Data Analysis and Visualization

Universität Konstanz

Information Visualization

Data Foundation



Dr. Johannes Fuchs

„Low-quality data will lead to low-quality mining results.“ [1]

Data Types
Data Cleaning
Normalization
Data Reduction

Data Analysis and Visualization

Universität Konstanz









Data Types



Data Types - Nominal

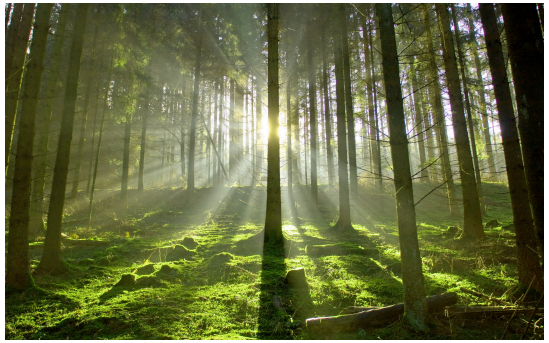








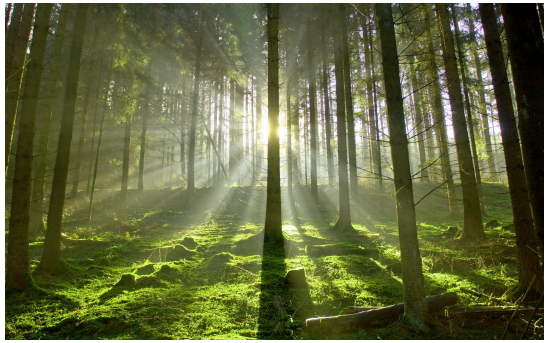
Data Types


≠

≠

≠


Data Types - Ordinal







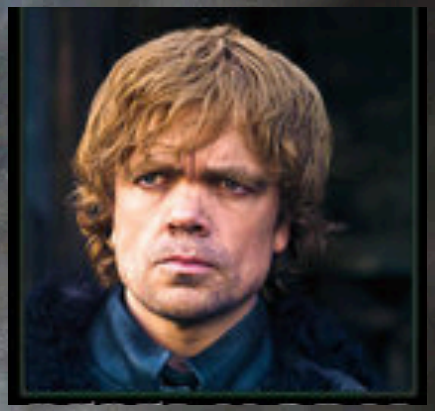
Data Types - Ordinal



Jon Snow



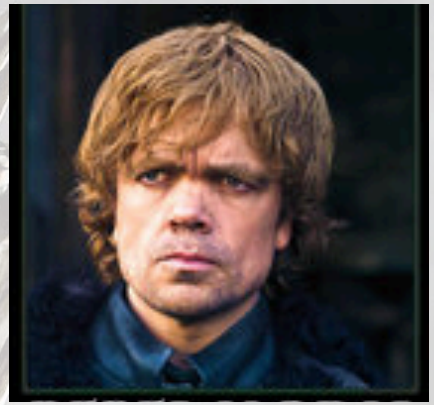
The Hound



Tyrion Lannister

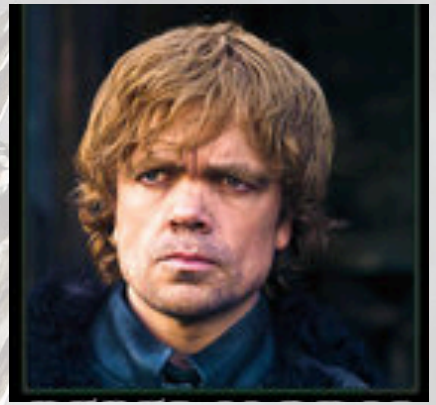
Data Types


≠

≠


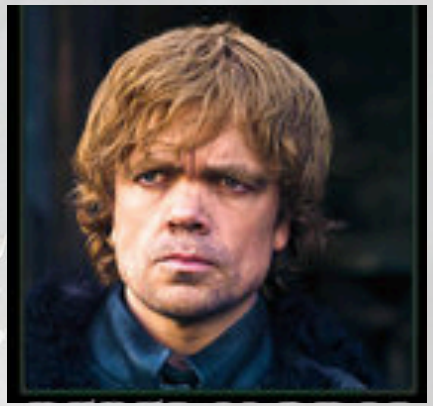
Data Types – Numeric





Data Types – Numeric


=(

+

/4)

Data Types – Numeric

Data Analysis and Visualization

Universität Konstanz



Hands-on-Session

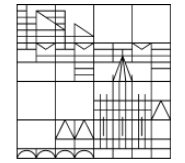
Unique ID	Gender	Age	Smoking Habits	Time
323	male	21	never	1h 40s
435	female	34	regularly	1h 55s
123	male	19	heavy	1h 23s
352	female	25	never	1h 43s
674	female	25	sometimes	1h 10s
865	male	18	heavy	1h 50s
341	female	41	regularly	1h 33s

Hands-on-Session

Unique ID	Gender	Age	Smoking Habits	Time
323	0	21	0	1h 40s
435	1	34	2	1h 55s
123	0	19	3	1h 23s
352	1	25	0	1h 43s
674	1	25	1	1h 10s
865	0	18	3	1h 50s
341	1	41	2	1h 33s

Hands-on-Session



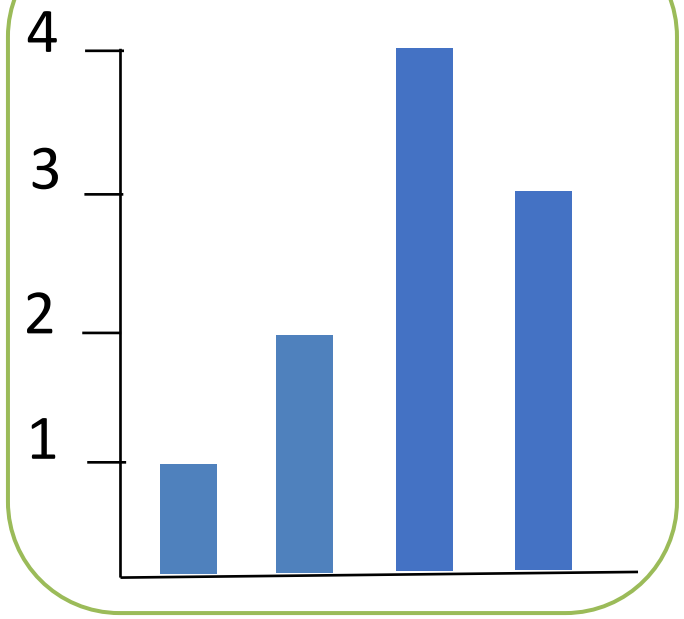


Hair color (Legend):

1. Blond
2. Red
3. Black
4. Green

Why is this important to know?

