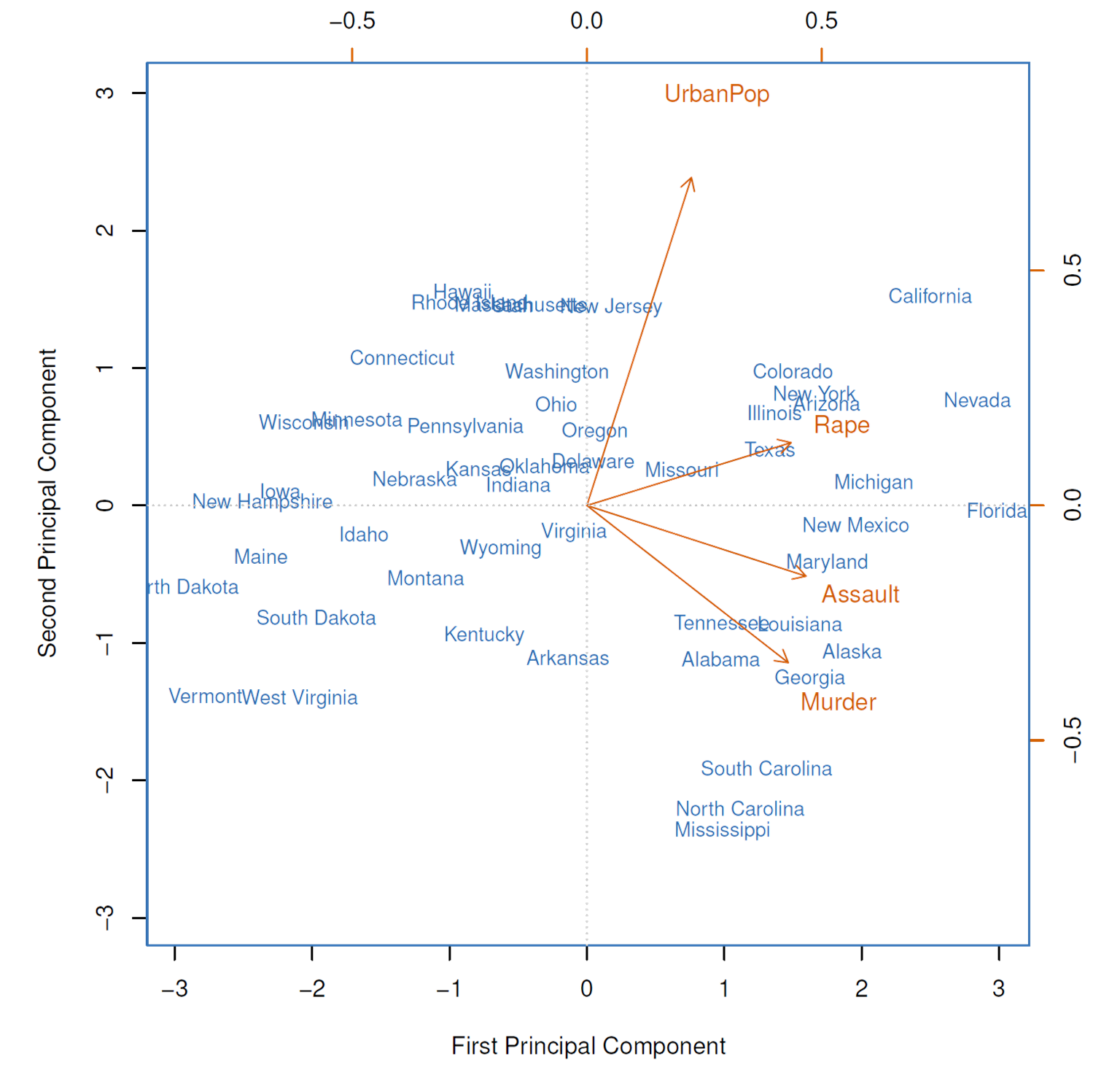
# Illustration

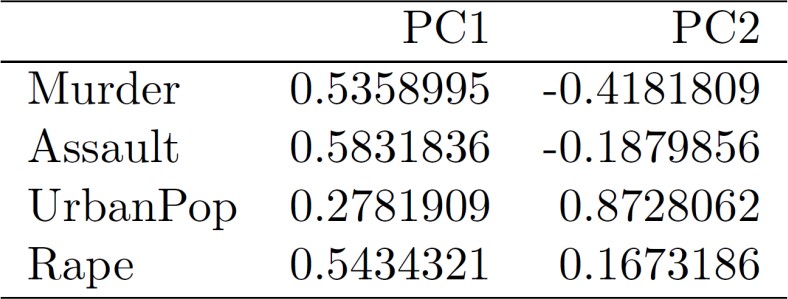
* USAarrests Data
  + For each of the fifty states in the United States, the data set contains the number of arrests per 100,000 residents for each of three crimes: Assault, Murder, and Rape
  + We also record UrbanPop (the percent of the population in each state living in urban areas)
* The principal component score vectors have length 𝑛 = 50, and the principal component loading vectors have length 𝑝 = 4
* PCA was performed after standardizing each variable to have mean zero and standard deviation one.

©copyright CIEL 2015 **22**

USAarrests **Data: PCA Plot**

[PCA loadings]





