BASIC STATISTICS FOR DATA SCIENCE

M. A. BENATIA,

DEC 2020

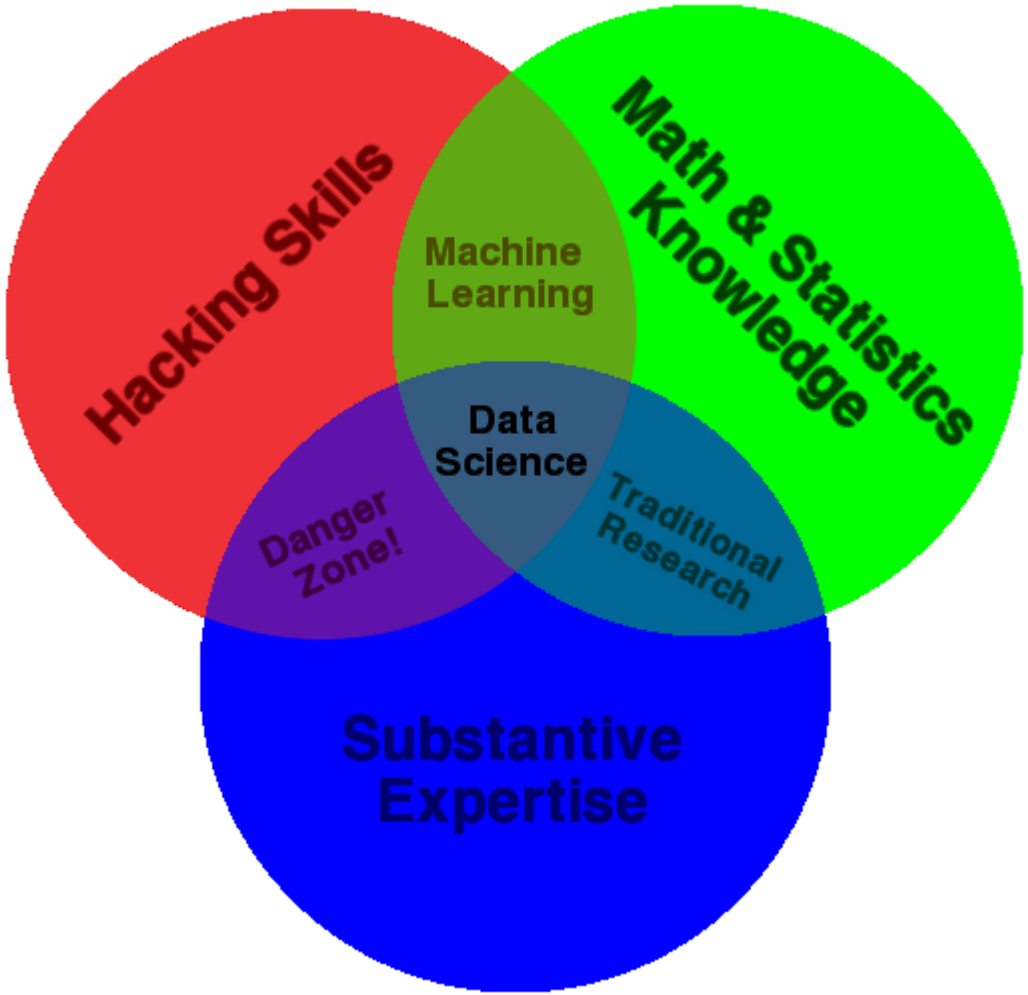
Chapter Topics

- ❑Why a Manager Needs to Know About Statistics
- ❑The Growth and Development of Modern Statistics
- ❑Some Important Definitions
- ❑Descriptive Versus Inferential Statistics
- ❑Why Data Are Needed
- ❑Types of Data and Their Sources
- ❑Organizing Numerical Data
- ❑Tabulating and Graphing Univariate Numerical Data
- ❑Measures of Central Tendency
- ❑Measures of Variation
- ❑Coefficient of Correlation

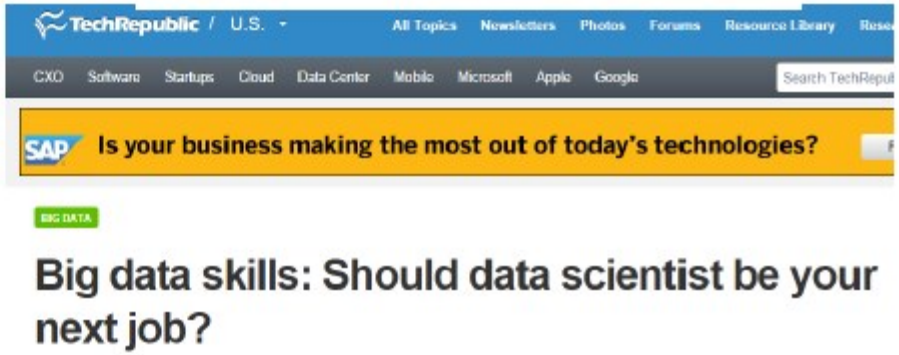
Part 0

Introduction to DataScience

Data Science – One Definition

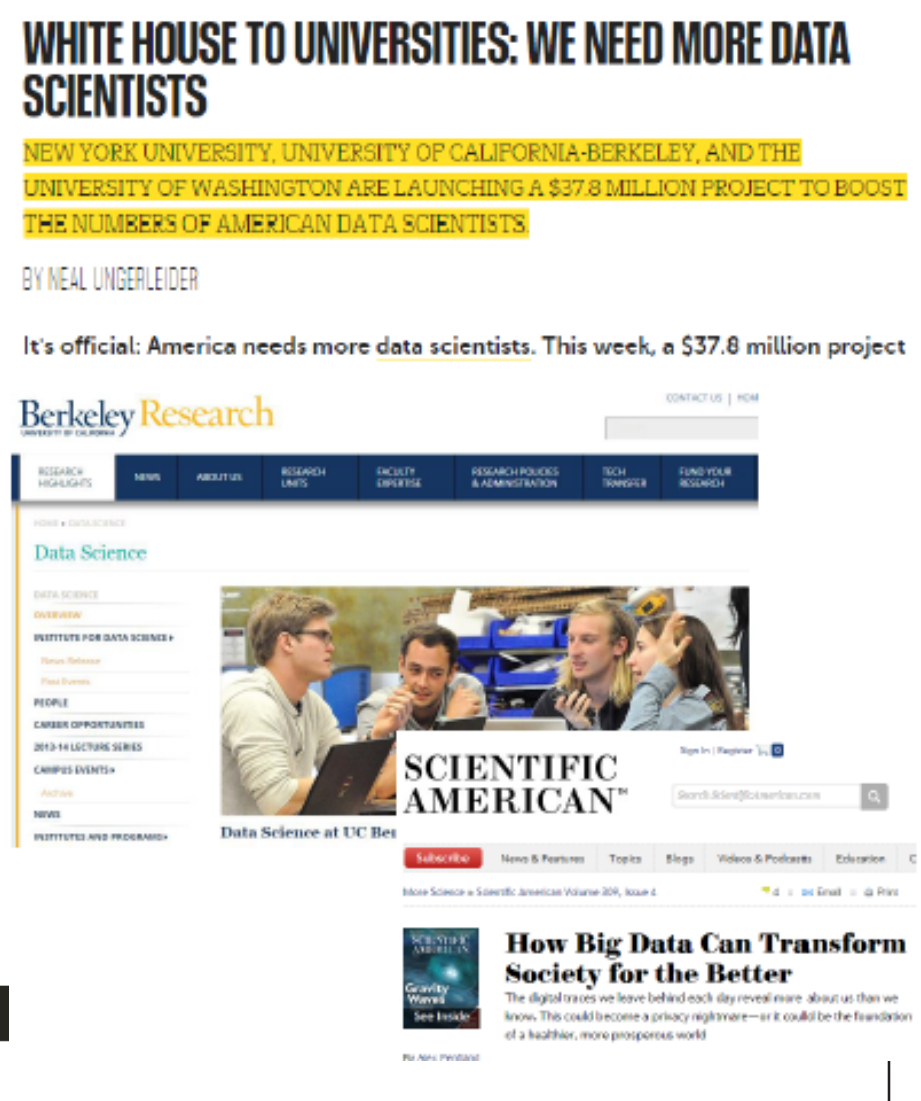


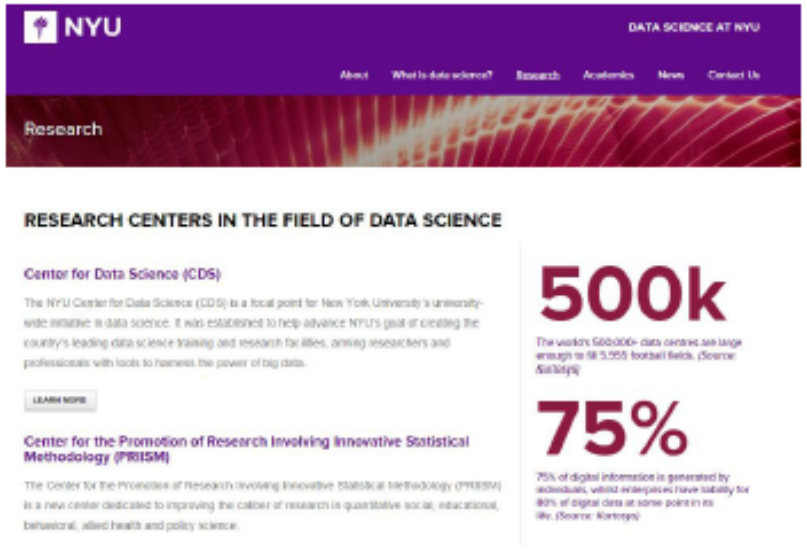
Data Scientists are in high demand



LINEACT cesi

Also in academia



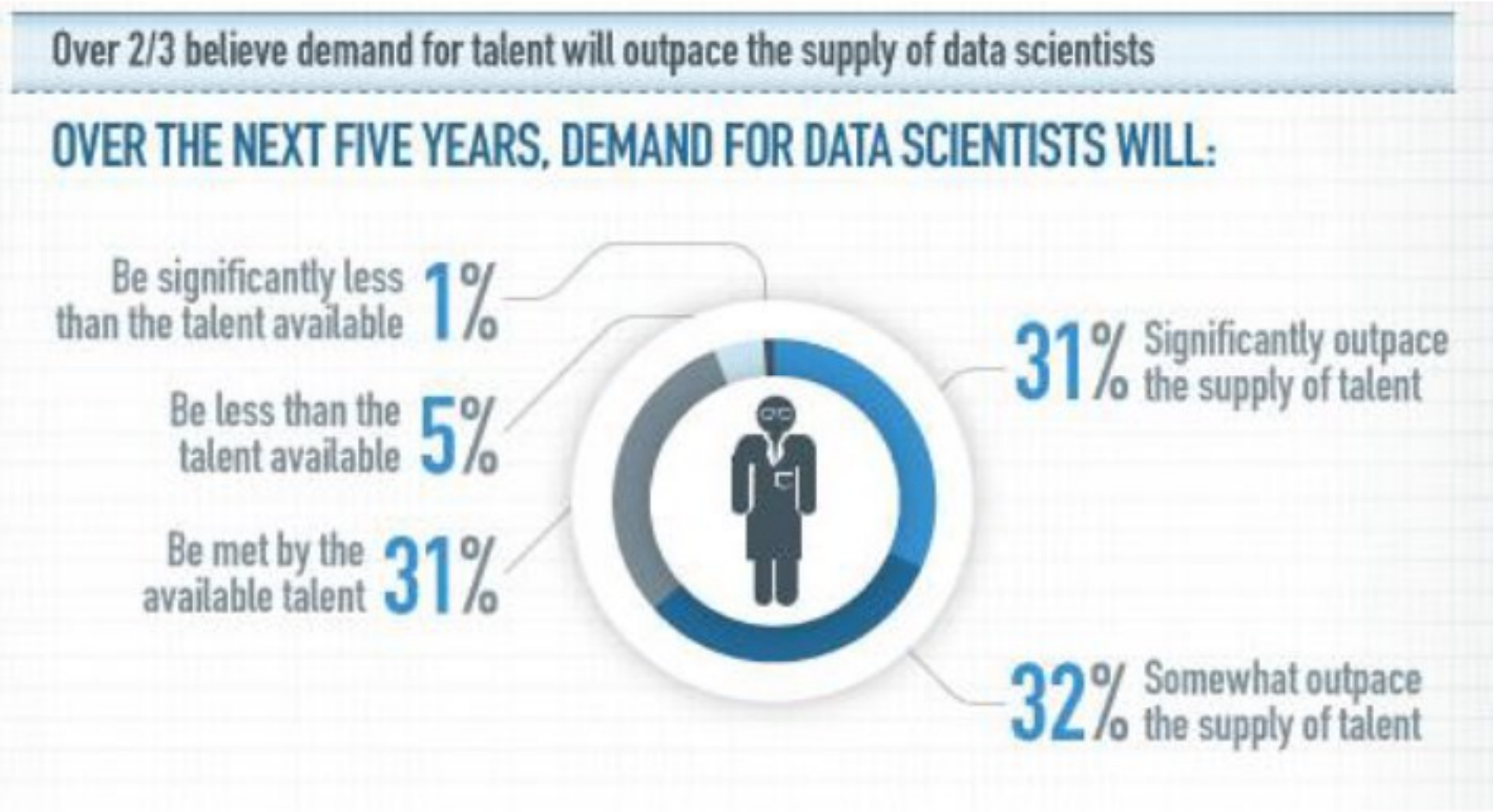


LINEACT cesi

Pays Well


Big Data, Big Paycheck
Median salary for analytics professionals and those specifically within data science, by level of experience.
Up to 3 years
Analytics professionals
$65,000
Data scientists
$80,000
4 to 8 years
$85,000
$120,000
9+ years
$115,000
$150,000
Note: Data do not include managers
Source: Burtch Works
The Wall Street Journal

Demand will outpace supply



Data Scientist Job Trend in last 3 years

Job postings

0.151%

Jobseeker interest

0.074%

LINEACT cesi

Source: indeed.com

Data Science: Why all the Excitement?

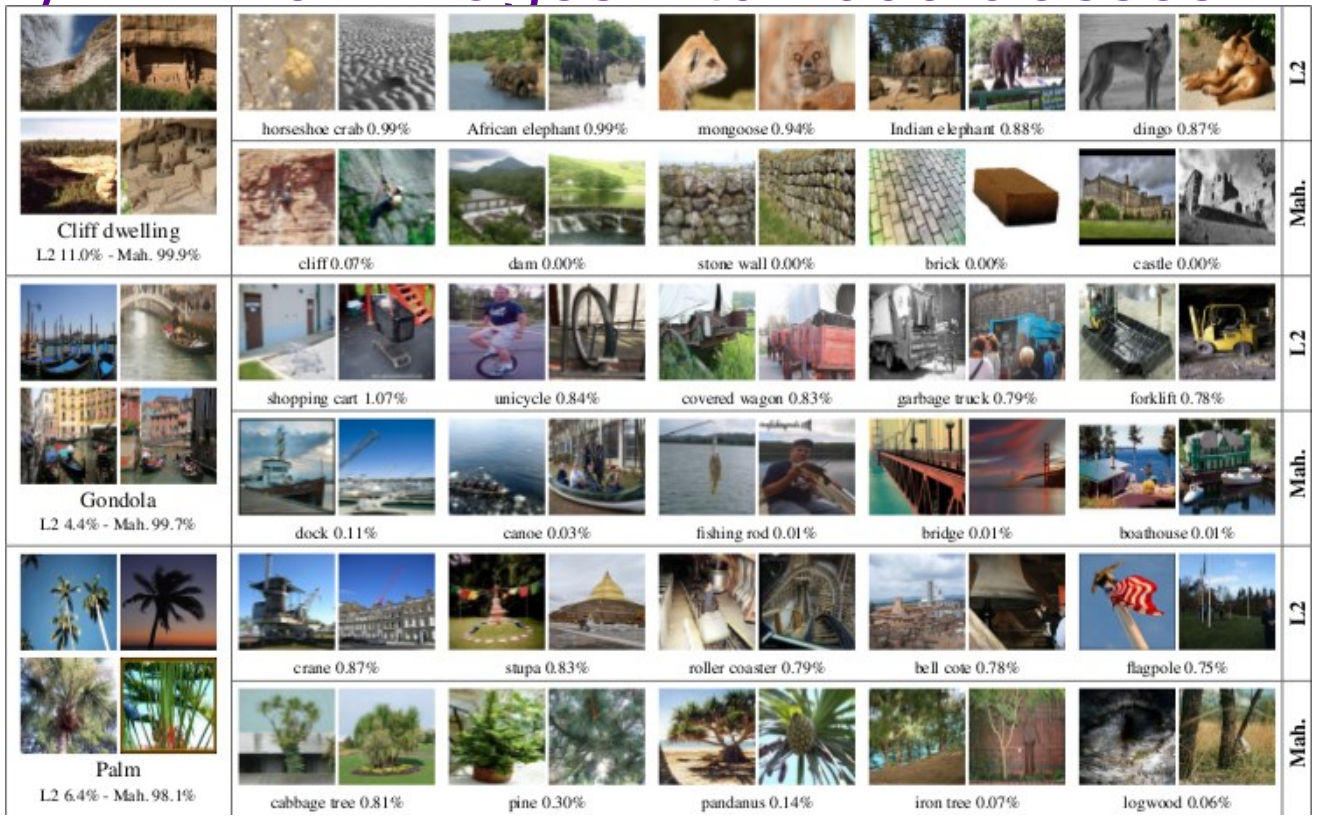
e.g.,
Google Flu Trends:

Detecting outbreaks
two weeks ahead
of CDC data

New models are estimating
which cities are most at risk
for spread of the Ebola virus.

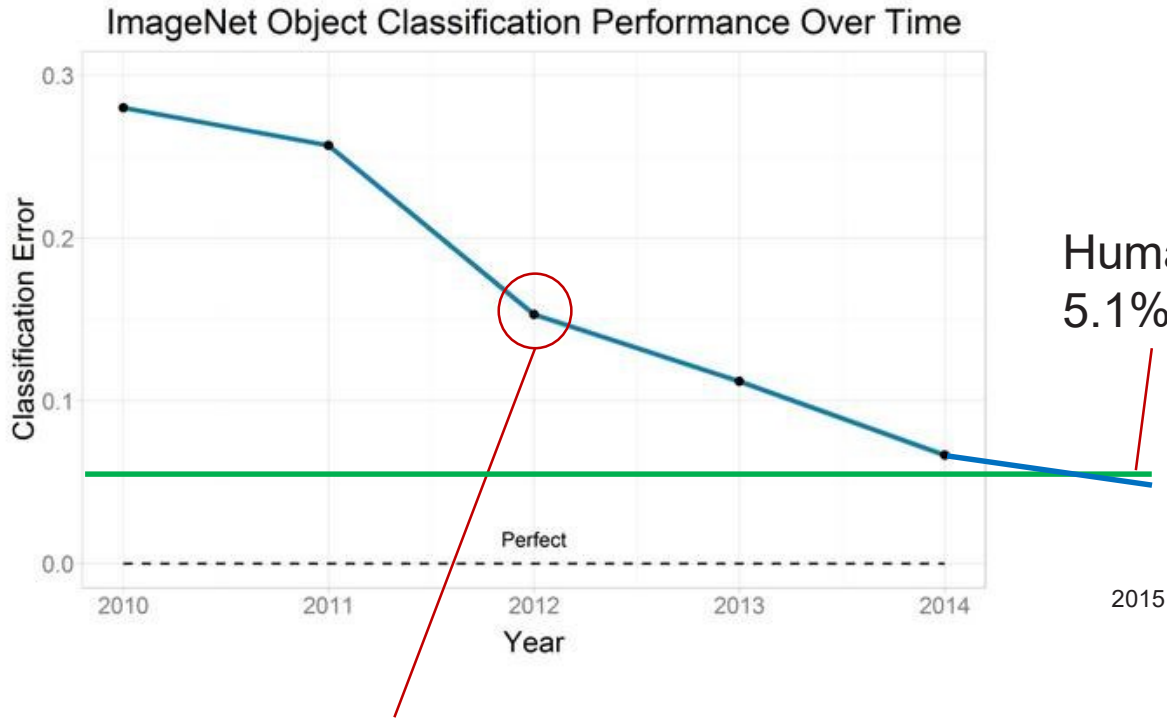
The unreasonable effectiveness of Deep Learning (CNNs)

2012 Imagenet challenge:
Classify 1 million images into 1000 classes.



The unreasonable effectiveness of Deep Learning (CNNs)

Performance of deep learning systems over time:

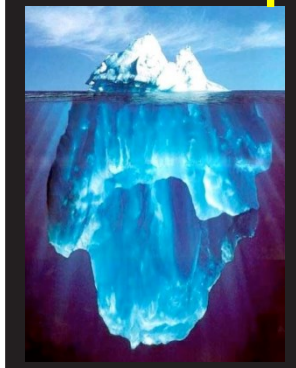


Krizhevsky, Sutskever, and Hinton, NIPS 2012

Where does data come from?

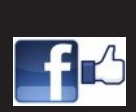
“Big Data” Sources

It’s All Happening On-line



Every:
Click
Ad impression
Billing event
Fast Forward, pause,…
Server request
Transaction
Network message
Fault
…

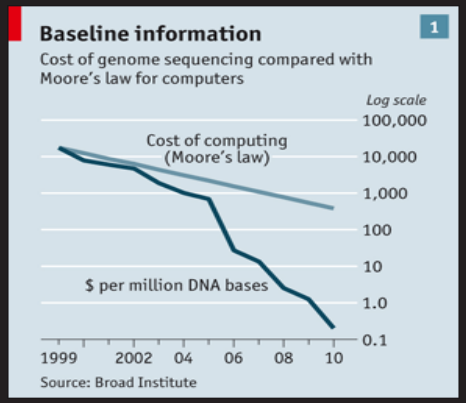
User Generated (Web & Mobile)




…
..


