Introduction to Data Science

Introduction

Kerstin Bunte and George Azzopardi

WMCS16002, semester Ia 2018

IDS staff 2018

`https://www.rug.nl/research/bernoulli`

Teachers

Kerstin Bunte

George Azzopardi

Teaching Assistants

Robert Bwana

Sreejita Ghosh

Henry Maathuis

Samira Rezaei

Course information

Period: Ia: 03.09.2018–23.10.2018

ECTS: 5 (140 work hours)

Workload: 8 lectures (incl. 1 flip classroom), 7 exercises + exam

Lectures: (mostly) Mon, 09:00–11:00 (check room)

Website: `http://www.rug.nl/ocasys/rug/vak/show?code=WMCS16002`

Exam: November 07, 09:00 – 12:00

Assignments

In order to pass the course, you must **pass at least** $\{all - 1\}$ **of the exercises**.[1]

All grades are fractional until rounded in the end.

In every exercise we expect a (short) **written report** and the corresponding **source code** which generates the results.
$\Rightarrow$ use the template available on Nestor!
Make running your code easy for the TAs. If they can't run your code, it is your fault.

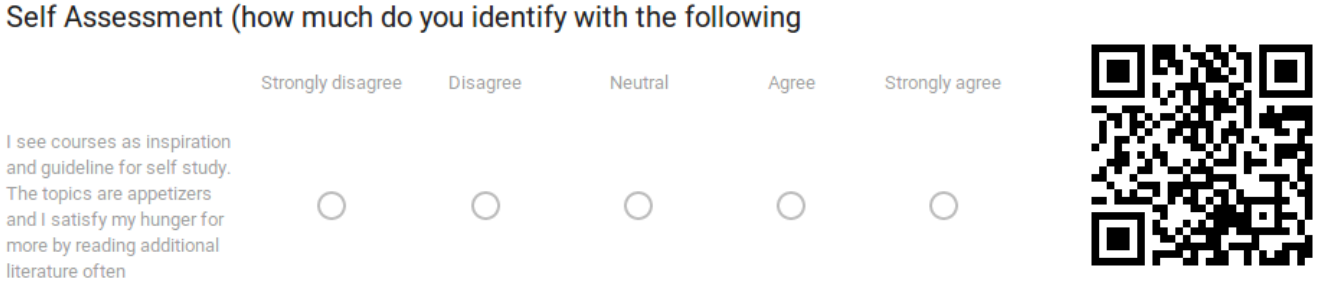
Check the **assignment deadlines!**
Every hour late decreases the grade with 10%.

[1] Passed grades are from 6 (sufficient) to 10 (outstanding). Grades <6 are not passing grades.

Group Work

All assignments (except the first) are submitted by the group

- $\approx$ 20 groups: membership assigned by us:
  - Interdisciplinary, but similar expectations!
  - Fill the questionnaire before Friday! (link on Nestor or QR)


Self Assessment (how much do you identify with the following

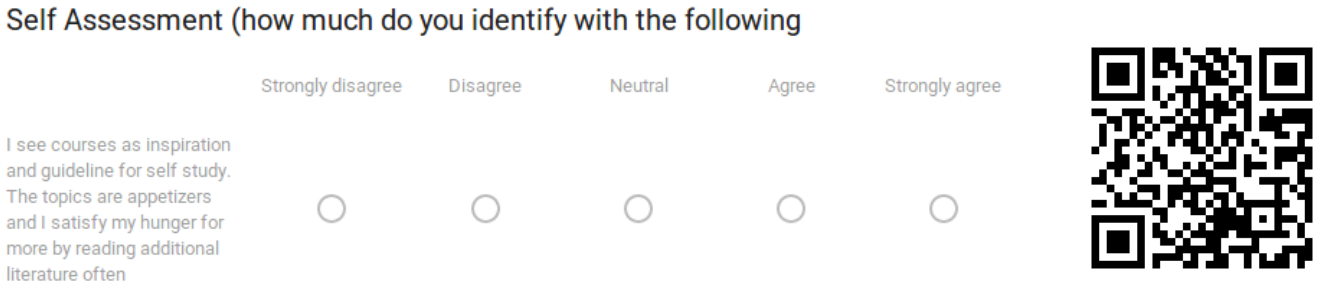
	Strongly disagree	Disagree	Neutral	Agree	Strongly agree
I see courses as inspiration and guideline for self study. The topics are appetizers and I satisfy my hunger for more by reading additional literature often	○	○	○	○	○

  - Do the homework before Friday! (Nestor) -ghost enrollments-

Group Work

All assignments (except the first) are submitted by the group

- $\approx$ 20 groups: membership assigned by us:
  - Interdisciplinary, but similar expectations!
  - Fill the questionnaire before Friday! (link on Nestor or QR)



  - Do the homework before Friday! (Nestor) -ghost enrollments-
- General assignment questions only in GitHub forum `https://github.com/RUG-IDS/Course-2018/issues`
  $\Rightarrow$ Don't be shy! If you can help your peers answer issues!
- Questions wrt. team grades and feedback in team repositories
- Start 2nd week: submission in GitHub (tutorial in 1st lab)

Grading

- The learning objectives are individually assessed by an exam.
- Final peer-review questionnaire: every team member evaluated by the peers wrt. professional behavior (quality, quantity, reliability) Nestor!
- The assignments (A) are worth together 60% of the grade; the exam (E) is worth 40% of the grade.
- The Final Grade (F) $= \min^2(A \cdot 0.6 + E \cdot 0.4, 10)$
- F should be above 6.0
- Both A and E grades must be above 5.0

The fine print: Presence in the lectures is not mandatory besides 1 exception for the topic presentation Oct. 1.

[2] Doing optional exercises makes it possible to have more than 10

Prerequisites

Bachelor level knowledge from one of the fields: Computing Science, Mathematics or Astronomy (Specialization Data Science)

- Basic mathematic courses
- Basic programming knowledge of *R*, *Matlab*, or *Python*
  because we use these languages in the course; for the exercises you are free to choose whatever language you want (and are able solve the exercises with)
  Note: disadvantageous to stick with one language (implementations vary in usefulness)

Content

We aim to
(1) present principles wrt. data science
(2) look at advanced analytical theory and methods, and
(3) give broad overview + reference to MSc courses for orientation.