Visualization



Hair color (Legend):

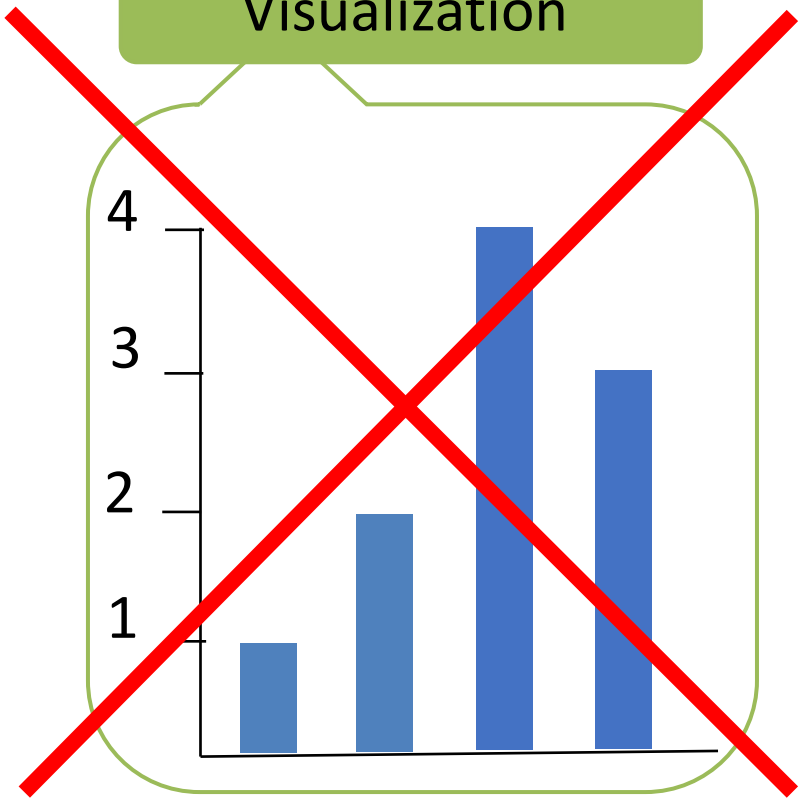
1. Blond
2. Red
3. Black
4. Green

Why is this important to know?

Hair color (Legend):

1. Blond
2. Red
3. Black
4. Green

Visualization



Calcuations

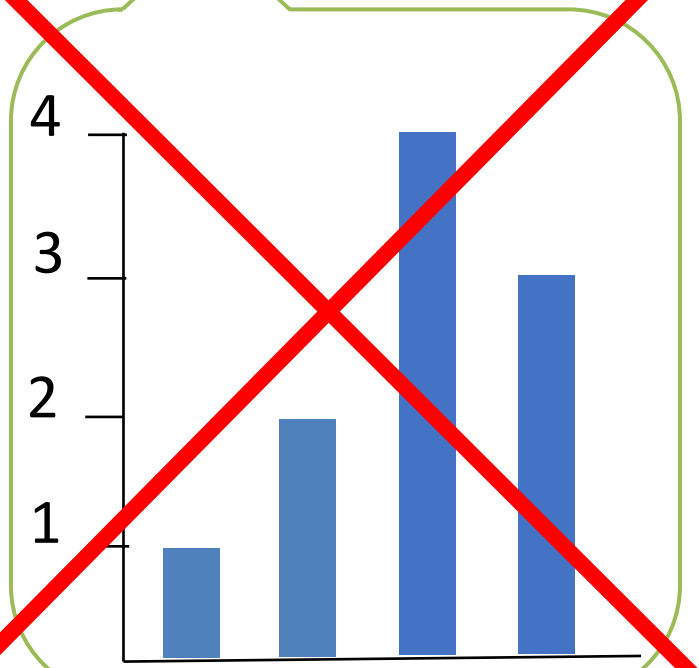
Avg(haircolor)
= (1+2+3+4)/4
= 2.5

Why is this important to know?

Hair color (Legend):

1. Blond
2. Red
3. Black
4. Green

Visualization



Calcuations

Avg(haircolor)
= (1+2+3+4)/4
= 2.5

Why is this important to know?

Nominal

- No quantitative relationship between categories
- Classification without ordering

Ordinal

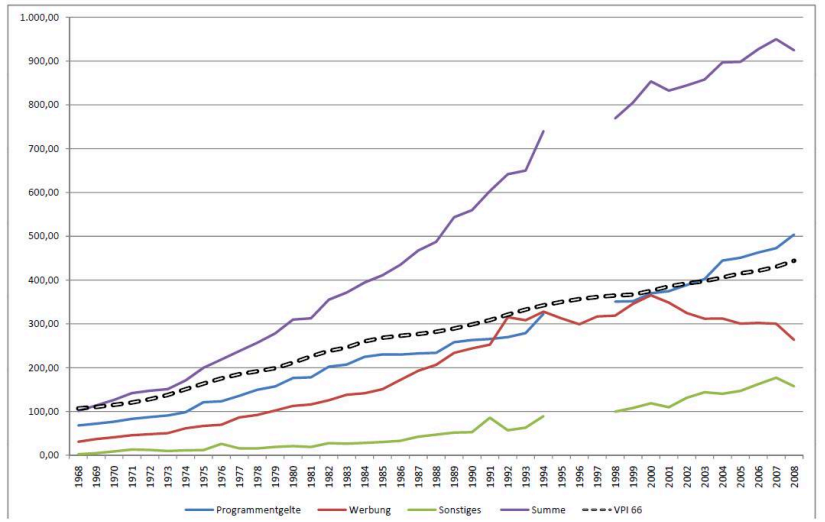
- Attributes can be rank-ordered
- Distances between values do not have any meaning

