# Segmentation and Cluster Analysis

MKTG 6279

Week 3

# Today

- Segmentation with regression (supervised)
- Example 1: Amazon online transactions
- Segmentation with cluster analysis (unsupervised)
- Example 2: Everlane retail sales

# Final Project

For the final presentation, pretend I am your client interested in a marketing problem.

It is up to you and your group to come up with the marketing problem and run the analysis.

In the final presentation (15 minutes) you will need to:

1) Outline the business problem
2) Describe the data
3) Talk about the model you used and why
4) Provide recommendations and **show the impact**

# More details

In 15 minutes, that may mean only about 10 slides. Less is more!

For those doing the practicum, you can use that dataset.

**Remember, I am an executive! Keep it relatively simple, succint, and actionable.**

Deliverables (besides the actual presentation): a writeup no more than 5 pages and your slides.

# Segmentation with regression

# Indicators and interactions

Most people don't realize how flexible linear regression is when the variables are used correctly

Indicator (dummy) variables and interaction effects control for differences between predefined segments of the data

- Dummy variables shift the expected mean up or down by group
- Interaction terms adjust the *marginal effect* of a particular variable of interest by group

# Classic amateur analyst idea

"In my data I think the effects of price on sales will be different by country, so I'm thinking of running a separate regression for each country"

There is no need to do this, just add an indicator variable by country

# Classic amateur analyst response

"But that won't work because the slopes might be different, not just the baseline demand (i.e., the intercept)"

Then add an interaction term between price and country

# Regression equation

In general:

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \ldots + \varepsilon, \varepsilon \sim N(0, \sigma^2)$$

Data for simple example:

```
load('data/simpleRegression.rdata')
head(myData)
```

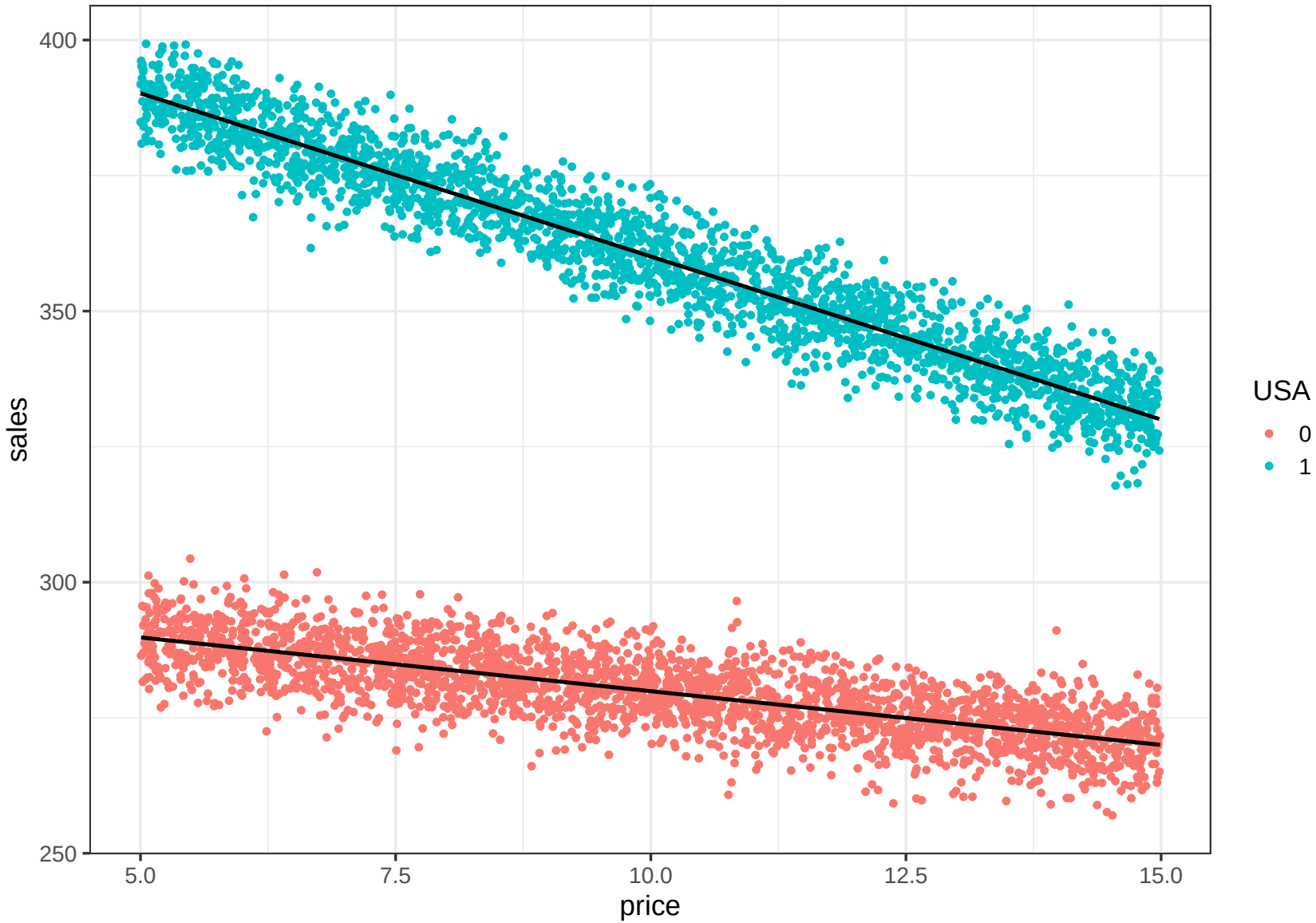
```
##   usa      price    sales
## 1   0  5.354995 286.1577
## 2   0  8.557036 283.8041
## 3   1  7.488729 370.8895
## 4   1 13.791804 345.2256
## 5   0  8.179789 285.2880
## 6   1  8.211205 366.6304
```

# Average sales by country

```
myData %>%
  group_by(usa) %>%
  summarise(avgSales = mean(sales))
```

```
## # A tibble: 2 x 2
##     usa avgSales
##   <dbl>    <dbl>
## 1     0     280.
## 2     1     360.
```

# Responsiveness to price



# Basic and lame amateur approach

```
data_mex = subset(myData,usa == 0)
out_mex  = lm(sales ~ price,data_mex)
round(coef(out_mex),2)
```

```
## (Intercept)       price
##      299.72       -1.98
```

```
data_usa = subset(myData,usa == 1)
out_usa  = lm(sales ~ price,data_usa)
round(coef(out_usa),2)
```

```
## (Intercept)       price
##      420.26       -6.02
```

# Professional approach

$$sales = \beta_0 + \beta_1 price + \beta_2 usa + \beta_3 (price \times usa) + \varepsilon$$

In this model you still allow for different intercepts and slopes by country

```
out = lm(sales ~ price * usa,myData)
round(coef(out),2)
```

```
## (Intercept)       price         usa   price:usa
##      299.72       -1.98      120.54       -4.04
```

# How? Walk through the math

When usa $= 0$ and price $= 5$

`sales = 299.72 - 1.98*5 + 120.54*0 - 4.04*5*0 = 289.82`

which is the same as `out_mex`:

`sales = 299.72 - 1.98*5 = 289.82`

# What about for USA observations?

When usa $= 1$ and price $= 5$

`sales = 299.72 - 1.98*5 + 120.54*1 - 4.04*5*1 = 390.16`

which is the same as:

`sales = (299.72 + 120.54) + (-1.98 - 4.04)*5 = 390.16`

which is the same as `out_usa`:

`sales = 420.26 - 6.02*5 = 390.16`

# Benefits

- You appear to know what you're doing
- Less work, only one model
- More elegant solution, easier to manage
- Coefficients that *don't* need to be different by segment are more precise because they can use all observations

# How do I know when to do this?

Your exploratory analysis: plots with relationships broken into main groups might highlight interactions

Your intuition: is there reason to believe the price sensitivity varies by country?

# Get REALLY comfortable with this

- Interaction terms allow for synergies between combinations of variables
- Indicator variables shift predictions up or down
- Polynomials allow for non-linear effects (not used as often, besides simple transformations)

**These three strategies will solve 95% of your modeling needs**

# Segmentation Example: Amazon Transactions

```
load('data/amazon_segmentation.rdata')
```

Transaction and demographic information from 400 Amazon users

# Cluster Analysis

# Cluster Analysis

In regression we talk about models of $y|x$

Clustering is all about models for $x$ alone (i.e. **un**supervised)

In clustering there are no pre-defined segments

Cluster analysis classifies a set of observations into mutually exclusive *unknown* groups based on combinations of variables

# Clustering: unsupervised dimension reduction

The purpose of cluster analysis is to organize observations (usually people) into groups, where members of a group share properties in common.