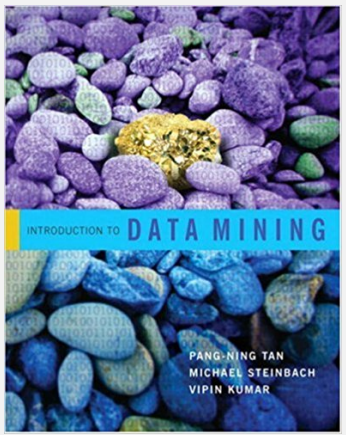
Schedule (tentative):

	Day	Date	Topic	Lecturer
1	Mon	03.09.	Introduction + Data	K
2	Mon	10.09.	Preprocessing	G
3	Tue	11.09.	Optimization	G
-	Mon	17.09.	*homework*	-
4	Mon	24.09.	Text Analysis	G
5	Mon	01.10.	Flip Classroom	K
6	Mon	08.10.	Unsupervised Methods	K
7	Mon	15.10.	Supervised Methods	K
8	Mon	22.10.	Time Series Analysis	G

Material

Course:

The course is mainly based on
Introduction to Data Mining, P. N. Tan, M. Steinbach and V. Kumar, PEARSON Addison Wesley
and (recent) scientific literature.
We will provide the slides and other course material (source code, data sets, etc.) via the Nestor portal.



Software:

Matlab: is installed in the RUG computers

R: `http://www.r-project.org/`

Python: `https://www.python.org/`

Git: is installed in RUG computers

Communication

We use Nestor for broadcasting announcements of potential relevance for everybody.

You use the GitHub issue tracking for questions of potential relevance to your colleagues

For face to face communication, we have office hours:

When: Tuesdays, 11am – 12am

Where: BB590 (K. Bunte), BB594 (G. Azzopardi)

Data Science

Or machine learning, data mining, etc.
Win real money! `https://www.kaggle.com/competitions`

An example

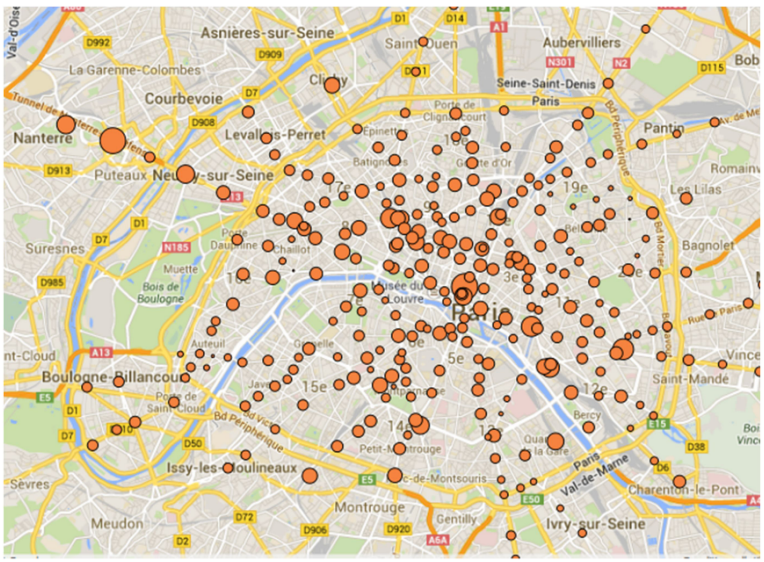
- Anonymized data from a french phone company:
  - \+ SIM ID
  - \+ GPS position
  - \+ Time of a call
  - \+ SIM ID called
  - \+ Length of a call
  - \- No Names
  - \- No addresses
  - \- No call content

Link with additional data, such as geography, traffic(-lights), etc.

An example

- Anonymized data from a french phone company:
  - \+ SIM ID
  - \+ GPS position
  - \+ Time of a call
  - \+ SIM ID called
  - \+ Length of a call
  - \- No Names
  - \- No addresses
  - \- No call content

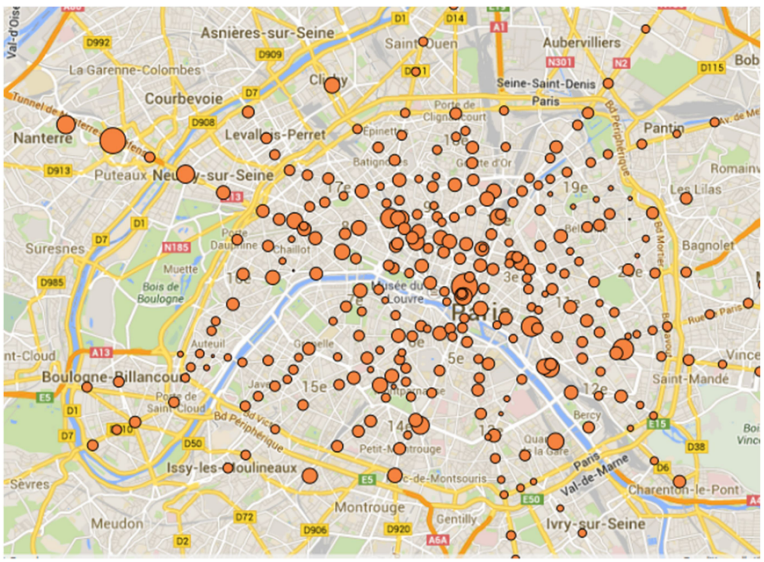
Link with additional data, such as geography, traffic(-lights), etc.

The bright side	[redacted]
- identify rush-hour bottlenecks - optimize traffic lights - save time and emissions! 	[redacted]

An example

- Anonymized data from a french phone company:
  - \+ SIM ID
  - \+ GPS position
  - \+ Time of a call
  - \+ SIM ID called
  - \+ Length of a call
  - \- No Names
  - \- No addresses
  - \- No call content

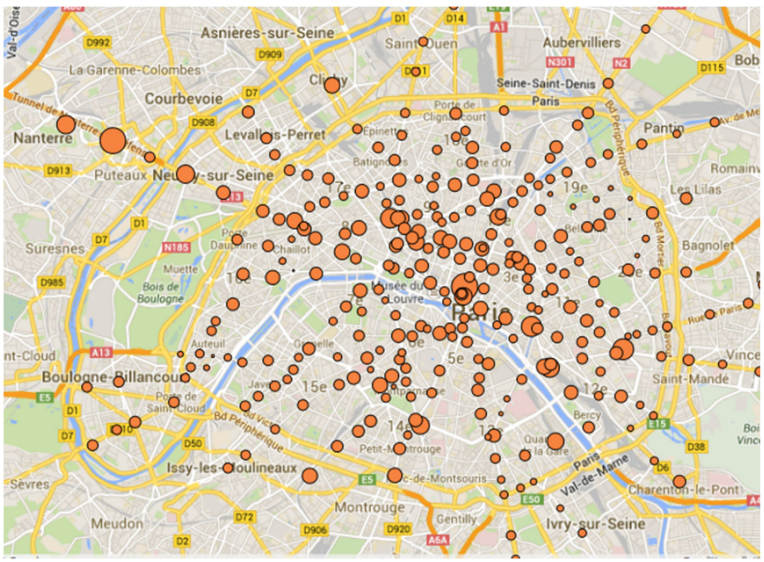
Link with additional data, such as geography, traffic(-lights), etc.