Internet of Things / M2M

Health/Scientific Computing



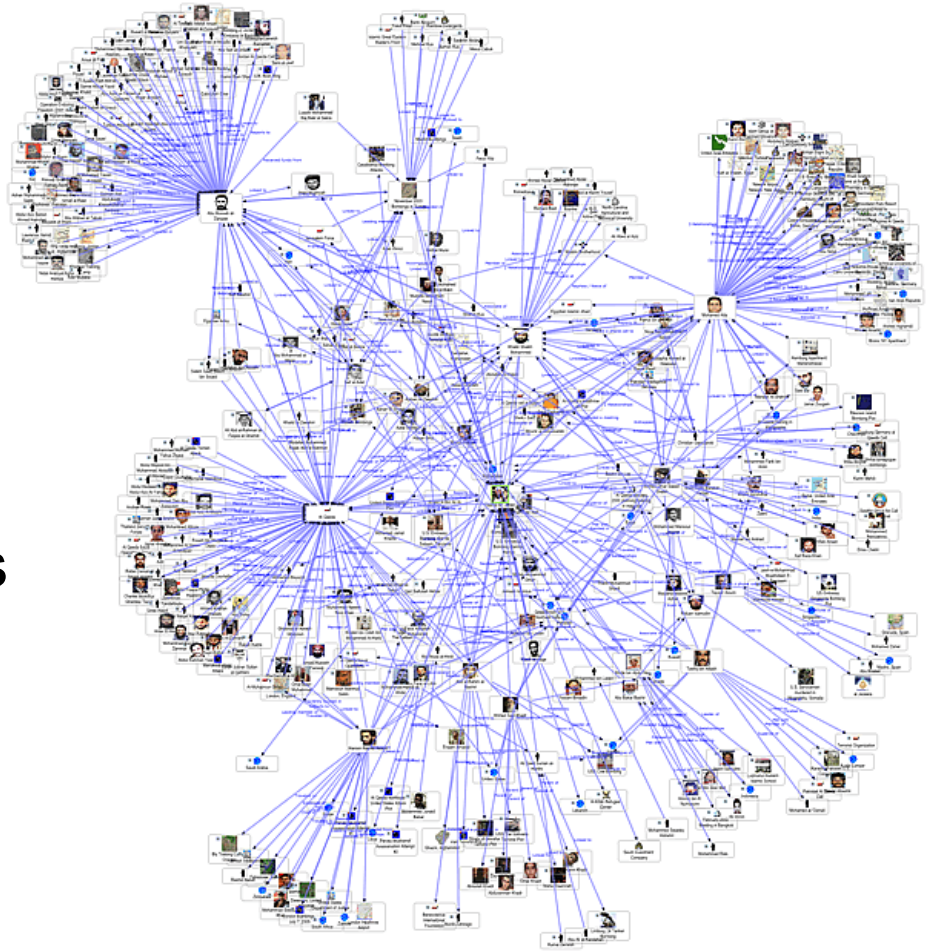
LINEACT cesi

Graph Data

Lots of interesting data has a graph structure:

- Social networks
- Communication networks
- Computer Networks
- Road networks
- Citations
- Collaborations/Relationships
- …

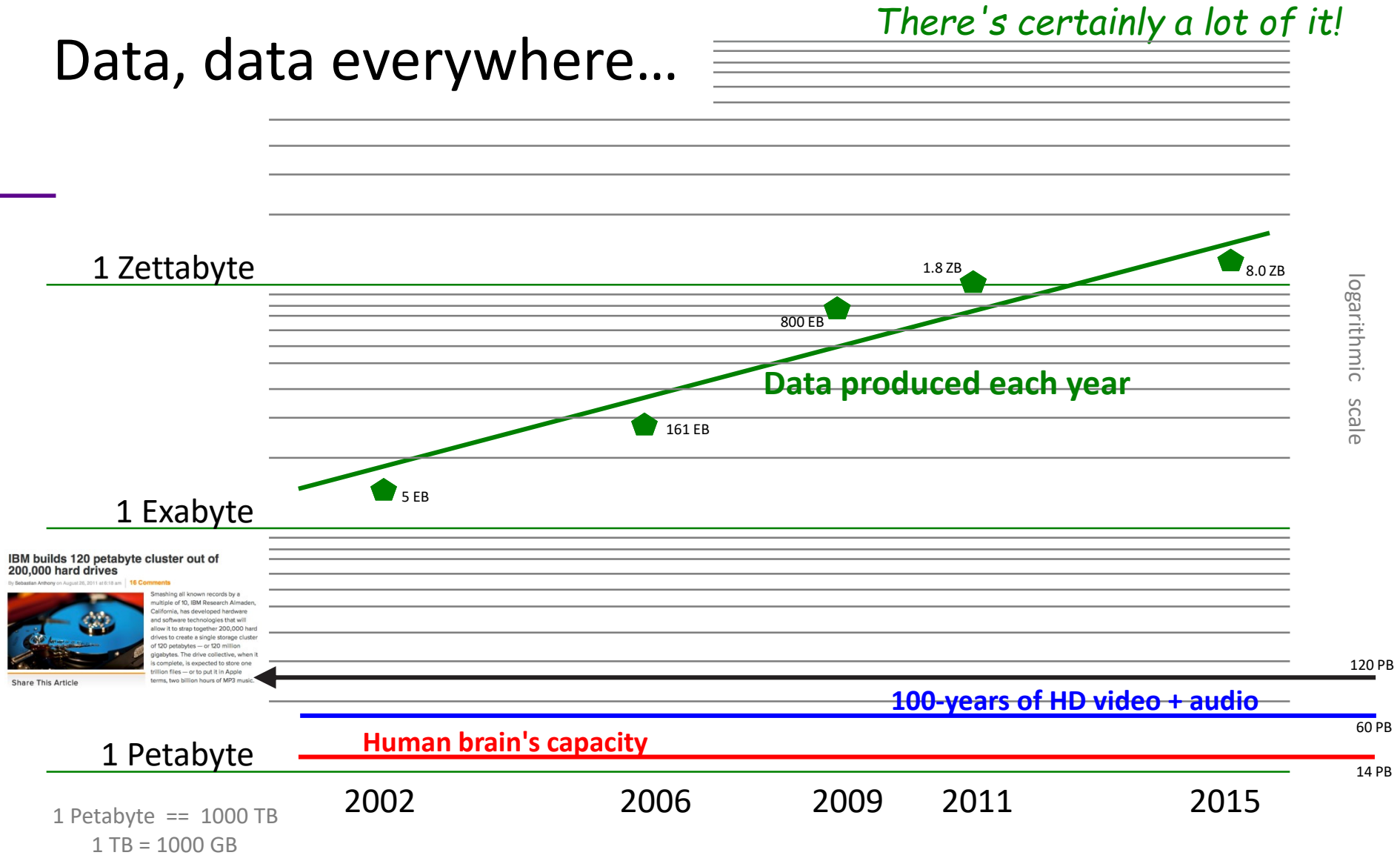
Some of these graphs can get quite large (e.g., Facebook* user graph)



LINEACT cesi

Data, data everywhere…

There's certainly a lot of it!



1 Petabyte == 1000 TB
1 TB = 1000 GB

References

(2015) 8 ZB: http://www.emc.com/collateral/analyst-reports/idc-extracting-value-from-chaos-ar.pdf

(2011) 1.8 ZB: http://www.emc.com/leadership/programs/digital-universe.htm

(2009) 800 EB: http://www.emc.com/collateral/analyst-reports/idc-digital-universe-are-you-ready.pdf

(2006) 161 EB: http://www.emc.com/collateral/analyst-reports/expanding-digital-idc-white-paper.pdf

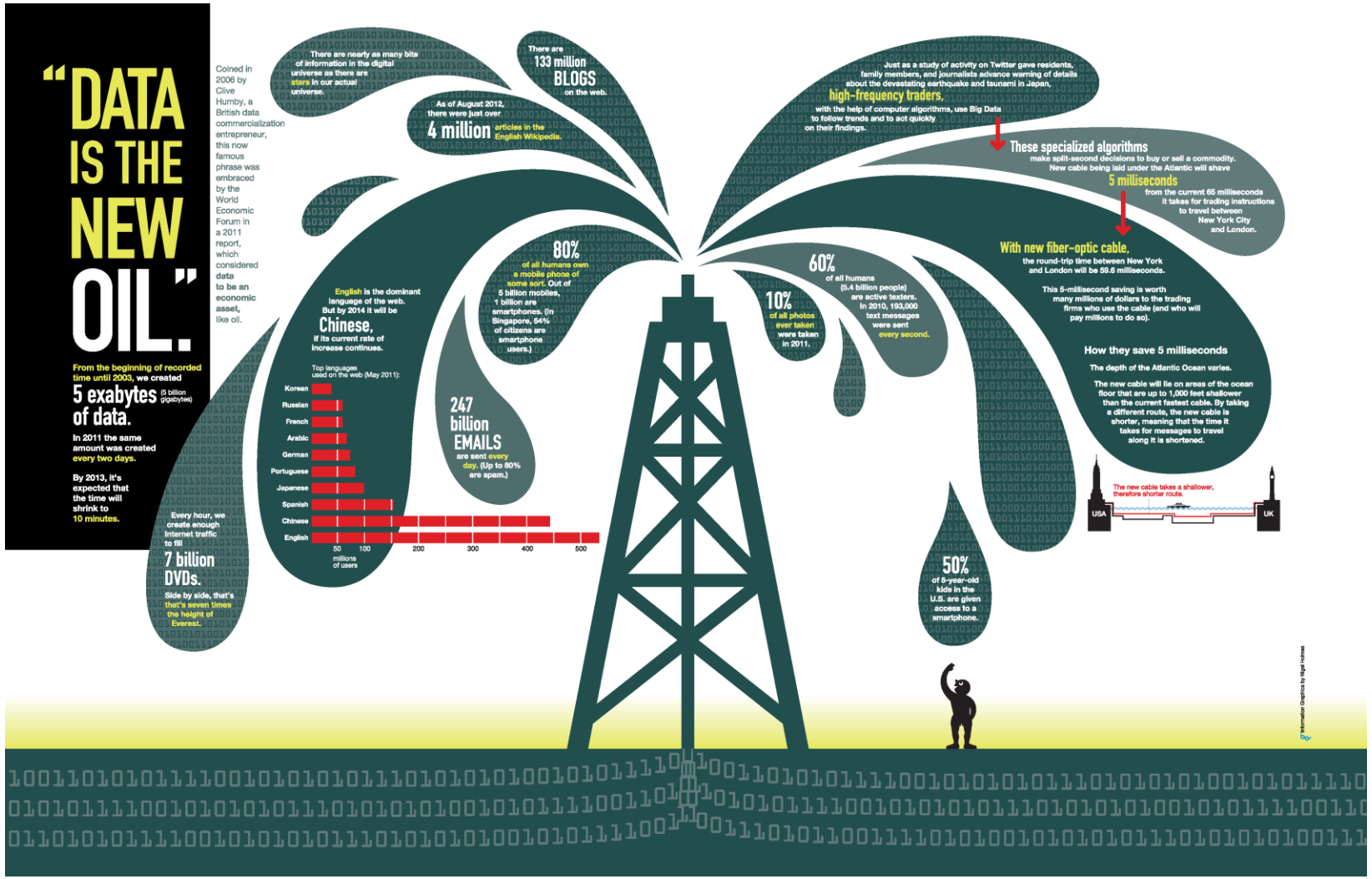
(2002) 5 EB: http://www2.sims.berkeley.edu/research/projects/how-much-info-2003/execsum.htm

(life in video) 60 PB: in 4320p resolution, extrapolated from 16MB for 1:21 of 640x480 video (w/sound) – almost certainly a gross overestimate, as sleep can be compressed significantly!

(brain) 14 PB: http://www.quora.com/Neuroscience-1/How-much-data-can-the-human-brain-store

LINEACT cesi

“Data is the New Oil” – World Economic Forum 2011



LINEACT cesi

“Data Science” an Emerging Field

O’Reilly Radar report, 2011

Data Science – A Definition

Data Science is the science which uses **computer science, statistics and machine learning, visualization and human-computer interactions** to **collect, clean, integrate, analyze, visualize, interact** with data to create data products.

collect, clean, integrate, analyze, visualize, interact

Goal of Data Science

Turn data into data products.

How to use data?

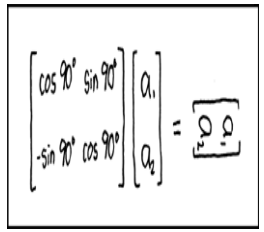
Data => exploratory analysis => knowledge models => product / decision marking

Data => predictive models => evaluate / interpret => product / decision making

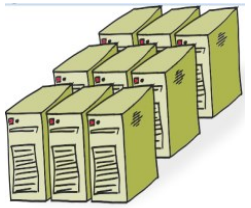
Data Scientist's Practice

Digging Around in Data

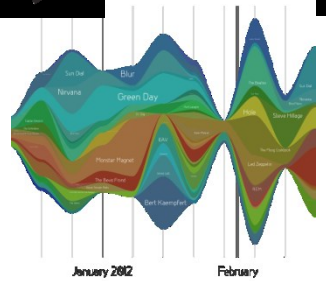
Clean, prep



Hypothesize Model



Large Scale Exploitation



Evaluate Interpret

Data Science *concerns*

Forbes | New Posts | Most Popular — Google's Driverless Car | Lists — Business Of Ba

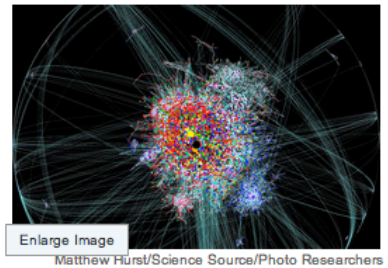
MARVELS | October 22, 2012, 11:28 a.m. ET

New Gumshoes Go Deep With Data

Article | Stock Quotes | Comments (4)

Email | Print | Save

A two-foot-long Lego model of the Imperial Star Destroyer from 'Star Wars' perches on the center table in the Hello Kitty-themed boardroom. Elsewhere in the building, taped-together cardboard boxes are piled to the ceiling. Amid the jumble of Care Bears, soda cans and packs of playing cards for "Magic: The Gathering" are camping tents and sleeping bags. This isn't kindergarten. It's "homesteading" week at Palantir, the "big-data" company that's the talk of Silicon Valley.



Enlarge Image

Matthew Hurst/Science Source/Photo Researchers

'Most of "big data" is a fraud, because it is really "dumb data," ' says Peter Thiel.

Once a year, employees (over 700 of them, 75% engineers) ward off corporate culture by living like this: a reminder of their rise from apartment start-up. Palantir occupies a growing number of buildings in downtown Palo Alto, and world-wide. It was recently valued at $4 billion.

Palantirians are the new Googlers. What search algorithms were to the 1990s, big data is today: a game changer. Imagine statistical analysis on steroids. Now multiply that.

ForbesBrandVoice Connecting marketers to the Forbes audience. What is this?

SAPVoice
RUN BETTER
+ Follow (450)

BUSINESS | 1/08/2013 @ 7:49AM | 2,383 views

Data Science: Buyer Beware

Ray Rivera, SAP

+ Comment Now + Follow Comments

Any field of study followed by the word "science", so goes the old wheeze, is not really a science, including computer science, climate science, police science, and investment science. And then there is the

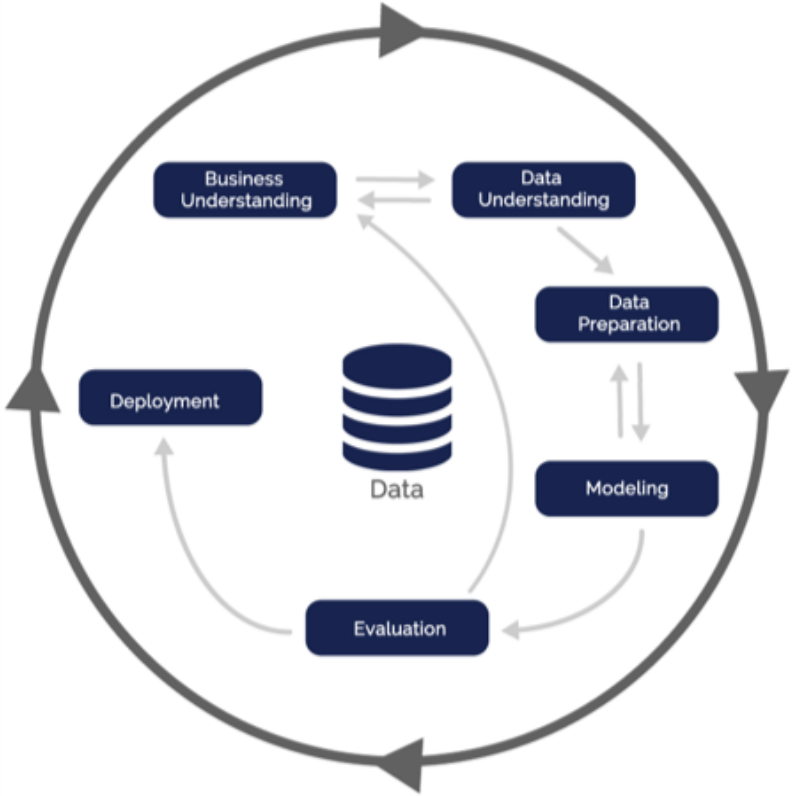
LINEACT cesi

Part 1

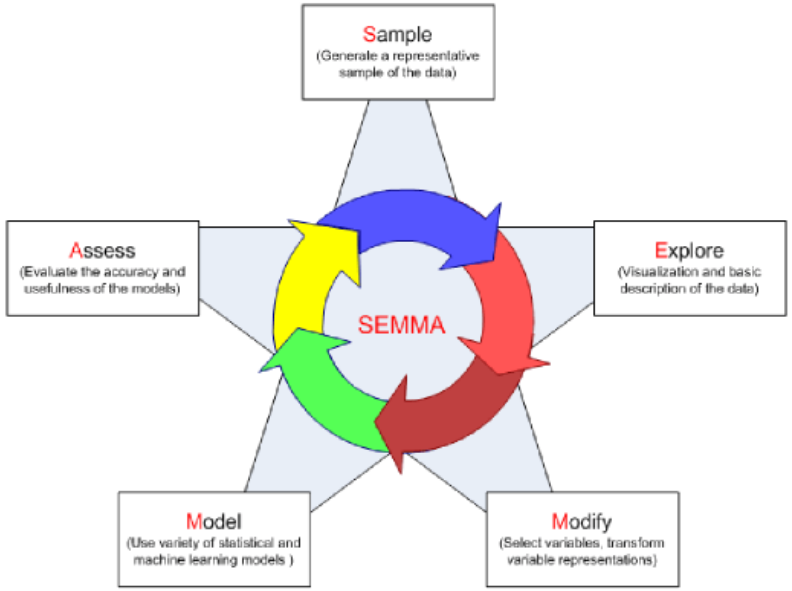
Data-Preprocessing

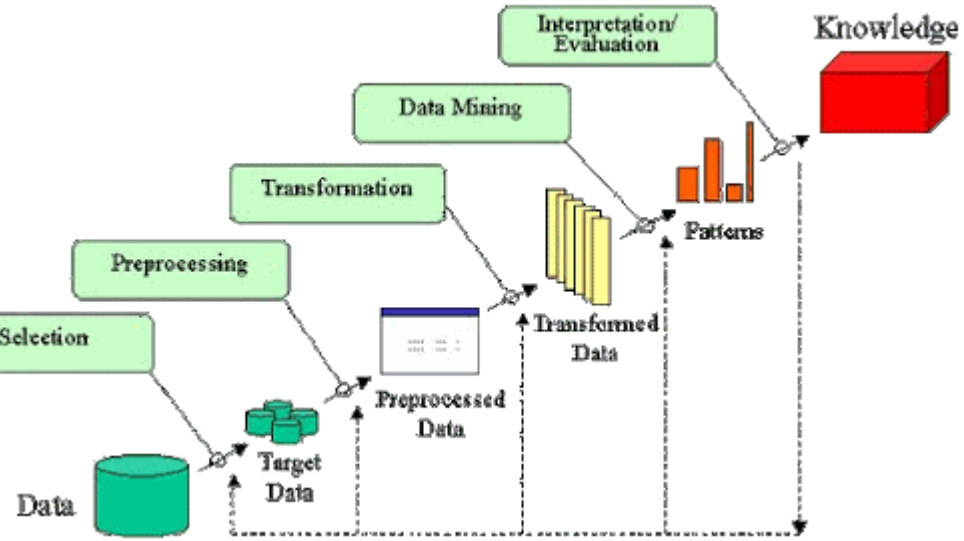
Why Data pre-processing?

CRISP-DM



SEMMA

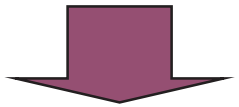




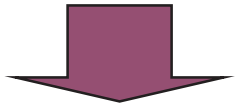
KDD

The Growth and Development of Modern Statistics

Needs of government to collect data on its citizenry



The development of the mathematics of probability theory

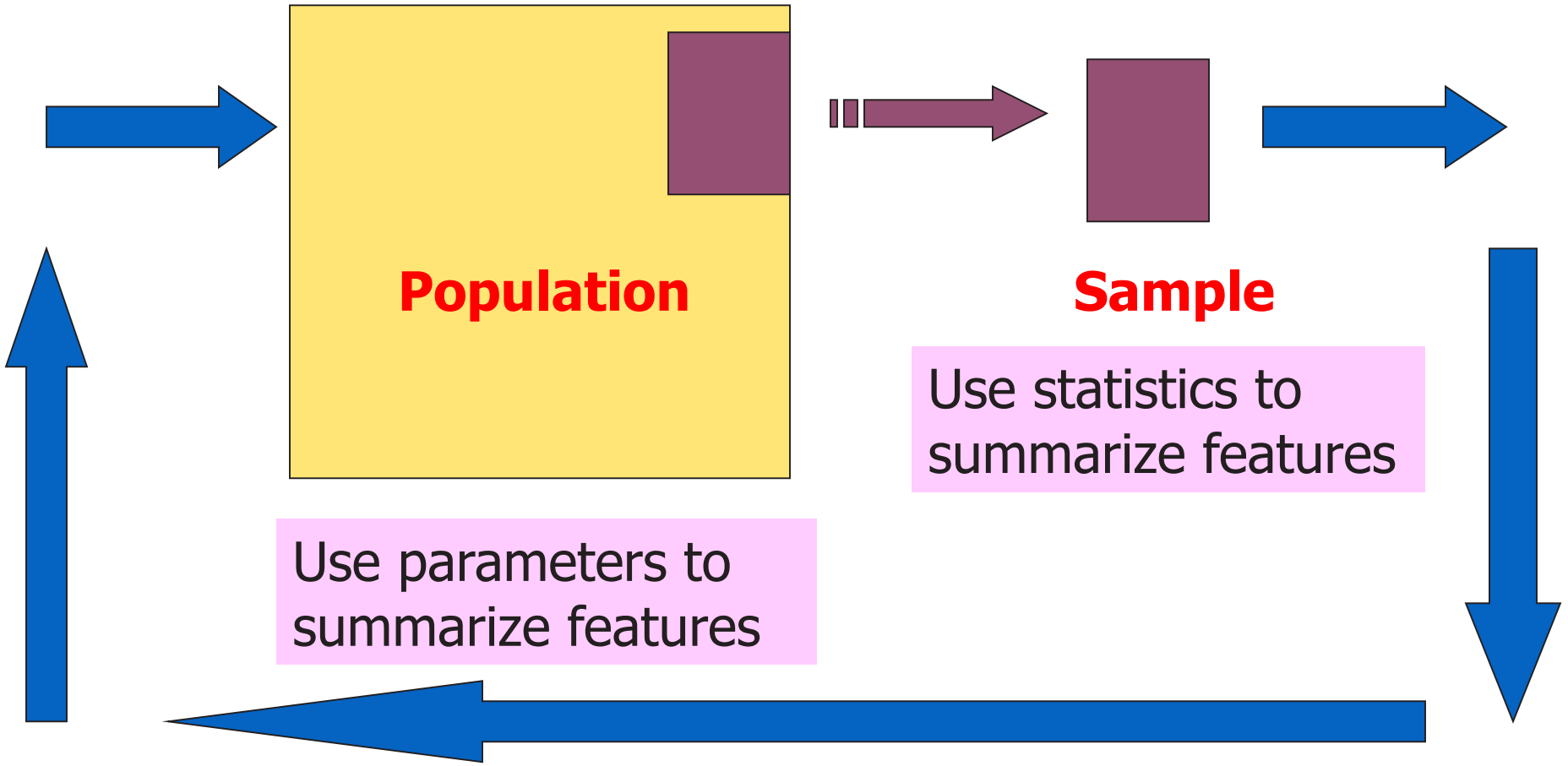


The advent of the computer

Some Important Definitions

- A Population (Universe) is the Whole Collection of Things Under Consideration
- A Sample is a Portion of the Population Selected for Analysis
- A Parameter is a Summary Measure Computed to Describe a Characteristic of the Population
- A Statistic is a Summary Measure Computed to Describe a Characteristic of the Sample

Population and Sample



Statistical Methods

Descriptive Statistics

- Collecting, presenting, and characterizing data

Inferential Statistics

- Drawing conclusions and/or making decisions concerning a population based only on sample data

Descriptive Statistics

Collect Data

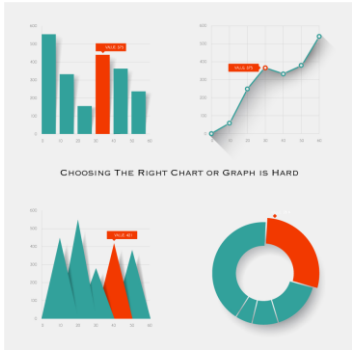
- E.g., Survey

Present Data

- E.g., Tables and graphs

Characterize Data

- E.g., Sample Mean $= \frac{\sum X_i}{n}$



Descriptive Statistics

Variable	Obs	Mean	Std.Dev.	Min	Max
price	74	6165.257	2949.496	3291	15906
mpg	74	21.297	5.786	12	41
rep78	69	3.406	.99	1	5
headroom	74	2.993	.846	.846	5
trunk	74	13.757	4.277	5	23
weight	74	3019.459	777.194	1760	4840
length	74	187.932	22.266	142	233
turn	74	39.649	4.399	31	51
displacement	74	197.297	91.837	79	425
gear_ratio	74	3.015	.456	2.19	3.89
foreign	74	.297	.46	0	1

Inferential Statistics

Drawing conclusions and/or making decisions concerning a population based on sample results.