- Demographic clusters: yuppies, hipsters, etc.
- Consumption clusters: frequent loyalists, infrequent loyalists

# Example: Claritas PRIZM

Enter a 5 digit zip code: 75205 Submit

PRIZM Premier | P$YCLE | ConneXions

The most common segments for ZIP Code 75205 Dallas, TX are:

**04 Young Digerati**
Wealthy Middle Age Mostly w/ Kids

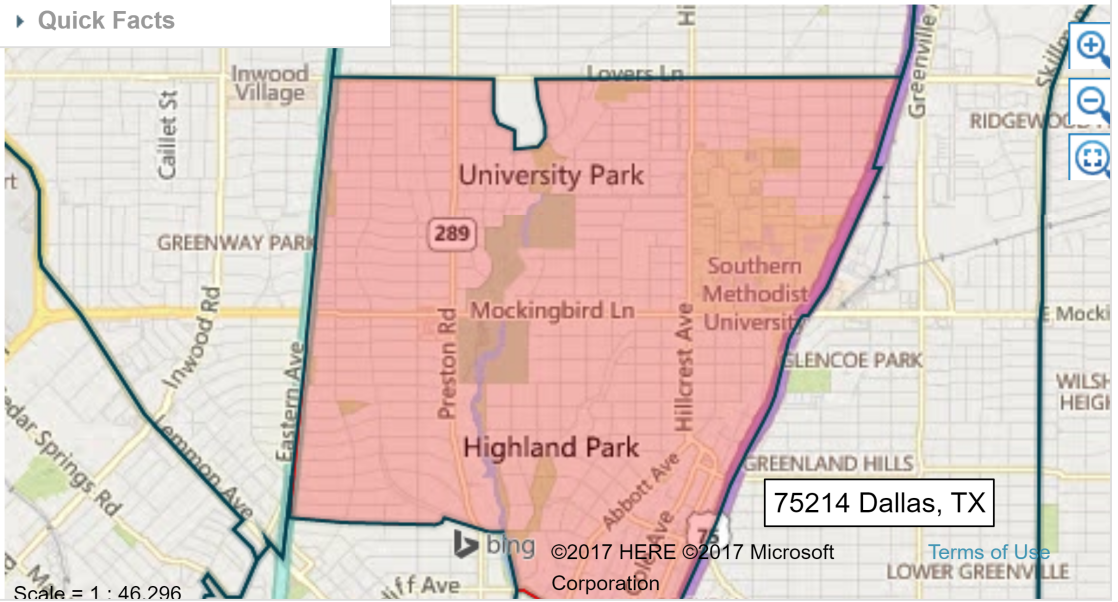
**07 Money & Brains**
Upscale Older Mostly w/o Kids

**17 Urban Elders**
Midscale Middle Age Mostly w/o Kids

**21 The Cosmopolitans**
Upscale Younger Family Mix

**31 Connected Bohemians**
Midscale Younger Mostly w/o Kids

Quick Facts




Population by Race & Ethnicity

# Young Digerati

## 04 Young Digerati

### Wealthy Middle Age Mostly w/ Kids

Young Digerati are tech-savvy and live in fashionable neighborhoods on the urban fringe. Affluent and highly educated, Young Digerati communities are typically filled with trendy apartments and condos, fitness clubs and clothing boutiques, casual restaurants and all types of bars, from juice to coffee to microbrew. Many have chosen to start families while remaining in an urban environment.

**Social Group:** 01 Urban Uptown
**Lifestage Group:** 01 Midlife Success

SnapShot | Neighborhood Demographics | Household Demographics | Lifestyles | Media | Premium

### 2018 Statistics

**US Households:** 1,841,200 (1.49%)
**Median Household Income:** $128,498

### Lifestyle & Media Traits

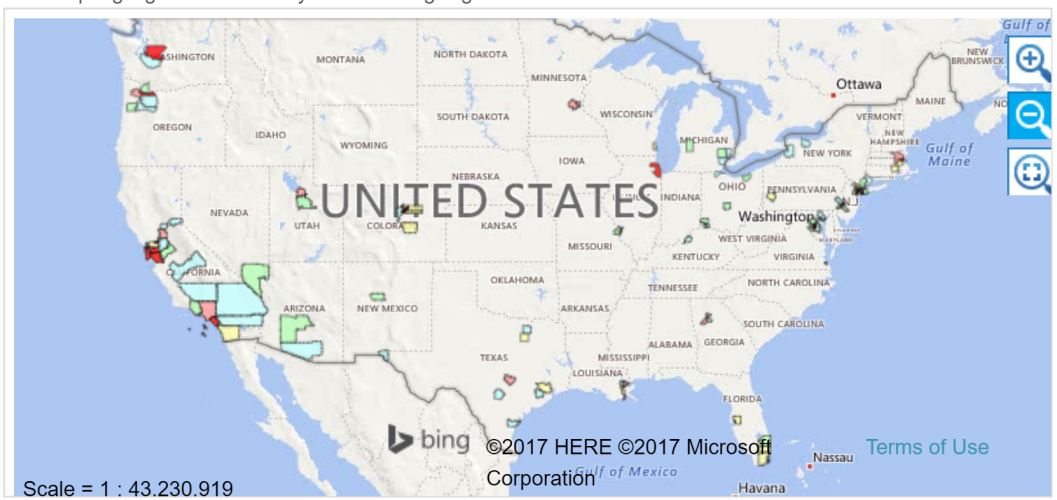
- Owns a Mercedes
- Eats at Chipotle
- Shops at Banana Republic
- Goes hiking/backpacking
- Visits Asia
- Uses Yelp
- Listens to Alternative

### Demographics Traits

- **Urbanicity:** Urban
- **Income:** Wealthy

### US by County

This map highlights each County where Young Digerati are found.




### Top 5 Counties

# Clustering methods

There are many algorithms to choose from in cluster analysis

- Each choice may result in a different grouping structure!

Hierarchical Methods

Easier to read and interpret, but tends to be unstable

- Agglomerative: start with all objects in their own cluster, and then combine
- Divisive: starts with all objects in one cluster and subdivides

*I won't go over hierarchical methods, but there are some slides in the back

# Non-Hierarchical methods

Allows objects to leave one cluster and join another as the clusters form

The output is more stable across samples, sometimes hard to interpret

Computationally faster for large datasets

- K-means clustering
- Probabilistic (K-Means Mixture)
- DBSCAN
- Self-organized map (forces multi-dimensions into two dimensions)

# Clustering and segmentation

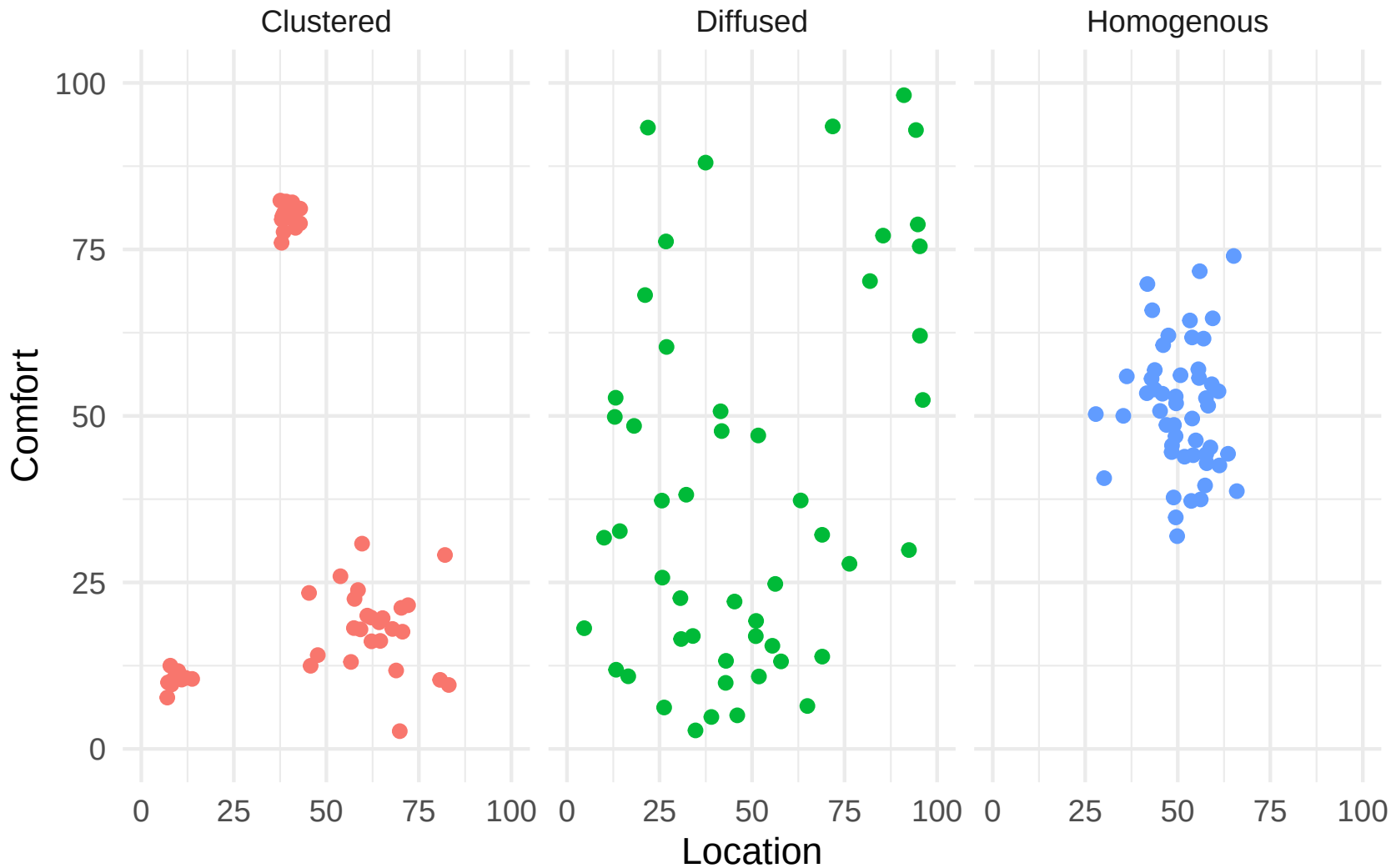
Market segmentation involves aggregating prospective buyers into groups that:

1) Have common needs
2) Will respond similarly to marketing actions

The groups that result from the market segmentation process are called market segments, a relatively homogeneous collection of prospective buyers.