The bright side	The dark side . . .
- identify rush-hour bottlenecks - optimize traffic lights - save time and emissions! 	

An example

- Anonymized data from a french phone company:
  - \+ SIM ID
  - \+ GPS position
  - \+ Time of a call
  - \+ SIM ID called
  - \+ Length of a call
  - \- No Names
  - \- No addresses
  - \- No call content

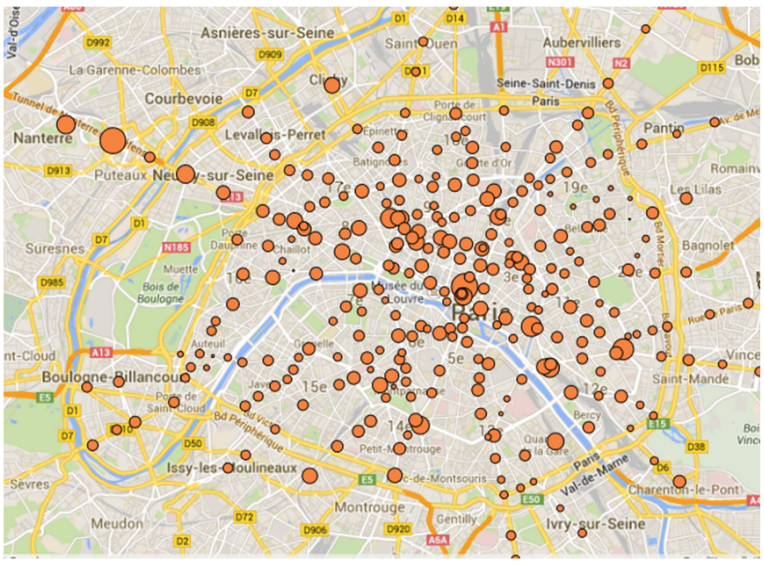
Link with additional data, such as geography, traffic(-lights), etc.

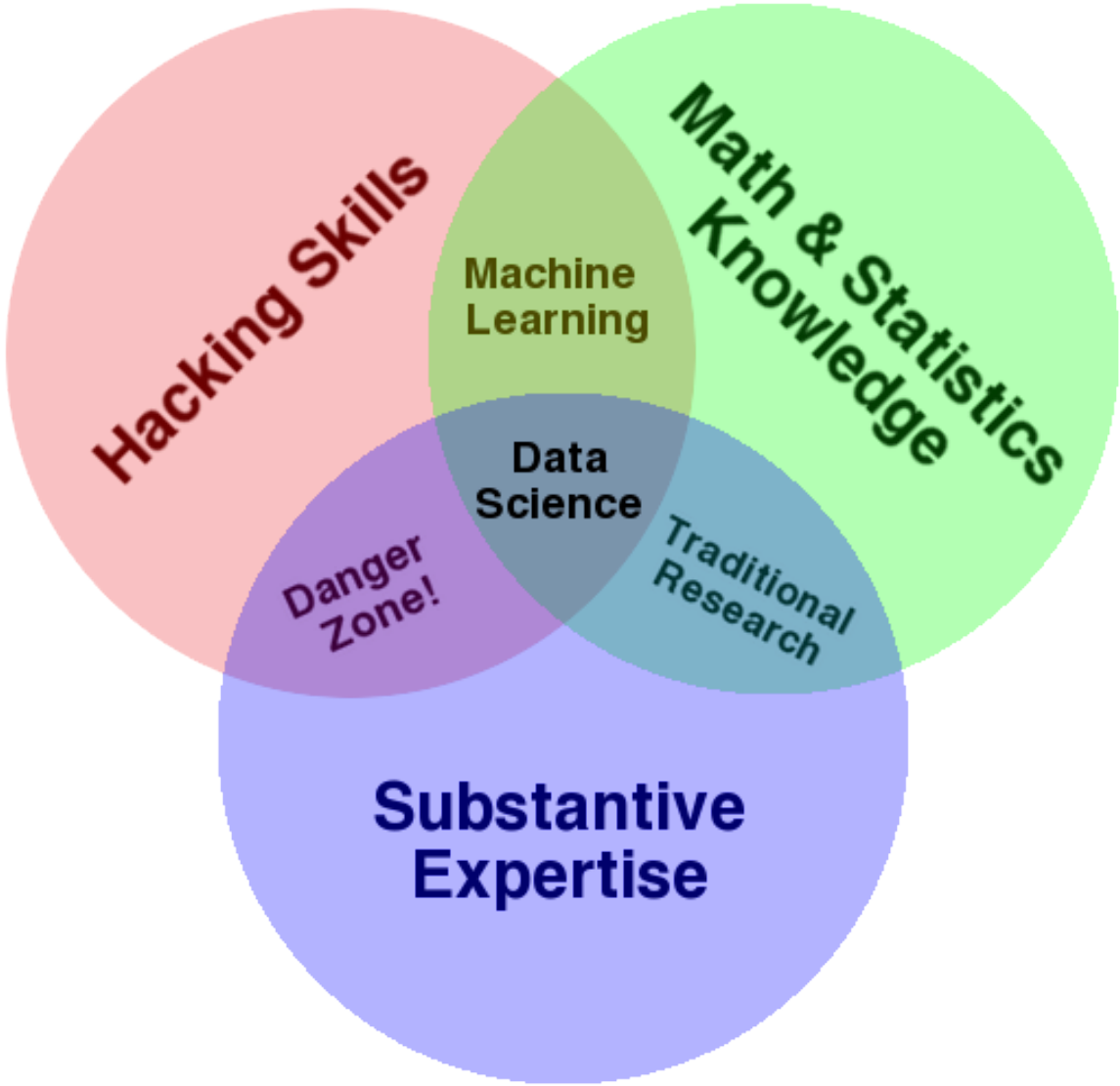
The bright side	The dark side . . .
- identify rush-hour bottlenecks - optimize traffic lights - save time and emissions! 	Uncover identifying information: - living address - number of phones/shared - hobbies/shopping behaviour $\Rightarrow$ interesting for advertising/burglary