Numeric

- Attributes can be rank-ordered
- Distances between values have a meaning
- Mathematical operations are possible

Data Types – Summary





Missing Values

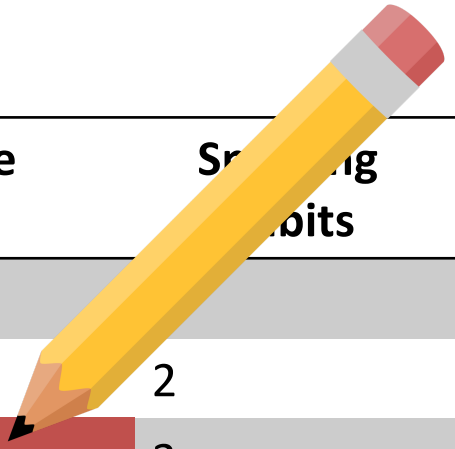
Unique ID	Gender	Age	Smoking Habits	Time
323	0	21	0	1h 40s
435	1	34	2	1h 55s
123	0		3	1h 23s
352	1	25	0	1h 43s
674	1	25	1	1h 10s
865	0	18	3	1h 50s
341	1	41	2	1h 33s

Missing Values

Ignore the tuple

Unique ID	Gender	Age	Smoking Habits	Time
323	0	21	0	1h 40s
435	1	34	2	1h 55s
~~123~~	~~0~~		~~3~~	~~1h 23s~~
352	1	25	0	1h 43s
674	1	25	1	1h 10s
865	0	18	3	1h 50s
341	1	41	2	1h 33s

Fill in the missing value manually



Unique ID	Gender	Age	Sr…ng …bits	Time
323	0	21		1h 40s
435	1	34	2	1h 55s
123	0		3	1h 23s
352	1	25	0	1h 43s
674	1	25	1	1h 10s
865	0	18	3	1h 50s
341	1	41	2	1h 33s

Use a global constant

Unique ID	Gender	Age	Smoking Habits	Time
323	0	21	0	1h 40s
435	1	34	2	1h 55s
123	0	-1	3	1h 23s
352	1	25	0	1h 43s
674	1	25	1	1h 10s
865	0	18	3	1h 50s
341	1	41	2	1h 33s

Use the attribute mean

Unique ID	Gender	Age	Smoking Habits	Time
323	0	21	0	1h 40s
435	1	34	2	1h 55s
123	0	27	3	1h 23s
352	1	25	0	1h 43s
674	1	25	1	1h 10s
865	0	18	3	1h 50s
341	1	41	2	1h 33s

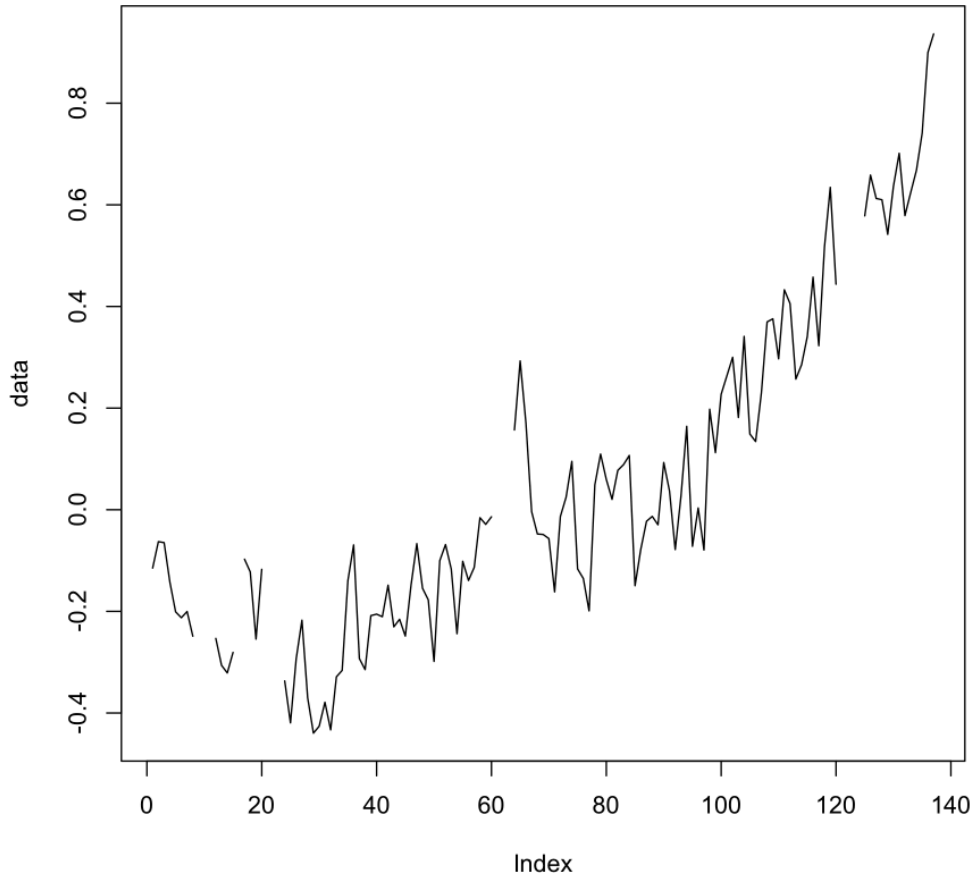
Use the most probable value

Unique ID	Gender	Age	Smoking Habits	Time
323	0	21	0	1h 40s
435	1	34	2	1h 55s
123	0		3	1h 23s
352	1	25	0	1h 43s
674	1	25	1	1h 10s
865	0	18	3	1h 50s
341	1	41	2	1h 33s

R-Code: 02a_MissingValue.r

R Konsole

```
> #Function to load the data
> loadData <- function(url)
+ {
+     data <- read.csv(url)
+     data <- data[,2]
+     data <- rev(data)
+     return(data)
+ }
> data <- loadData("./R/dataFoundation/temperature.csv")
> plot(data, type="l")
>
```



Enter value manually

+ For small datasets effective
- Need to know the value
- Time consuming
- Not feasible with large datasets