# Basic market-preference patterns

## Hotel clustering: amenities vs. location



# Segmention is important

1) Consumers are not homogeneous
2) Opportunity for competitive advantage
3) Greater marketing effectiveness

# The clustering process

1) Select variables on which to cluster
2) Select similarity measure and scale the variables
3) Select a clustering method
4) Determine the number of clusters
5) Conduct the analysis
6) Define and name clusters

# Selecting the variables

Which variables do you want to use?

- Benefits sought: "How important is price to you?"
- Demographics: age, income, etc.
- Behavior: recency, frequency, monetary
- Etc: CLV, share of wallet, etc.

Depends on the *strategic reason* for segmentation

- For a new product application, probably want to cluster on benefits sought

Including irrelevant variables can strongly affect the solution

# Similarity measures

Broadly there are two types:

**Distance**

- By far the most common (our focus)
- Appropriate when variables are measured on a common metric

**Matching**

- Used for categorical variables
- Ratio of number of matching attributes to total number of attributes
- Most software packages don't even have this

# Measuring similarity

Data Matrix:

| . | $x_1$ | $x_2$ | $x_3$ | $x_4$ |
|---|---|---|---|---|
| $c_1$ | | | | |
| $c_2$ | | | | |
| $c_3$ | | | | |

Proximity Matrix:

| . | $c_1$ | $c_2$ | $c_3$ |
|---|---|---|---|
| $c_1$ | | | |
| $c_2$ | $d_{21}$ | | |
| $c_3$ | | | |

$d_{ij}$ is the **proximity** or **distance** between customers $i$ and $j$

We collapse the four data variables into one measure

# Distance measure

The most popular distance measure is the Euclidean distance:

$$d_{ij} = \sqrt{\sum_{p=1}^{P} (x_{ip} - x_{jp})^2}$$

$P$ is the dimension of $X$

If the variables are measured in different units, one variable could overwhelm the other

Standardizing is a possible solution (subtract the mean, divide by standard deviation)

Weights can also be applied

# *Grouping individuals

Several methods for grouping customers into clusters, here are the most common ones:

- Single linkage: nearest-neighbor approach, measure based on whomever is closest
- Complete linkage: furthest-neighbor approach, criterion based on furthest distance
- Average linkage: as similar as the average similarity to all objects between clusters
- Centroid method: distance between two clusters is the distance between their centroids
- Ward's method: minimizes within-cluster variation

# Choose the clustering algorithm: K-means

Most popular clustering method

User supplies $K$, the number of clusters, and each observation is assigned to one of the $K$ clusters

Compare solutions for different values of $K$ based on interpretability, managerial usefulness, and goodness of fit

The measure of goodness of fit is based on:

$$\frac{\text{avg. distance between all pairs of objects within a cluster}}{\text{total distance between clusters}}$$

Within group variance **must** decrease as $K$ increases

# Basic K-Means algorithm

- Begin with initial random partitioning of all respondents into $K$ clusters
- Compute the mean (called the centroid) of each cluster
- Reassign respondents based on some criterion (e.g., assign to the cluster with the closest centroid) such that within-cluster variability reduces
- Recompute means
- Continue until no objects change cluster assignments

# Example: The Home Depot

You work as an analyst for The Home Depot

You are trying to segment customers based on standard RFM metrics

- Recency, frequency, monetary

Interest is only in customers from the past 3 years, so ignore the recency metric

# The data

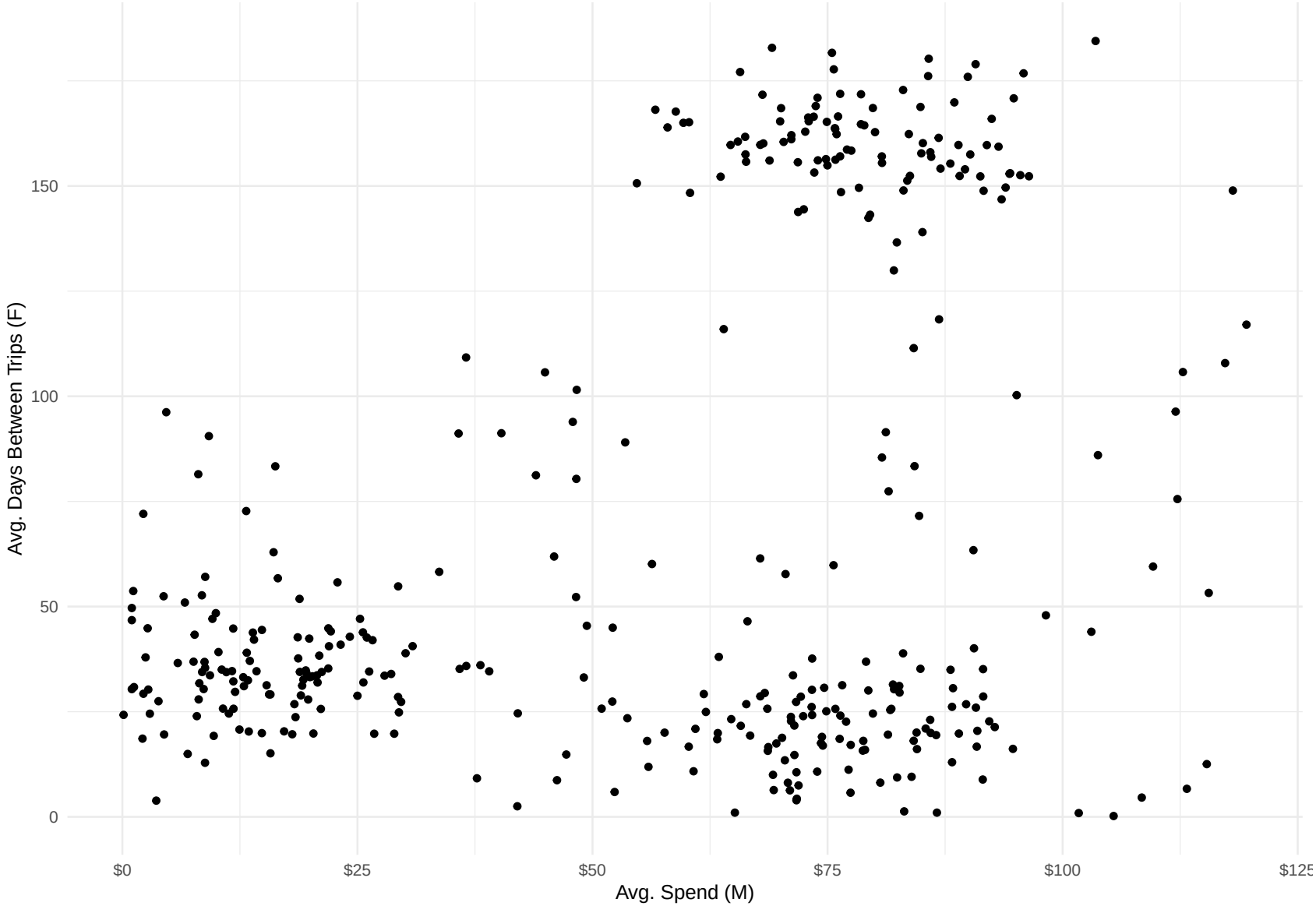
Data from 375 customers:

- Average spend per trip
- Average days between purchases

```
load('data/homedepot.rdata')
str(homedepot)
```

```
## 'data.frame':    375 obs. of  2 variables:
##  $ spend       : num  8.74 6.64 18.3 19.87 20.76 ...
##  $ daysBetween: num  36.8 51 26.8 42.4 31.9 ...
```

# Simple Plot


150
100
50
0
Avg. Days Between Trips (F)
$0
$25
$50
$75
$100
$125
Avg. Spend (M)

# Basic K-means in R

Let's start with $k = 2$

```
#recenter the variables
hd_scaled = scale(homedepot)
kfit = kmeans(hd_scaled,centers = 2)

#total variance
kfit$tot.withinss
```

```
## [1] 325.8823
```