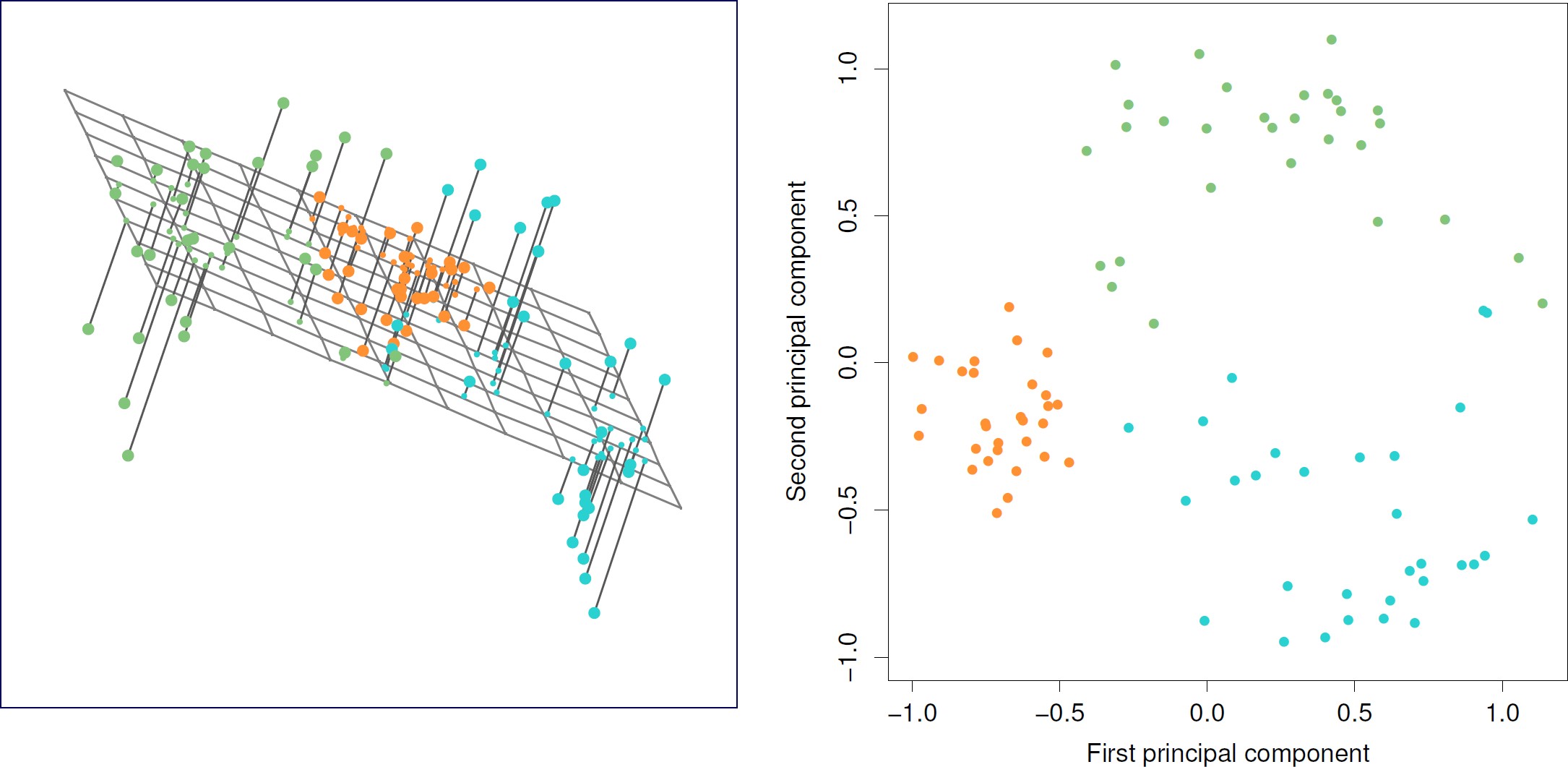
©copyright CIEL 2015 **23**

USAarrests **Data: PCA Plot [cont.]**

* The first two principal components for the USArrests data
  + The blue state names represent the scores for the first two principal components
  + The orange arrows indicate the first two principal component loading vectors (with axes on the top and right). For example, the loading for Rape on the first component is 0.54, and its loading on the second principal component 0,17 [the word Rape is centered at the point (0.54, 0.17)].
  + This figure is known as a biplot, because it displays both the principal component scores and the principal component loadings

©copyright CIEL 2015 **24**

# Another Interpretation of Principal Components



©copyright CIEL 2015 **25**

# PCA Find the Hyperplane Closest to Observations

* The first principal component loading vector has a very special property: it defines the line in 𝑝- dimensional space that is closest to the 𝑛 observations (using average squared Euclidean distance as a measure of closeness)
* The notion of principal components as the dimensions that are closest to the 𝑛 observations extends beyond just the first principal component
* For instance, the first two principal components of a data set span the plane that is closest to

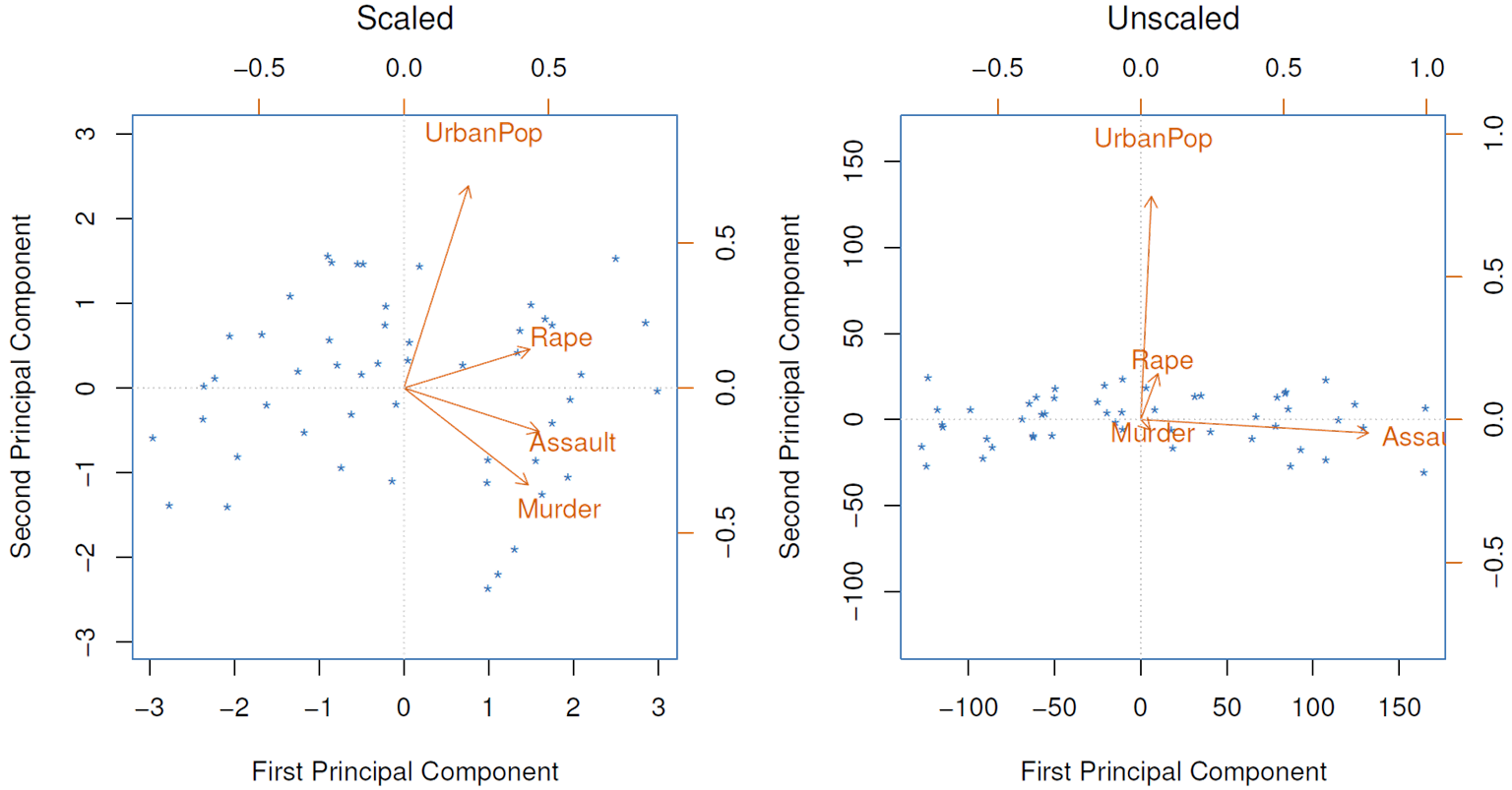
the 𝑛 observations, in terms of average squared Euclidean distance.