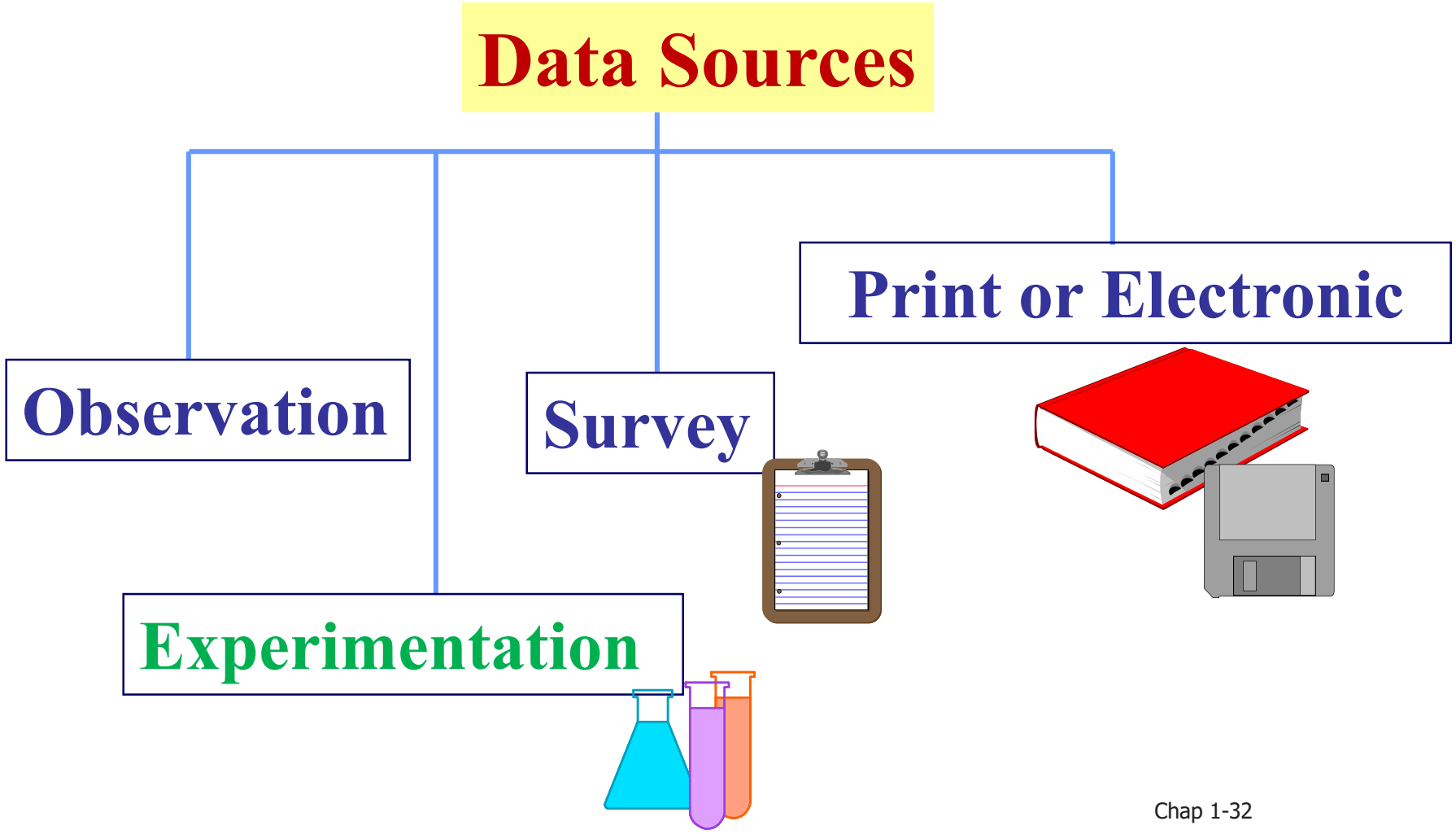
Estimation

- E.g. Estimate the population mean weight using the sample mean weight

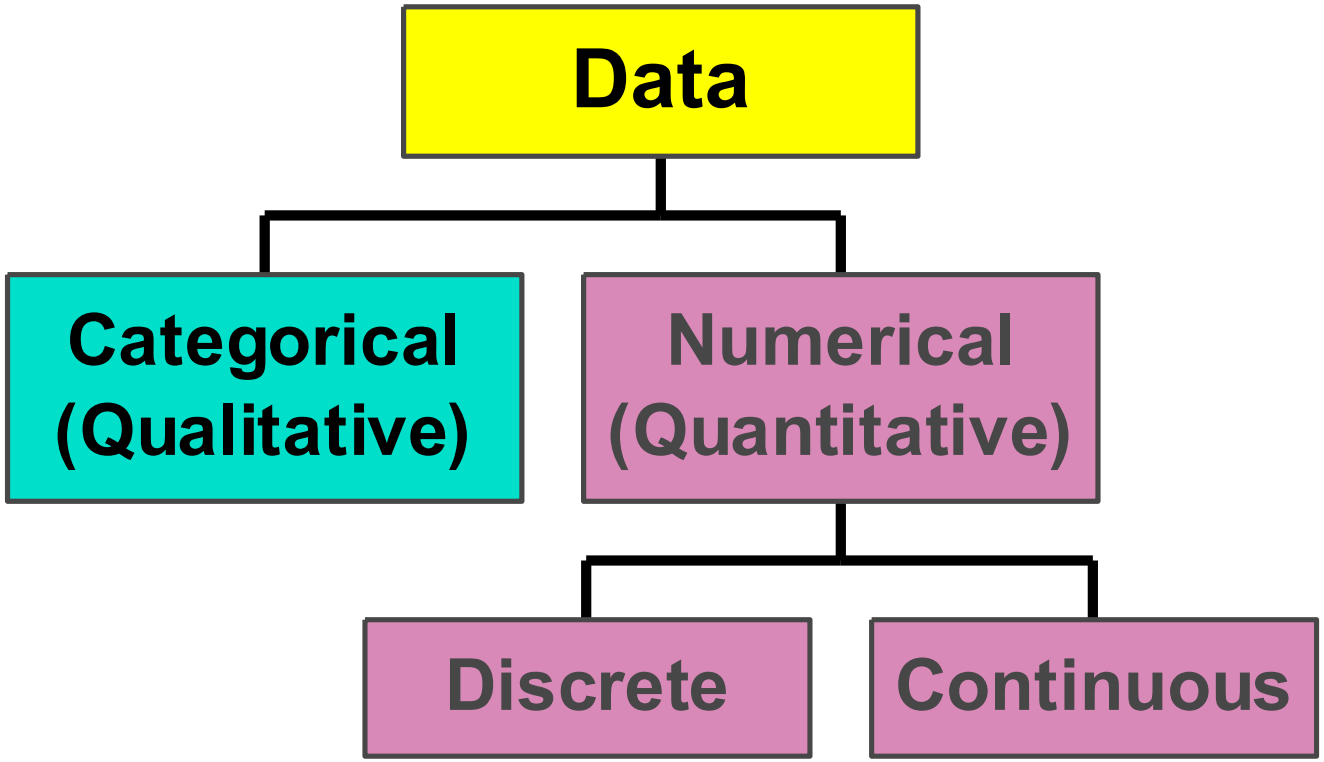
Hypothesis Testing

- E.g. Test the claim that the population mean weight value is 120 pounds

Data Sources



Types of Data



Type of Data (Continued)

Categorical random variables yield categorical responses

- **E.g. Are you married? Yes or No**

Numerical random variables yield numerical responses

- **Discrete random variables** yield numerical response that arise from a counting process
  - **E.g. How many cars do you own? 3 cars**
- **Continuous random variables** yield numerical responses that arise from a measuring process
  - **E.g. What is your weight? 130 pounds**

Levels of Measurement and Types of Measurement Scales

Nominal Scale – distinct categories in which *no ordering* is implied

- E.g. Type of stocks invested: growth, income, other and none

Ordinal Scale – distinct categories in which *ordering* is implied

- E.g. Student grades: A, B, C, D or F

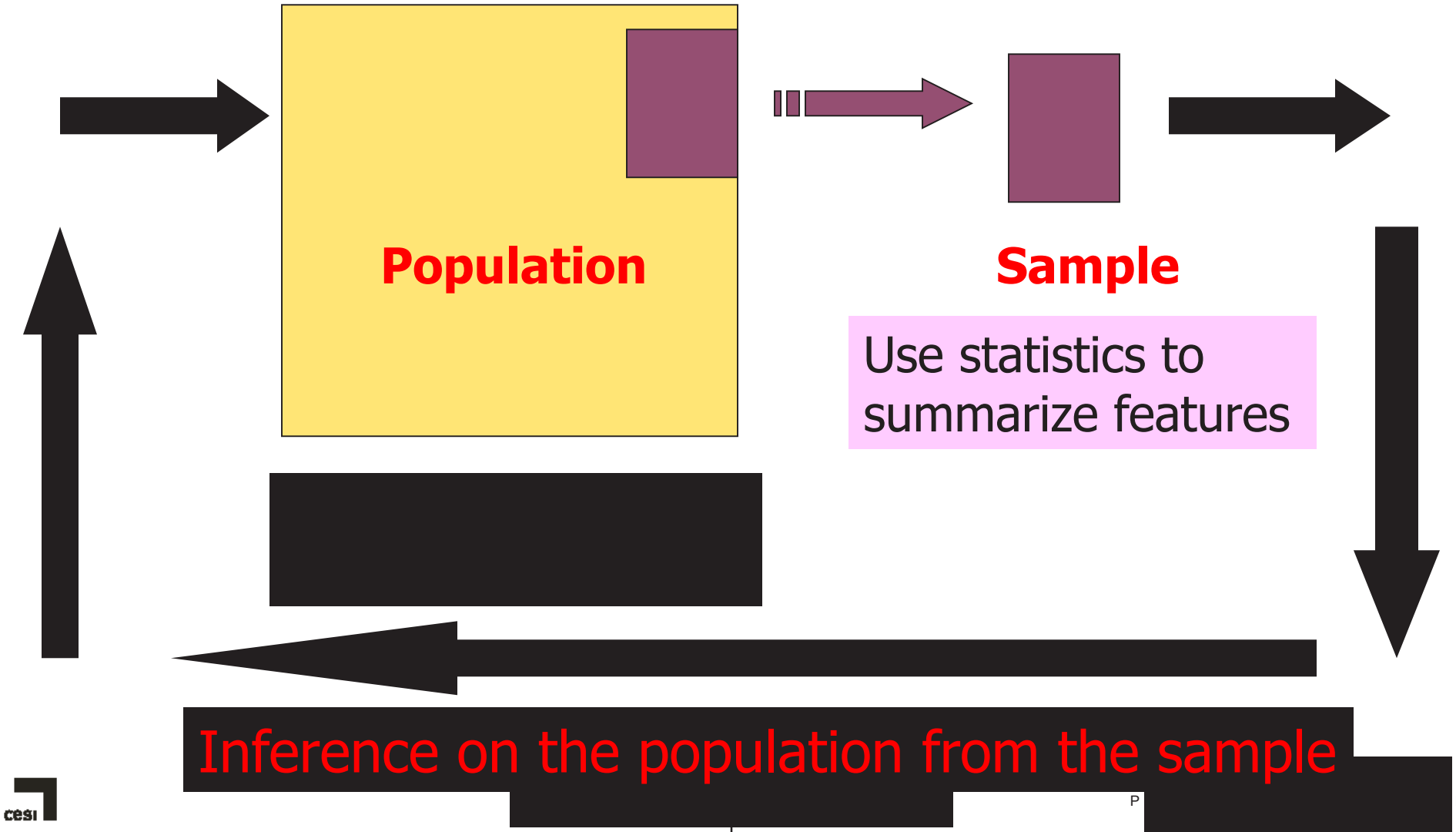
Interval Scale – an ordered scale in which the difference between the measurements *does not involve* a true zero point

- E.g. Temperature in degrees Celsius

Ratio Scale – an ordered scale in which the difference between the measurements *involves* a true zero point

- E.g. Weight in pounds

Population and Sample



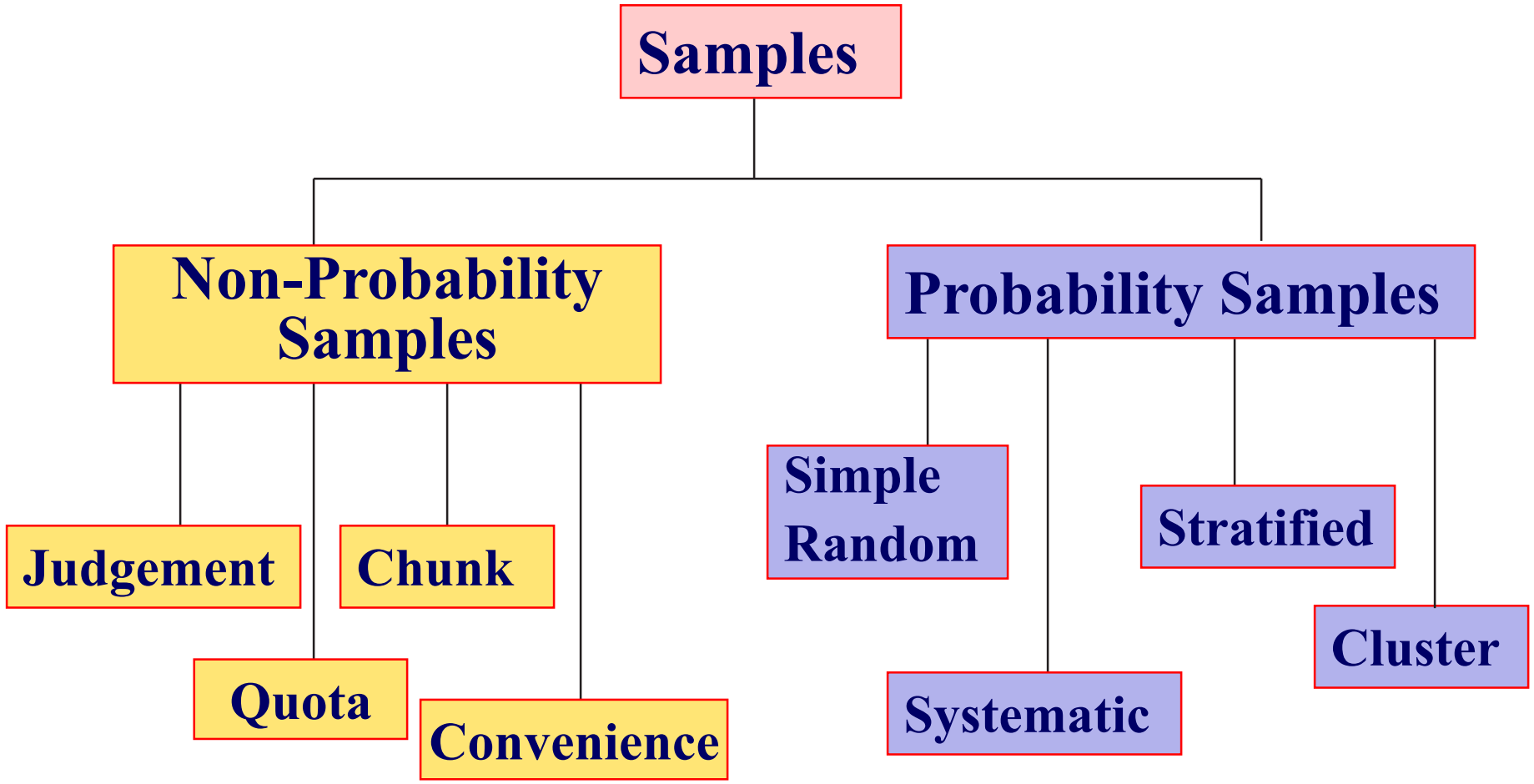
Reasons for Drawing a Sample

Less Time Consuming Than a Census

Less Costly to Administer Than a Census

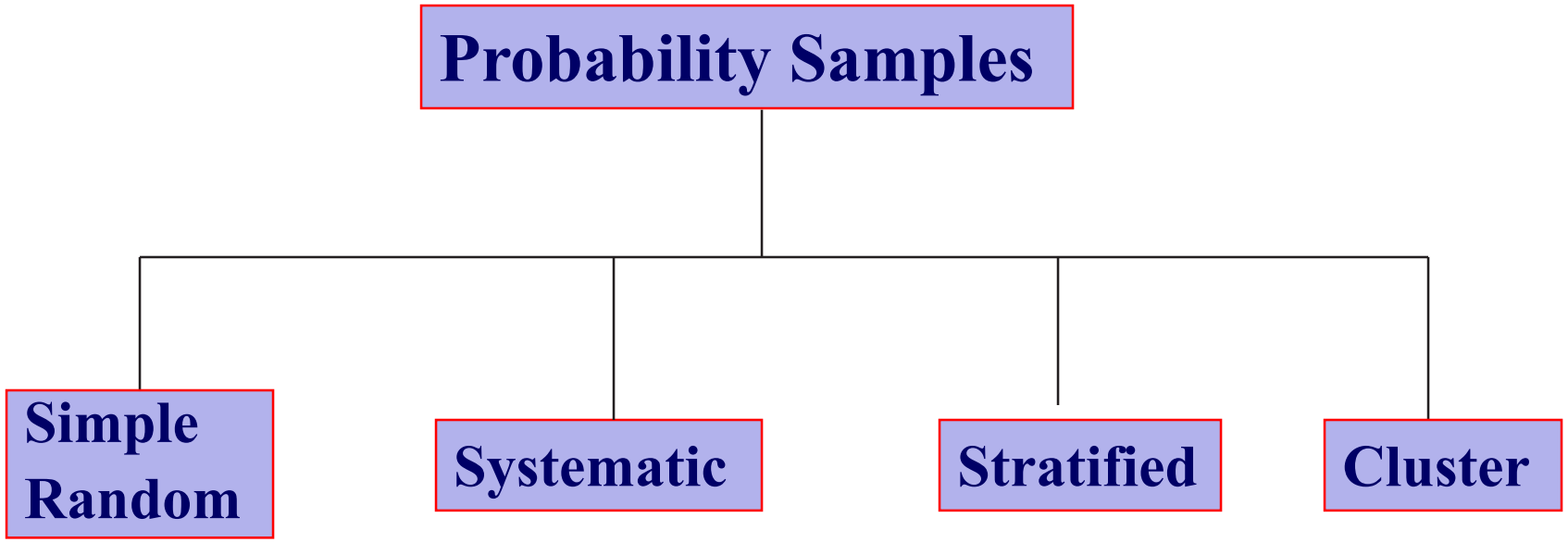
Less Cumbersome and More Practical to Administer Than a Census of the Population

Types of Sampling Methods



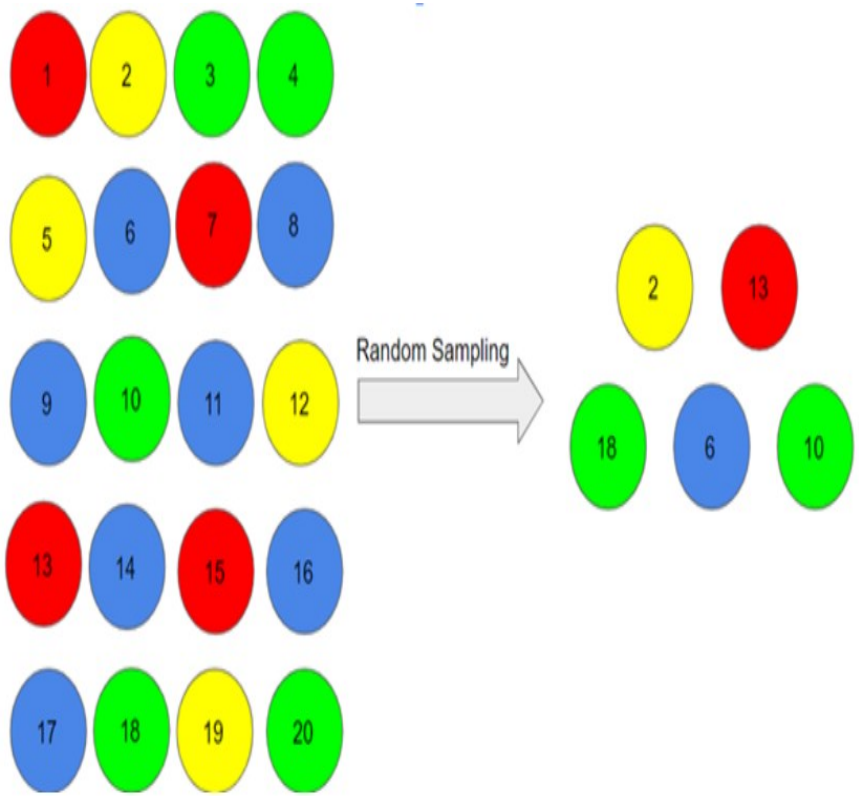
Probability Sampling

Subjects of the Sample are Chosen Based on Known Probabilities



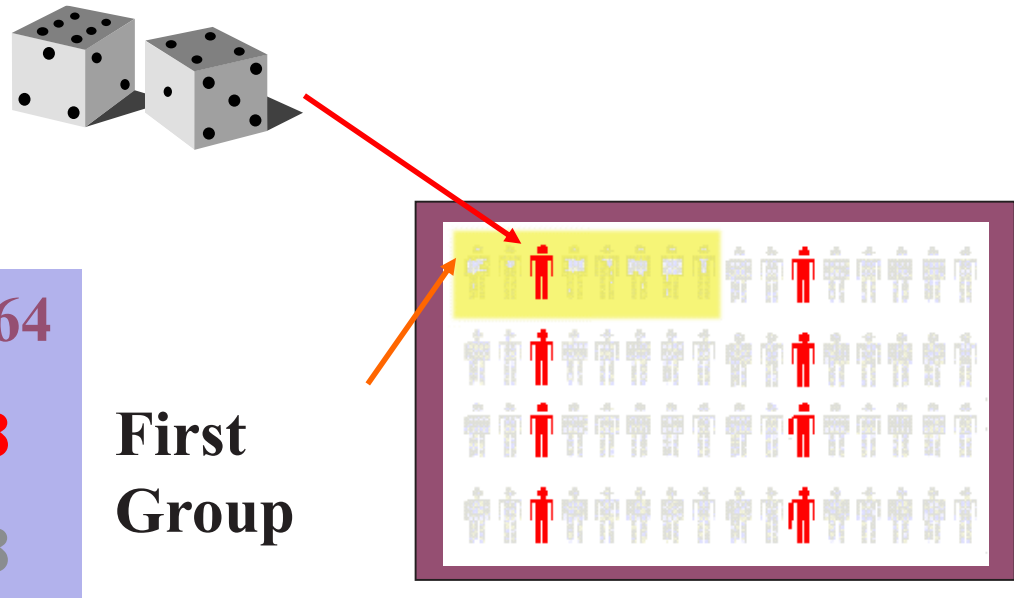
Simple Random Samples

- Every Individual or Item from the Frame Has an Equal Chance of Being Selected
- Selection May Be With Replacement or Without Replacement
- One May Use Table of Random Numbers or Computer Random Number Generators to Obtain Samples