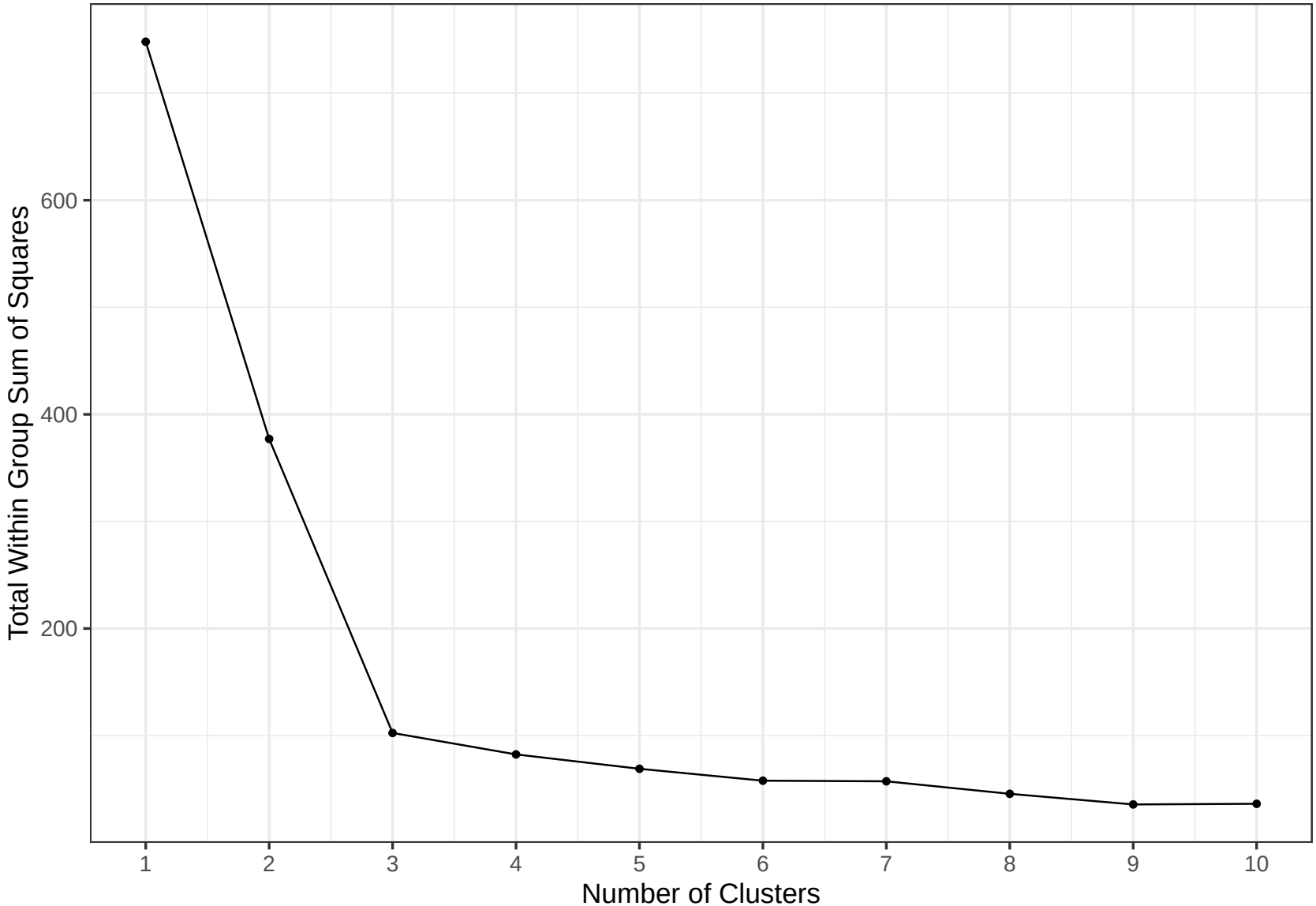
```
#sum((hd_scaled - kfit$centers[kfit$cluster,])^2)

#size of each cluster
kfit$size
```

```
## [1] 257 118
```

# Selecting the number of clusters: use a scree plot

```
screedf = data.frame(k = 1:10,tot.withinss = NA)
for(k in 1:nrow(screedf)){
  screedf$tot.withinss[k] =
    kmeans(hd_scaled,centers = k)$tot.withinss
}

ggOut = ggplot(screedf,aes(k,tot.withinss)) +
  geom_line() + theme_bw(15) + geom_point() +
  scale_x_continuous(breaks = 1:10) +
  xlab("Number of Clusters") +
  ylab("Total Within Group Sum of Squares")
```

# Scree Plot: look for the "elbow"



# Looks like 3 clusters

```
kfit = kmeans(hd_scaled,centers = 3)
homedepot$cluster = kfit$cluster
```

# Visualize


150
100
50
0
Avg. Days Between Trips (F)
$0
$25
$50
$75
$100
$125
Avg. Spend (M)
Cluster
1
2
3

# Describe the clusters and label...

```
homedepot %>%
  group_by(cluster) %>%
  summarise(across(everything(),~round(mean(.),2)))
```

```
## # A tibble: 3 x 3
##   cluster spend daysBetween
##     <int> <dbl>       <dbl>
## 1       1  17.8        39.2
## 2       2  77.1        26.3
## 3       3  79.8       155.
```

# The K-Means mixture

AHC and K-means have a critical limitation: each customer can only be in one cluster.

Because of this, they tend to be sensitive to outliers and don't perform well with "overlapping" clusters.

**Probabilistic clustering** overcomes these issues by incorporating uncertainty about a customer's cluster membership.

# The K-Means mixture

The population consists of a number of subpopulations ("clusters"), each having variables with a different multivariate probability density function.

This results in a *finite mixture density* for the population as a whole.

By using finite mixture densities, we now focus on estimating the parameters of the assumed mixture model and then using the estimated parameters to calculate the posterior probability of cluster membership.

In other words, what is the probability that an observation belongs to one of $K$ clusters?

# *Finite mixture of normals

For a mixture of multivariate normals, they take the form:

$$f(x; \boldsymbol{p}, \boldsymbol{\mu}, \boldsymbol{\Sigma}) = \sum_{k=1}^{K} p_k f_k(x; \mu_k, \Sigma_k)$$

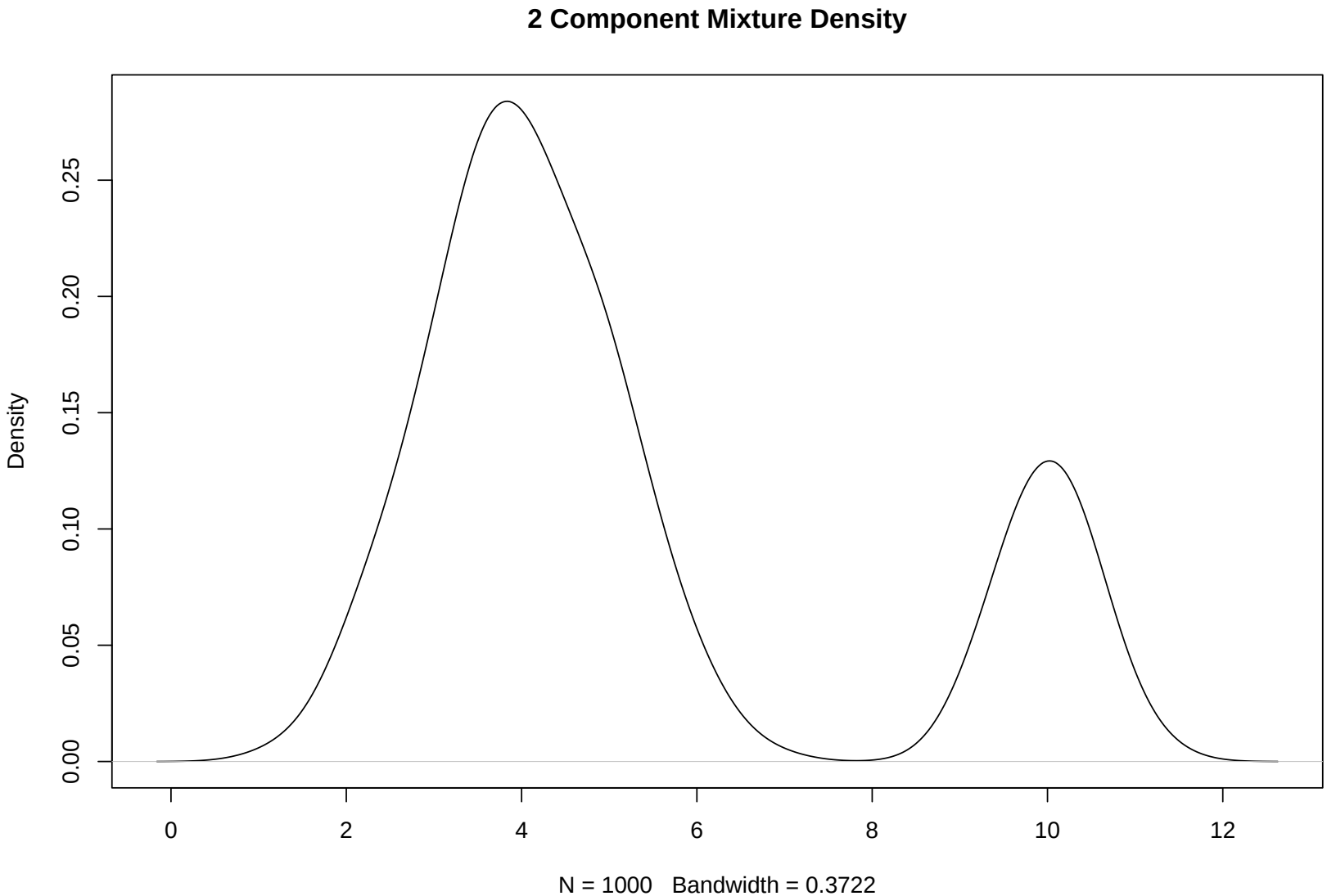
- $x$ is a vector of data (the length equals the number of columns in $X$)
- $p$ is a $K$-dimensional vector of probabilities of being in each cluster (these sum to 1)
- $\mu_k$ and $\Sigma_k$ are the parameters for the associated parameters for the $k$th density

# *Simulating mixture densities

Let's start by mixing two univariate normals

```
K = 2
mu    = c(4,10)
sigma = c(1,.5)
p     = c(.8,.2)

#observations to simulate
n = 1000
z = sample(1:K,n,TRUE,p)

X = rep(NA,n)
for(k in 1:K){
  X[z == k] = rnorm(sum(z==k),mu[k],sigma[k])
}
```