©copyright CIEL 2015 **26**

# Scaling of Variables Matters

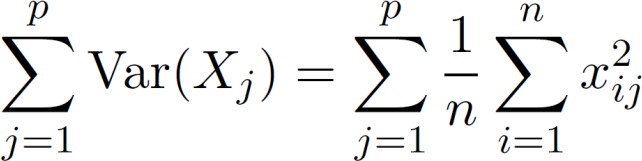
* If the variables are in different units, scaling each to have standard deviation equal to one is recommended
* If they are in the same units, you might or might not scale the variables

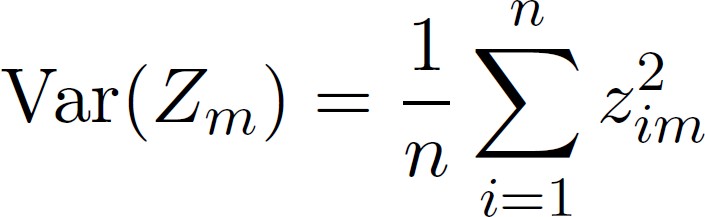
©copyright CIEL 2015 **27**



# Proportion Variance Explained

* To understand the strength of each component, we are interested in knowing the proportion of variance explained (PVE) by each one
* The total variance present in a data set (assuming that the variables have been centered to have

mean zero) is defined as

* + The variance explained by the 𝑚th principal component is
* It can be shown that σ𝑝 Var(𝑋𝑗) = σ𝑀 Var(𝑍𝑚), with 𝑀 = min(𝑛 − 1, 𝑝)

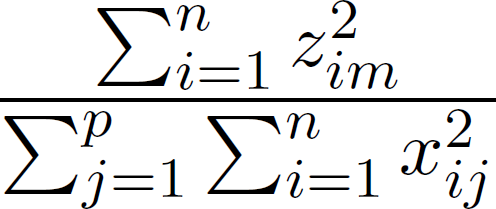
𝑗=1

𝑚=1

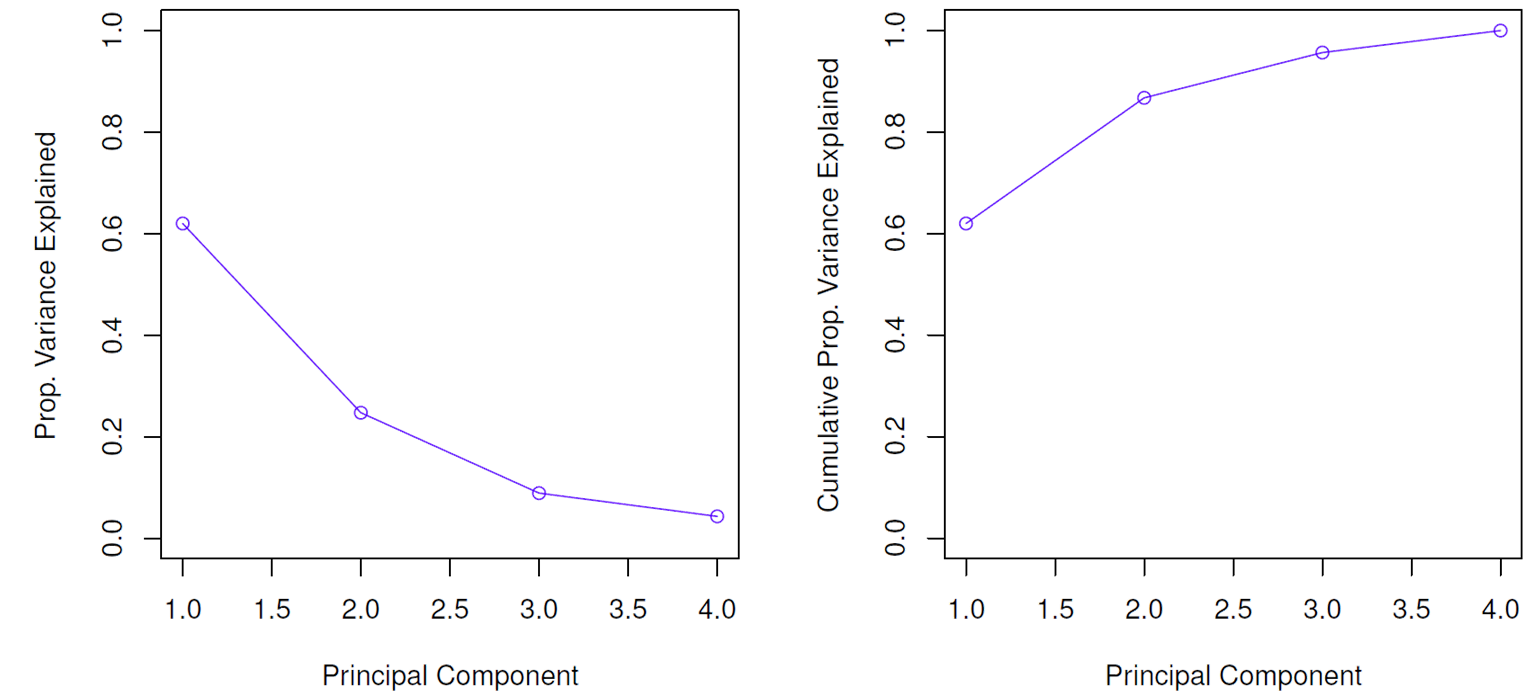


©copyright CIEL 2015 **28**

# Proportion Variance Explained [cont.]

* Therefore, the PVE of the 𝑚th principal component is given by the positive quantity between 0 and 1
* The PVEs sum to one. We sometimes display the cumulative PVEs

**29**



©copyright CIEL 2015

# How Many Principal Components Should We Use?

* If we use principal components as a summary of our data, how many components are sufficient?
  + No simple answer to this question, as cross-validation is not available for this purpose
* The “scree plot” on the previous slide can be used as a guide: we look for an “elbow”.

©copyright CIEL 2015 **30**

# Clustering

* Clustering refers to a very broad set of techniques for finding subgroups, or clusters, in a data set
* We seek a partition of the data into distinct groups so that the observations within each group

are quite similar to each other

* It make this concrete, we must dene what it means for two or more observations to be similar or

different

* Indeed, this is often a domain-specific consideration that must be made based on knowledge of the data being studied

©copyright CIEL 2015 **31**

# PCA vs Clustering

* PCA looks for a low-dimensional representation of the observations that explains a good fraction of the variance
* Clustering looks for homogeneous subgroups among the observations.

©copyright CIEL 2015 **32**

# Clustering for Market Segmentation

* Suppose we have access to a large number of measurements (e.g. median household income, occupation, distance from nearest urban area, and so forth) for a large number of people
* Our goal is to perform market segmentation by identifying subgroups of people who might be more receptive to a particular form of advertising, or more likely to purchase a particular product
* The task of performing market segmentation amounts to clustering the people in the data set.