Use attribute mean

+ Simple to implement
- Not the most accurate approximation of the value

Ignore the tuple

+ Can be easily done
+ No computational effort
- Loss of information
- Unnecessary if the attribute is not needed

Use a global constant

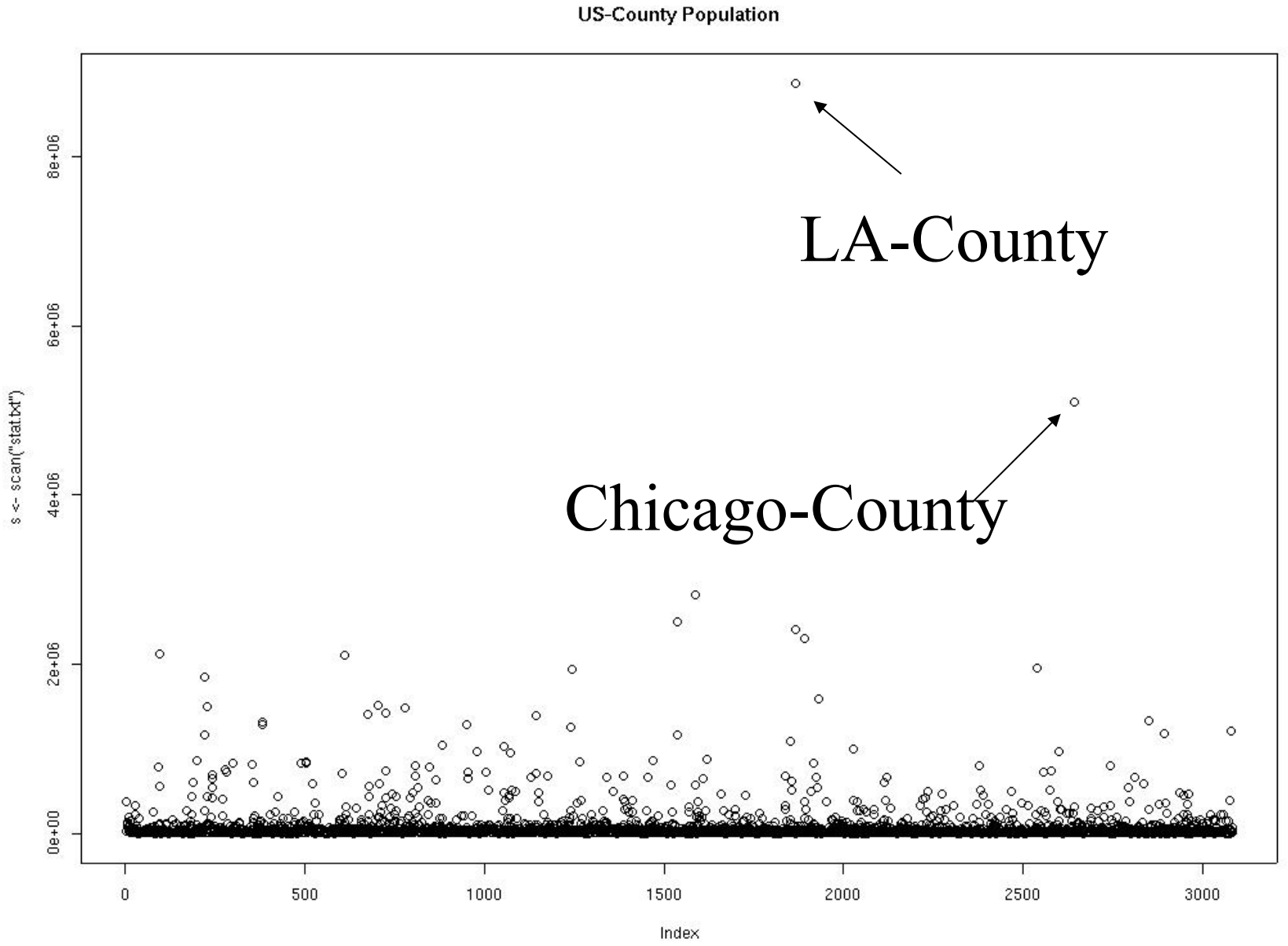
+ Can be easily done
+ Perhaps interesting to know the missing values

Your algorithm should not use this value for further calculations (be careful with "-1")