An example

- Anonymized data from a french phone company:
  - \+ SIM ID
  - \+ GPS position
  - \+ Time of a call
  - \+ SIM ID called
  - \+ Length of a call
  - \- No Names
  - \- No addresses
  - \- No call content

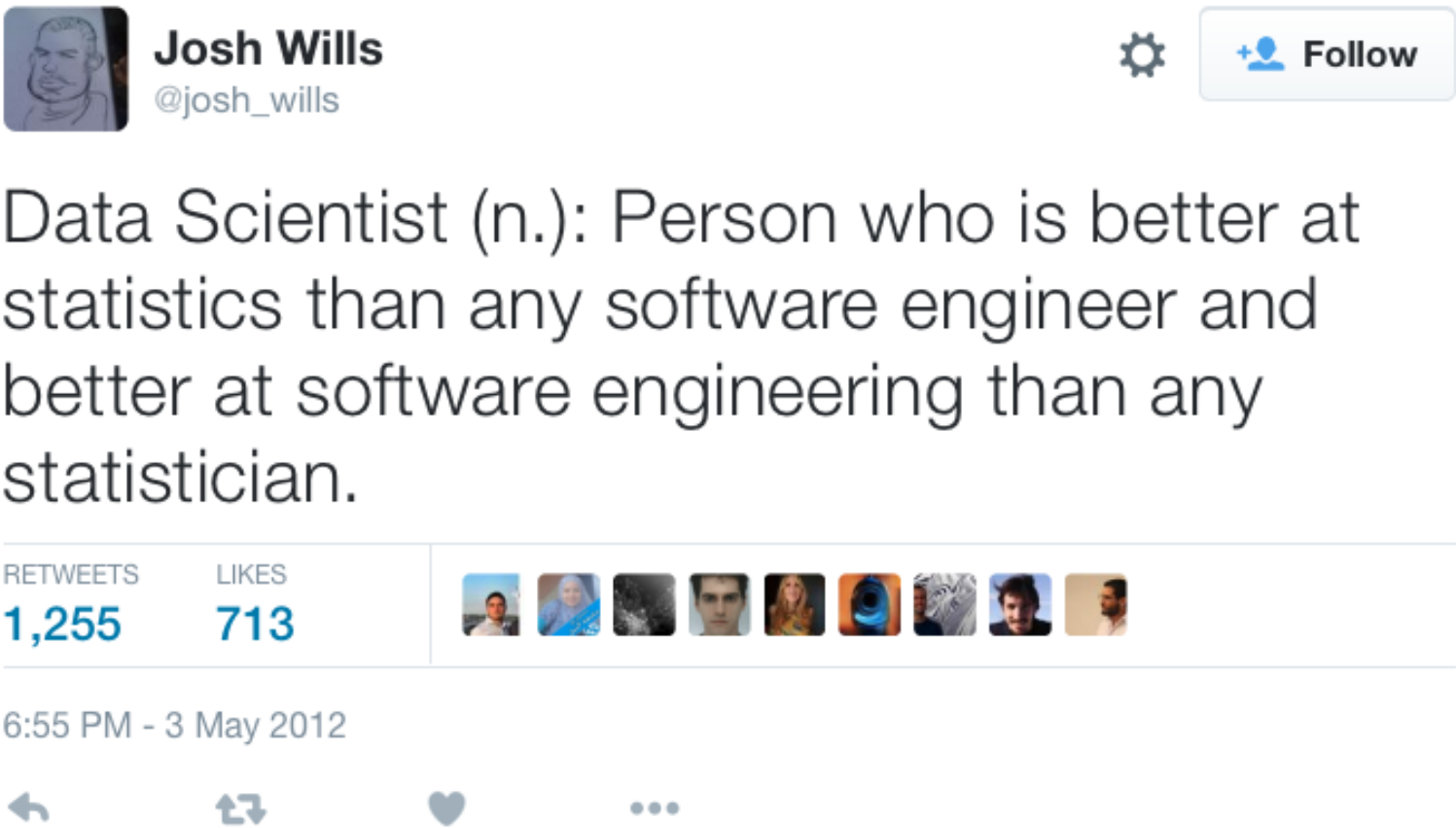
Link with additional data, such as geography, traffic(-lights), etc.

The bright side	The dark side . . .
- identify rush-hour bottlenecks - optimize traffic lights - save time and emissions! 	Uncover identifying information: - living address - number of phones/shared - hobbies/shopping behaviour ⇒ interesting for advertising/burglary Discover “unusual” behaviour: - no out-, only incoming calls - short calls - no movement ⇒ interesting for law enforcement, totalitarian leaders, etc.



http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram

What is a Data Scientist?



A Data Science Project

Asking the right question **first** is very important. Types of questions (in increasing order of difficulty):