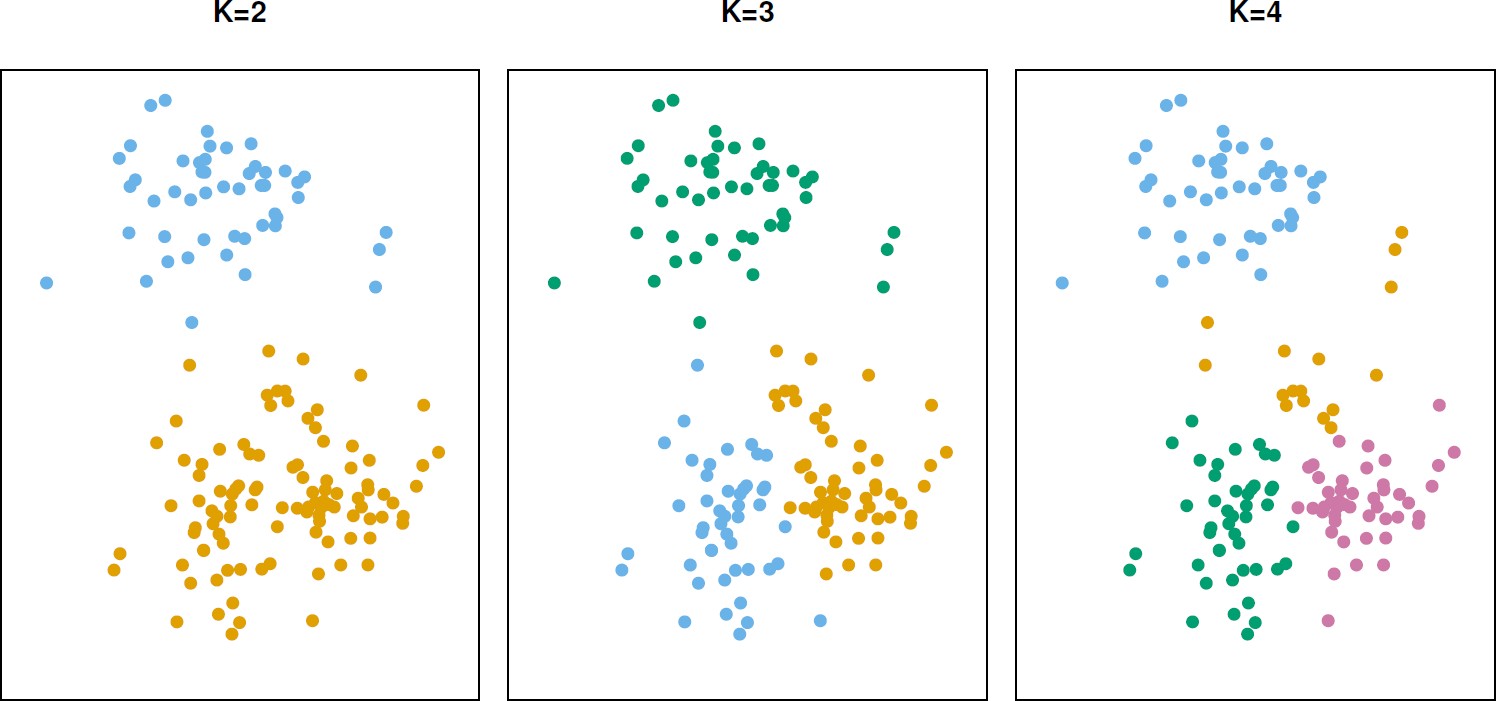
©copyright CIEL 2015 **33**

# Two Clustering 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; in fact, we end up with a tree-like visual representation of the observations, called a dendrogram, that allows us to view at once the clusterings obtained for each possible number of clusters, from 1 to n

©copyright CIEL 2015 **34**

# 𝐾-Means Clustering



* 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 assigned 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

©copyright CIEL 2015 **35**

# 𝐾-Means Clustering [cont.]

* Let 𝐶1, ⋯ 𝐶𝐾 denote sets containing the indices of the observations in each cluster. These sets satisfy two properties:

1. 𝐶1 ∪ 𝐶2 ∪ ⋯ ∪ 𝐶𝐾 = {1, ⋯ , 𝑛}. In other words, each observation belongs to at least one of the *K* clusters
2. 𝐶𝑘 ∩ 𝐶𝑘′ = ∅ for all 𝑘 ≠ 𝑘′. In other words, the clusters are non-overlapping: no observation belongs to more than one cluster

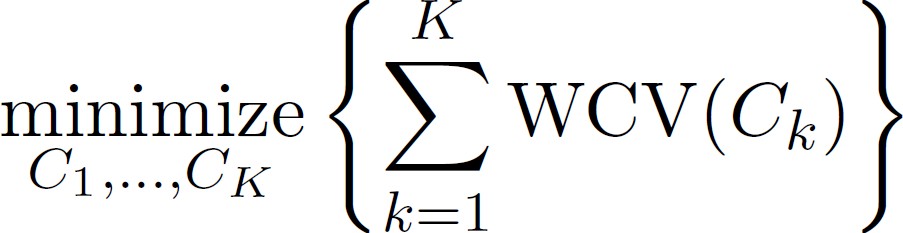
* For instance, if the 𝑖th observation is in the 𝑘th cluster, then 𝑖 ∈ 𝐶𝑘

©copyright CIEL 2015 **36**

# 𝐾-Means Clustering [cont.]

* The idea behind 𝐾-means clustering is that a good clustering is one for which the within-cluster variation is as small as possible
* The within-cluster variation for cluster 𝐶𝑘 is a measure WCV(𝐶𝑘) of the amount by which the

observations within a cluster differ from each other

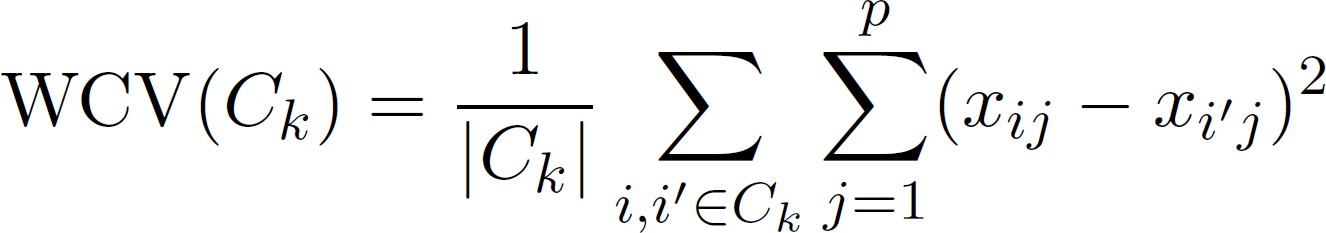
* Hence we want to solve the problem

(2)

* In words, this formula says that we want to partition the observations into 𝐾 clusters such that the total within-cluster variation, summed over all 𝐾 clusters, is as small as possible

©copyright CIEL 2015 **37**

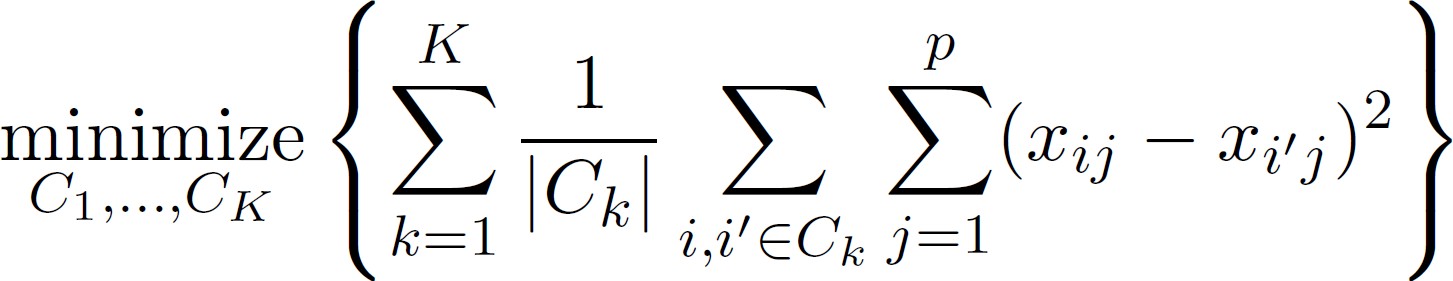
# How to Define Within-Cluster Variation?