Use most probable value

+ Most accurate approximation of the value
- Most computational effort

Missing Values - Summary



Normalization

$$f_{lin}(v) = \frac{v - min}{max - min}$$

Linear Normalization

index	1	2	3	4	5	6	7	8	9	10
data	8	7	5	6	9	2	15	11	8	6

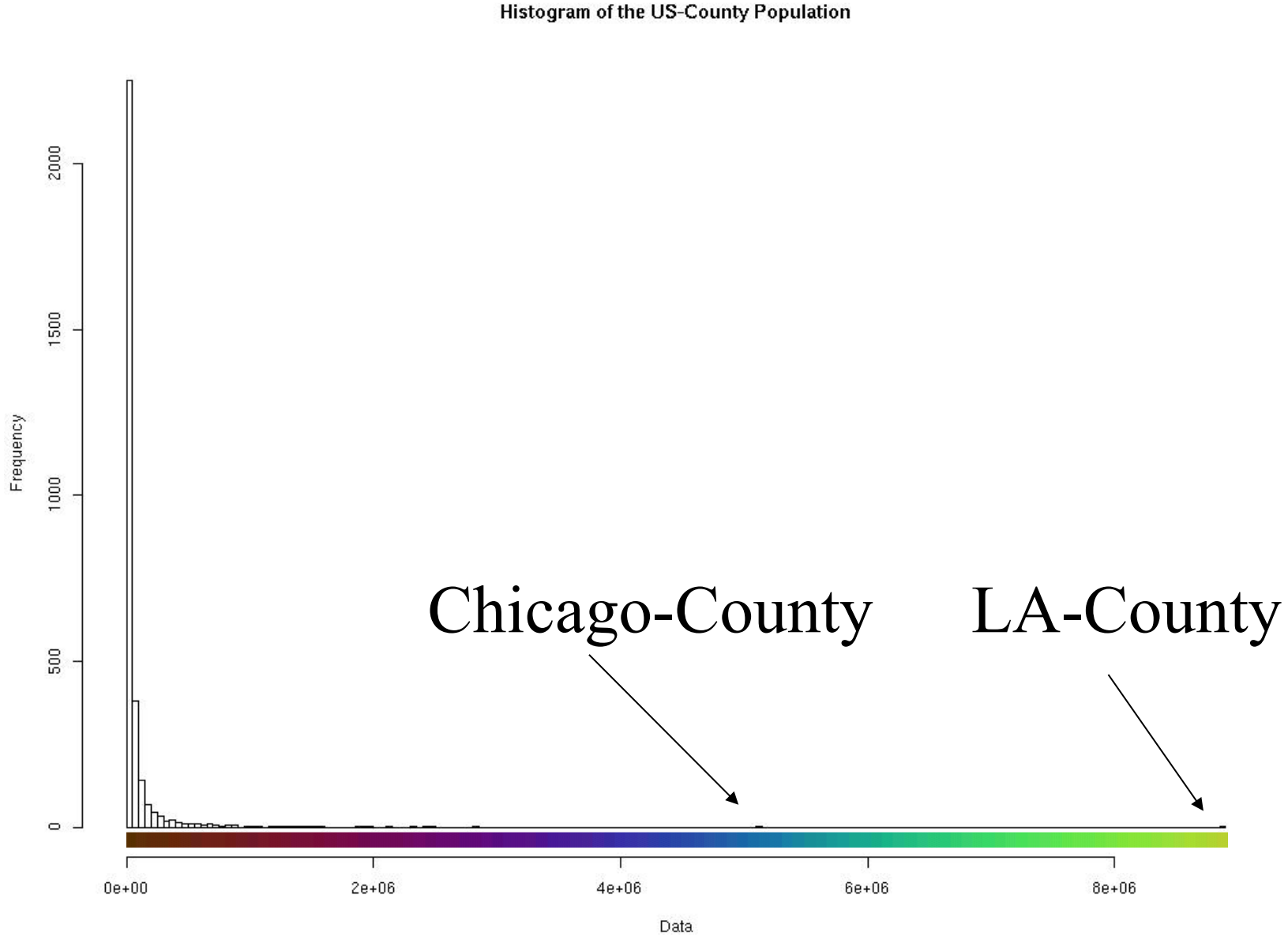
$$f_{lin}(v) = \frac{v - min}{max - min}$$

Linear Normalization

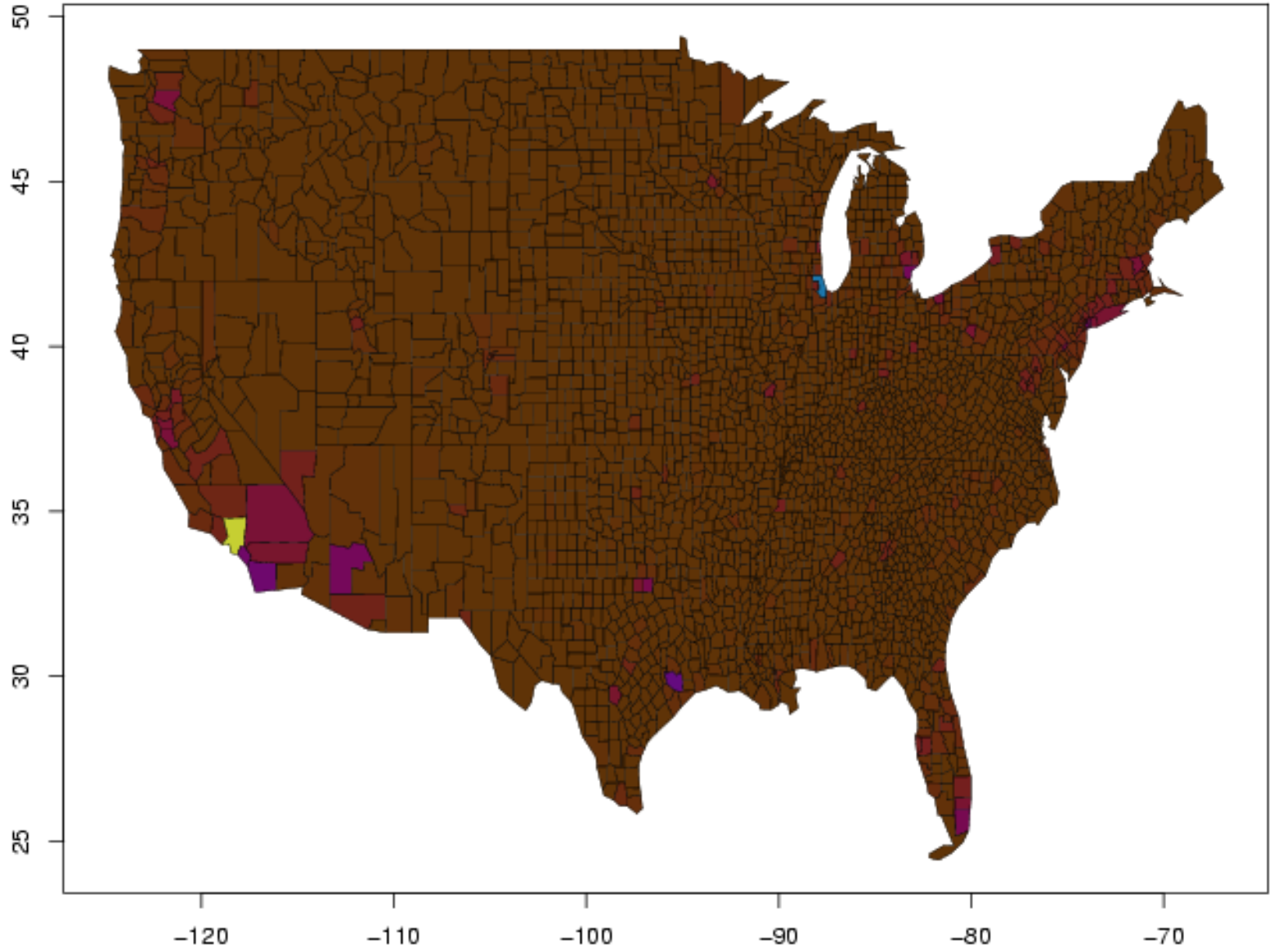
index	1	2	3	4	5	6	7	8	9	10
data	8	7	5	6	9	2	15	11	8	6