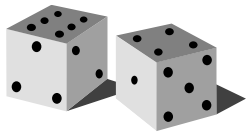
Systematic Samples

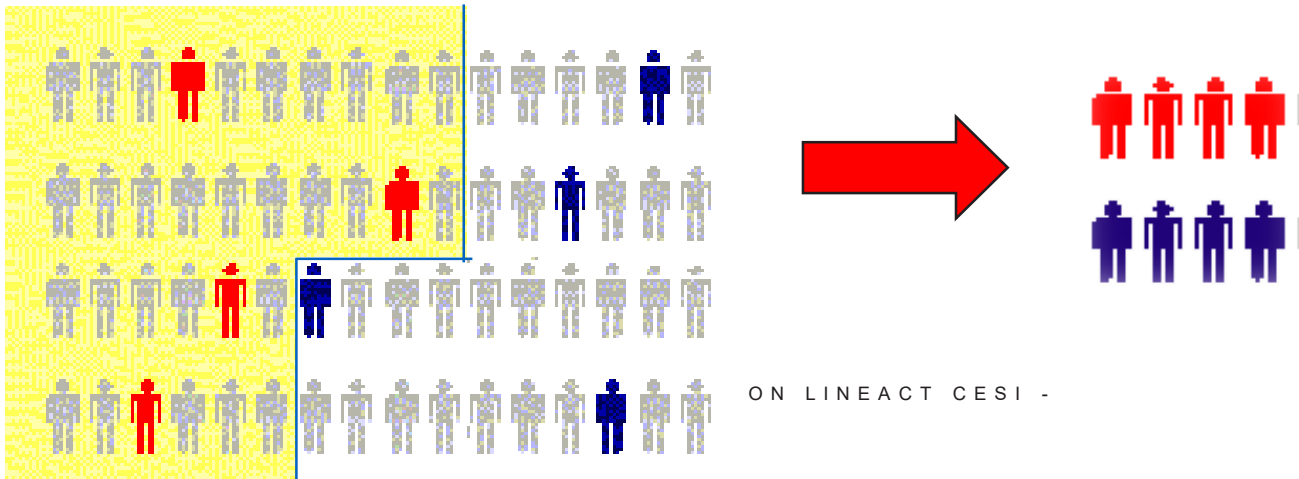
- Decide on Sample Size: n
- Divide Frame of N individuals into Groups of k Individuals: k=N/n
- Randomly Select One Individual from the 1st Group
- Select Every k-th Individual Thereafter



Stratified Samples

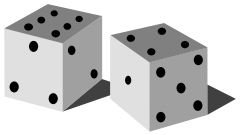
- Population Divided into 2 or More Groups According to Some Common Characteristic
- Simple Random Sample Selected from Each Group
- The Two or More Samples are Combined into One

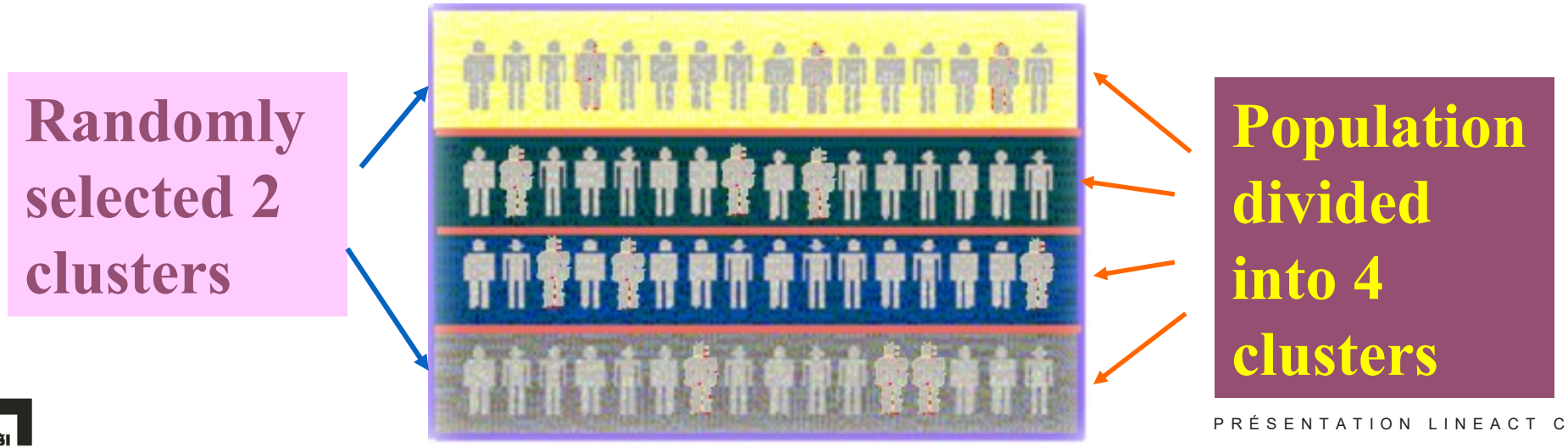




Cluster Samples

- Population Divided into Several “Clusters,” Each Representative of the Population
- A Random Sampling of Clusters is Taken
- All Items in the Selected Clusters are Studied





Advantages and Disadvantages

Simple Random Sample & Systematic Sample

- Simple to use
- May not be a good representation of the population's underlying characteristics

Stratified Sample

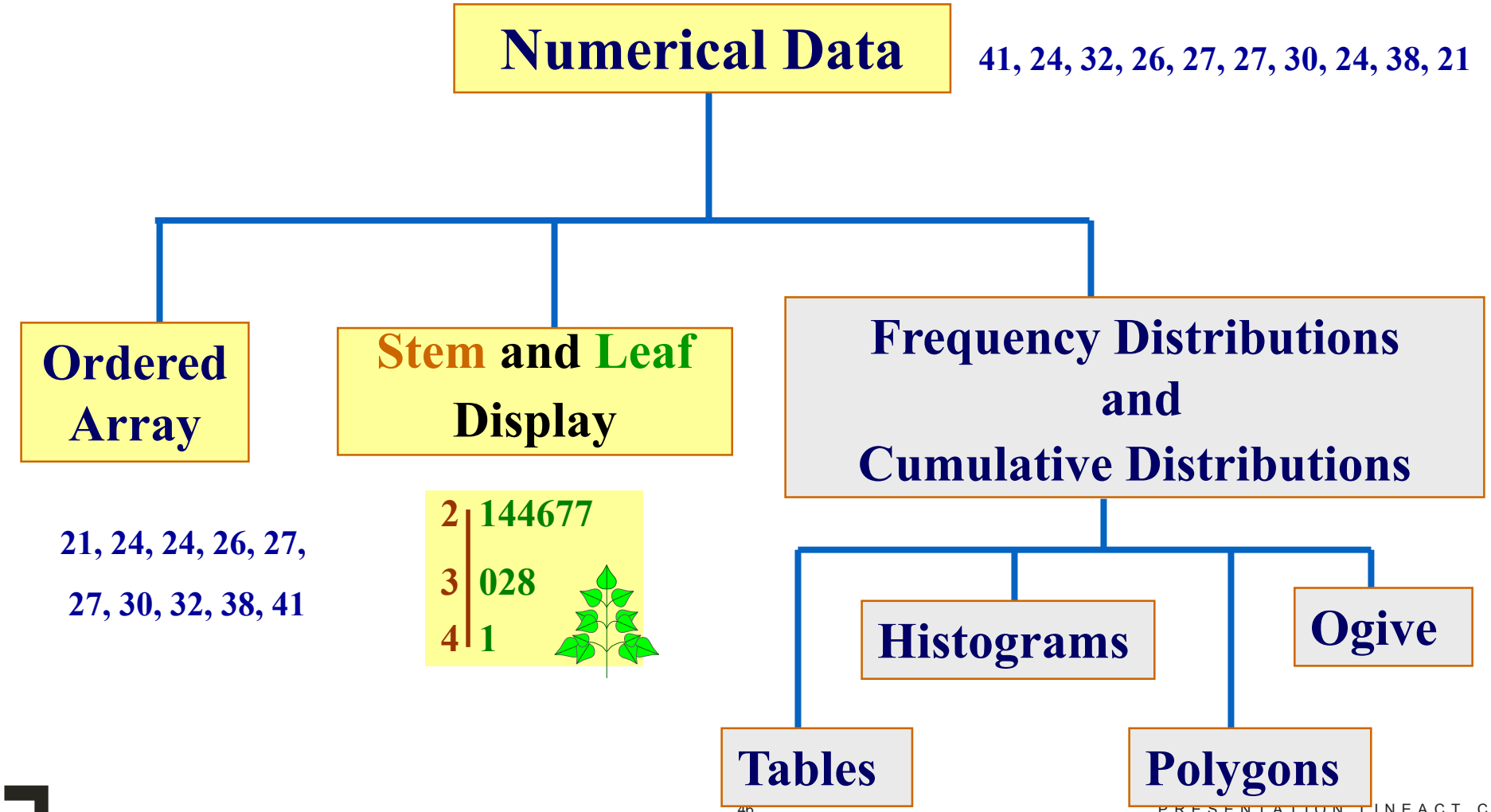
- Ensures representation of individuals across the entire population

Cluster Sample

- More cost effective
- Less efficient (need larger sample to acquire the same level of precision)

Part 2
Presenting Data in Tables and Charts

Organizing Numerical Data



Organizing Numerical Data *(continued)*

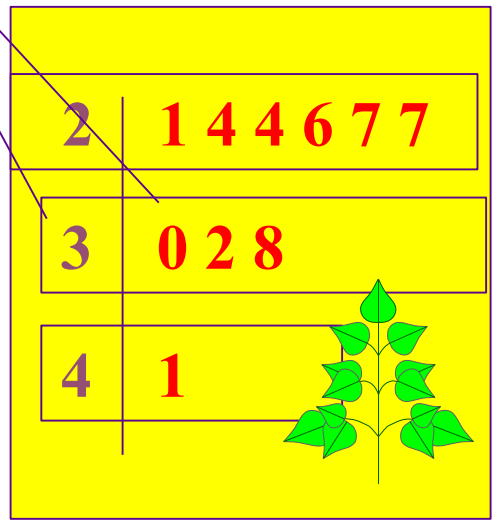
Data in *Raw* Form (as Collected):

24, 26, 24, 21, 27, 27, 30, 41, 32, 38

Data in ***Ordered Array* from *Smallest to Largest*:**

21, 24, 24, 26, 27, 27, **30**, 32, 38, 41

Stem-and-Leaf Display:



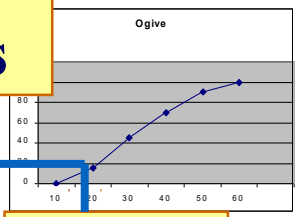
Tabulating and Graphing Numerical Data

Numerical Data 41, 24, 32, 26, 27, 27, 30, 24, 38, 21

- **Ordered Array**
  - 21, 24, 24, 26, 27, 27, 30, 32, 38, 41
- **Stem and Leaf Display**

Stem	Leaf
2	144677
3	028
4	1

- **Frequency Distributions and Cumulative Distributions**
  - **Tables**
  - **Histograms**
  - **Ogive**



  - **Polygons**

Tabulating Numerical Data: Frequency Distributions

- Sort Raw Data in Ascending Order

12, 13, 17, 21, 24, 24, 26, 27, 27, 30, 32, 35, 37, 38, 41, 43, 44, 46, 53, 58

- Find Range: **58 - 12 = 46**
- Select Number of Classes: **5 (usually between 5 and 15)**
- Compute Class Interval (Width): **10 (46/5 then round up)**
- Determine Class Boundaries (Limits):**10, 20, 30, 40, 50, 60**
- Compute Class Midpoints: **15, 25, 35, 45, 55**
- Count Observations & Assign to Classes

Frequency Distributions, Relative Frequency Distributions and Percentage Distributions

Data in Ordered Array:

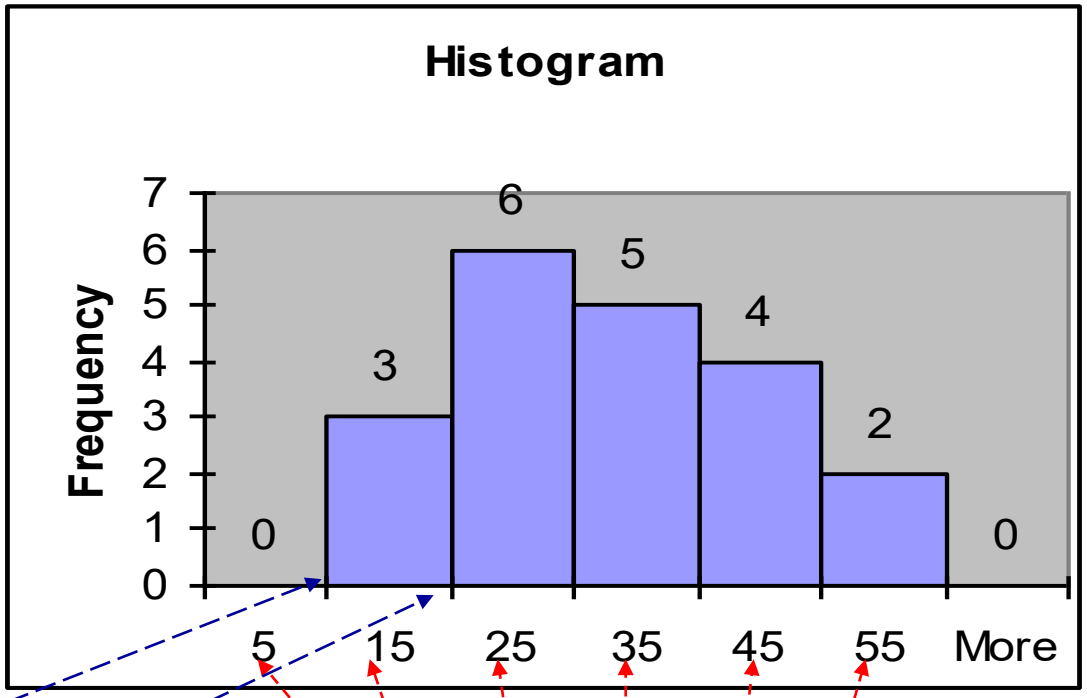
12, 13, 17, 21, 24, 24, 26, 27, 27, 30, 32, 35, 37, 38, 41, 43, 44, 46, 53, 58

Class	Frequency	Relative Frequency	Percentage
10 but under 20	3	.15	15
20 but under 30	6	.30	30
30 but under 40	5	.25	25
40 but under 50	4	.20	20
50 but under 60	2	.10	10
Total	20	1	100

Graphing Numerical Data: The Histogram

Data in Ordered Array:

12, 13, 17, 21, 24, 24, 26, 27, 27, 30, 32, 35, 37, 38, 41, 43, 44, 46, 53, 58



No Gaps Between Bars

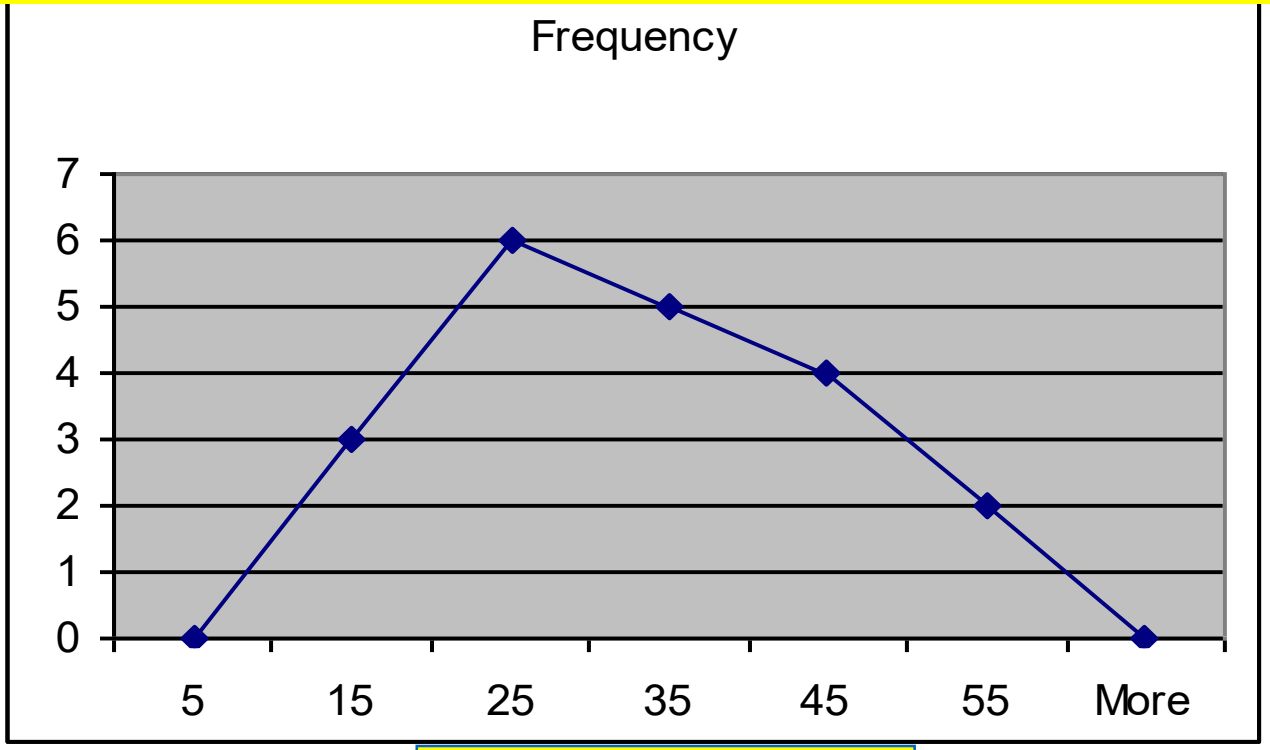
Class Boundaries

Class Midpoints

Graphing Numerical Data: The Frequency Polygon

Data in Ordered Array:

12, 13, 17, 21, 24, 24, 26, 27, 27, 30, 32, 35, 37, 38, 41, 43, 44, 46, 53, 58



Tabulating Numerical Data: Cumulative Frequency

Data in Ordered Array:

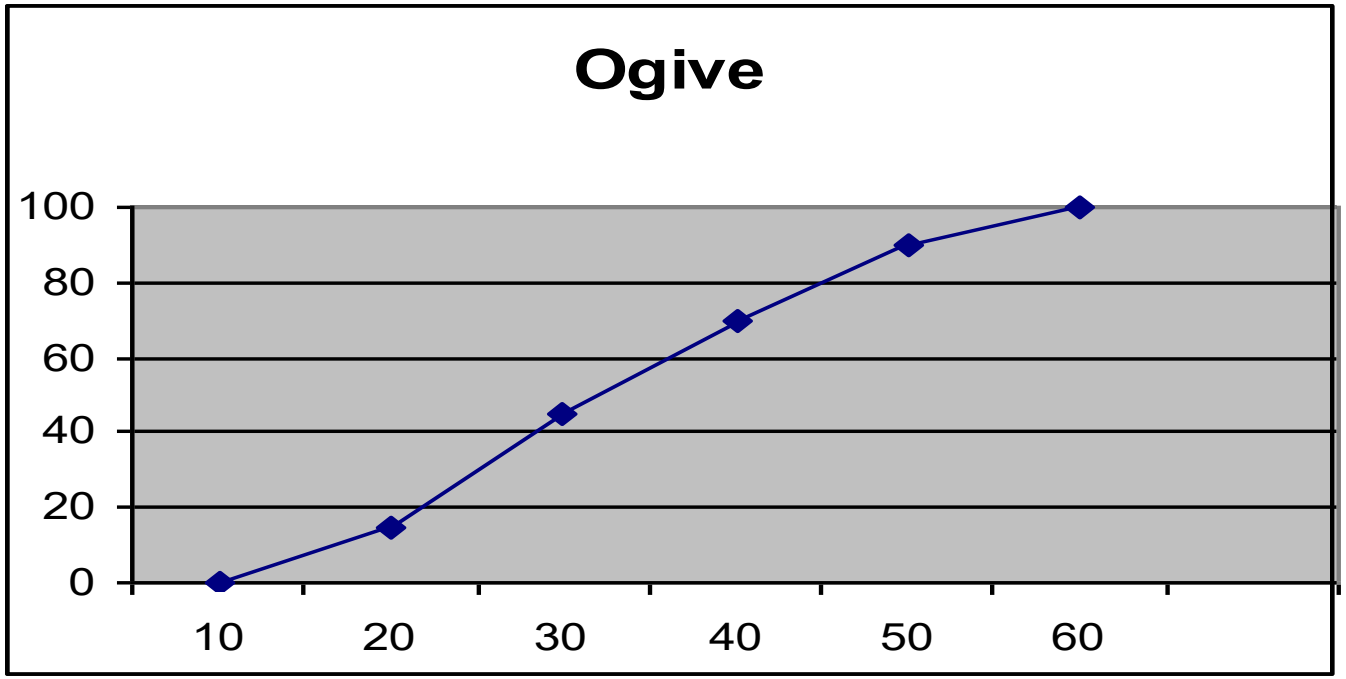
12, 13, 17, 21, 24, 24, 26, 27, 27, 30, 32, 35, 37, 38, 41, 43, 44, 46, 53, 58

Lower Limit	Cumulative Frequency	Cumulative % Frequency
10	0	0
20	3	15
30	9	45
40	14	70
50	18	90
60	20	100

Graphing Numerical Data: The Ogive (Cumulative % Polygon)

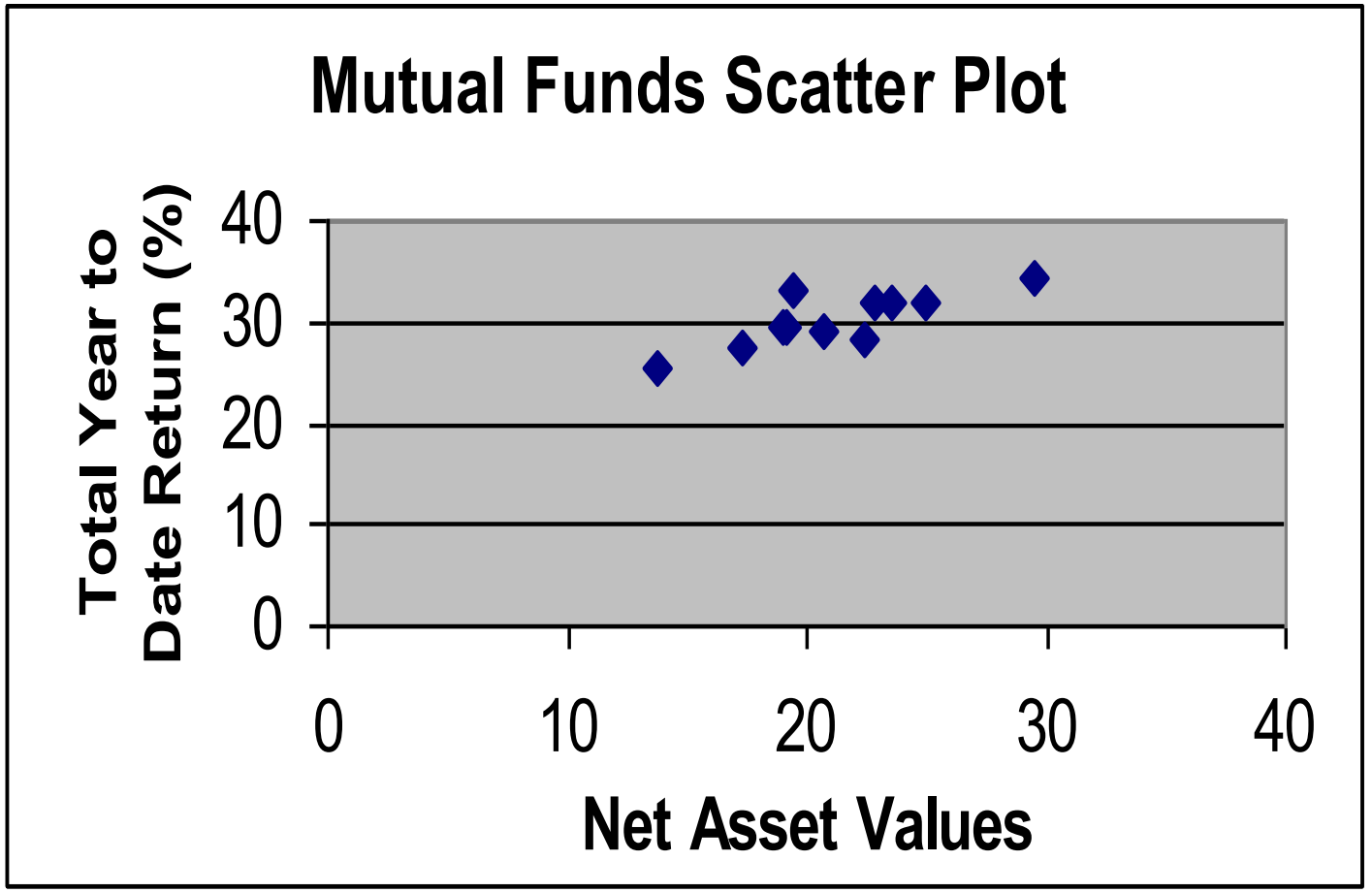
Data in Ordered Array :