* Typically we use Euclidean distance

(3)



* + 𝐶𝑘 denotes the number of observations in the 𝑘th cluster
* Combining (2) and (3) gives the optimization problem that denes K-means clustering,

(4)

©copyright CIEL 2015 **38**

# 𝐾-Means Clustering 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:
   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

* 1. Assign each observation to the cluster whose centroid is closest (where closest is defined using Euclidean distance)

©copyright CIEL 2015 **39**

# Properties of Algorithm

* This algorithm is guaranteed to decrease the value of the objective (4) at each step. Why? Note that
  + ��

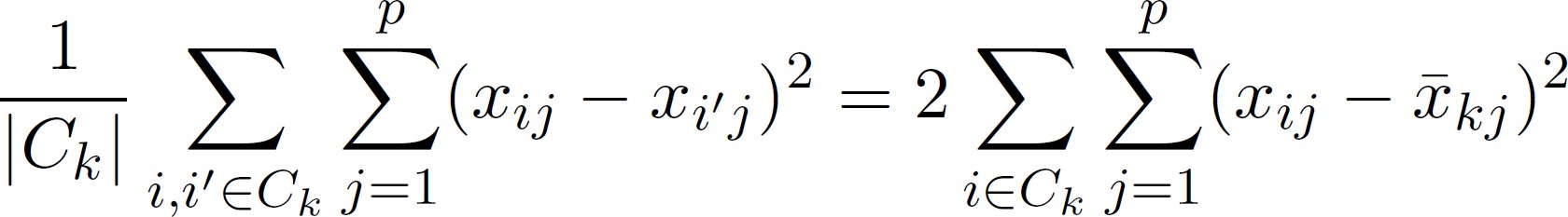
𝑘𝑗

= 1

|𝐶𝑘|

σ𝑖∈𝐶𝑘

𝑥𝑖𝑗



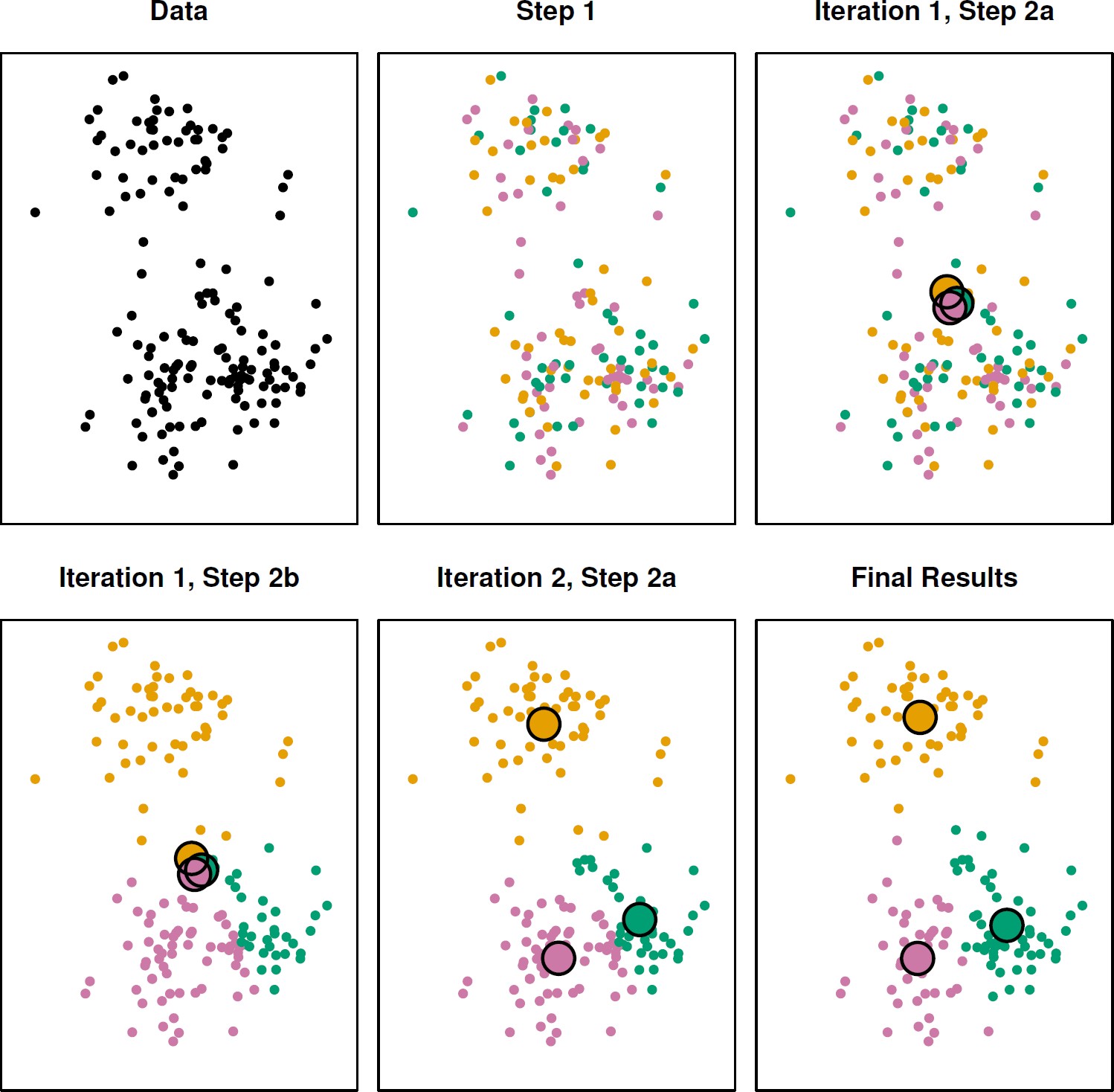
is the mean for feature 𝑗 in cluster 𝐶𝑘



* However it is not guaranteed to give the global minimum

©copyright CIEL 2015 **40**

# Properties of Algorithm: Example

* The progress of the 𝐾-means algorithm with 𝐾 = 3
  + Top left: The observations are shown
  + Top center: In Step 1 of the algorithm, each

observation is randomly assigned to a cluster

* + Top right: In Step 2(a), the cluster centroids are computed. These are shown as large colored disks. Initially the centroids are almost completely overlapping because the initial cluster assignments were chosen at random
  + Bottom left: In Step 2(b), each observation is assigned to the nearest centroid
  + Bottom center: Step 2(a) is once again performed,