1. Descriptive
2. Exploratory
3. Inferential
4. Predictive
5. Causal

1. Descriptive Analysis

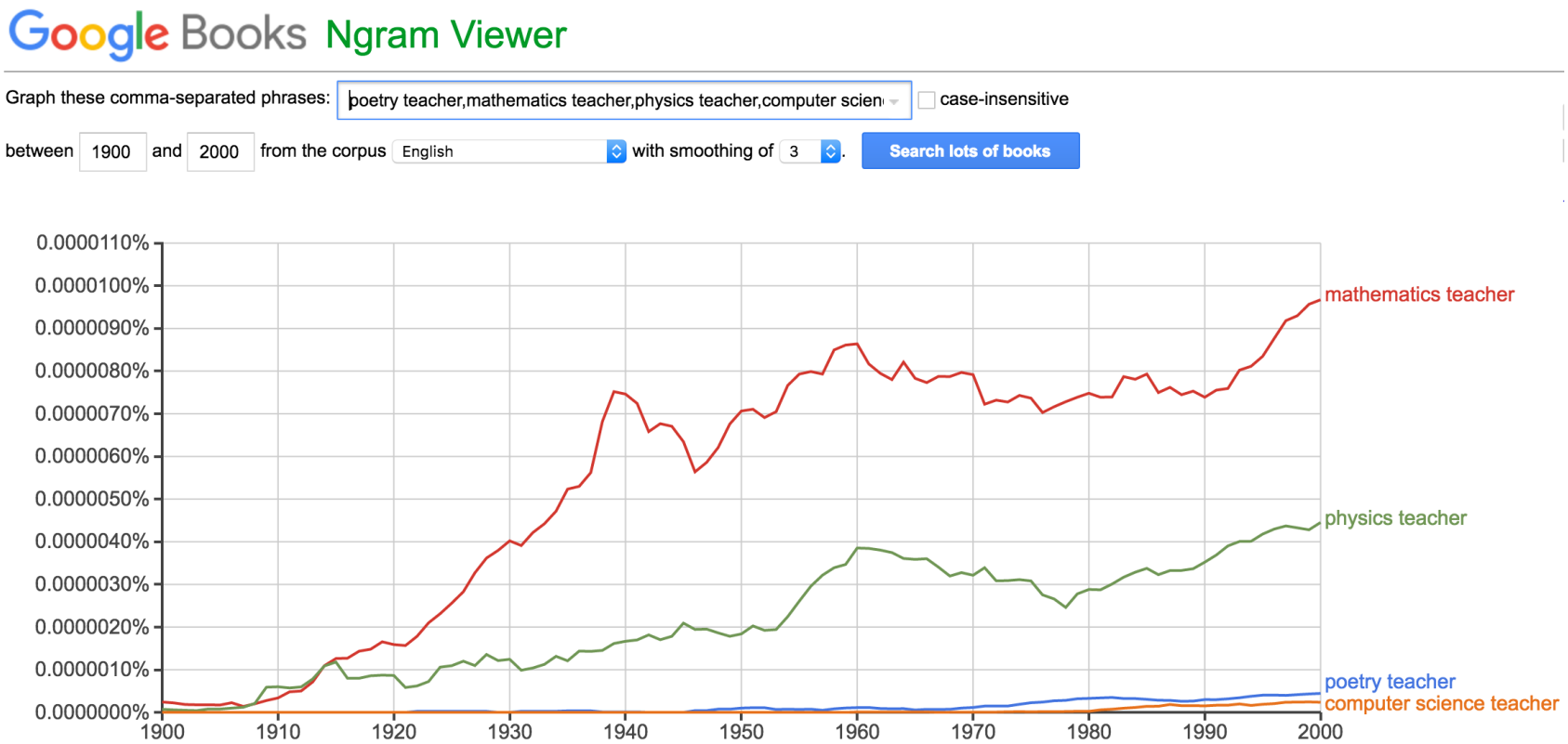
Goal: describe a set of data.

- Numbers used to summarize and describe data
- Multiple numbers represeting descriptive statistics can be used simultaneously
- They do not assume **generalizing** beyond the data at hand

Avg. Salary (USD)	Job
124,310	dentist
87,890	physicist
64,410	architect
49,710	police
43,482	teacher

US, 2016

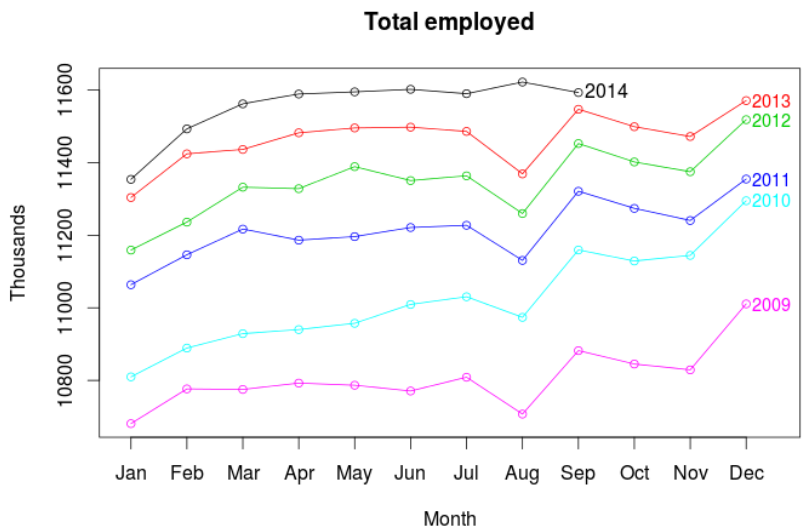
1. Descriptive Analysis

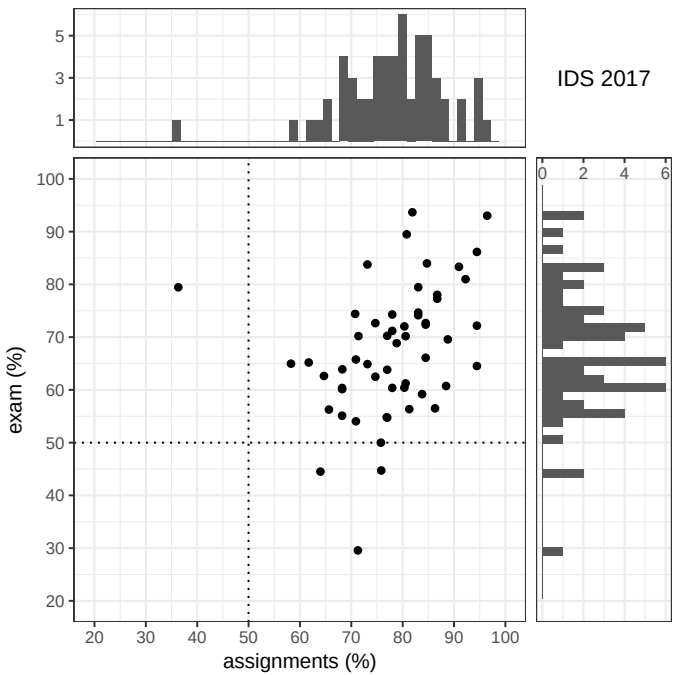


2. Exploratory Analysis

Goal: Find relationships that you didn't know about

- Good for defining future studies
- Should not be used for generalizing/predicting
- Correlation does not imply causation