12, 13, 17, 21, 24, 24, 26, 27, 27, 30, 32, 35, 37, 38, 41, 43, 44, 46, 53, 58

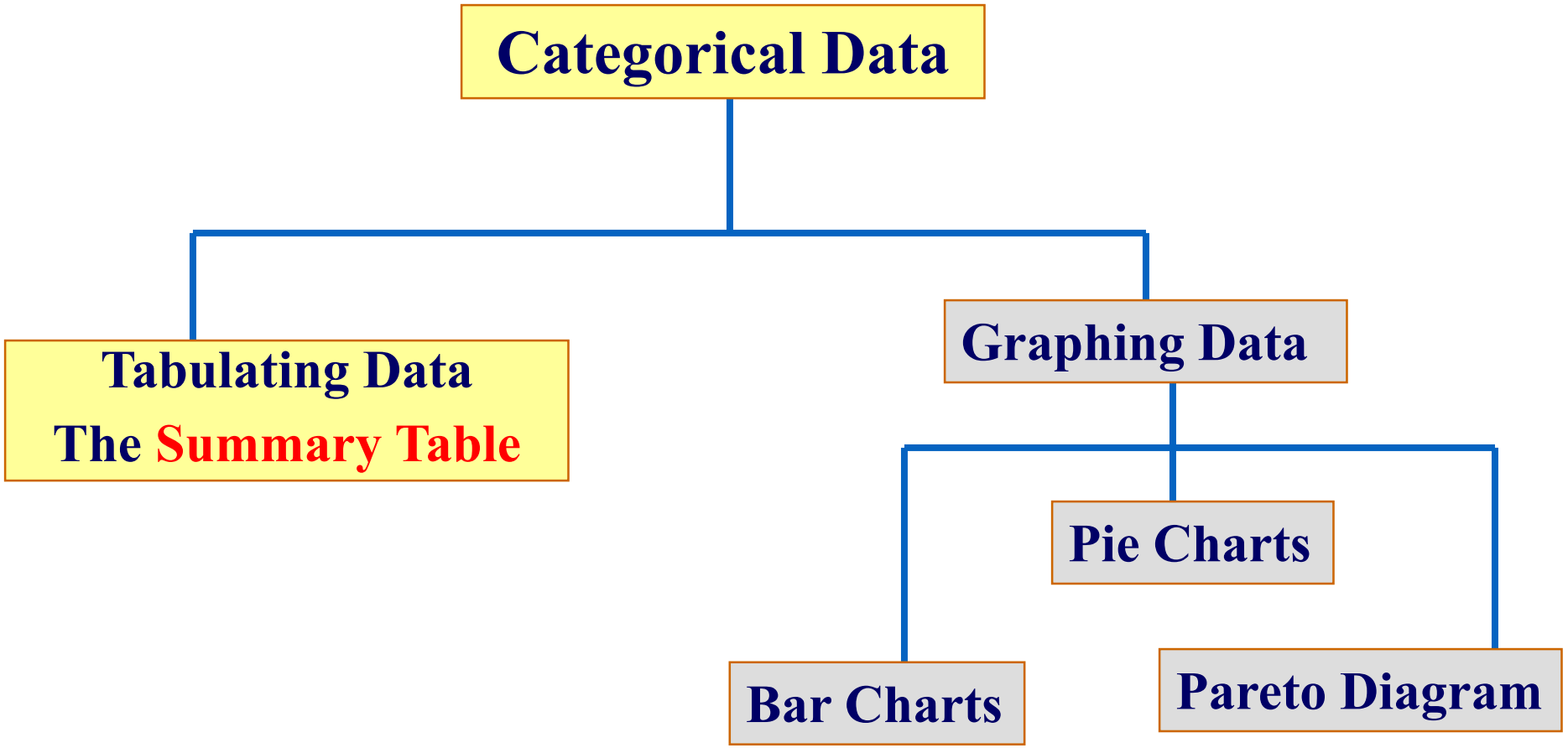


Class Boundaries (*Not Midpoints*)

Graphing Bivariate Numerical Data (Scatter Plot)



Tabulating and Graphing Univariate Categorical Data



Summary Table (for an Investor's Portfolio)

Investment Category	Amount (in thousands $)	Percentage
Stocks	46.5	42.27
Bonds	32	29.09
CD	15.5	14.09
Savings	16	14.55
Total	**110**	**100**

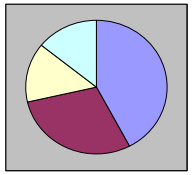
Variables are Categorical

Graphing Univariate Categorical Data

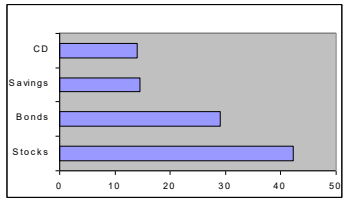
Categorical Data

Tabulating Data
The Summary Table

Graphing Data



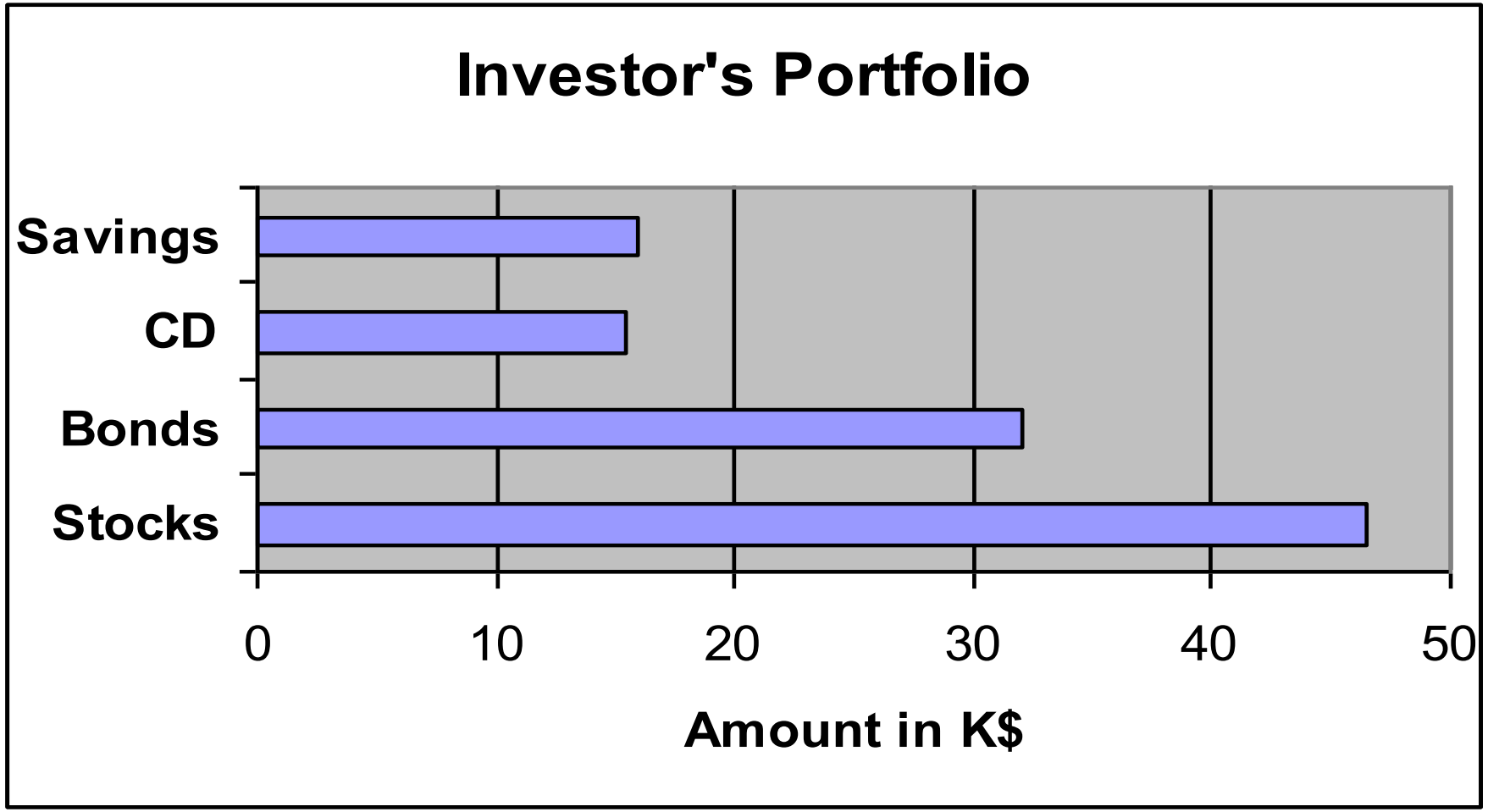
Pie Charts



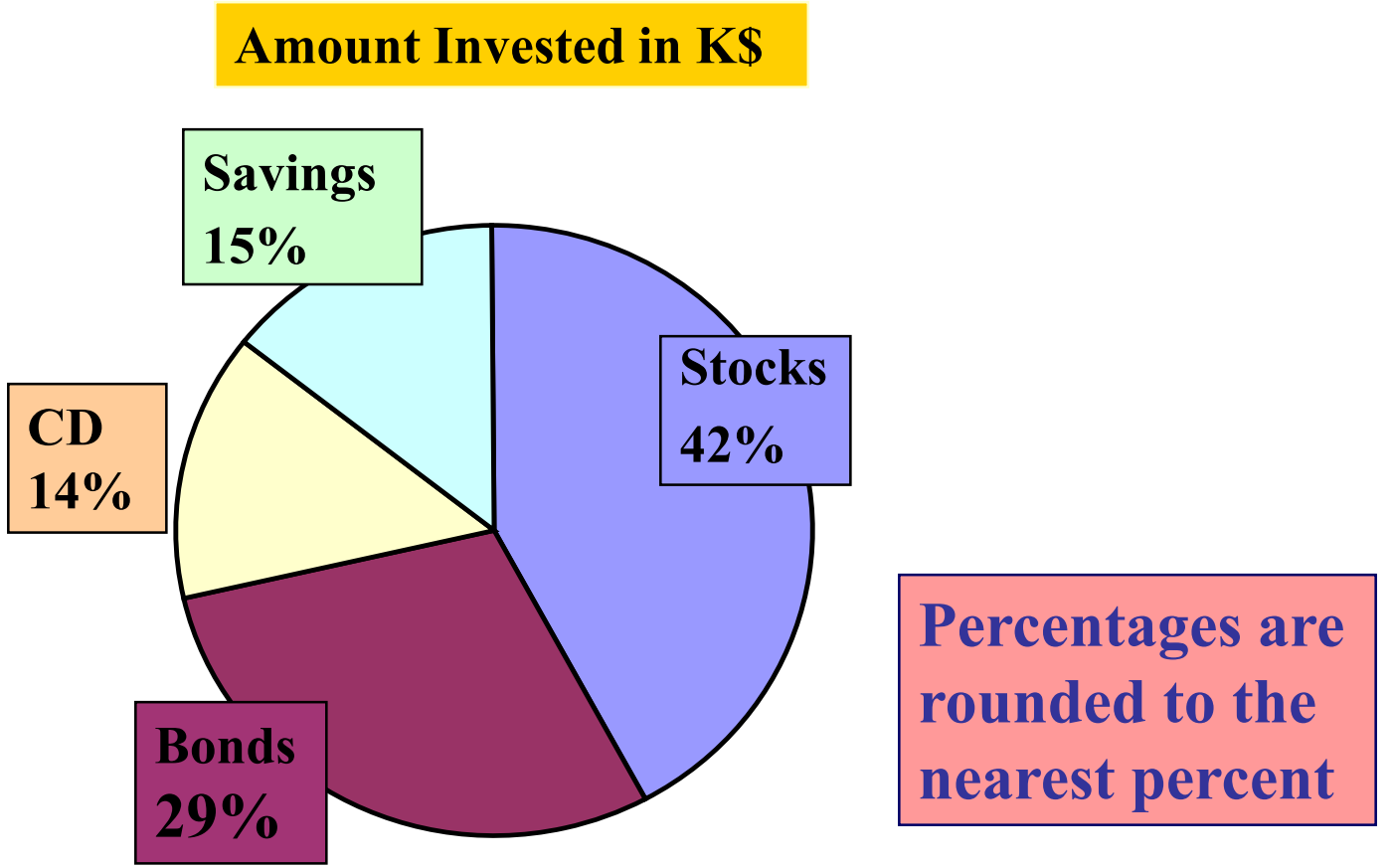
Bar Charts

Pareto Diagram

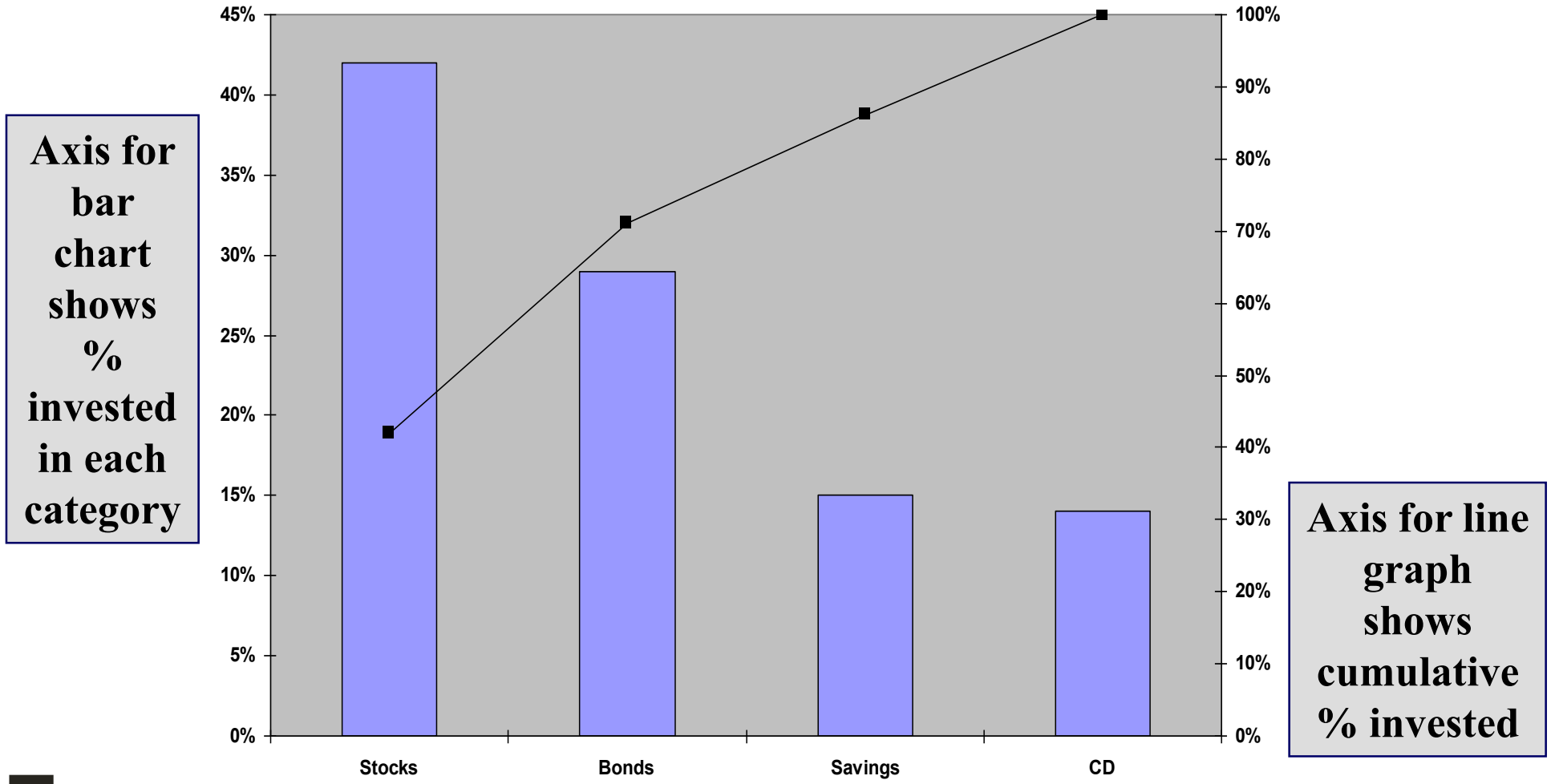
Bar Chart (for an Investor’s Portfolio)



Pie Chart (for an Investor's Portfolio)



Pareto Diagram



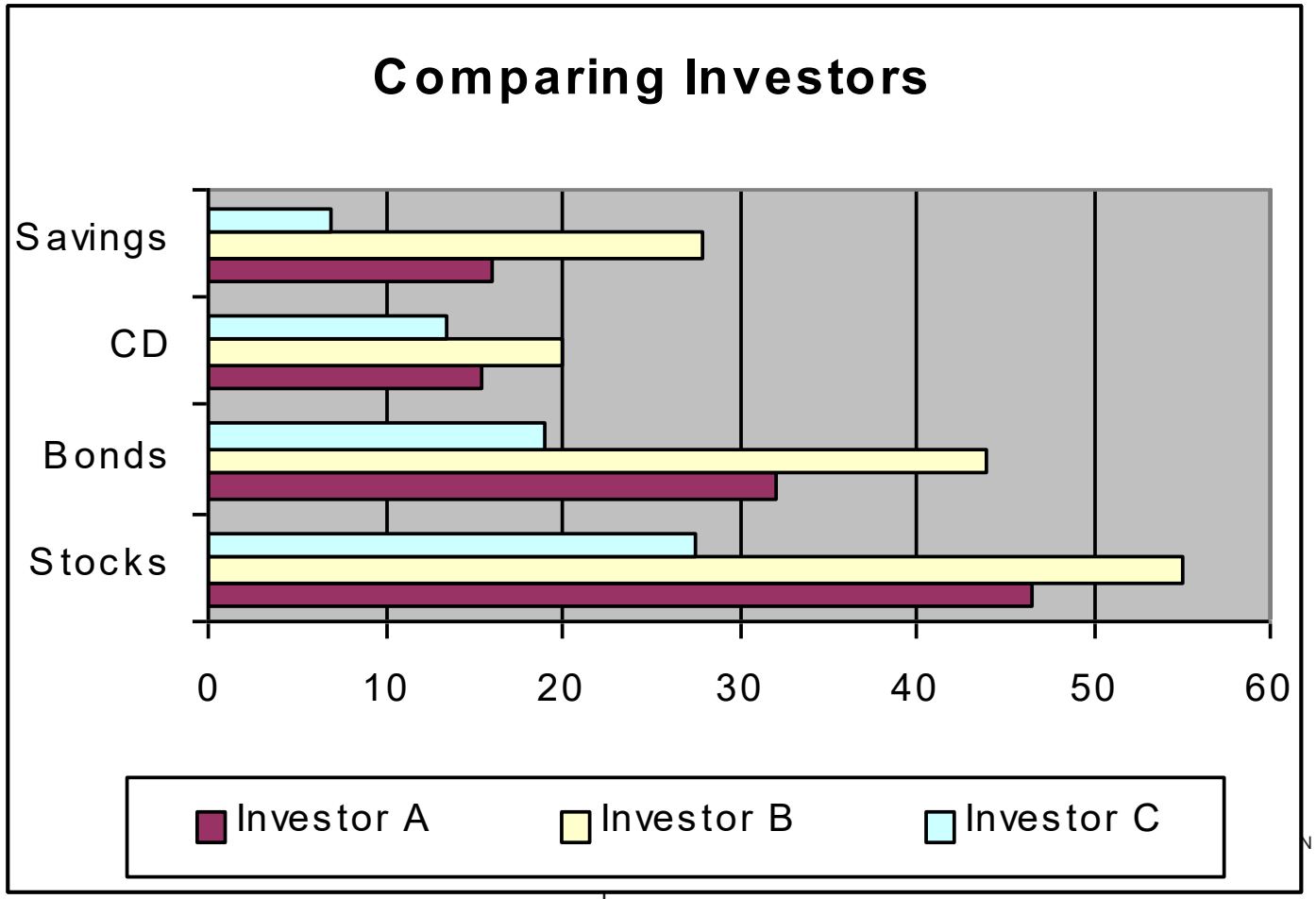
Tabulating and Graphing Bivariate Categorical Data

Contingency Tables: Investment in Thousands of Dollars

Investment Category	Investor A	Investor B	Investor C	Total
Stocks	46.5	55	27.5	**129**
Bonds	32	44	19	**95**
CD	15.5	20	13.5	**49**
Savings	16	28	7	**51**
Total	**110**	**147**	**67**	**324**