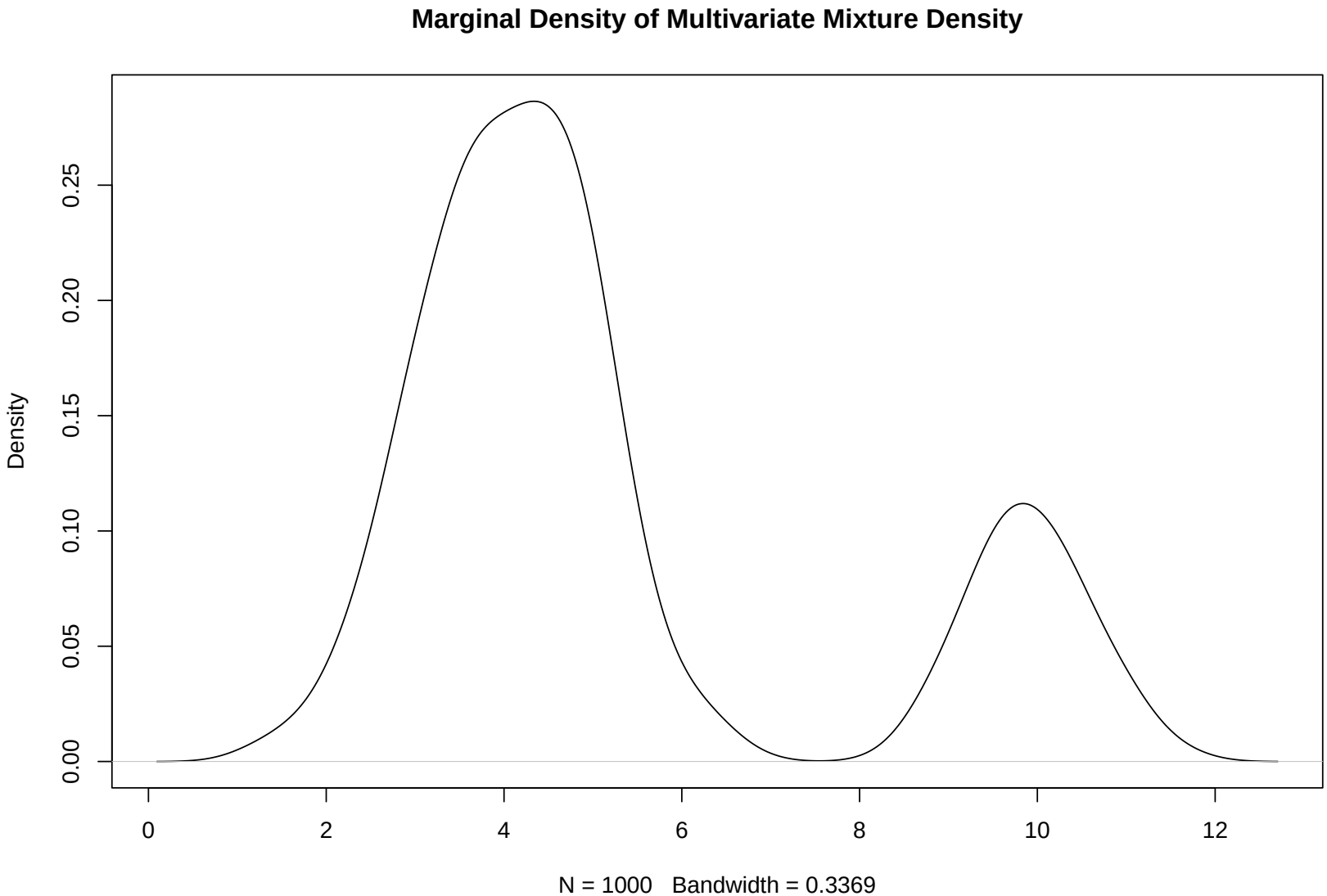
# *Density plot of $X$



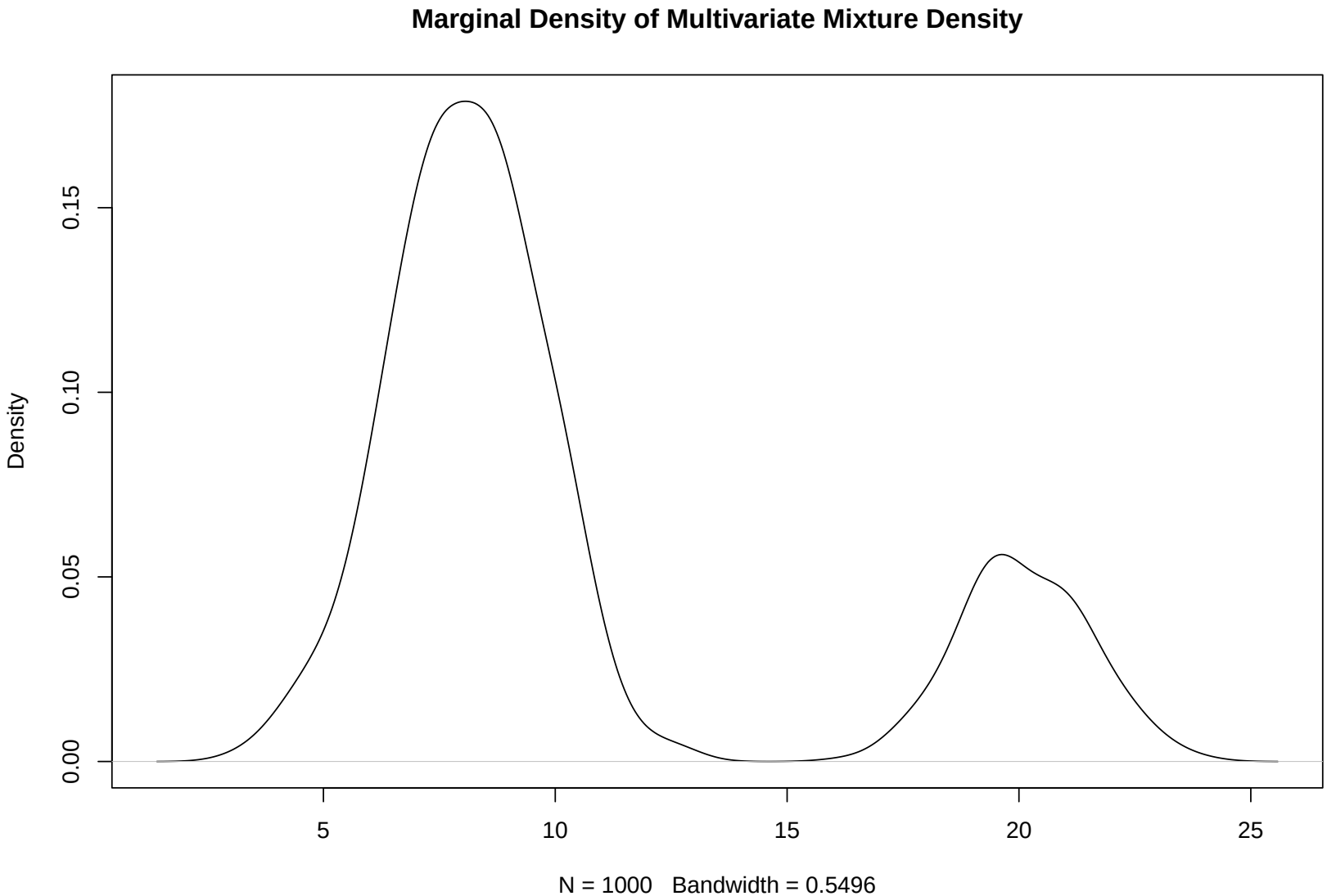
# *Simulating multivariate mixture densities

```
K    = 2 #number of components
dim = 2 #dimension of X
parms = list()
parms[[1]] = list(mu = c(4,8),
                  Sigma=matrix(c(1,.9,.9,3),ncol=dim))
parms[[2]] = list(mu = c(10,20),
                  Sigma=matrix(c(.5,-.3,-.3,2),ncol=dim))
p      = c(.8,.2)

#observations to simulate
n = 1000
z = sample(1:K,n,TRUE,p)

X = matrix(NA,n,dim)
for(k in 1:K){
  X[z == k,] = mvrnorm(sum(z==k),parms[[k]]$mu,
                       parms[[k]]$Sigma)
}
```