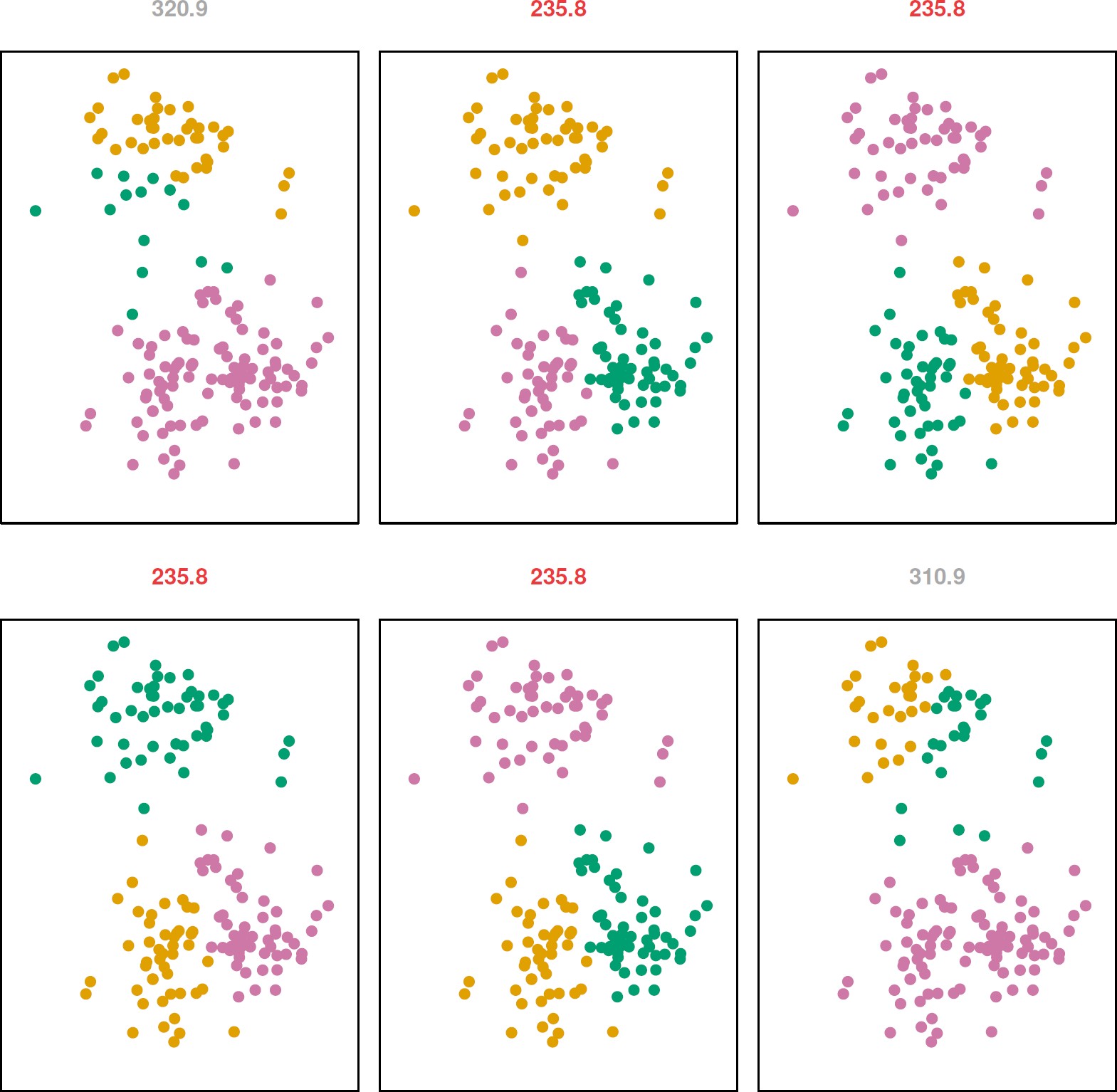
leading to new cluster centroids

* + Bottom right: The results obtained after 10 iterations

©copyright CIEL 2015 **41**

# Properties of Algorithm: Example – Different Starting Values

* 𝐾-means clustering performed six times on the data from previous figure with 𝐾 = 3, each time with a different random assignment of the observations in Step 1 of the 𝐾 –means algorithm
* Above each plot is the value of the objective (4)
* Three different local optima were obtained, one of which resulted in a smaller value of the objective and provides better separation between the clusters
* Those labeled in red all achieved the same best solution, with an objective value of 235.8



©copyright CIEL 2015 **42**

# Hierarchical Clustering

* 𝐾-means clustering requires us to pre-specify the number of clusters 𝐾
  + This can be a disadvantage (later we discuss strategies for choosing 𝐾)
* Hierarchical clustering is an alternative approach which does not require that we commit to a

particular choice of 𝐾

* In this section, we describe bottom-up or agglomerative clustering
  + This is the most common type of hierarchical clustering, and refers to the fact that a dendrogram is built

starting from the leaves and combining clusters up to the trunk

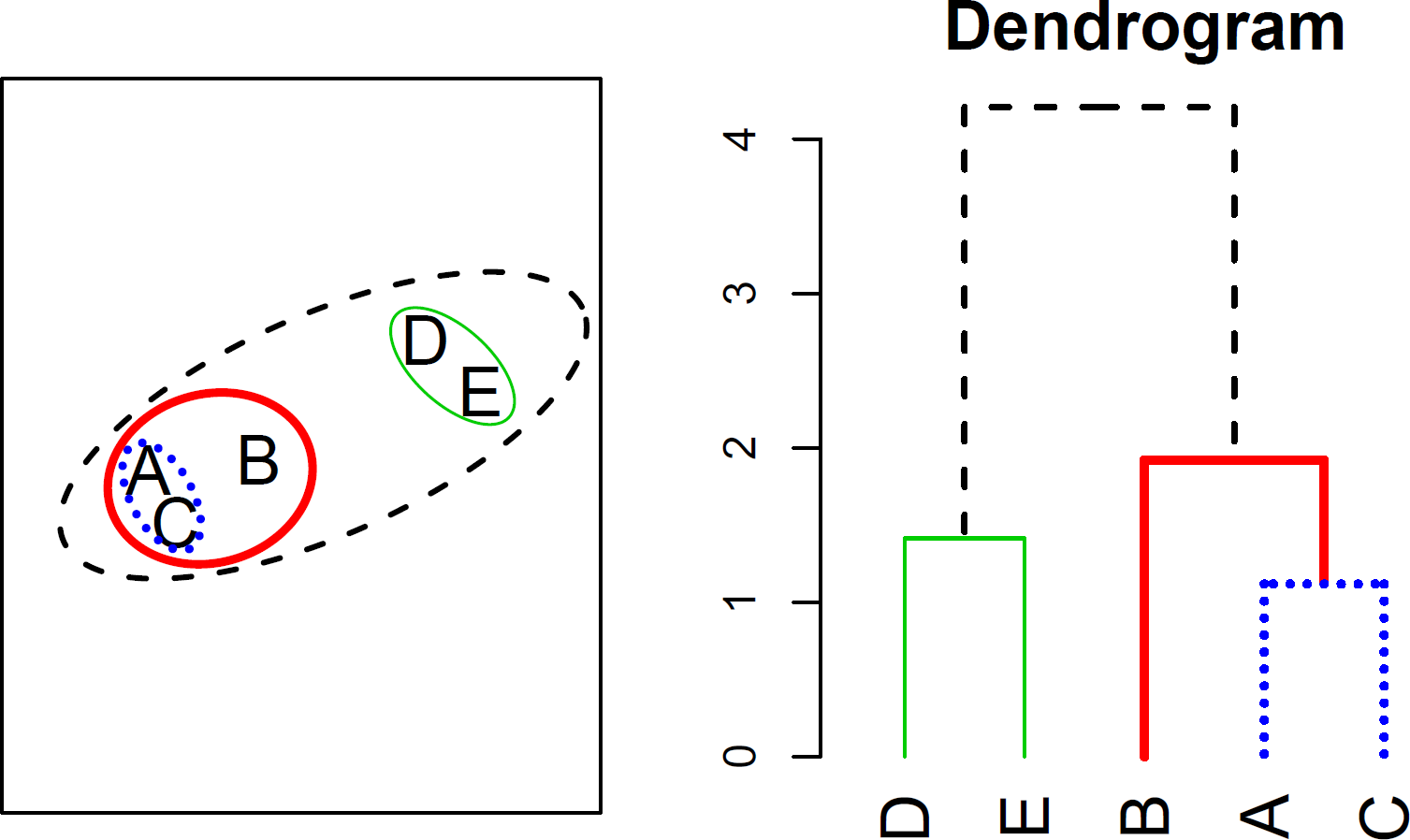
©copyright CIEL 2015 **43**

# Hierarchical Clustering: Idea

* Builds a hierarchy in a “bottom-up” fashion...