Tabulating and Graphing Bivariate Categorical Data

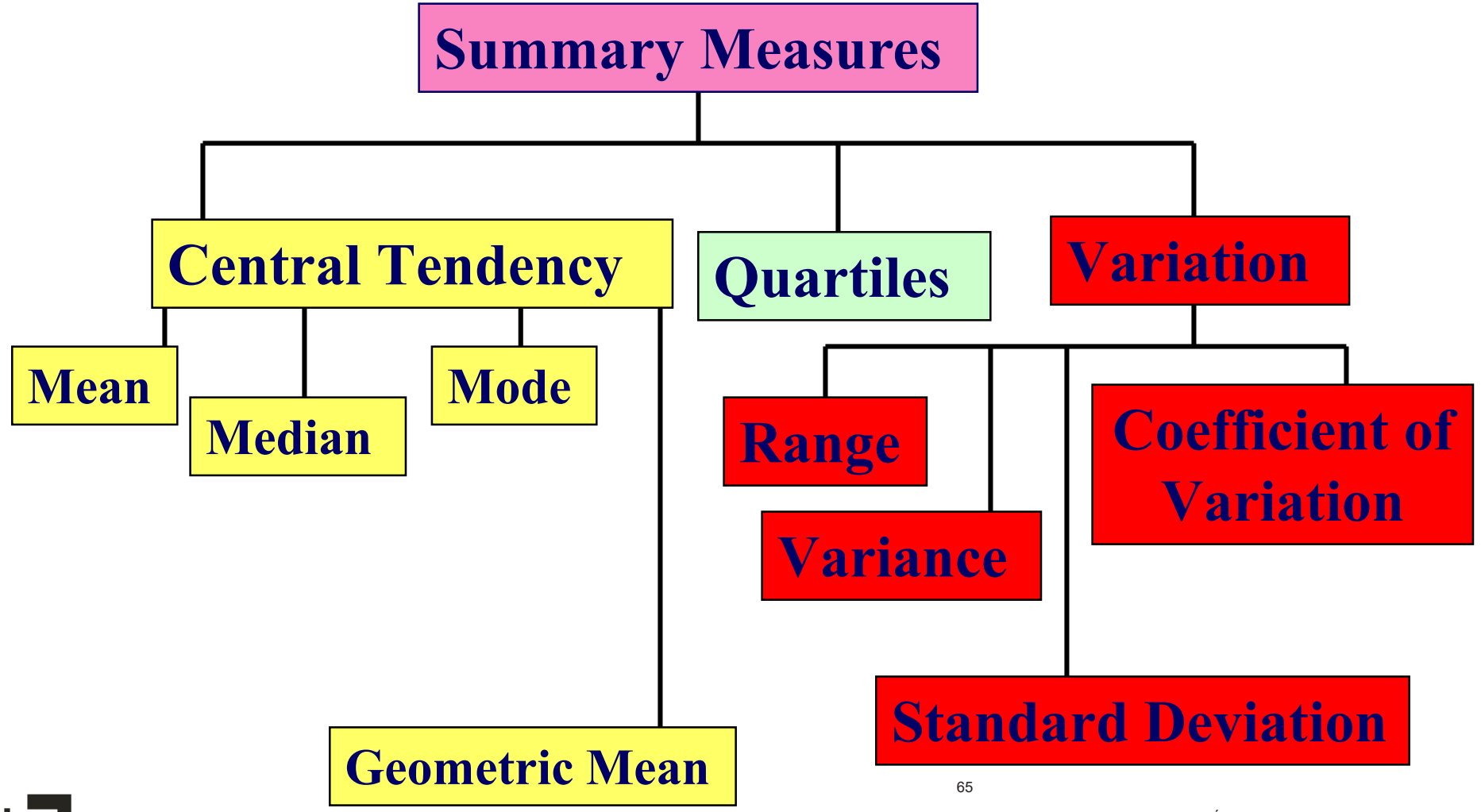
Side by Side Charts



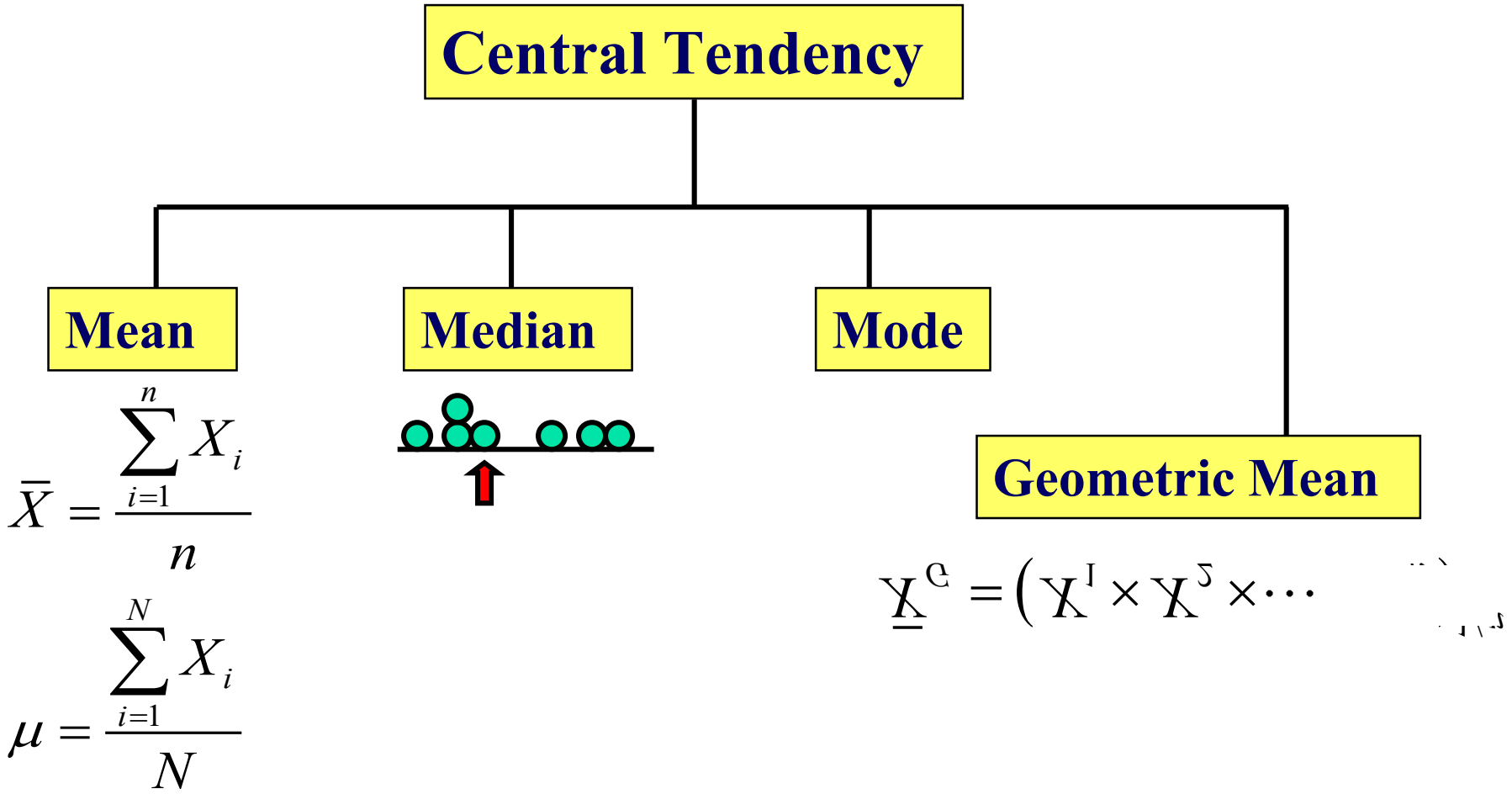
Part 3

Numerical Descriptive Measures

Summary Measures



Measures of Central Tendency



Mean (Arithmetic Mean)

Mean (Arithmetic Mean) of Data Values

- Sample mean

Sample Size

$$\bar{X} = \frac{\sum_{i=1}^{n} X_i}{n} = \frac{X_1 + X_2 + \cdots}{n}$$

- Population mean

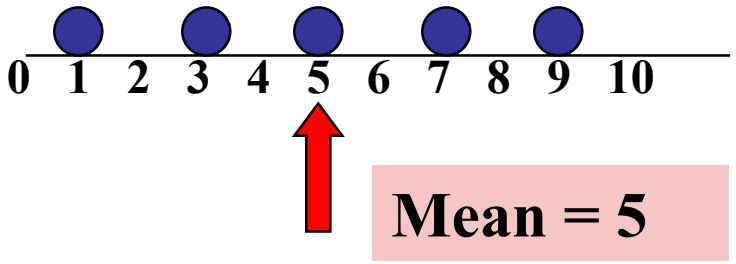
Population Size

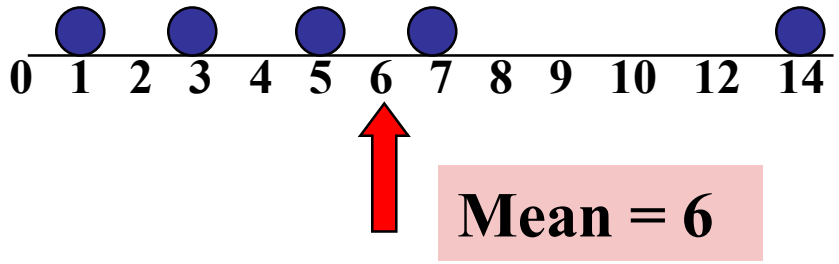
$$\mu = \frac{\sum_{i=1}^{N} X_i}{N} = \frac{X_1 + X_2 + \cdots}{N}$$

Mean (Arithmetic Mean) *(continued)*

The Most Common Measure of Central Tendency

Affected by Extreme Values (Outliers)





Mean (Arithmetic Mean) From a Frequency Distribution *(continued)*

Approximating the Arithmetic Mean

- Used when raw data are not available
- $\bar{X} = \dfrac{\sum_{j=1}^{c} m_j f_j}{n}$

n = sample size

c = number of classes in the frequency distribution

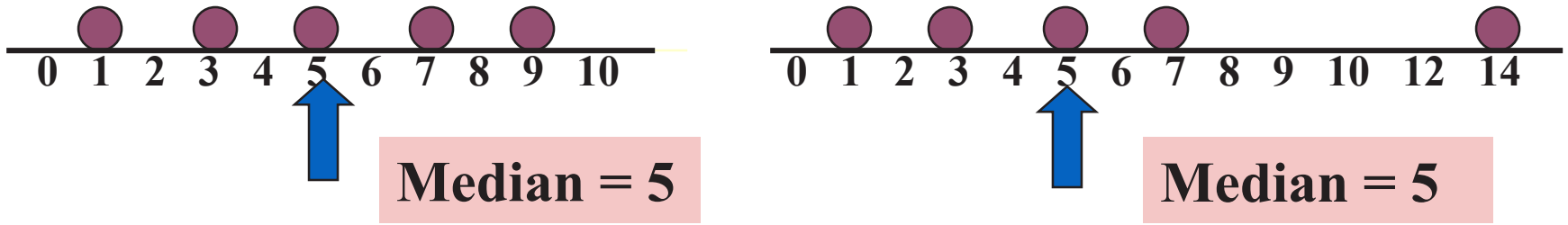
m_j = midpoint of the jth class

f_j = frequencies of the jth class

Median

Robust Measure of Central Tendency

Not Affected by Extreme Values



In an Ordered Array, the Median is the 'Middle' Number

- If n or N is odd, the median is the middle number
- If n or N is even, the median is the average of the 2 middle numbers

Mode

A Measure of Central Tendency

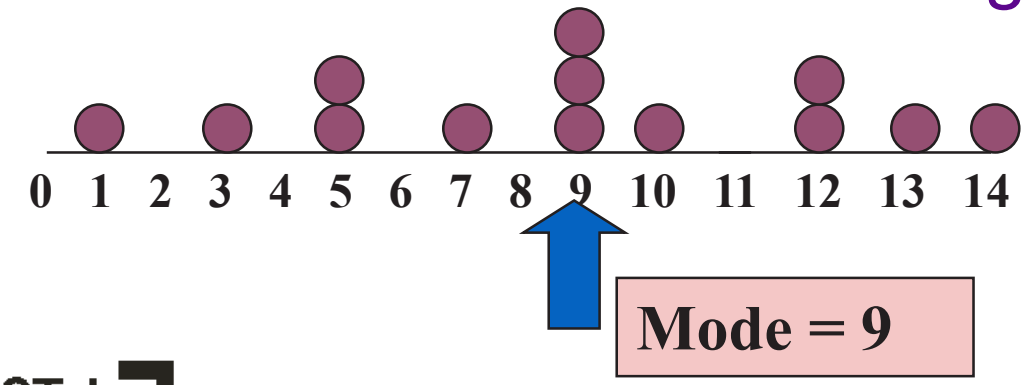
Value that Occurs Most Often

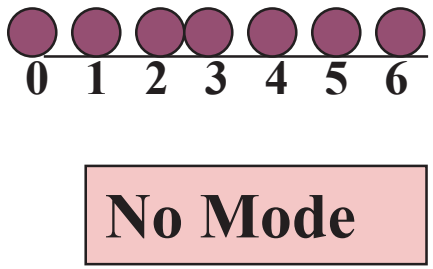
Not Affected by Extreme Values

There May Not Be a Mode

There May Be Several Modes

Used for Either Numerical or Categorical Data





Geometric Mean

Useful in the Measure of Rate of Change of a Variable Over Time

$$\bar{X}_G = \left(X_1 \times X_2 \times \cdots \right.$$

Geometric Mean Rate of Return

- Measures the status of an investment over time

$$\bar{R}_G = \left[\left(1 + R_1\right) \times \left(1 + R_2\right) \times \cdots \right]^{1/n} - 1$$

Example

An investment of $100,000 declined to $50,000 at the end of year one and rebounded back to $100,000 at end of year two:

$$R_1 = -0.5 \text{ (or } -50\%) \qquad R_2 = 1 \text{ (or } 100\% \text{)}$$

Average rate of return:

$$\bar{R} = \frac{(-0.5)+(1)}{2} = 0.25 \text{ (or } 25\%)$$

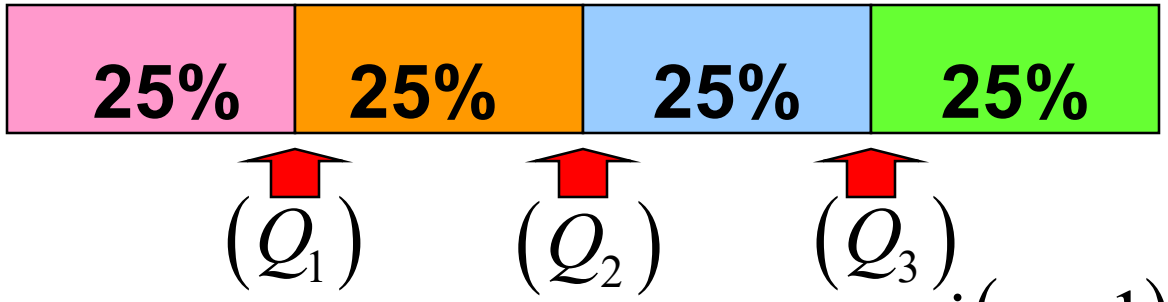
Geometric rate of return:

$$\bar{R}_G = \left[(1-0.5)\times(1+1)\right]^{1/2} - 1$$

$$= \left[(0.5)\times(2)\right]^{1/2} - 1 = 1^{1/2} - 1 = 0 \text{ (or } 0\%)$$

Quartiles

Split Ordered Data into 4 Quarters



Position of i-th Quartile

$$(Q_i) = \frac{i(n+1)}{4}$$

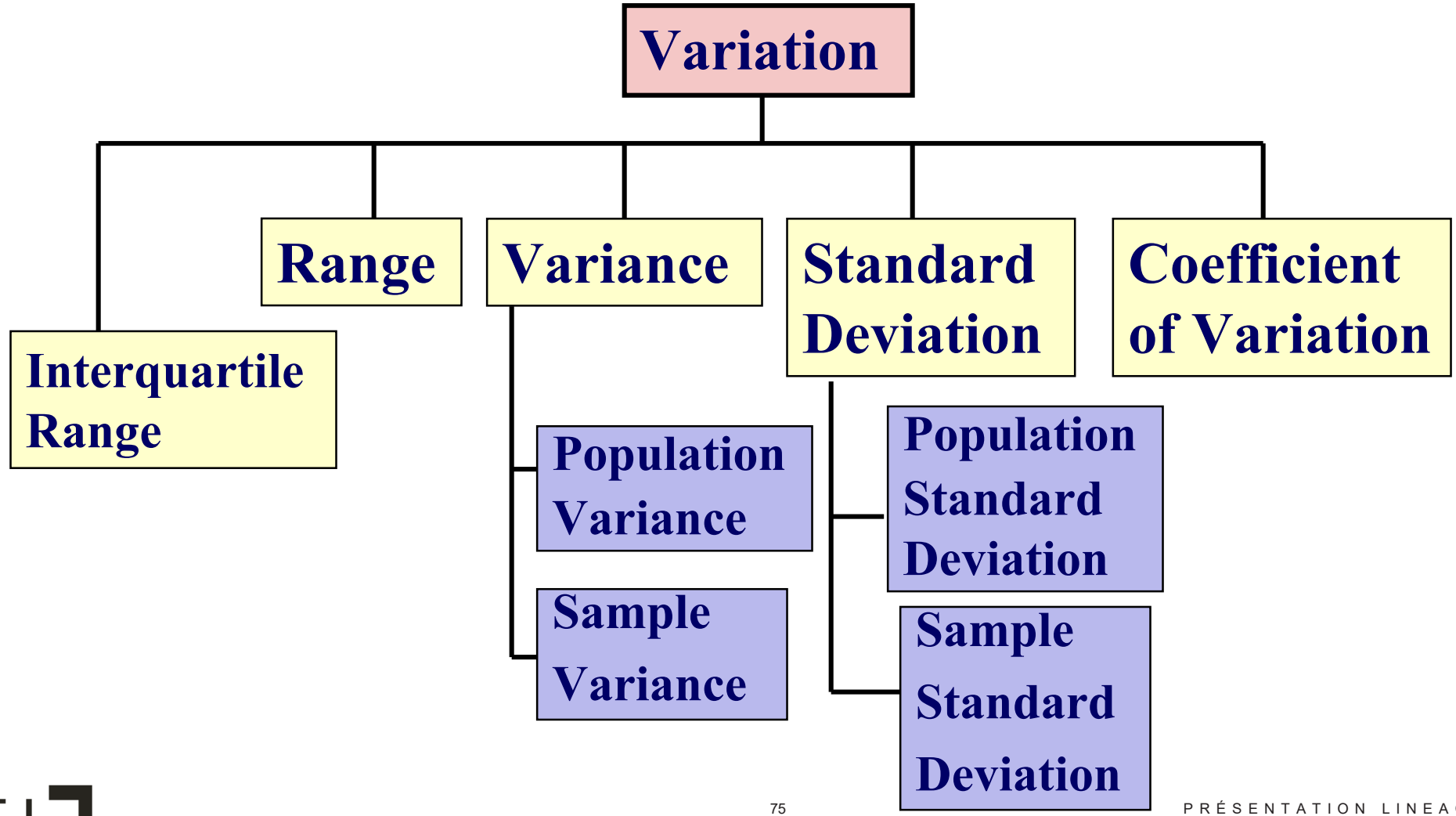
Data in Ordered Array: 11 12 13 16 16 17 18 21 22

$$\text{Position of } Q_1 = \frac{1(9+1)}{4} = 2.5 \qquad Q_1 = \frac{(12+13)}{2} = 12.5$$

Q_1 and Q_3 are Measures of Noncentral Location

Q_2 = Median, a Measure of Central Tendency

Measures of Variation



Range

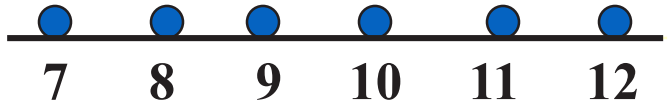
Measure of Variation

Difference between the Largest and the Smallest Observations:

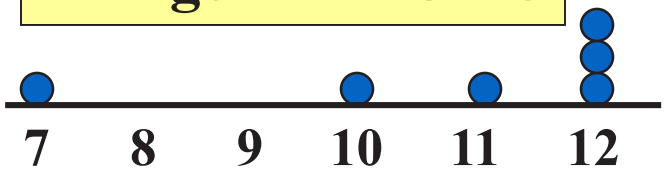
$$\text{Range} = X_{\text{Largest}} - X_{\text{Smallest}}$$

Ignores How Data are Distributed

Range = 12 - 7 = 5



Range = 12 - 7 = 5



Interquartile Range

Measure of Variation

Also Known as Midspread

- Spread in the middle 50%

Difference between the First and Third Quartiles

Data in Ordered Array: 11 12 13 16 16 17 17 18 21

$$\text{Interquartile Range} = Q_3 - Q_1 = 17.5 - 12.5 = 5$$

Not Affected by Extreme Values

Variance

Important Measure of Variation

Shows Variation about the Mean

- Sample Variance:

$$S^2 = \frac{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2}{n-1}$$

- Population Variance:

$$\sigma^2 = \frac{\sum_{i=1}^{N}\left(X_i - \mu\right)^2}{N}$$

Standard Deviation

- Most Important Measure of Variation
- Shows Variation about the Mean
- Has the Same Units as the Original Data
  - Sample Standard Deviation:

$$S = \sqrt{\frac{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2}{n-1}}$$

  - Population Standard Deviation

$$\sigma = \sqrt{\frac{\sum_{i=1}^{N}\left(X_i - \mu\right)^2}{N}}$$

Standard Deviation From a Frequency Distribution

Approximating the Standard Deviation

- Used when the raw data are not available and the only source of data is a frequency distribution
- $$S = \sqrt{\frac{\sum_{j=1}^{c}\left(m_j - \bar{X}\right)^2 f_j}{n-1}}$$

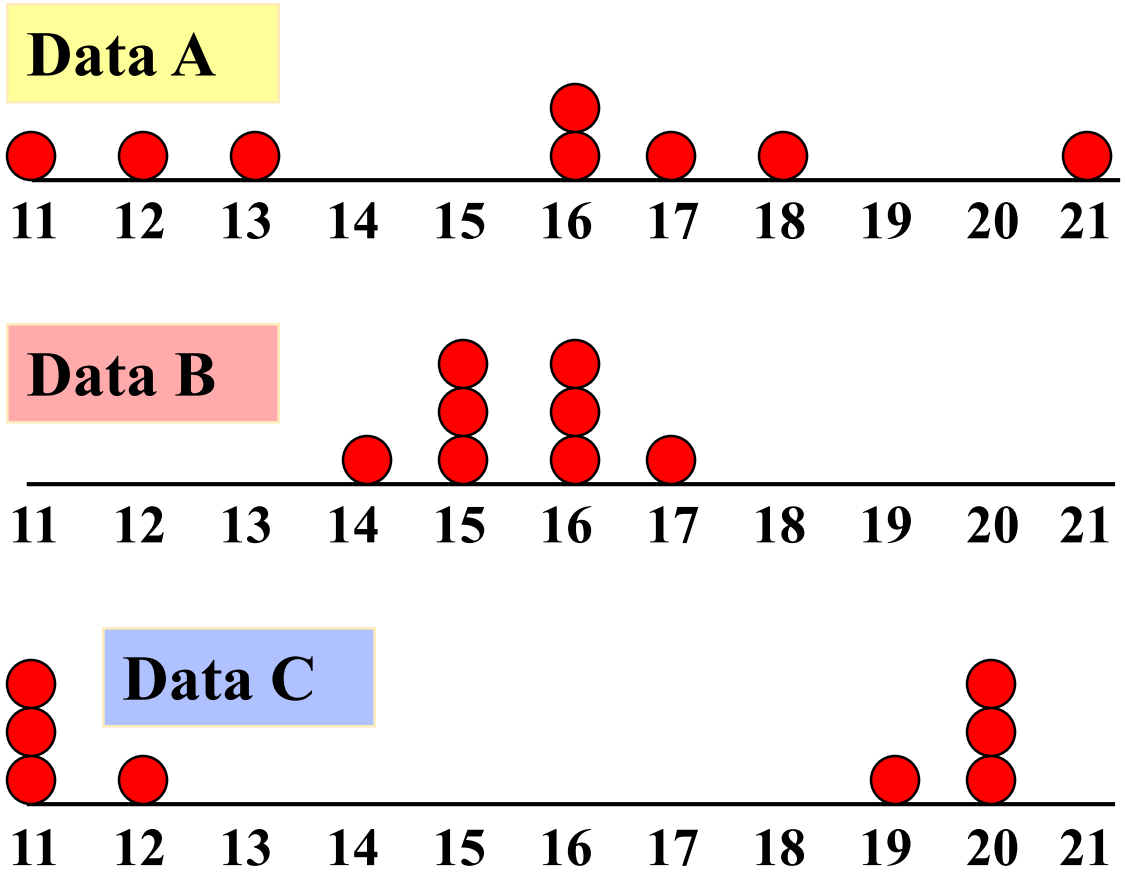
n = sample size

c = number of classes in the frequency distribution

m_j = midpoint of the jth class

f_j = frequencies of the jth class

Comparing Standard Deviations



Mean = 15.5
s = 3.338

Mean = 15.5
s = .9258

Mean = 15.5
s = 4.57

Coefficient of Variation

- Measure of Relative Variation
- Always in Percentage (%)
- Shows Variation Relative to the Mean
- Used to Compare Two or More Sets of Data Measured in Different Units
- $$CV = \left(\frac{S}{\overline{X}} \right) 100\%$$
- Sensitive to Outliers

Comparing Coefficient of Variation

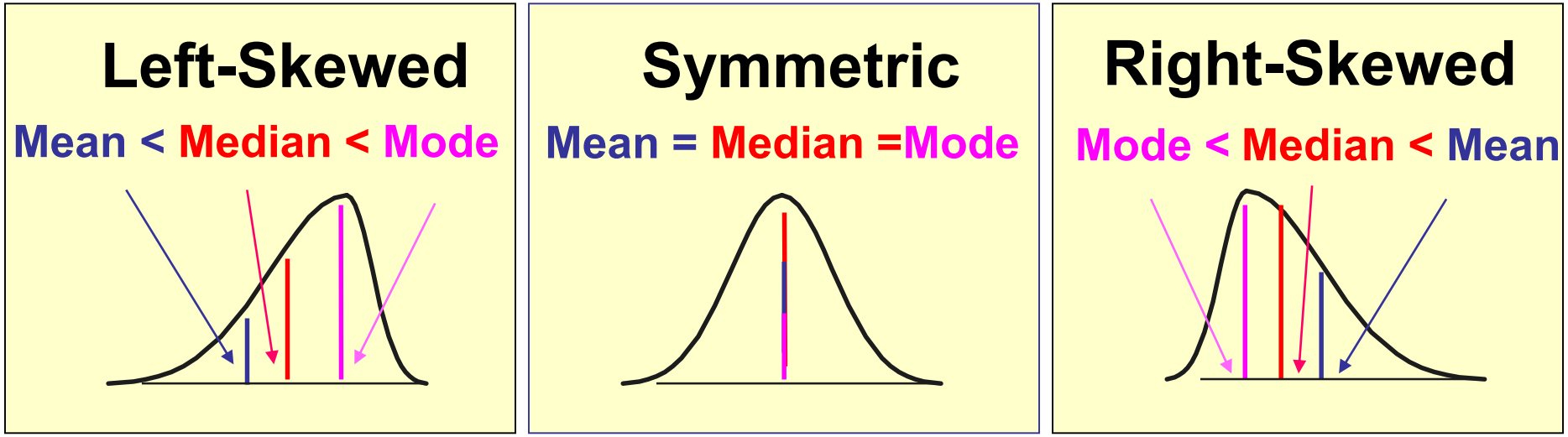
- Stock A:
  - Average price last year = $50
  - Standard deviation = $2
- Stock B:
  - Average price last year = $100
  - Standard deviation = $5
- Coefficient of Variation:
  - Stock A: $CV = \left(\frac{S}{\bar{X}}\right)100\% = \left(\frac{\$2}{\$50}\right)100\% = 4\%$
  - Stock B: $CV = \left(\frac{S}{\bar{X}}\right)100\% = \left(\frac{\$5}{\$100}\right)100\% = 5\%$