3. Inferential Analysis

Goal: Use sample data to say something about larger population

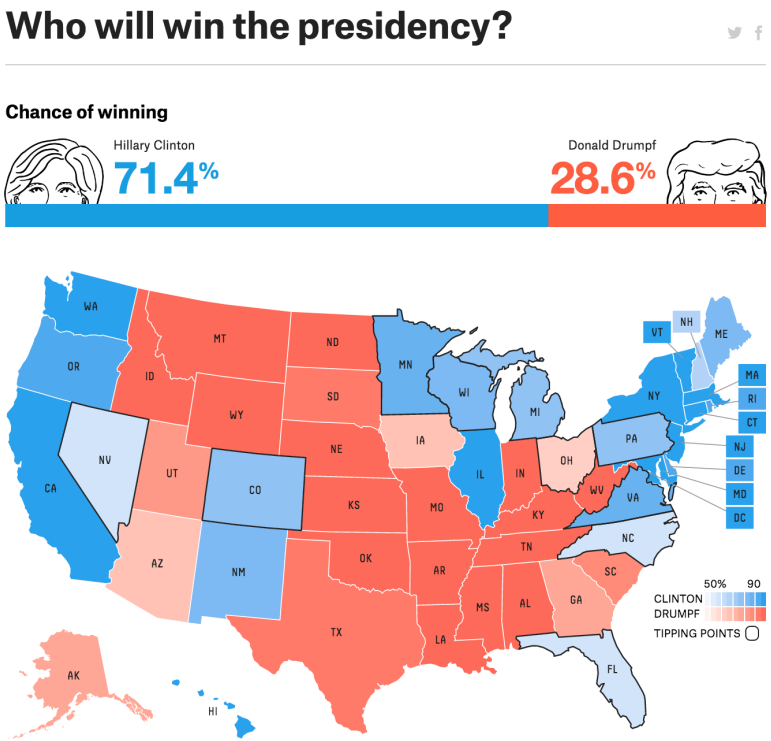
- Usually the goal of statistical models
- Involves estimating the quantity of interest and uncertainty
- Relies on using adequate sampling

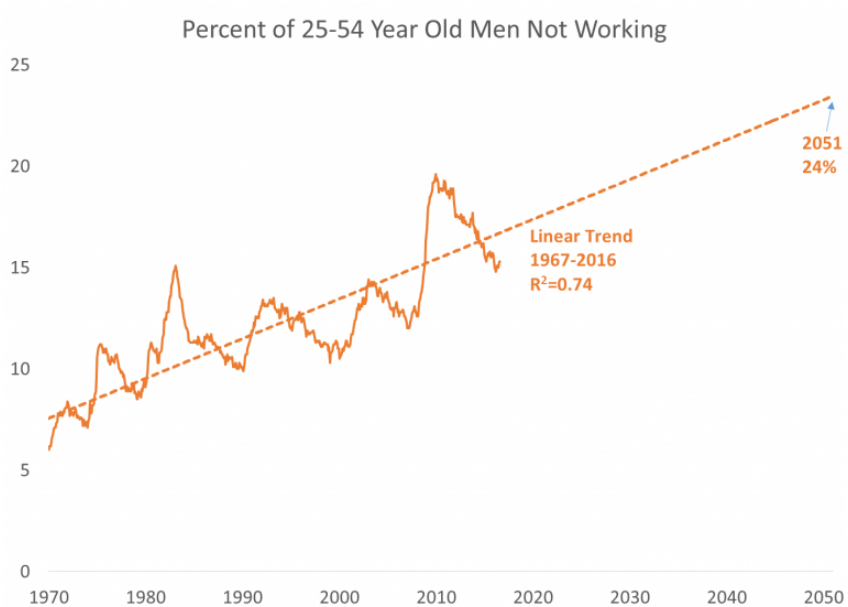
Example: You have been hired by the National Election Commission to examine how the Dutch people feel about the fairness of the voting procedures in Netherland.
Whom will you ask?

4. Predictive Analysis

Goal: use data on some objects to predict values on another object

- if X predicts Y it does not mean X causes Y
- focus is on "What will happen"?
- prediction is very hard, especially about the future (N. Bohr)





Data

- **Type of Data**:
  - Quantitative or Qualitative
  - Special characterisitics:
    (time-series, objects with explicit relations)
  - Determines which tools to use for analysis
- **Quality of Data**:
  - Far from perfect!
    (most methods tolerate some level of imperfection)
  - Improving data often improves quality of resulting analysis
  - Quality issues:

- **Type of Data**:
  - Quantitative or Qualitative
  - Special characterisitics:
    (time-series, objects with explicit relations)
  - Determines which tools to use for analysis
- **Quality of Data**:
  - Far from perfect!
    (most methods tolerate some level of imperfection)
  - Improving data often improves quality of resulting analysis
  - Quality issues:
    - presence of *noise* and *outliers*
    - *missing*, inconsistent, or *duplicate*
    - data is *biased* or *unrepresentative* of phenomenon or population that's supposed to be described

Data Types

Attribute Type		Description	Examples	Operations
Categorical (Qualitative)	Nominal			
	Ordinal			
Numeric (Quantitative)	Interval			
	Ratio Scale			

Data Types

Attribute Type		Description	Examples	Operations
Categorical (Qualitative)	Nominal	No natural ordering; only distinguishing information $(=, \neq)$	favorite color, gender, ID number	mode, entropy, correlation, χ^2 test
	Ordinal			
Numeric (Quantitative)	Interval			
	Ratio Scale			

Data Types

Attribute Type		Description	Examples	Operations
Categorical (Qualitative)	Nominal	No natural ordering; only distinguishing information $(=, \neq)$	favorite color, gender, ID number	mode, entropy, correlation, χ^2 test
	Ordinal	categories can be ordered $(>, <)$	t-shirt size, attitude	median, percentiles, rank correlation
Numeric (Quantitative)	Interval			
	Ratio Scale			

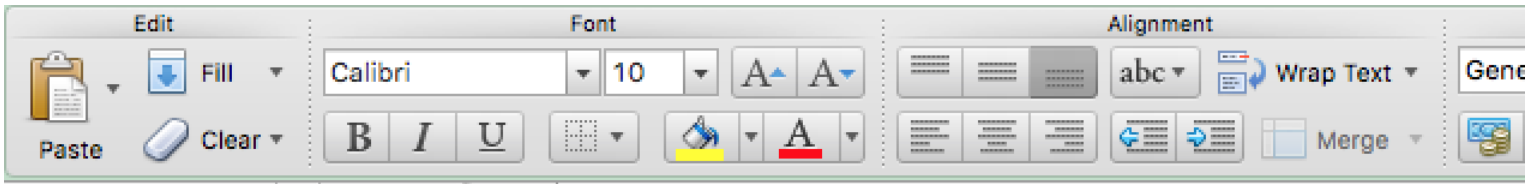
Data Types

Attribute Type		Description	Examples	Operations
Categorical (Qualitative)	Nominal	No natural ordering; only distinguishing information $(=, \neq)$	favorite color, gender, ID number	mode, entropy, correlation, χ^2 test
	Ordinal	categories can be ordered $(>, <)$	t-shirt size, attitude	median, percentiles, rank correlation
Numeric (Quantitative)	Interval	numerical, equidistant measure, differences are meaningful $(+, -)$	calendar dates, temperature in Celsius or Fahrenheit	mean, standard deviation, Pearson's correlation, t and F tests
	Ratio Scale			