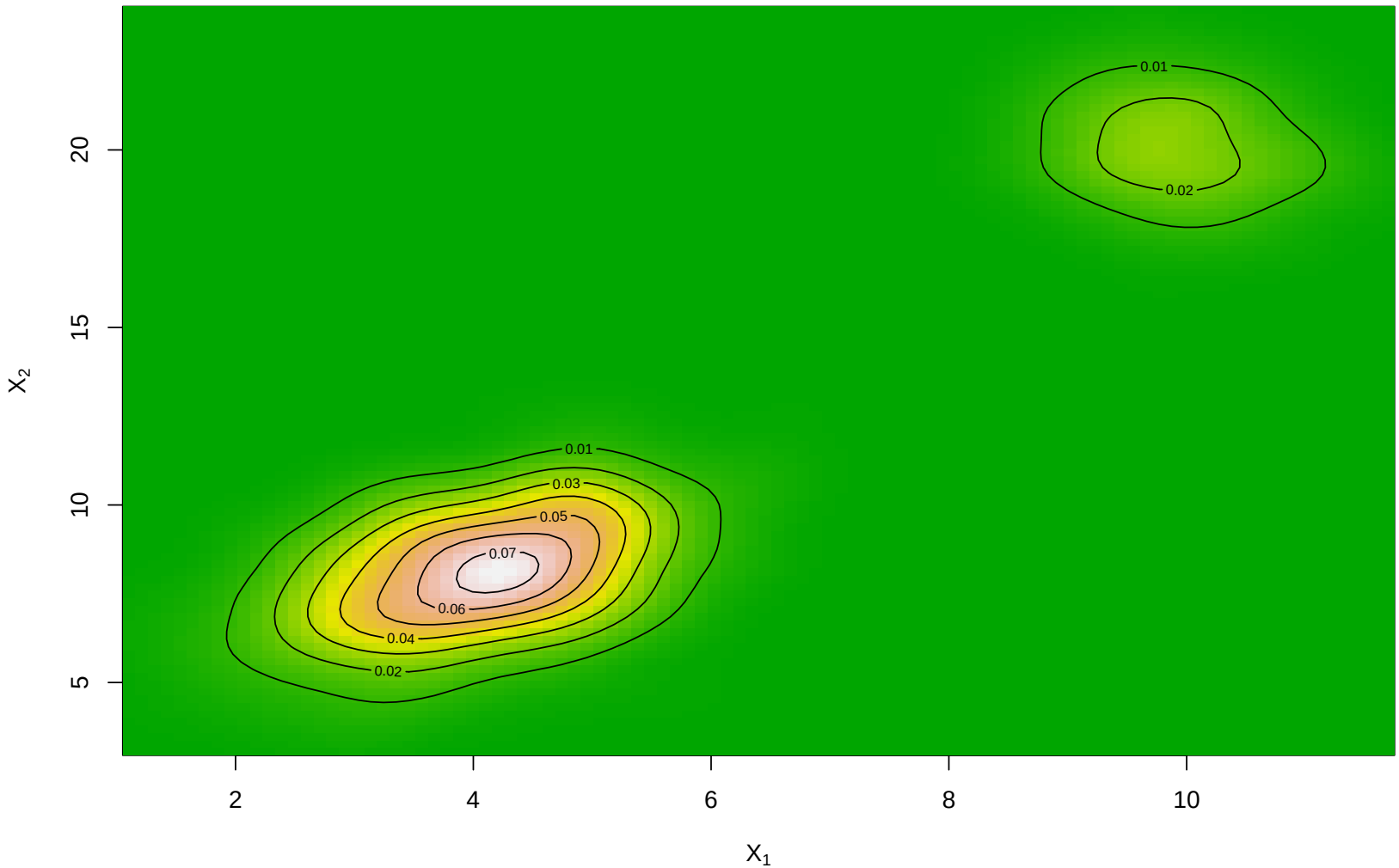
# *Density plot of $X_1$



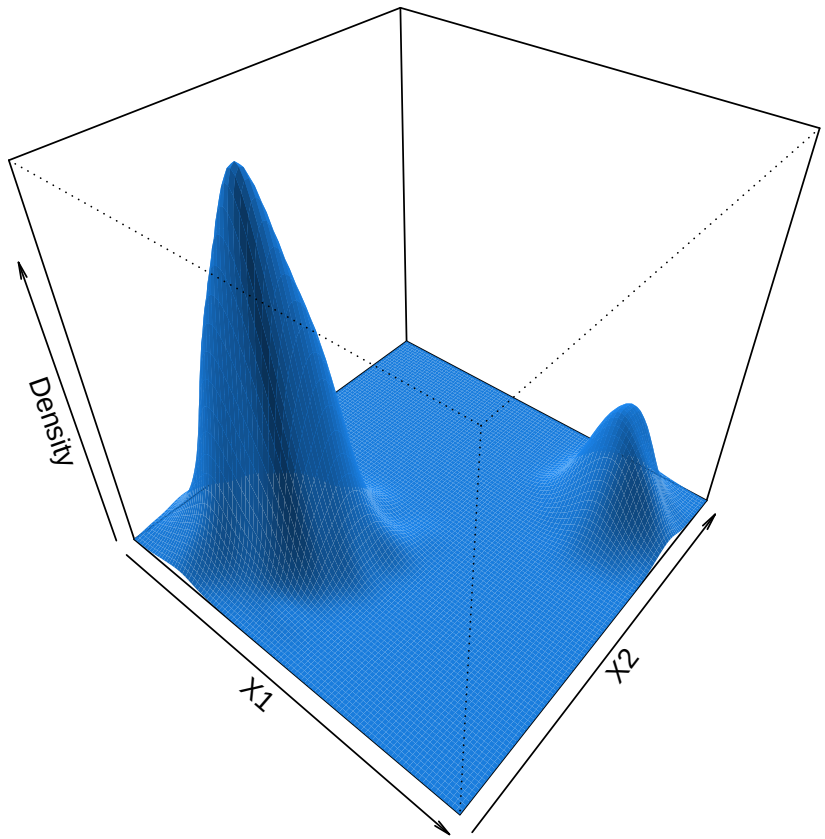
# *Density plot of $X_2$



# *Joint density



# *Another view



# Using the estimates

Observations are associated with the most likely cluster:

$$\Pr(\text{cluster} = k|x_i) = \frac{p_k f_k(x_i; \mu_k, \Sigma_k)}{f(x_i; \boldsymbol{p}, \boldsymbol{\mu}, \boldsymbol{\Sigma})}, k = 1 \ldots K$$

For a particular observation $x_i$, what is the likelihood of observing that observation in the $k$th cluster?

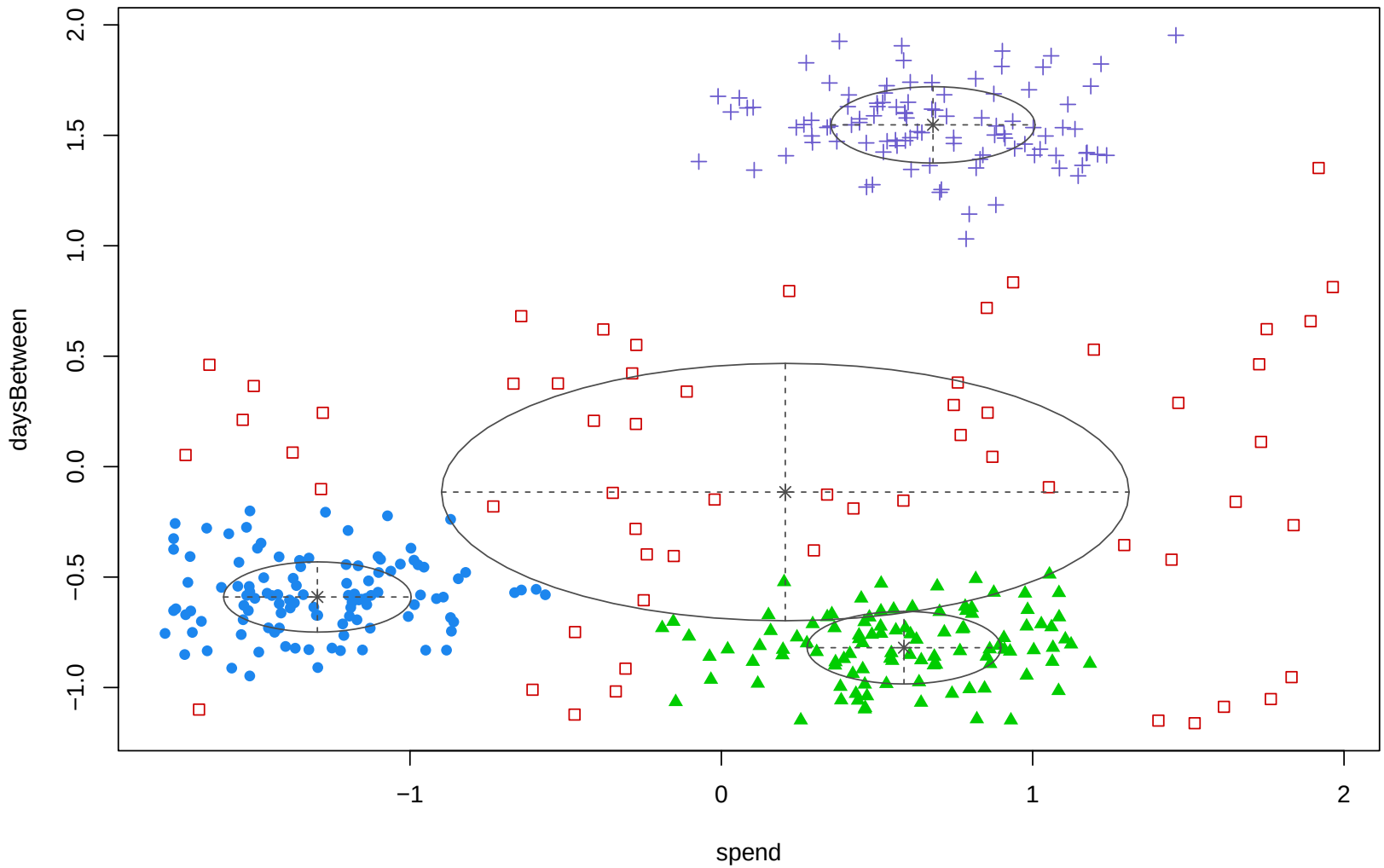
The denominator is simply the sum of the mixture weights and densities across all of the clusters (see the formula few slides back).