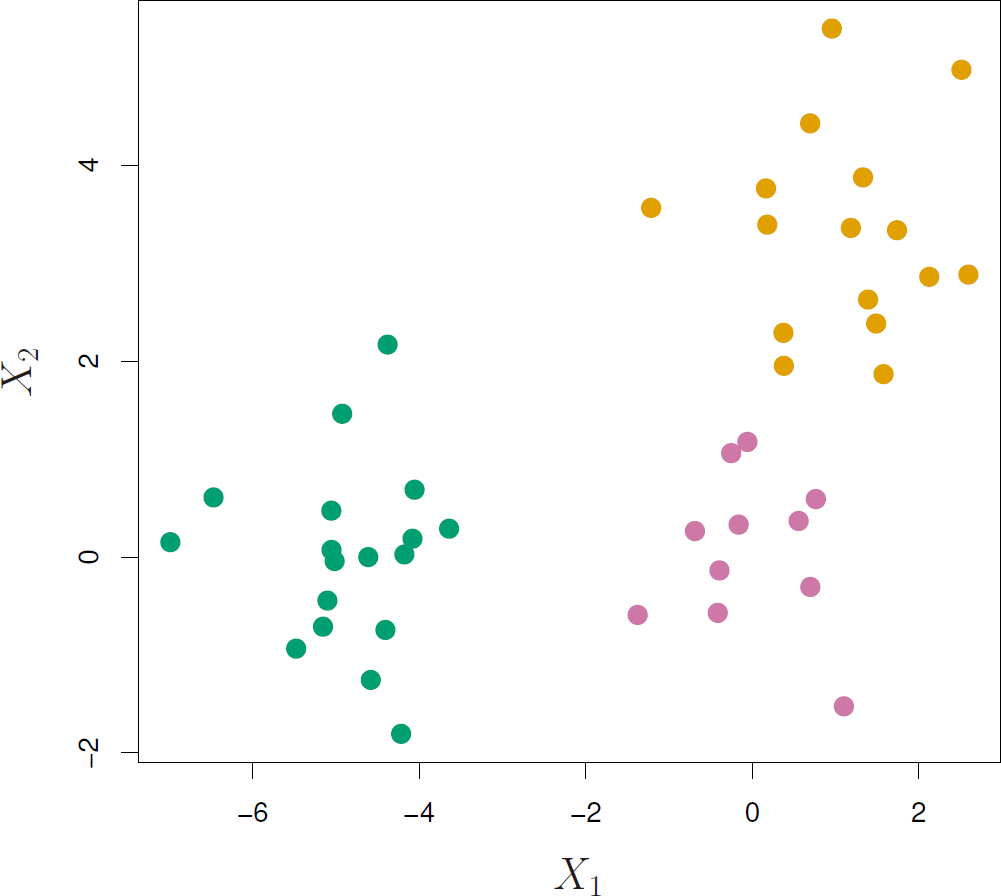
©copyright CIEL 2015 **44**

# Hierarchical Clustering Algorithm

* The approach in words:
  + Start with each point in its own cluster
  + Identify the closest two clusters and merge them
  + Repeat
  + Ends when all points are in a single cluster

©copyright CIEL 2015 **45**

# An Example

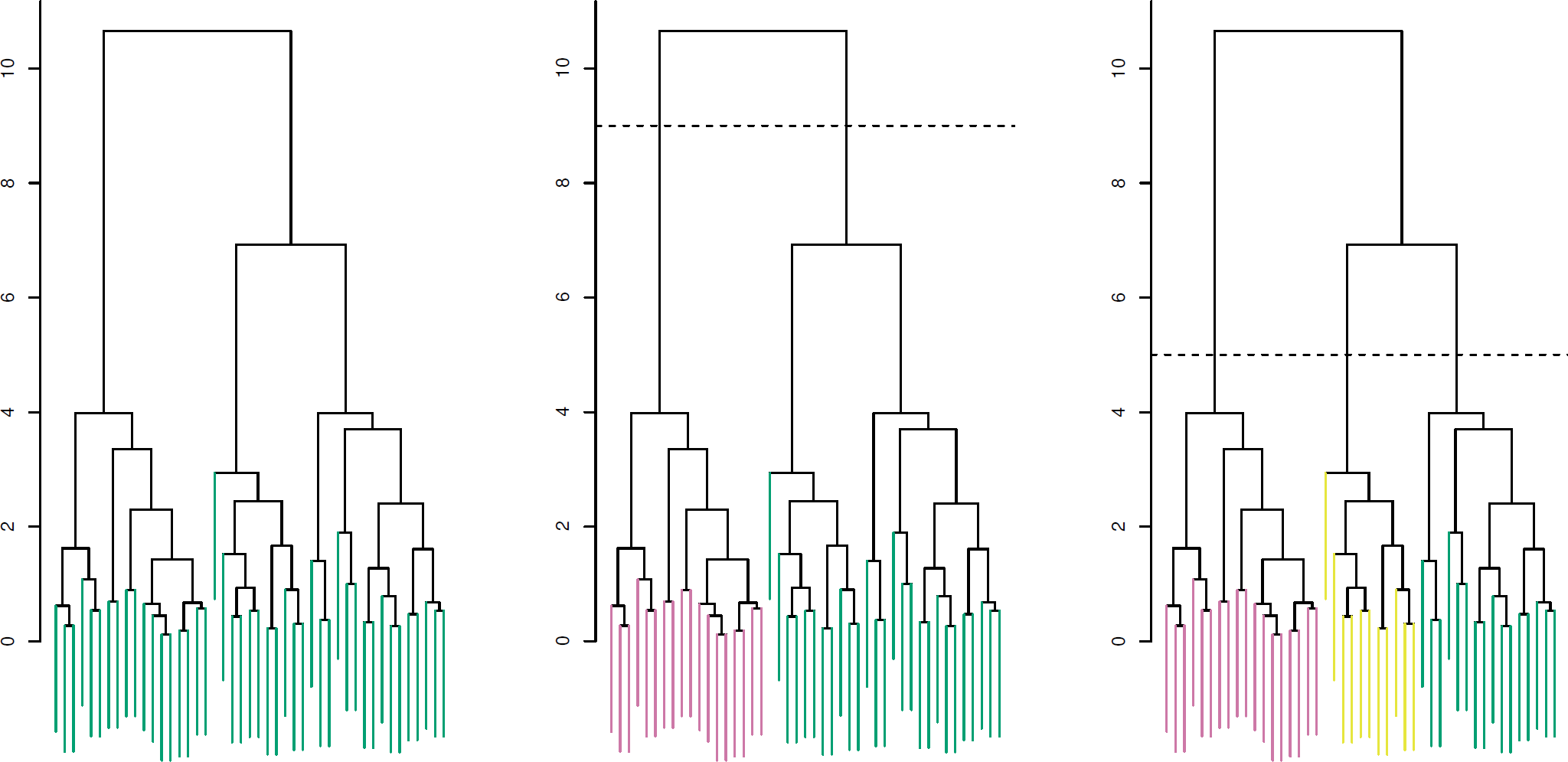
* 45 observations generated in 2- dimensional space
* In reality there are three distinct classes,

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.