Data Types

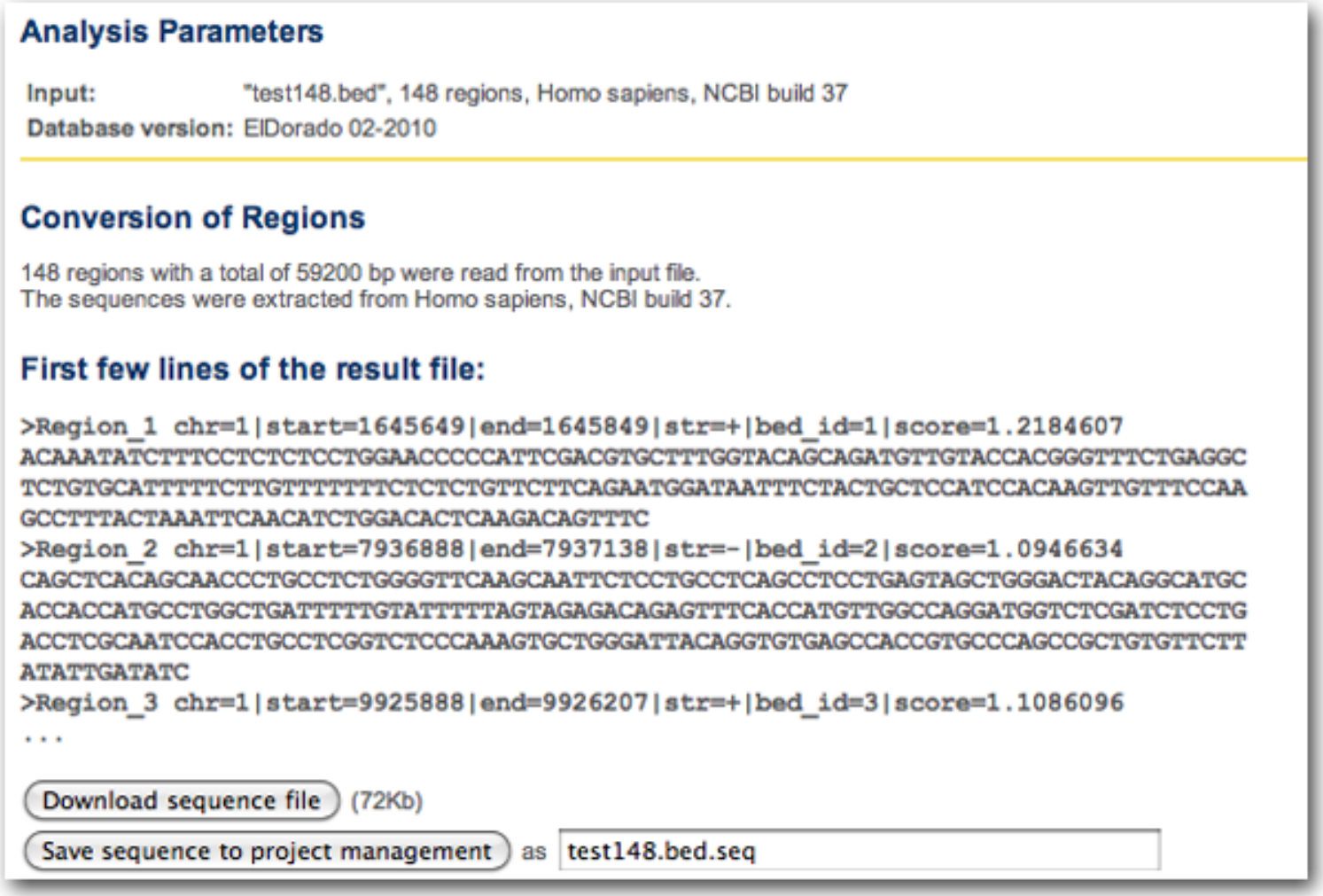
Attribute Type		Description	Examples	Operations
Categorical (Qualitative)	Nominal	No natural ordering; only distinguishing information $(=, \neq)$	favorite color, gender, ID number	mode, entropy, correlation, χ^2 test
	Ordinal	categories can be ordered $(>, <)$	t-shirt size, attitude	median, percentiles, rank correlation
Numeric (Quantitative)	Interval	numerical, equidistant measure, differences are meaningful $(+, -)$	calendar dates, temperature in Celsius or Fahrenheit	mean, standard deviation, Pearson's correlation, t and F tests
	Ratio Scale	Interval Scale, both difference and ratios are meaningful $(*, /)$	age, temperatur in Kelvin, True rations exist (e.g. I swim half as fast as Phelps)	geometric/ harmonic mean, percent variation

Data in the Wild: Spreadsheets



	A	B	C	D	E	F	G	H
1	Title	Directors	AVG rating	IMDb Ratin	Rotten Tor	Metacriti	Runt	Year
2	Planet Earth		95	9.5			550	2006
3	Breaking Bad		95	9.5			45	2008
4	The Shawshank Redemption	Frank Darabont	87.666667	9.3	90	80	142	1994
5	Sherlock	Mark Gatiss, Steven Moffat	93	9.3			90	2010
6	The Godfather	Francis Ford Coppola	97.333333	9.2	100	100	175	1972
7	The Blue Planet	Alastair Fothergill	92	9.2	n/a		400	2001
8	Human Planet		84.5	9.2	n/a	77	480	2011
9	The Good, the Bad and the U	Sergio Leone	92.333333	9	97	90	161	1966
10	The Godfather: Part II	Francis Ford Coppola	89.666667	9	99	80	200	1974
11	Pulp Fiction	Quentin Tarantino	92.666667	9	94	94	154	1994
12	The Dark Knight	Christopher Nolan	90	9			152	2008
13	Life		90	9			585	2009
14	Valley Uprising		90	9				2014
15	Mad Max: Fury Road	George Miller	92.333333	9	98	89	120	2015
16	Interstellar	Christopher Nolan	90	9				
17	12 Angry Men	Sidney Lumet	94.5	8.9	100	n/a	96	1957
18	The Decalogue	Krzysztof Kieślowski	94.5	8.9	100	n/a	55	1988
19	Schindler's List	Steven Spielberg	93	8.9	97	93	195	1993
20	Fight Club	David Fincher	78.666667	8.9	81	66	139	1999
21	The Lord of the Rings: The Re	Peter Jackson	89	8.9			201	2003
22	Boyhood	Richard Linklater	89	8.9			165	2014
23	Seven Samurai	Akira Kurosawa	95.666667	8.8	100	99	207	1954
24	One Flew Over the Cuckoo's	Milos Forman	88	8.8			133	1975
25	Star Wars	George Lucas	88	8.8			121	1977
26	Star Wars: Episode V - The E	Irvin Kershner	88	8.8			124	1980
27	Goodfellas	Martin Scorsese	91.333333	8.8	97	89	146	1990
28	The Lord of the Rings: The Fe	Peter Jackson	88	8.8			178	2001
29	Inception	Christopher Nolan	81	8.8		74	148	2010
30	Gravity	Alfonso Cuarón	93.666667	8.8	97	96	90	2013
31	House of Cards		88	8.8			60	2013
32	Casablanca	Michael Curtiz	87	8.7			102	1942
33	It's a Wonderful Life	Frank Capra	87	8.7			130	1946