# Back to The Home Depot

```
library("mclust")
mcfit = Mclust(hd_scaled)
```

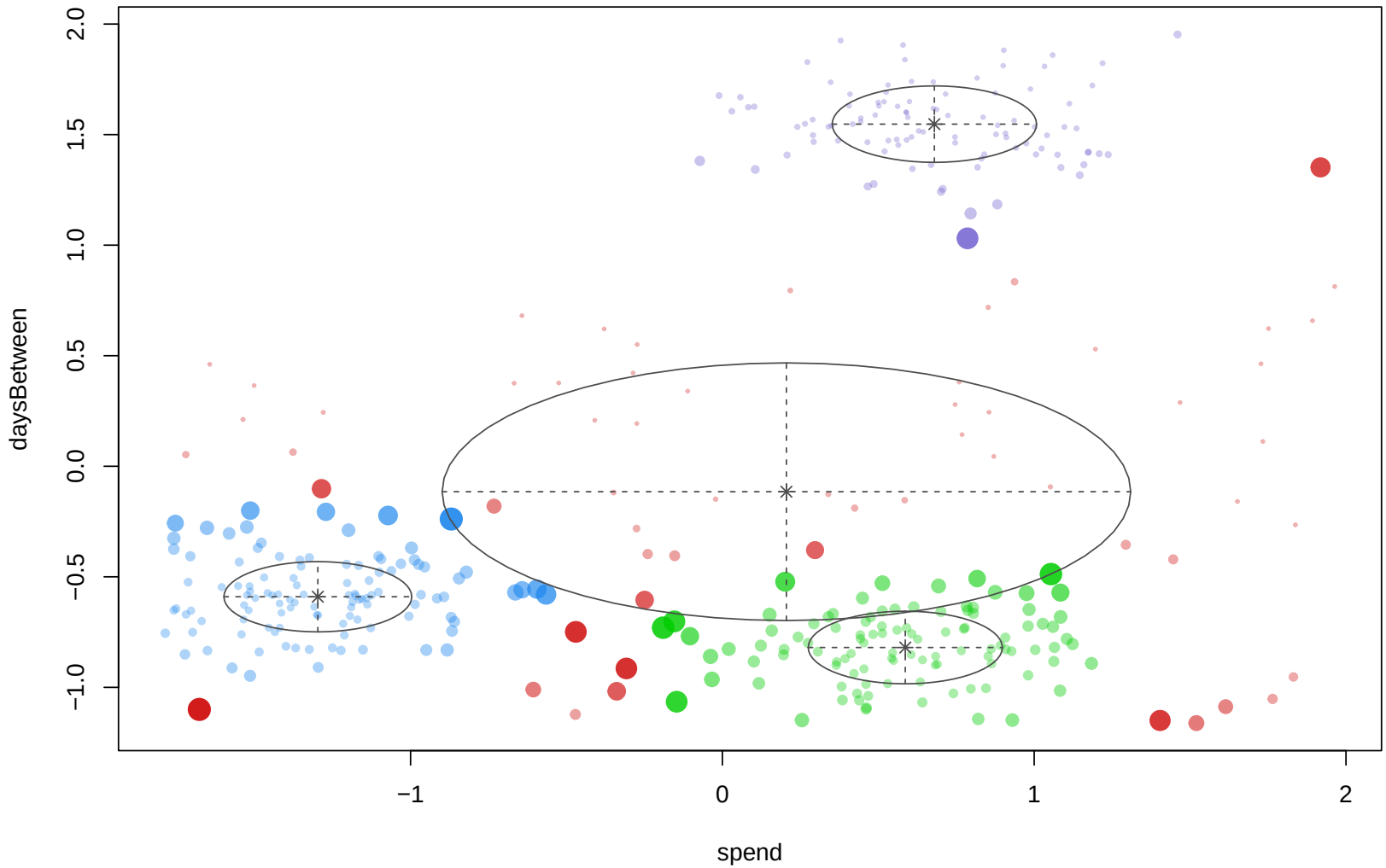
# Mixture classification

The mixture model found 4 classes



# Mixture uncertainty

Notice that further away from the centers there is more uncertainty



# Why care about uncertainty?

If there is a costly mistake on classifications (e.g., a high offer value), maybe make offer value a function of uncertainty

If a lot of observations within a cluster have a lot of uncertainty, it could just be noisy data (e.g., the newly created cluster on the last slide)

Focus A/B tests on those that are "quintessential" to the cluster to reduce variance

Depends on the context...

# DBSCAN

Groups data points based on their density, forming clusters where points are closely packed together

Unlike K-means, DBSCAN can identify clusters with irregular shapes, not just spherical ones

It effectively identifies outliers as “noise”, separating them from the core clusters

DBSCAN does not require you to predefine the number of clusters; it discovers them based on data density

# In R: `dbscan`

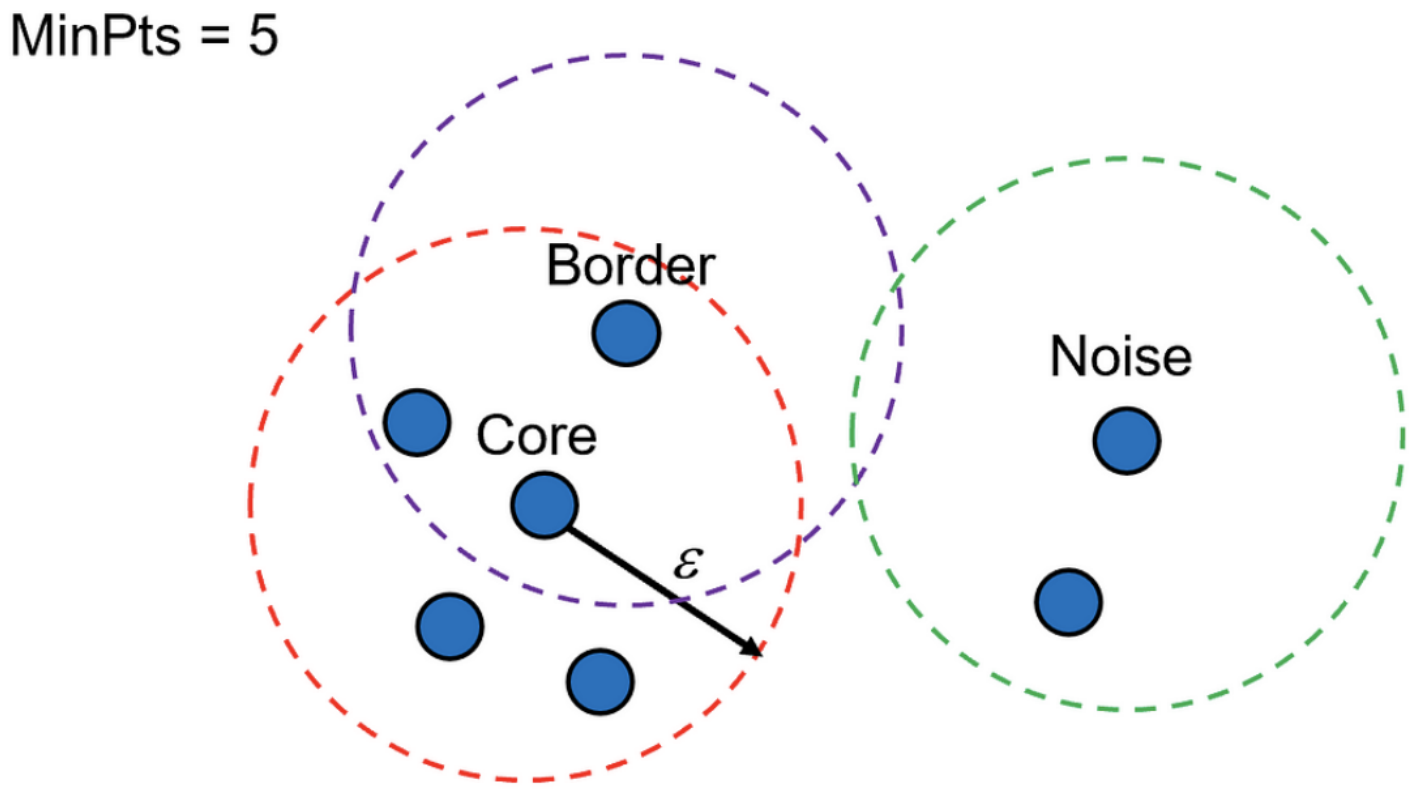
Epsilon (`eps`):

- The size of the neighborhood around each point
- A larger value means that more points will be considered neighbors, potentially leading to larger clusters

Minimum Points (`minPts`)

- Minimum number of points to form a "core point"
- Higher values result in denser clusters

# Intuition



# In R

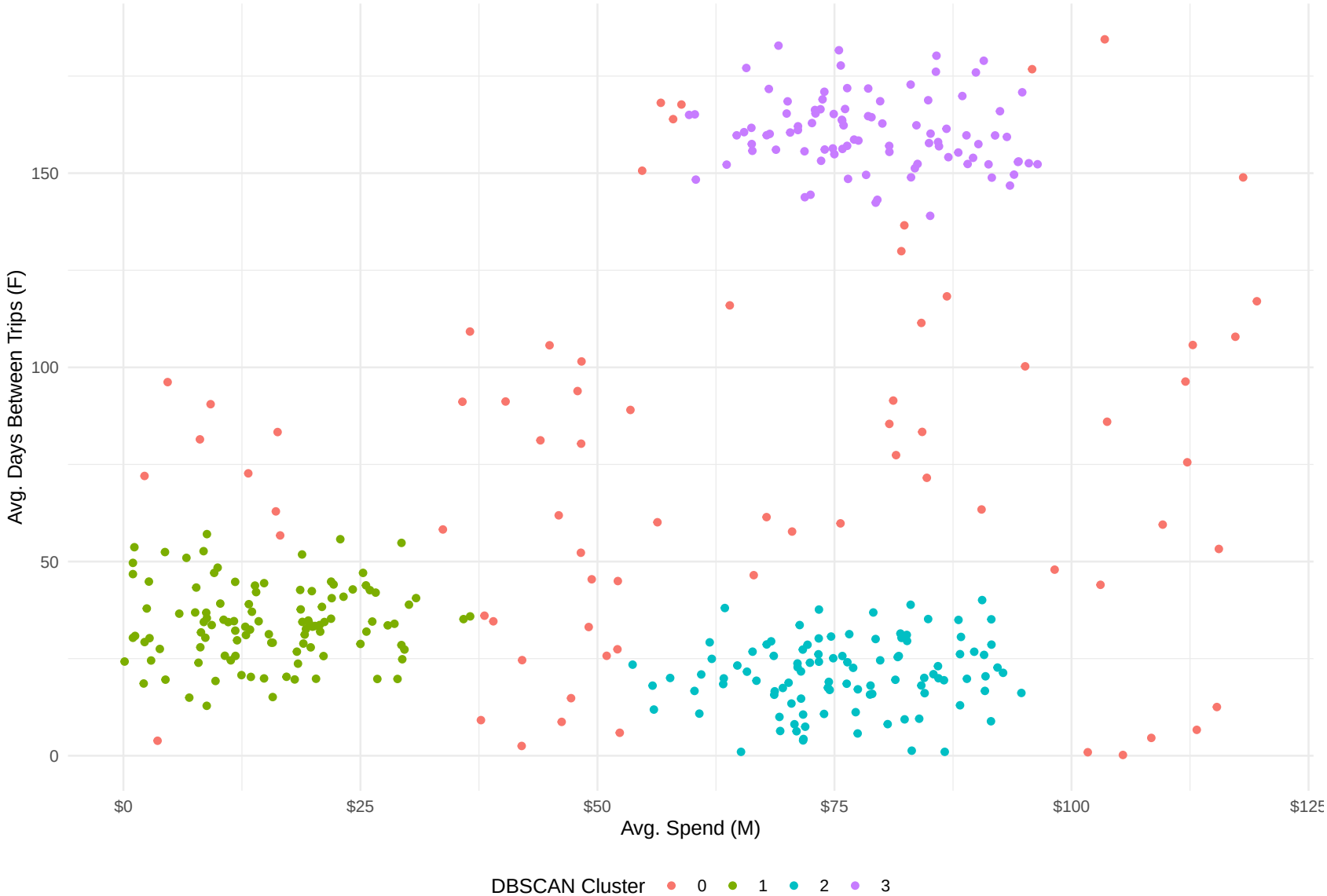
Using the scaled data...

```
library(dbscan)
dbscan_out = dbscan(hd_scaled, eps = .5, minPts = 10)
homedepot$dbs1 = dbscan_out$cluster
```