$$f_{lin}(6) = \frac{6-2}{15-2} = 0.31$$

Linear Normalization



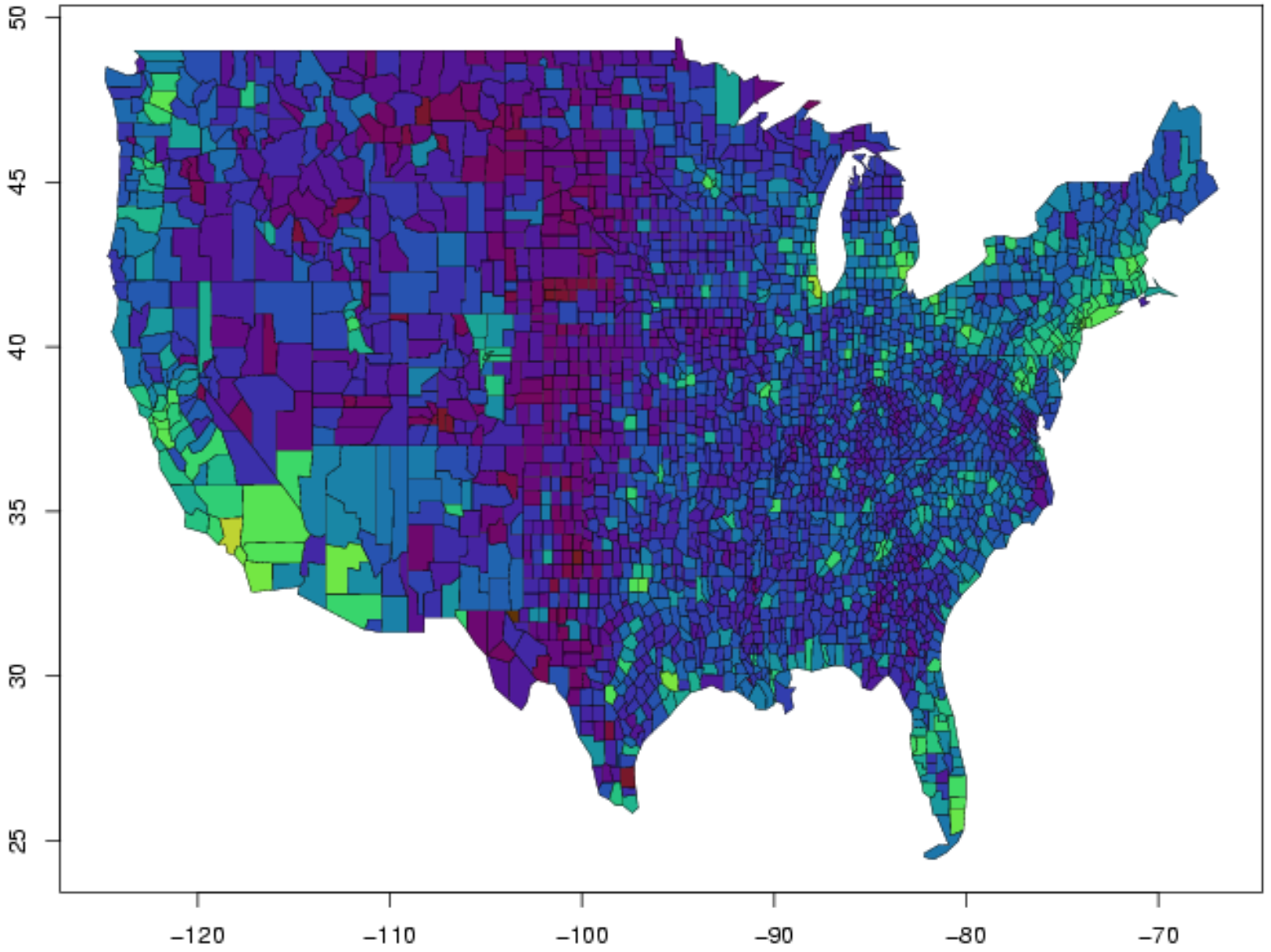
Linear Normalization



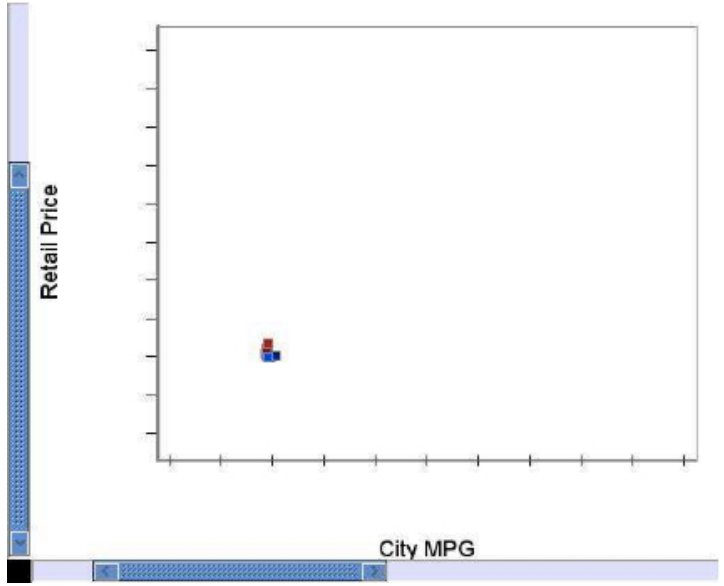
Linear Normalization

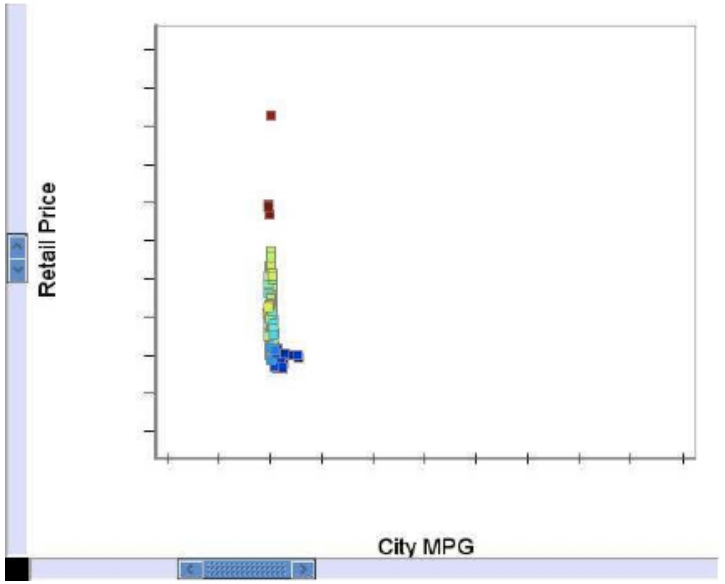
$$f_{\ln}(v) = \frac{\ln(v) - \ln(\min)}{\ln(\max) - \ln(\min)}$$