Shape of a Distribution

Describe How Data are Distributed

Measures of Shape

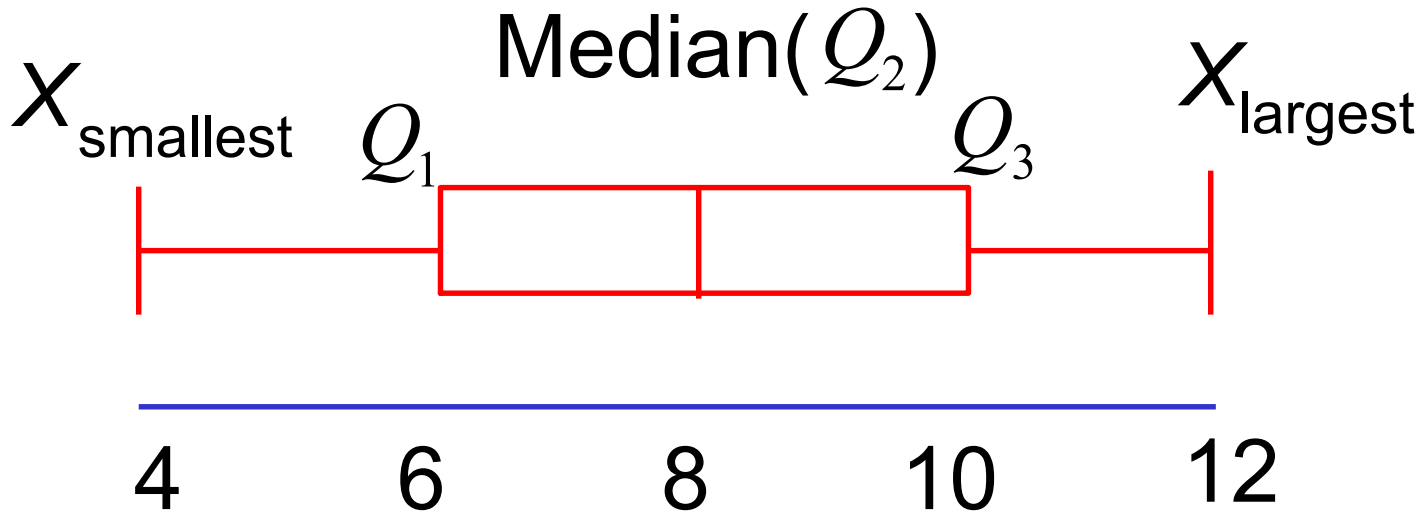
- Symmetric or skewed



Exploratory Data Analysis

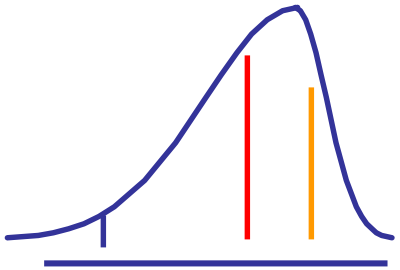
Box-and-Whisker Plot

- Graphical display of data using 5-number summary

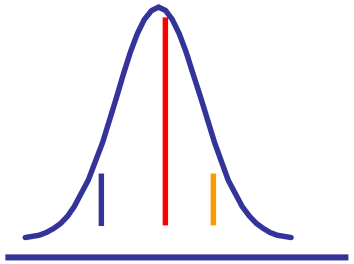


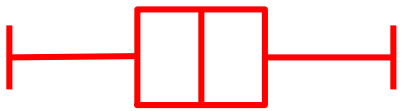
Distribution Shape & Box-and-Whisker Plot

Left-Skewed

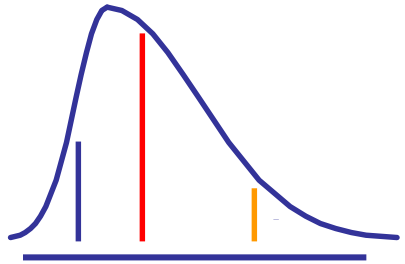


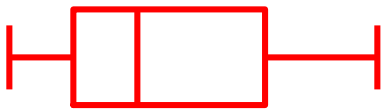
Symmetric





Right-Skewed





The Empirical Rule

For Data Sets That Are Approximately Bell-shaped:

- Roughly 68% of the Observations Fall Within 1 Standard Deviation Around the Mean
- Roughly 95% of the Observations Fall Within 2 Standard Deviations Around the Mean
- Roughly 99.7% of the Observations Fall Within 3 Standard Deviations Around the Mean

The Bienayme-Chebyshev Rule

- The Percentage of Observations Contained Within Distances of k Standard Deviations Around the Mean Must Be at Least
  - Applies regardless of the shape of the data set $\left(1-1/k^2\right)100\%$
  - At least 75% of the observations must be contained within distances of 2 standard deviations around the mean
  - At least 88.89% of the observations must be contained within distances of 3 standard deviations around the mean
  - At least 93.75% of the observations must be contained within distances of 4 standard deviations around the mean

Coefficient of Correlation

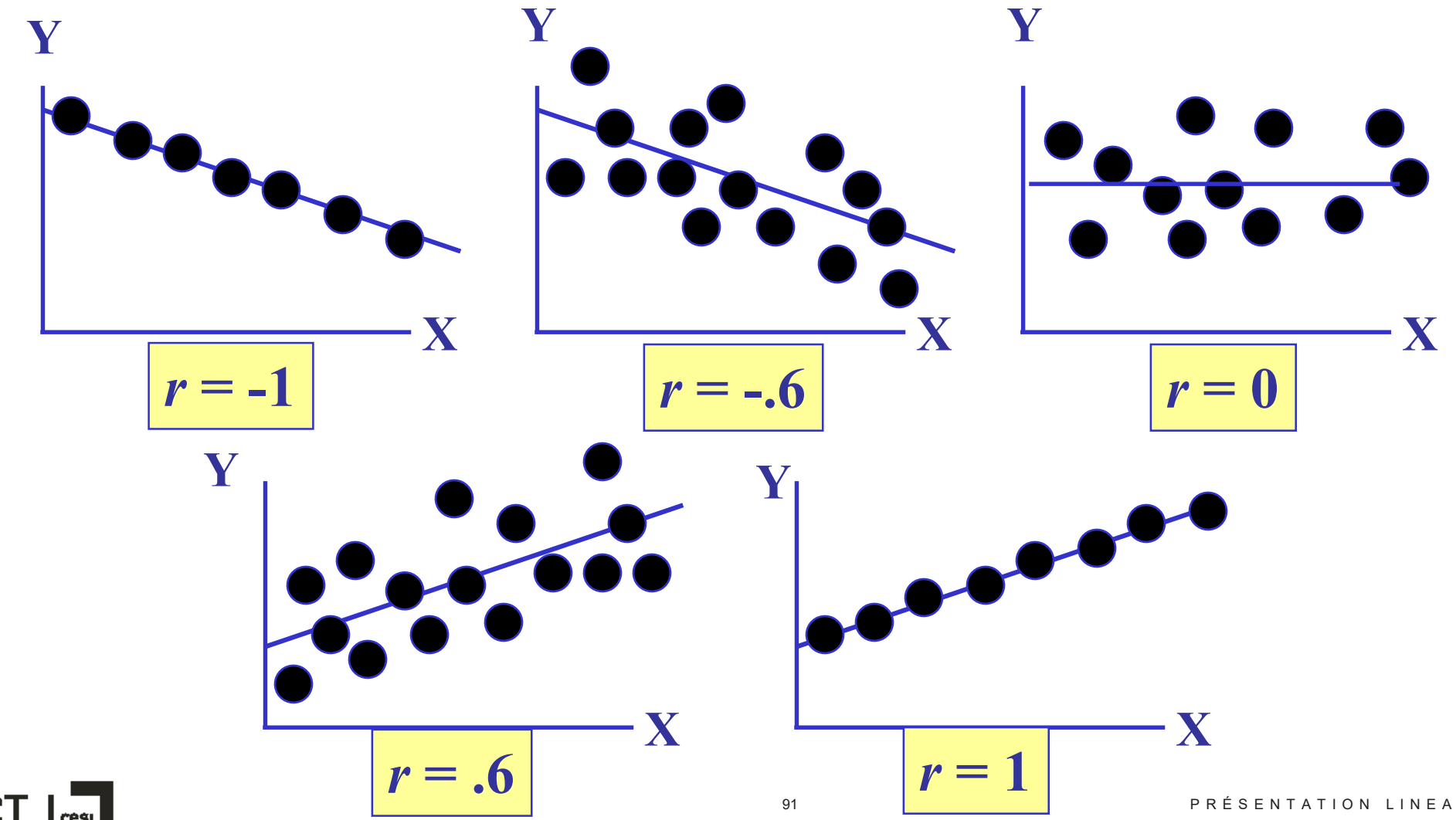
- Measures the Strength of the Linear Relationship between 2 Quantitative Variables

$$r = \frac{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)\left(Y_i - \bar{Y}\right)}{\sqrt{\sum_{i=1}^{n}\left(X_i - \bar{X}\right)^2}\sqrt{\sum_{i=1}^{n}\left(Y_i - \bar{Y}\right)^2}}$$

Features of Correlation Coefficient

- Unit Free
- Ranges between –1 and 1
- The Closer to –1, the Stronger the Negative Linear Relationship
- The Closer to 1, the Stronger the Positive Linear Relationship
- The Closer to 0, the Weaker Any Linear Relationship

Scatter Plots of Data with Various Correlation Coefficients



Pitfalls in Numerical Descriptive Measures and Ethical Issues

Data Analysis is Objective

- Should report the summary measures that best meet the assumptions about the data set

Data Interpretation is Subjective

- Should be done in a fair, neutral and clear manner

Ethical Issues

- Should document both good and bad results
- Presentation should be fair, objective and neutral
- Should not use inappropriate summary measures to distort the facts

Data Transformation: Normalization

- Particularly useful for classification (NNs, distance measurements, nn classification, etc)
  - min-max normalization

$$v' = \frac{v - Min_A}{Max_A - Min_A}$$

  - z-score normalization

$$v' = \frac{v - mean_A}{stand\_dev_A}$$

  - normalization by decimal scaling

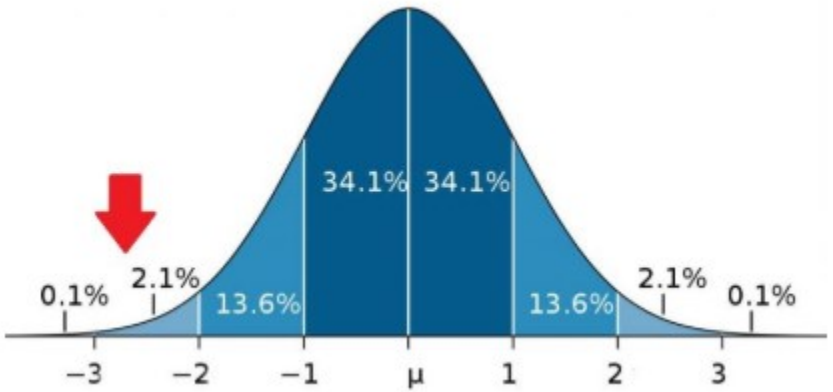
$$v' = \frac{v}{10^j}$$ Where j is the smallest integer such that $\text{Max}(|v'|)<1$

Example 1

The minimum and maximum values for the price of the house are respectively 125,000 € and 925,000 €. We need to normalize that price range in between (0,1). We can use min-max normalization to transform any value between them (say, 300,000).

Example 2: Z-Score

The normative reference sample evaluated with the Wechsler Adult Intelligence Scale (I.Q) has a distribution with a mean (X) 100 and a standard deviation (σ) of 15. The threshold of intellectual slowness is defined by a score below 70. How can this point be expressed as a Z-score?



LINEACT cesi

Part 4

Application in Python (Pandas)

DataFrame functions

mean()

- Mean(axis=0, skipna=True)

sum()

cumsum()

describe(): return summary statistics of each column

- for numeric data: mean, std, max, min, 25%, 50%, 75%, etc.
- For non-numeric data: count, uniq, most-frequent item, etc.

corr(): correlation between two Series, or between columns of a DataFrame

corr_with(): correlation between columns of DataFram and a series or between the columns of another DataFrame

Handling missing data

Filtering out missing values

```
In [1204]: data.notnull()
Out[1204]:
0 True
1 False
2 True
3 False
4 True
dtype: bool

In [1205]: data[data.notnull()]
Out[1205]:
0 1.0
2 2.5
4 6.0
dtype: float64
```

LINEACT cesi

```
In [1198]: from numpy import nan as NaN

In [1199]: data = Series([1, NaN, 2.5, NaN, 6])

In [1200]: data.dropna()
Out[1200]:
0 1.0
2 2.5
4 6.0
dtype: float64

In [1201]: data
Out[1201]:
0 1.0
1 NaN
2 2.5
3 NaN
4 6.0
dtype: float64
```

Handling missing data - 2

In [1206]: data = DataFrame([[1, 2, 3], [1, NaN, NaN], [NaN, NaN, NaN], [NaN, 4, 5]])

In [1207]: data
Out[1207]:
0 1 2
0 1.0 2.0 3.0
1 1.0 NaN NaN
2 NaN NaN NaN
3 NaN 4.0 5.0

In [1208]: data.dropna()
Out[1208]:
0 1 2
0 1.0 2.0 3.0

In [1209]: data.dropna(how='all')
Out[1209]:
0 1 2
0 1.0 2.0 3.0
1 1.0 NaN NaN
3 NaN 4.0 5.0

In [1210]: data.dropna(axis=1, how='all')
Out[1210]:
0 1 2
0 1.0 2.0 3.0
1 1.0 NaN NaN
2 NaN NaN NaN
3 NaN 4.0 5.0

In [1215]: data[4]=NaN

In [1216]: data
Out[1216]:
0 1 2 4
0 1.0 2.0 3.0 NaN
1 1.0 NaN NaN NaN
2 NaN NaN NaN NaN
3 NaN 4.0 5.0 NaN

In [1217]: data.dropna(axis=1, how='all')
Out[1217]:
0 1 2
0 1.0 2.0 3.0
1 1.0 NaN NaN
2 NaN NaN NaN
3 NaN 4.0 5.0

Filling in missing data

In [**1218**]: data
Out[**1218**]:
0 1 2 4
0 1.0 2.0 3.0 NaN
1 1.0 NaN NaN NaN
2 NaN NaN NaN NaN
3 NaN 4.0 5.0 NaN

In [**1219**]: data.fillna(0)
Out[**1219**]:
0 1 2 4
0 1.0 2.0 3.0 0.0
1 1.0 0.0 0.0 0.0
2 0.0 0.0 0.0 0.0
3 0.0 4.0 5.0 0.0

Modify the dataframe instead of retunring a new object (default)

In [**1220**]: data.fillna(0, inplace=True)

In [**1221**]: data
Out[**1221**]:
0 1 2 4
0 1.0 2.0 3.0 0.0
1 1.0 0.0 0.0 0.0
2 0.0 0.0 0.0 0.0
3 0.0 4.0 5.0 0.0

In [**1227**]: data
Out[**1227**]:
0 1 2
0 NaN 9 9.0
1 NaN 7 2.0
2 4.0 8 9.0
3 3.0 4 NaN

replace nan with column mean

In [**1228**]:
data.fillna(data.mean(skipna=True))
Out[**1228**]:
0 1 2
0 3.5 9 9.000000
1 3.5 7 2.000000
2 4.0 8 9.000000
3 3.0 4 6.666667

LINEACT cesi

Data transformation and normalization

Use boxplot to take a quick book

Transform data to obtain a certain distribution

- e.g. from lognormal to normal
- Normalize data so different columns became comparable / compatible

Typical normalization approach:

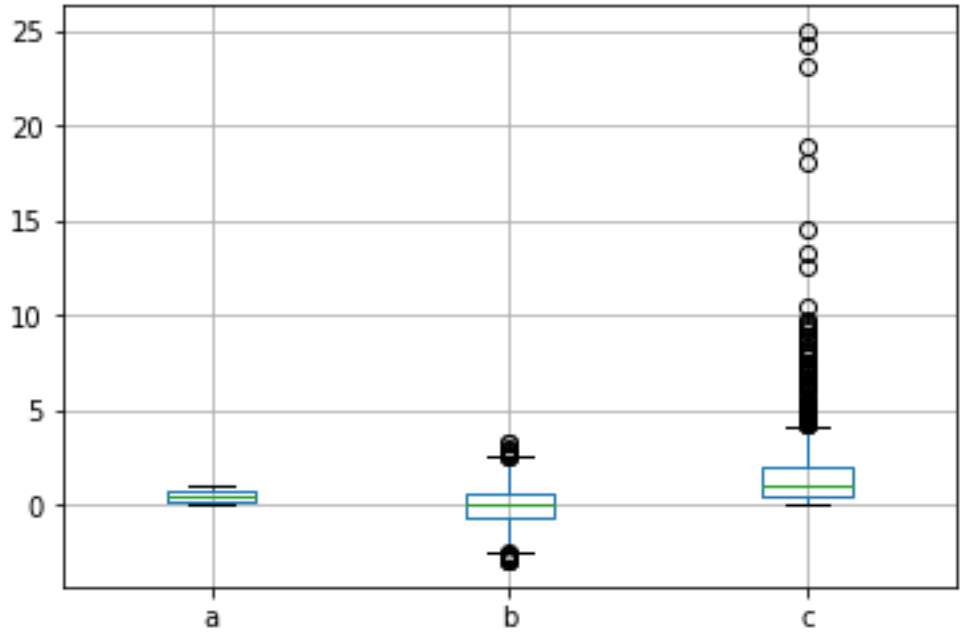
- Z-score transformation
- Scale to between 0 and 1
- Trimmed mean normalization
- Vector length transformation
- Quantilenorm

Boxplot example

```
In [1867]: df=DataFrame({'a':
np.random.rand(1000),       'b':
np.random.randn(1000, ),
                'c': np.random.lognormal(size=(1000,))})
```

```
In [1868]: df.head()
Out[1868]:
a b c
0 0.356627 1.406655 3.288161
1 0.472792 -1.247858 2.499727
2 0.467848 0.406503 2.215045
3 0.341257 1.457440 0.390666
4 0.236013 0.026771 1.295106
```

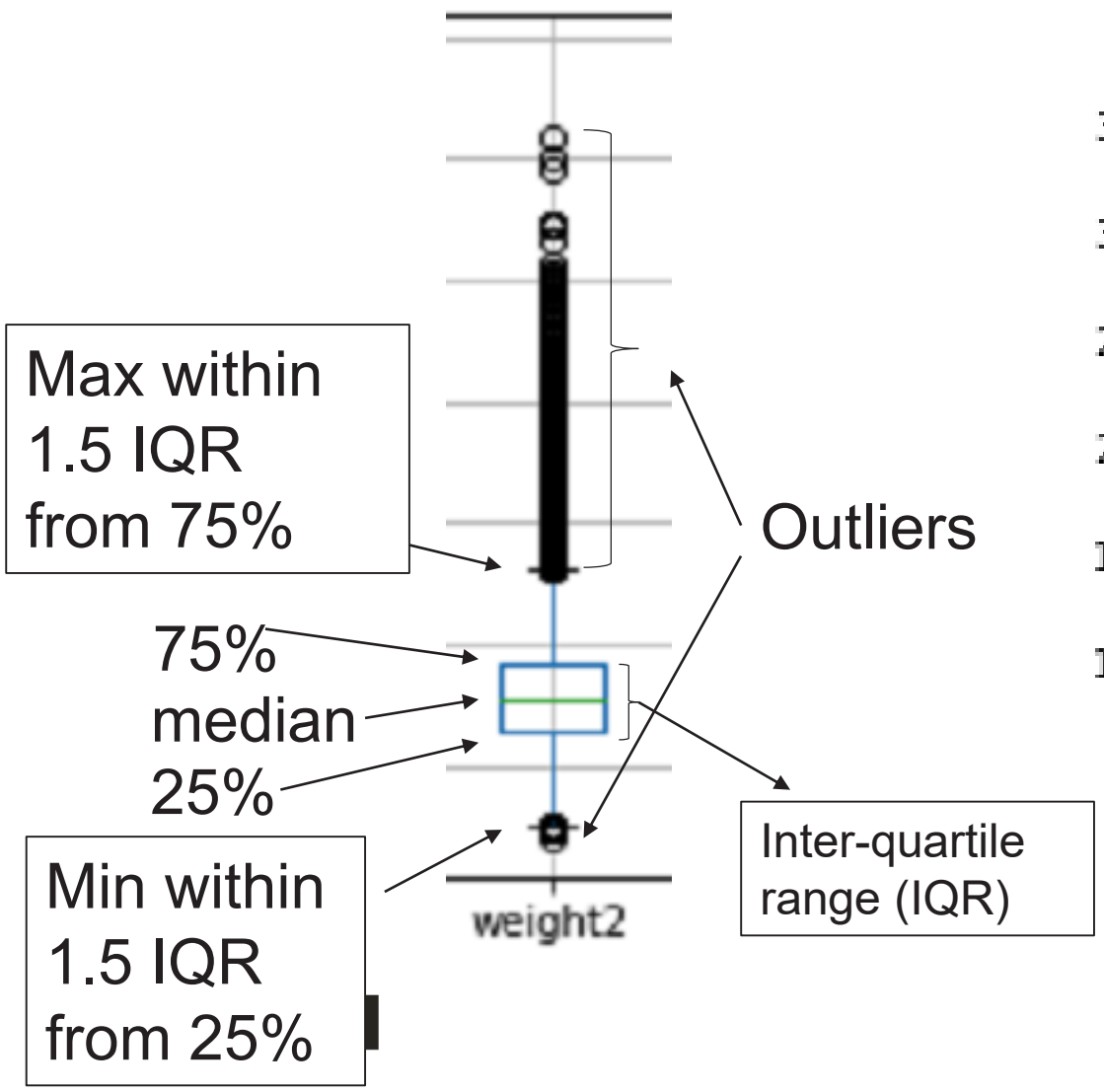
```
In [1869]: df.boxplot()
Out[1869]:
<matplotlib.axes._subplots.AxesSubplot at
```