# eps = .5 and minPts = 10

# `eps = .2` and `minPts = 10`

Very sensitive to these settings...



# Using cluster analysis

Used properly, cluster analysis can help provide structure to what appears to be hopelessly complex data.

The solution is sensitive to the choice of variables, units of measurement, distance metric, and clustering algorithm.

There is no precise or objective way to choose between alternative solutions. Managerial judgement is a critical guide. Beware of "finding what you are looking for" by trying many combinations.

# How many clusters?

Are they interpretable for a manager? Is there a label that makes sense?

Are they of sufficient size?

There are no statistical measures for k-means and agglomerative methods, but a scree plot is often a useful (and good enough) guide.

# Clustering Summary

- K-means and AHC (using Ward's method) have been found to outperform other methods in simulations.
- Partitioning methods (K-means) seeks to find the *best* K-group classification of the objects in question.
- Hierarchical methods seek to find a nested pattern or grouping of the objects - they provide a more dynamic picture of the data.
- Mixture models have a definite statistical advantage because the clustering is based on sensible models of the data, as opposed to an exploratory data summary.
- DBSCAN's strength is in dealing with arbitrary shapes of clusters and removing noise.

# Extensions of Cluster Analysis

Cluster Regression

- One use of *unsupervised* clustering is to throw the results into a *supervised* regression model.
- For example, use the cluster associated with each hotel guest to predict the probability of churning.

# Example: customer segmentation at Everlane

```
load('data/everlane.rdata')
str(everlane)
```

```
## 'data.frame':    1875 obs. of  6 variables:
##  $ id       : int  1 1 1 2 2 2 2 3 3 3 ...
##  $ transdate: Date, format: "2018-05-28" "2018-08-09" ...
##  $ spend    : num  72.2 62.3 57.3 112.6 130.5 ...
##  $ age      : int  56 56 56 30 30 30 30 19 19 19 ...
##  $ female   : int  1 1 1 0 0 0 0 1 1 1 ...
##  $ promo    : num  0 0 0 15 15 0 25 0 0 0 ...
```

# Extra: Hierarchical Clustering

# *Hierarchical methods

Instead of a single solution, a hierarchy of possible solutions is formed in a tree-like structure

Thus there are 1...$n$ possible clusters, where $n$ is the number of customers

The output is a tree or dendrogram, which is "pruned" at a level that appears useful

# *Hierarchical clustering process

1) Start with an $n$ cluster solution
2) Produce an $n-1$ cluster solution by combining the two most similar customers
3) Join the next two most similar items, either a previously formed cluster or individuals
4) Continue until all items are clustered (i.e., one cluster)

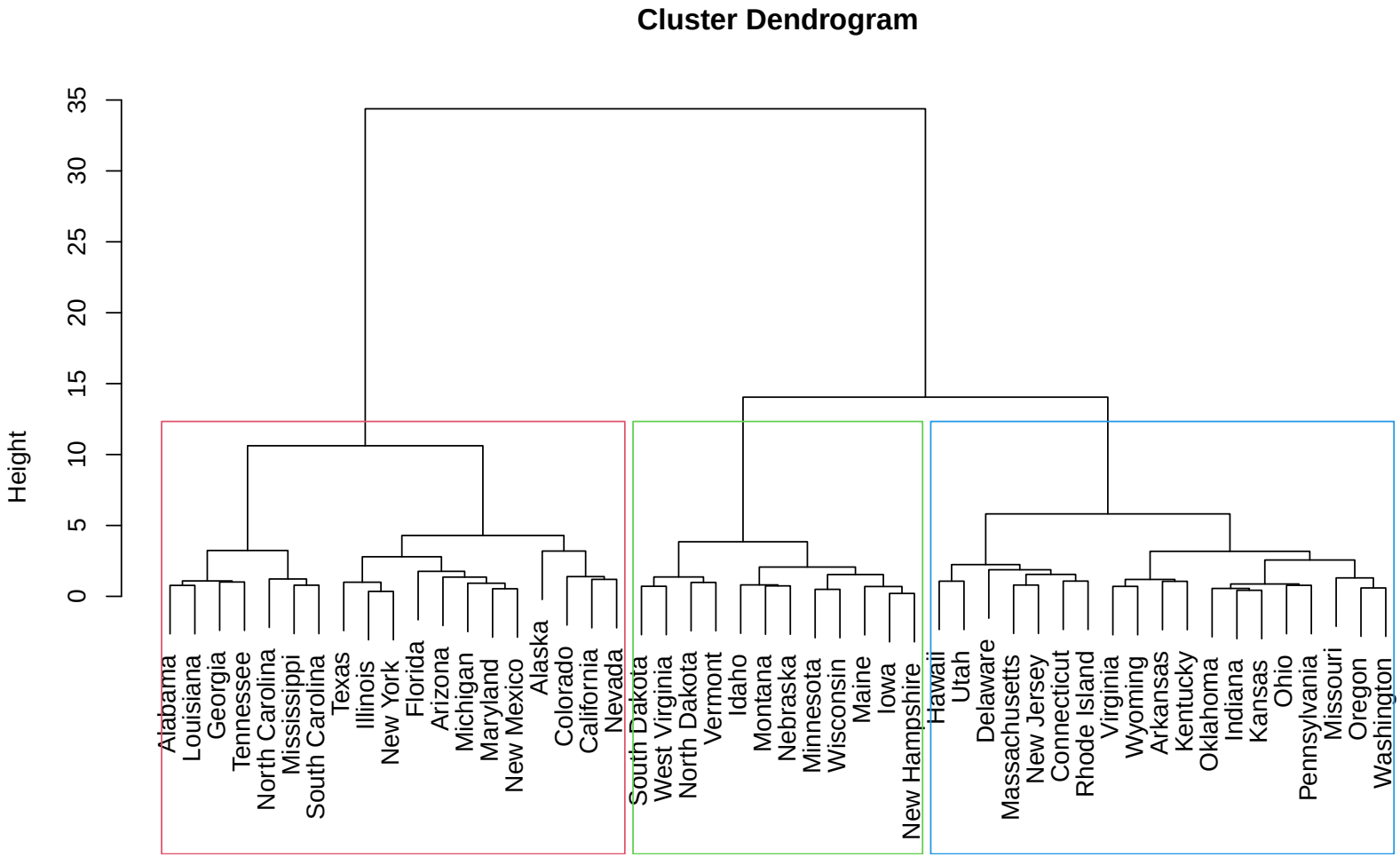
# *Agglomerative clustering example: US Arrests

Arrests per 100,000 residents for various crimes in 1973:

```
##               Murder Assault UrbanPop Rape
## Alabama         13.2     236       58 21.2
## Alaska          10.0     263       48 44.5
## Arizona          8.1     294       80 31.0
## Arkansas         8.8     190       50 19.5
## California       9.0     276       91 40.6
## Colorado         7.9     204       78 38.7
```

# *Agglomerative clustering in R

```
plot(hclustout,labels = row.names(USArrests))
rect.hclust(hclustout, k=3,border = 2:5)
```



# *Interpreting a dendrogram

- Nodes are objects being clustered
- Branches indicate when the cluster method joins subgroups containing that object
- The length of the branch indicates the distances between the subgroups when they are joined
- The two most similar objects are combined first and linked at the bottom rung of the hierarchy; the next two most similar objects are combined and linked at the next rung, and so on

# *So what can we say about the cities?

```r
# Cut tree into 3 groups
sub_grp = cutree(hclustout, k = 3)
```

```
## # A tibble: 4 x 4
##   name       `1`   `2`   `3`
##   <chr>    <dbl> <dbl> <dbl>
## 1 Assault    259   142    76
## 2 Murder      12     6     3
## 3 Rape        29    19    12
## 4 UrbanPop    68    71    52
```

# *More references

Check out the `mclust` package vignette or *An Introduction to Applied Multivariate Analysis in R* for for more applications and additional details on mixture clustering

- Parameter estimation methods (EM algorithm)
- Setting constraints on the shape of the covariance matrices
- Additional plots for model evaluation

For a really deep dive here is a whole book on the topic:

*Everitt, B. S., Landau, S., Leese, M., and Stahl, D. (2011), Cluster Analysis, Chichester, UK: John Wiley & Sons, 5th edition.*