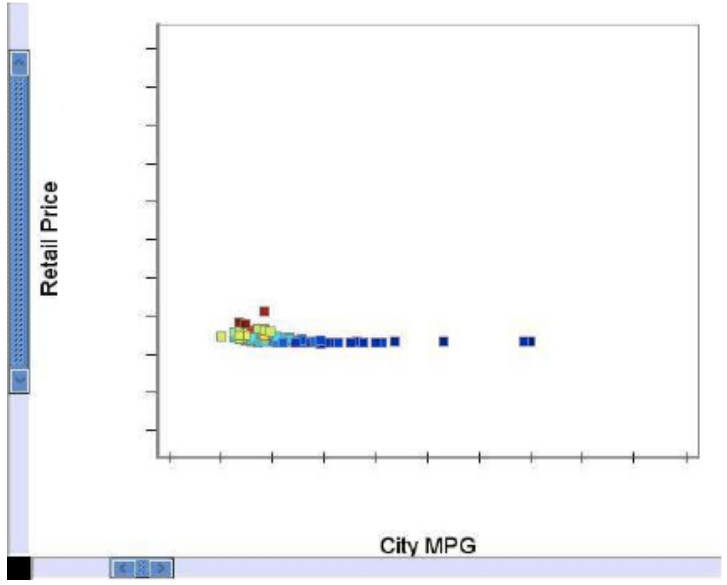
Logarithmic Normalization

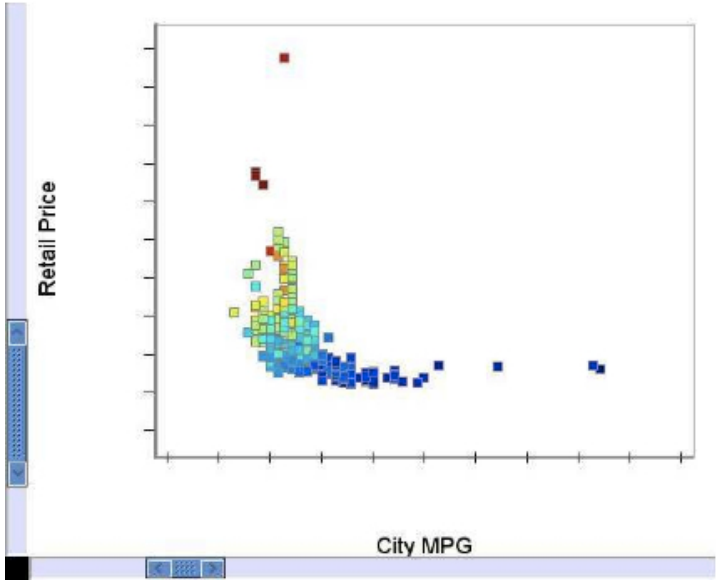


Logarithmic Normalization









Normalization

	MPG	HP	Cylinder	Weight	Gear
Car A	21	110	4	2500	4
Car B	18.7	93	6	2320	4
Car C	10.4	175	6	3440	3
Car D	32.4	245	8	5250	3

Hands-On-Session

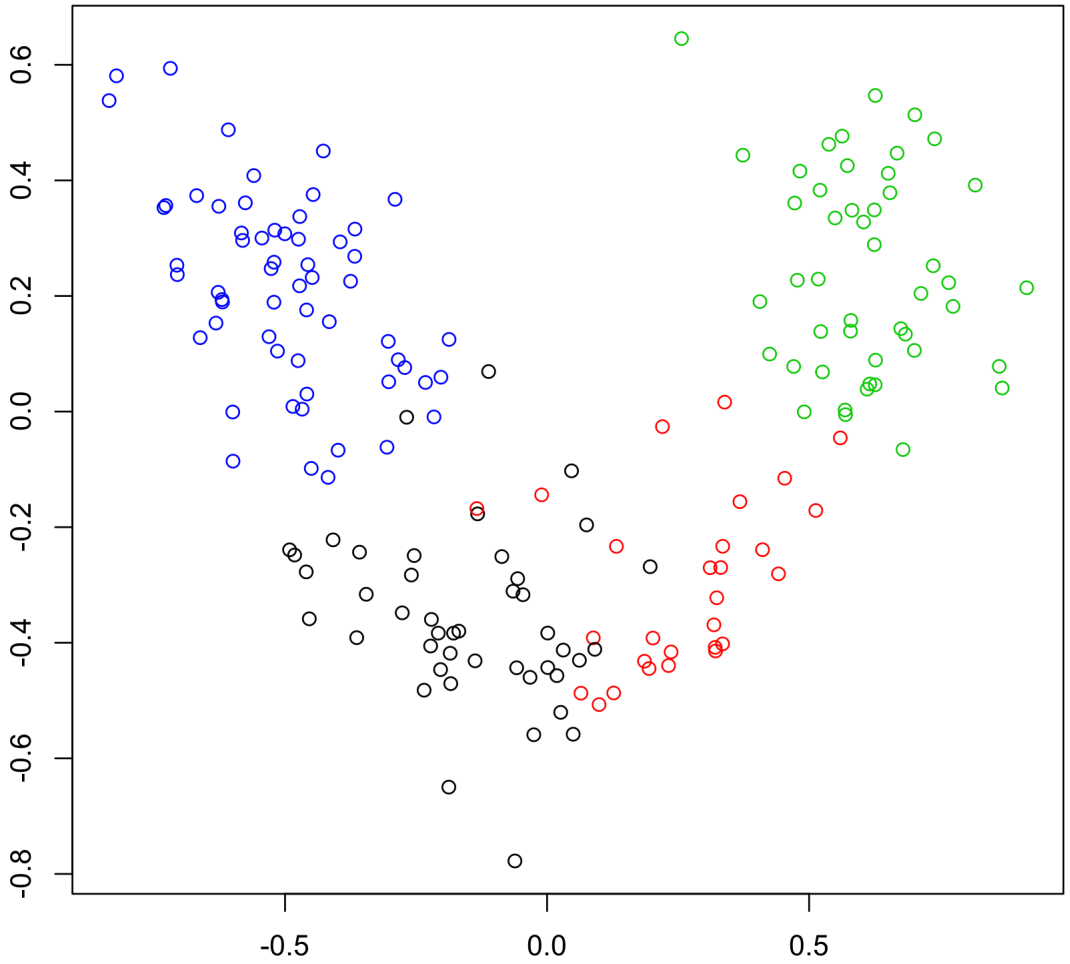
	MPG	HP	Cylinder	Weight	Gear
Car A	21	110	4	2500	4
Car B	18.7	93	6	2320	4
Car C	10.4	175	6	3440	3
Car D	32.4	245	8	5250	3