©copyright CIEL 2015 **46**

# Application of Hierarchical Clustering

* Left: Dendrogram obtained from hierarchically clustering the data from previous slide, with complete linkage and Euclidean distance
* Center: The dendrogram from the left-hand panel, cut at a height of 9 (indicated by the dashed line)
  + This cut results in two distinct clusters, shown in different colors
* Right: The dendrogram from the left-hand panel, now cut at a height of 5
  + This cut results in three distinct clusters, shown in different colors.
  + Note that the colors were not used in clustering, but are simply used for display purposes in this figure

©copyright CIEL 2015 **47**

# Types of Linkage

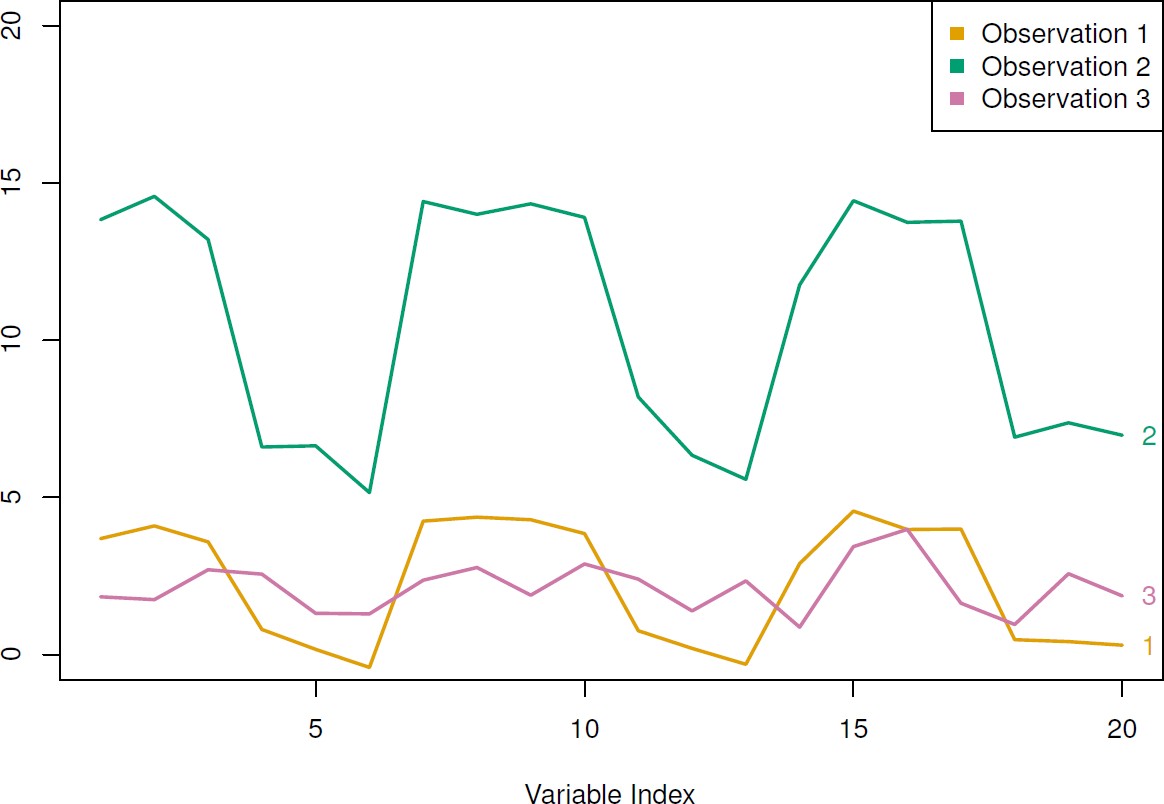
|  |  |
| --- | --- |
| **Linkage** | **Description** |
| Complete | Maximal inter-cluster dissimilarity. Compute all pairwise dissimilarities between the observations in cluster A and the observations in cluster B, and record the largest of these dissimilarities |
| Single | Minimal inter-cluster dissimilarity. 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 |

©copyright CIEL 2015 **48**

# Choice of Dissimilarity Measure

* So far have used Euclidean distance
* An alternative is correlation-based distance which considers two observations to be similar if

their features are highly correlated



©copyright CIEL 2015 **49**

# Practical Issues

* Scaling of the variables matters! Should the observations or 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

©copyright CIEL 2015 **50**



**Summary & Next Class**

* Unsupervised learning

## Summary & Next class

©copyright CIEL 2015 https://ciel.seoultech.ac.kr/

# Summary

* Unsupervised learning is important for understanding the variation and grouping structure of a set of unlabeled data, and can be a useful pre-processor for supervised learning
  + Principle component analysis
  + 𝐾-means clustering
  + Hierarchical clustering
* It is intrinsically more difficult than supervised learning because there is no gold standard (like an outcome variable) and no single objective (like test set accuracy).
* It is an active field of research, with many recently developed tools such as self-organizing maps, independent components analysis and spectral clustering.
  + See The Elements of Statistical Learning, chapter 14.

©copyright CIEL 2015 **52**

# Course Schedule (Tentative)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Week** | **Topics** | **Note** | **Date (W)** | **Date (M)** |
| 1 | Orientation, Statistical Learning (Ch2) | Online | 03/03 | 03/08 |
| 2 | Statistical Learning (Ch2), Python Programming | Online | 03/10 | 03/15 |
| 3 | Probability & Statistics | Online | 03/17 | 03/22 |
| 4 | Probability & Statistics | Online | 03/24 | 03/29 |
| 5 | Linear Regression (Ch3) | Online | 03/31 | 04/05 |
| 6 | Linear Regression (Ch3) | Online | 04/07 | 04/12 |
| 7 | Classification (Ch4) | Online | 04/14 | 04/19 |
| 8 | **Midterm exam** | **Class hours (W1-W7)** | **04/21** | **04/26** |
| 9 | Resampling Methods (Ch5) | Online | 04/28 | 05/03 |
| 10 | Linear Model Selection and Regularization (Ch6) | Online | 05/05 | 05/10 |
| 11 | Moving Beyond Linearity (Ch7) | Online | 05/12 | 05/17 |
| 12 | Tree-Based Methods (Ch8) | Online | 05/19 | 05/24 |
| 13 | Support Vector Machines (Ch9) | Online | 05/26 | 05/31 |
| 14 | Unsupervised Learning (Ch10) | Online | 06/02 | 06/07 |
| **15** | **Final exam** | **6pm (W9-W14)** | **06/10Th** | **06/10Th** |

©copyright CIEL 2015 **53**