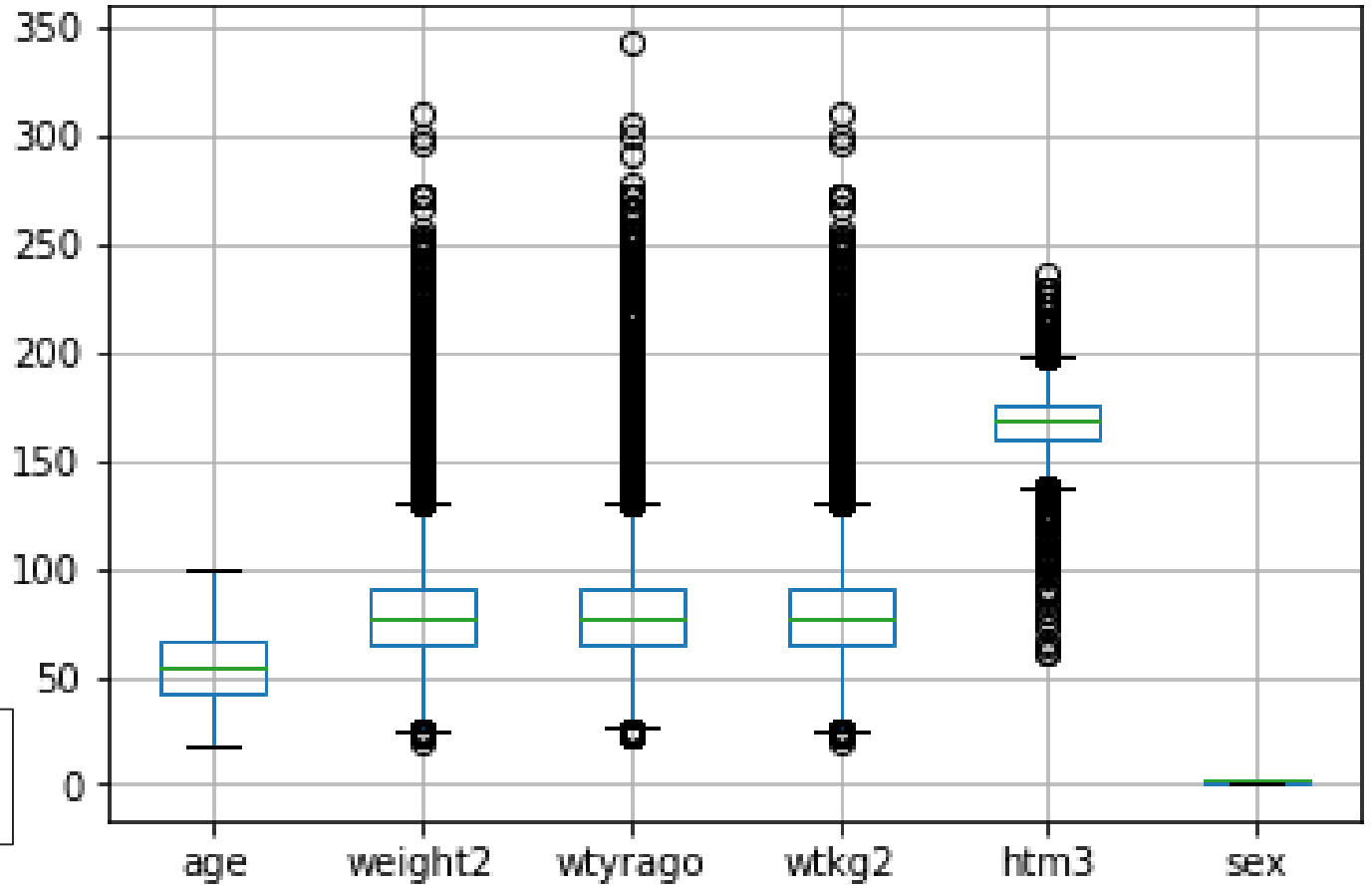


Boxplot example 2

```
In [1876]: df2 = pd.read_csv('brfss.csv', index_col=0)

In [1877]: df2.boxplot()
Out[1877]: <matplotlib.axes._subplots.AxesSubplot at
0x4ebcf588>
```



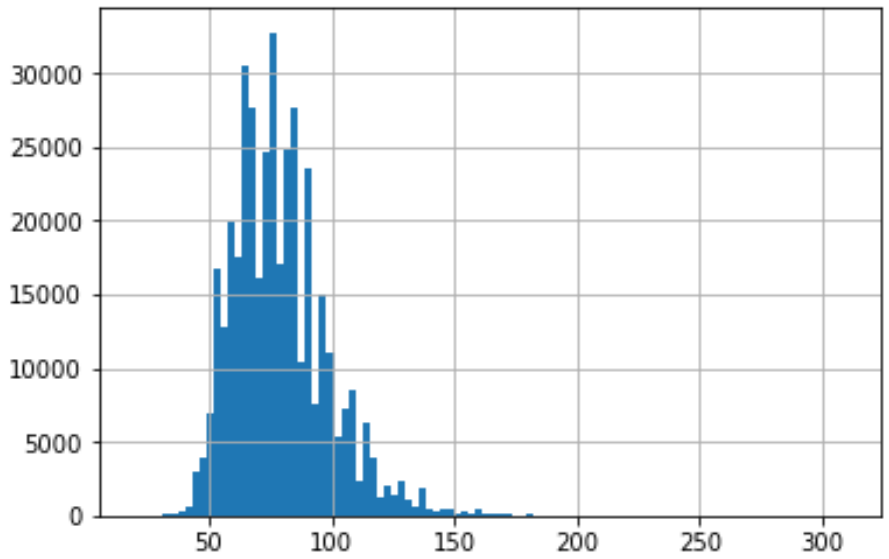


Other useful pandas plotting functions

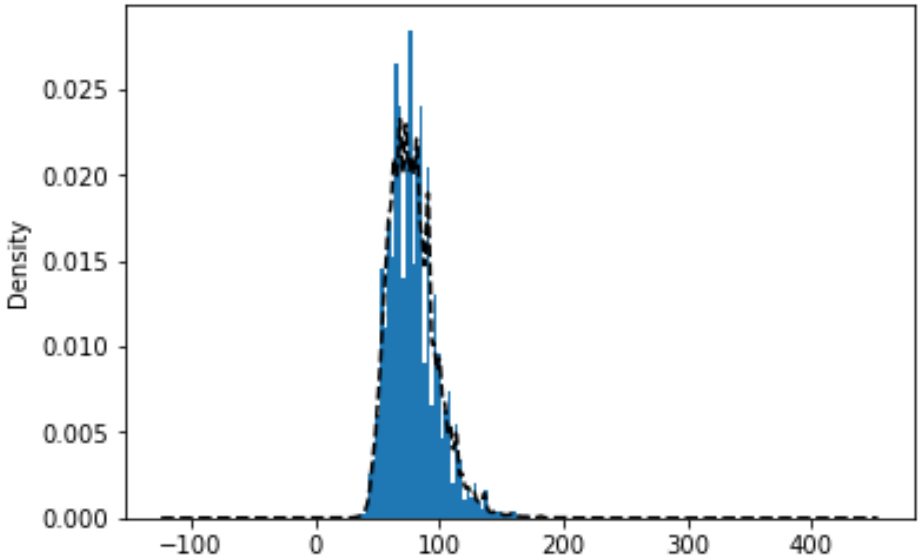
hist, plot, scatter, etc.

Use kernel density estimate to approximate the distribution with a mixture of normal distributions

In [1891]: df2['weight2'].hist(bins=100)
Out[1891]: <matplotlib.axes._subplots.AxesSubplot at 0x52197fd0>

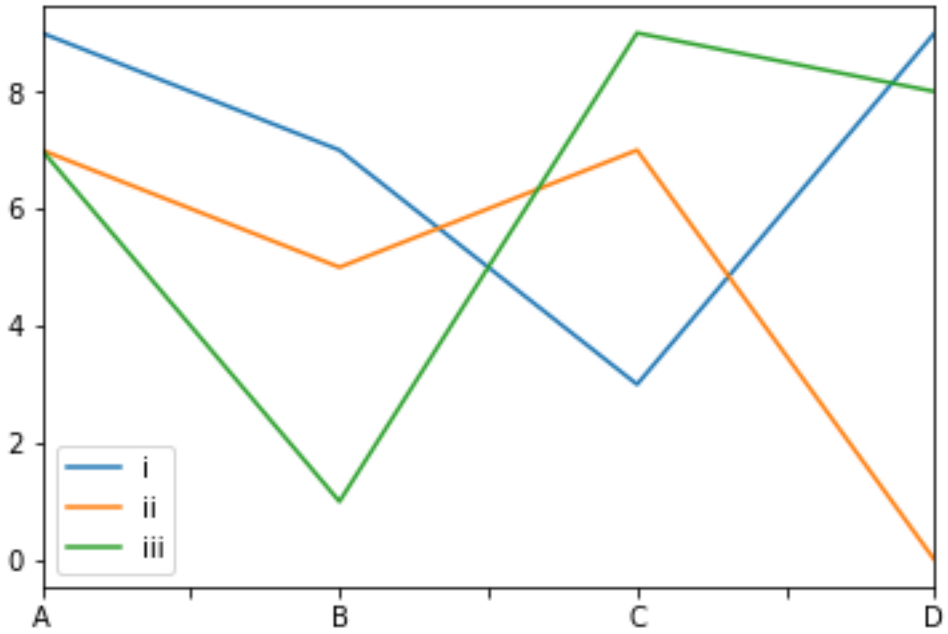


In [1893]: df2['weight2'].hist(bins=10[redacted]ue)
df2['weight2'].plot[redacted]', style='k--')
Out[1893]: <matplotlib.axes._subplots.AxesSubplot
0x53ddc828>

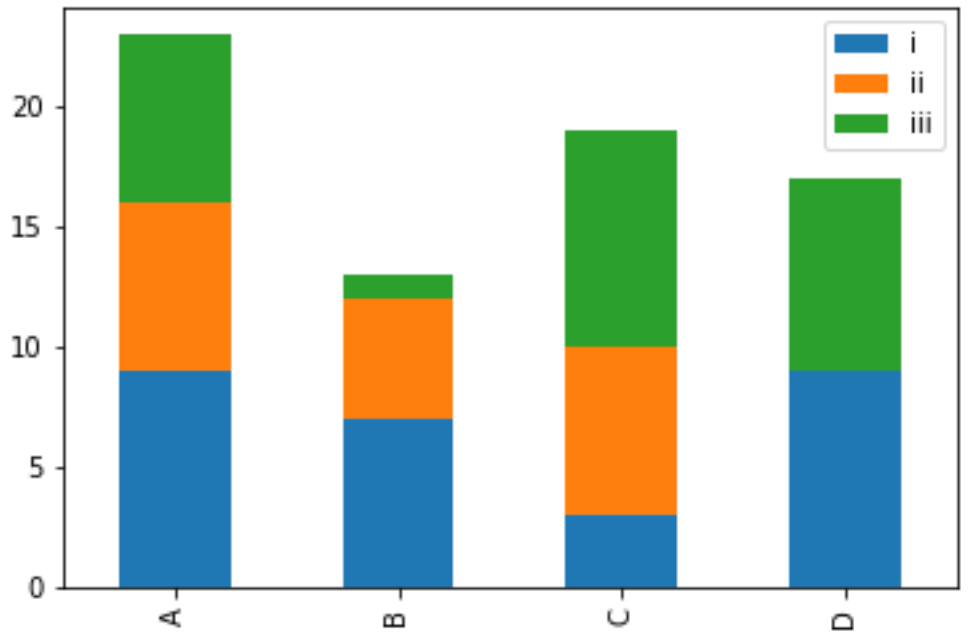


```
In [1911]: df3 = DataFrame(np.random.randint(0, 10, (4,
3)), index=['A', 'B', 'C', 'D'], columns=['i', 'ii', 'iii'])

In [1912]: df3.plot()
Out[1912]: <matplotlib.axes._subplots.AxesSubplot at
0x519a07b8>
```




```
In [1913]: df3.plot(kind='bar', stacked=True)
Out[1913]: <matplotlib.axes._subplots.AxesSubplot a
0x51afad68>
```



Why normalization (re-scaling)

	Height (inches)	Heights (feet)	Heights (cm)	Weight (LB)
A	63	5.25	160.0	150
B	64	5.33	162.6	155
C	72	6.00	182.9	156

```
In [1961]: def distance(ser1, ser2): return ((ser1-ser2)**2).sum()**0.5
```

A-B

```
In [1958]: distance(df6.loc['A',['inch','lb']],df6.loc['B',['inch','lb']])
Out[1958]: 5.099019513592784
```

A-C

```
In [1959]: distance(df6.loc['A',['inch','lb']],df6.loc['C',['inch','lb']])
Out[1959]: 10.816653826391969
```

B-C

```
In [1960]: distance(df6.loc['B',['inch','lb']],df6.loc['C',['inch','lb']])
Out[1960]: 8.06225774829855
```

A-B

```
In [1963]: distance(df6.loc['A',['foot','lb']],df6.loc['B',['foot','lb']])
Out[1963]: 5.000639959045242
```

A-C

```
In [1964]: distance(df6.loc['A',['foot','lb']],df6.loc['C',['foot','lb']])
Out[1964]: 6.046693311223912
```

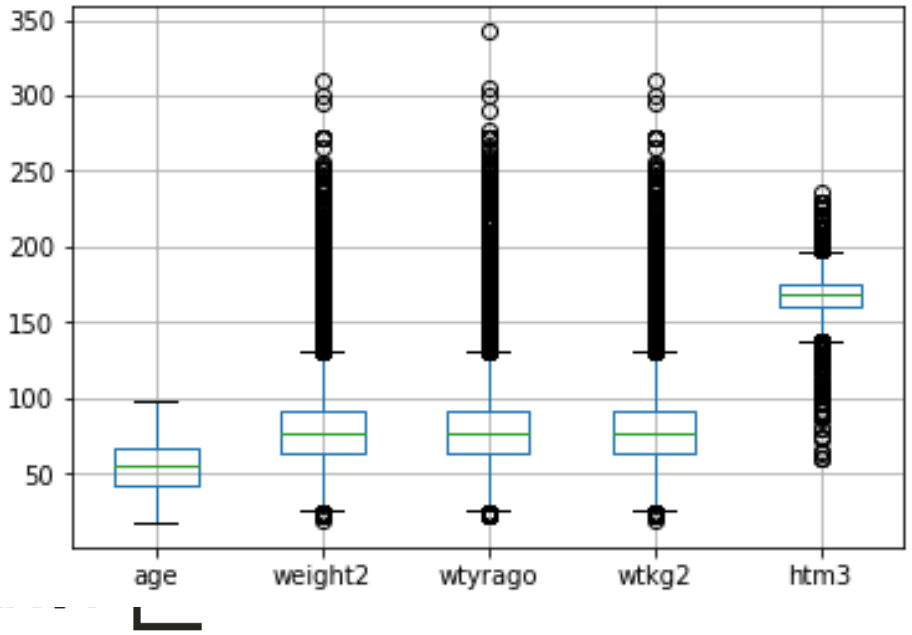
B-C

```
In [1965]: distance(df6.loc['B',['foot','lb']],df6.loc['C',['foot','lb']])
Out[1965]: 1.203702621082134
```

Z-score transformation

In [1929]: df4 = df.drop('sex', axis=1)

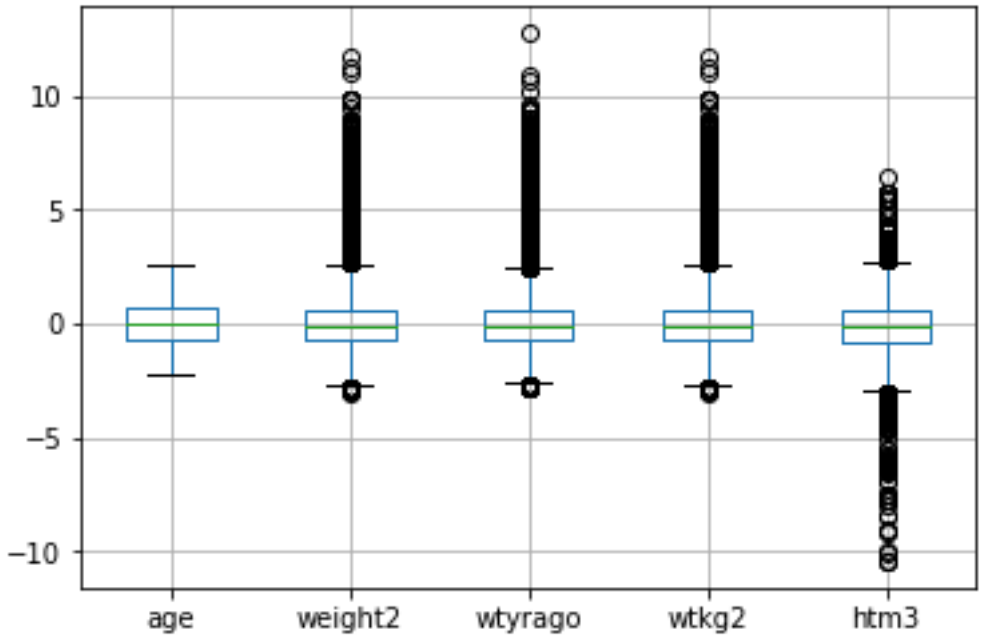
In [1930]: df4.boxplot()
Out[1930]: <matplotlib.axes._subplots.AxesSubplot at 0x51e9cb00>



In [1931]: def zscore(series): return (series - series.mean(skipna=True)) / series.std(skipna=True);

In [1932]: df5 = df4.apply(zscore)

In [1933]: df5.boxplot()
Out[1933]: <matplotlib.axes._subplots.AxesSubplot at 0x51e52ac8>

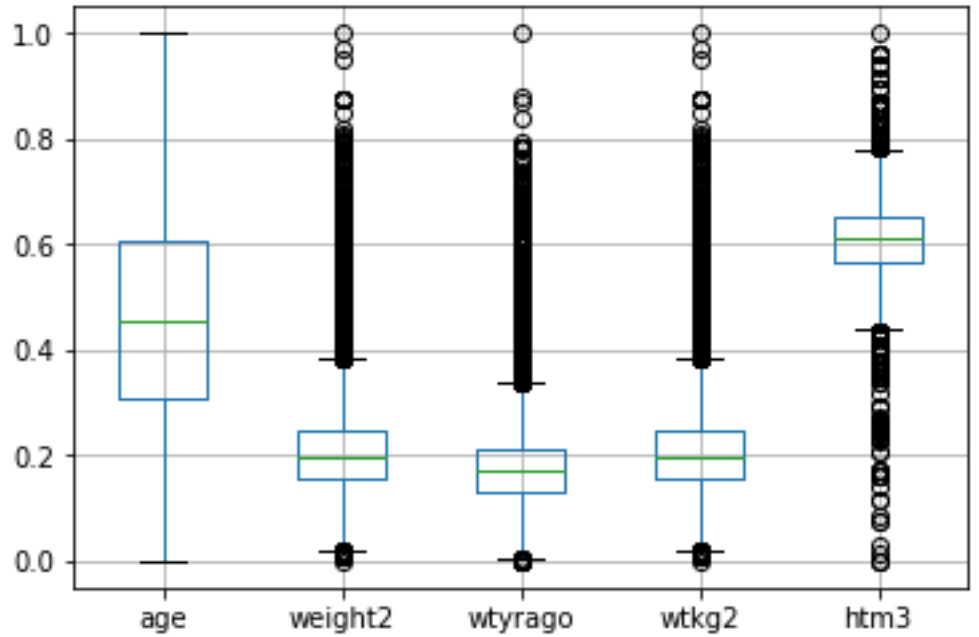


Scaling to between 0 and 1

In [**2181**]: def scaling(series):
    return (series - series.min()) / (series.max() - series.min())

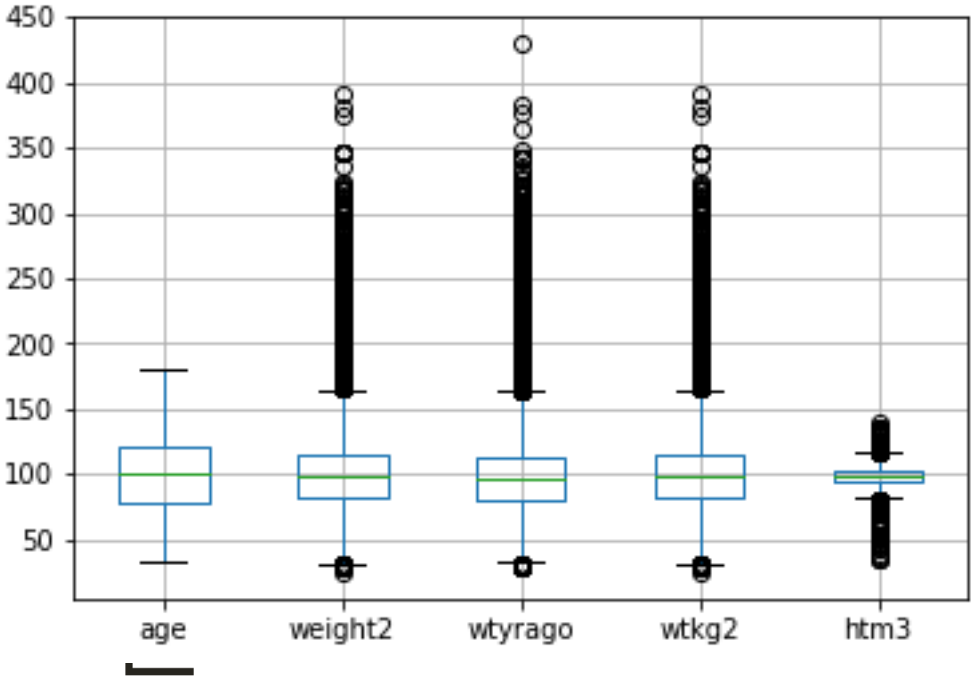
In [**2182**]: df7 = df4.apply(scaling)

In [**2183**]: boxplot(df7)



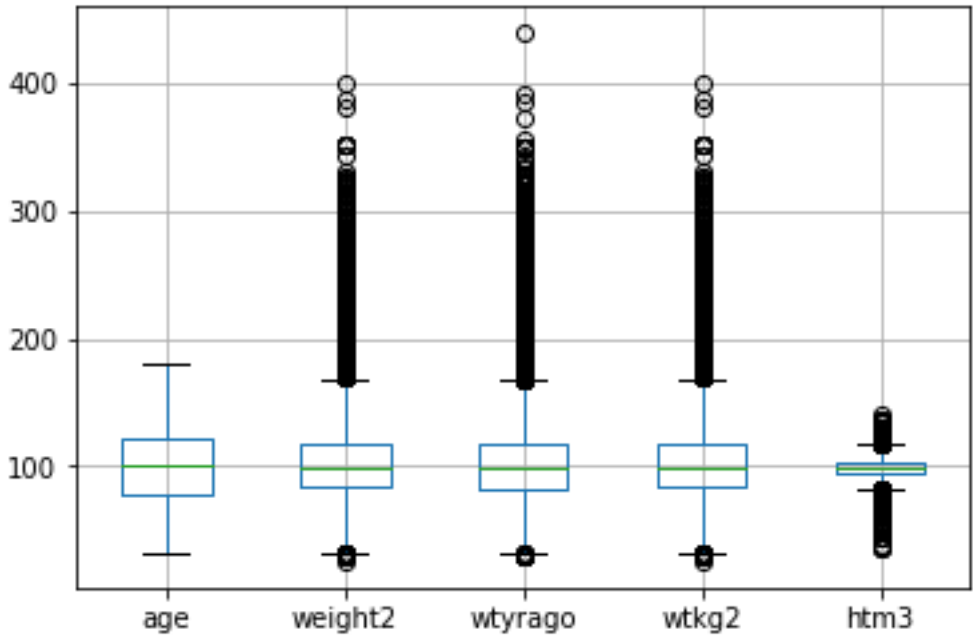
Mean-based scaling

```
In [2188]: def meanScaling(series):
      ...:         return series / series.mean()
      ...: df8 = df4.apply(meanScaling) * 100
      ...: df8.boxplot()
      ...:
```




```
In [2229]: def trimMeanScale(series, proportionToCut=0):
      ...:         return series /
stats.trim_mean(series.dropna(), proportionToCut)
In [2230]: df8 = df4.apply(trimMeanScale,
proportionToCut=0.1)*100
In [2231]: df8.boxplot()
```

Mean after removing largest and smallest proportionToCut data



Transform and normalize

```
In [2242]: df9 = df4.transform({'age': np.copy, 'weight2': np.log, 'wtyrago': np.log, 'wtkg2': np.log, 'htm3': np.copy})

In [2243]: df10 = df9.apply(zscore);

In [2244]: df10.boxplot()
```

Transform each column with a different function