Distance(Car C und Car D) = (10.4 – 32.4)^2 + (175 - 245)^2 + (6 - 8)^2 + (3440 - 5250)^2 + (3-3)^2

R-Code: 02b_Clustering.r

R Konsole

```
> #normalize data
> norm <- function(data)
+ {
+     temp <-(data - min(data)) / (max(data) - min(data));
+   return(temp);
+ }
>
> #load wine dataset
> loadDataWine <- function()
+ {
+     data(wine, package="rattle.data")
+     data <- wine[2:14]
+     return(data)
+ }
> #k-means clustering
> kMeans <- function(dataNorm, nrOfClusters)
+ {
+     data <- kmeans(dataNorm, centers=nrOfClusters)
+     return(data$cluster)
+ }
> data <- loadDataWine()
> dataNorm <- apply(data, 2, norm)
> dataCluster <- kMeans(dataNorm, 4)
> pca <- prcomp(dataNorm)
> plot(pca$x, col=dataCluster)
>
```



Data reduction:

- Reduction of number of **data points**
- Reduction of number of **dimensions**

Motivation:

- Represent a large dataset by a much smaller subset
- Speed up automatic calculations performed in a later step
- The subset should sufficiently represent the whole data set

index	1	2	3	4	5	6	7	8	9	10
income	800	700	500	600	900	200	105	110	800	600
gender	1	0	1	0	1	0	0	1	0	1
education	BA	BA	MA	BA	PHD	MA	PHD	BA	MA	BA

Simple Random Sampling

index	1	2	3	4	5	6	7	8	9	10
income	800	700	500	600	900	200	105	110	800	600
gender	1	0	1	0	1	0	0	1	0	1
education	BA	BA	MA	BA	PHD	MA	PHD	BA	MA	BA

Simple Random Sampling

index	1	2	3	4	5	6	7	8	9	10
income	800	700	500	600	900	200	105	110	800	600
gender	1	0	1	0	1	0	0	1	0	1
education	BA	BA	MA	BA	PHD	MA	PHD	BA	MA	BA

Systematic Random Sampling

index	7	8	6	3	4	10	2	1	9	5
income	105	110	200	500	600	600	700	800	800	900
gender	0	1	0	1	0	1	0	1	0	1
education	PHD	BA	MA	MA	BA	BA	BA	BA	MA	PHD

Systematic Random Sampling

index	7	8	6	3	4	10	2	1	9	5
income	105	110	200	500	600	600	700	800	800	900
gender	0	1	0	1	0	1	0	1	0	1
education	PHD	BA	MA	MA	BA	BA	BA	BA	MA	PHD

Systematic Random Sampling

index	1	2	3	4	5	6	7	8	9	10
income	800	700	500	600	900	200	105	110	800	600
gender	1	0	1	0	1	0	0	1	0	1
education	BA	BA	MA	BA	PHD	MA	PHD	BA	MA	BA

Stratified Random Sampling

index	1	2	4	8	10	3	6	9	7	5
income	800	700	600	110	600	500	200	800	105	900
gender	1	0	0	1	1	1	0	0	0	1
education	BA	BA	BA	BA	BA	MA	MA	MA	PHD	PHD

Stratified Random Sampling

index	1	2	4	8	10	3	6	9	7	5
income	800	700	600	110	600	500	200	800	105	900
gender	1	0	0	1	1	1	0	0	0	1
education	BA	BA	BA	BA	BA	MA	MA	MA	PHD	PHD

Stratified Random Sampling

index	8	2	4	1	10	3	6	9	7	5
income	110	700	600	800	600	500	200	800	105	900
gender	1	0	0	1	1	1	0	0	0	1
education	BA	BA	BA	BA	BA	MA	MA	MA	PHD	PHD

Cluster Random Sampling

index	8	2	4	1	10	3	6	9	7	5
income	110	700	600	800	600	500	200	800	105	900
gender	1	0	0	1	1	1	0	0	0	1
education	BA	BA	BA	BA	BA	MA	MA	MA	PHD	PHD

Cluster Random Sampling

Random Sampling

+ Can be easily done
+ Least biased sampling method
- Chance that some characteristics are not taken into account

Stratified Random Sampling

+ Good sample of the data
+ Different sample strategies within a strata can be used
- Computationally more expensive

Systematic Random Sampling

+ Can be easily done
+ Good sample for the ordered attribute
- Possible bias towards the other attributes
- Problem with periodicities

Cluster Random Sampling

+ Cheap method when it is geographically convenient
+ Easy to increase sample size
- Least representative of the population

Sampling – Summary

- Think about the underlying data types
- Normalize your data to avoid wrong calculations/visual judgements
- Deal with missing values in an appropriate way
- Sample your data only if necessary (calculations take too long)

Guidelines – Data Foundation

[1] Jiawei Han, and Micheline Kamber, ”*Data mining: concepts and techniques*", 2001, Morgan Kauffmann.

References