Data in the Wild: Files



Analysis Parameters

Input: "test148.bed", 148 regions, Homo sapiens, NCBI build 37
Database version: ElDorado 02-2010

Conversion of Regions

148 regions with a total of 59200 bp were read from the input file.
The sequences were extracted from Homo sapiens, NCBI build 37.

First few lines of the result file:

```
>Region_1 chr=1|start=1645649|end=1645849|str=+|bed_id=1|score=1.2184607
ACAAATATCTTTCCTCTCTCCTGGAACCCCCATTCGACGTGCTTTGGTACAGCAGATGTTGTACCACGGGTTTCTGAGGC
TCTGTGCATTTTTCTTGTTTTTTTCTCTCTGTTCTTCAGAATGGATAATTTCTACTGCTCCATCCACAAGTTGTTTCCAA
GCCTTTACTAAATTCAACATCTGGACACTCAAGACAGTTTC
>Region_2 chr=1|start=7936888|end=7937138|str=-|bed_id=2|score=1.0946634
CAGCTCACAGCAACCCTGCCTCTGGGGTTCAAGCAATTCTCCTGCCTCAGCCTCCTGAGTAGCTGGGACTACAGGCATGC
ACCACCATGCCTGGCTGATTTTTGTATTTTTAGTAGAGACAGAGTTTCACCATGTTGGCCAGGATGGTCTCGATCTCCTG
ACCTCGCAATCCACCTGCCTCGGTCTCCCAAAGTGCTGGGATTACAGGTGTGAGCCACCGTGCCCAGCCGCTGTGTTCTT
ATATTGATATC
>Region_3 chr=1|start=9925888|end=9926207|str=+|bed_id=3|score=1.1086096
...
```

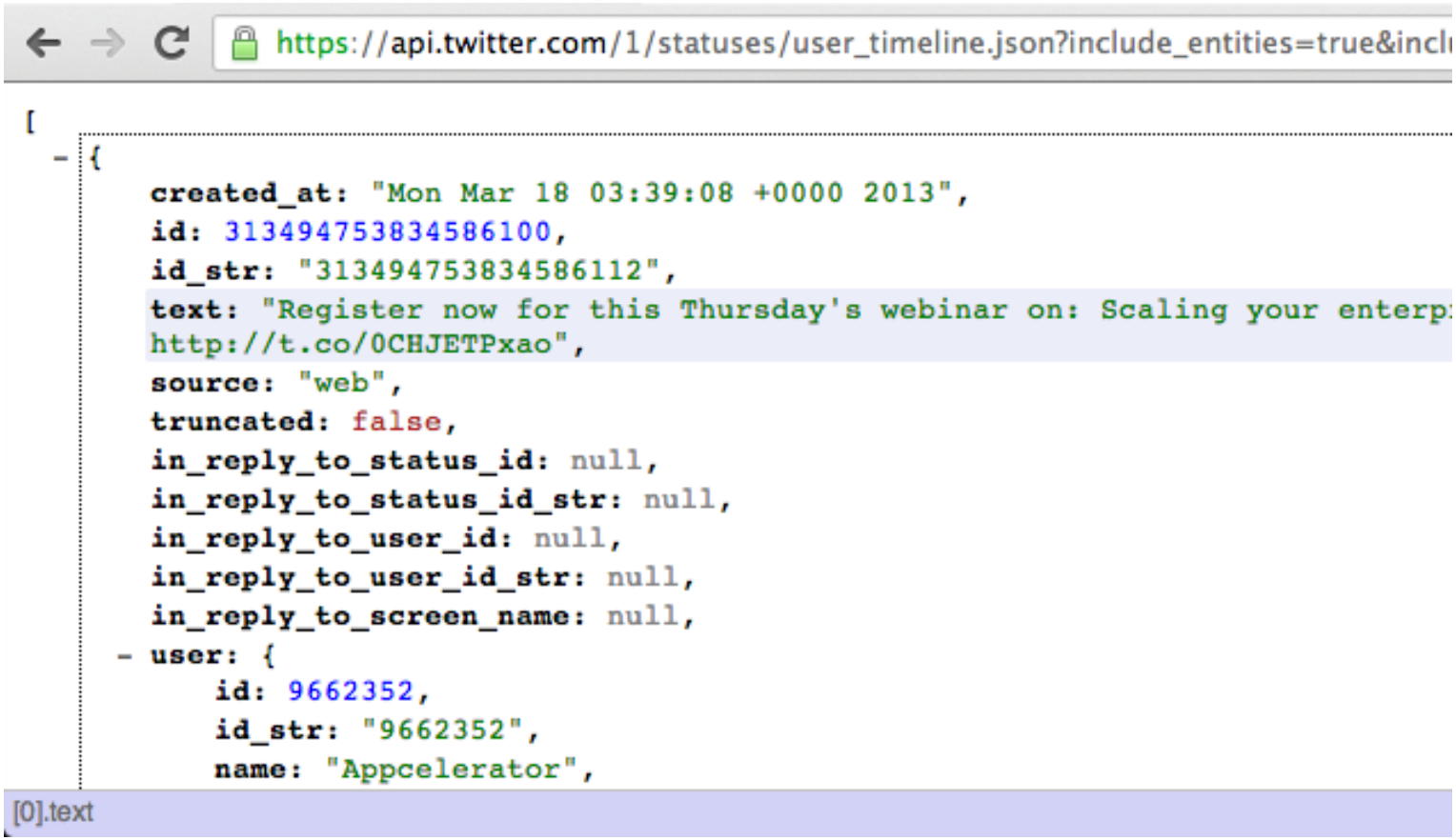
Download sequence file (72Kb)

Save sequence to project management as test148.bed.seq

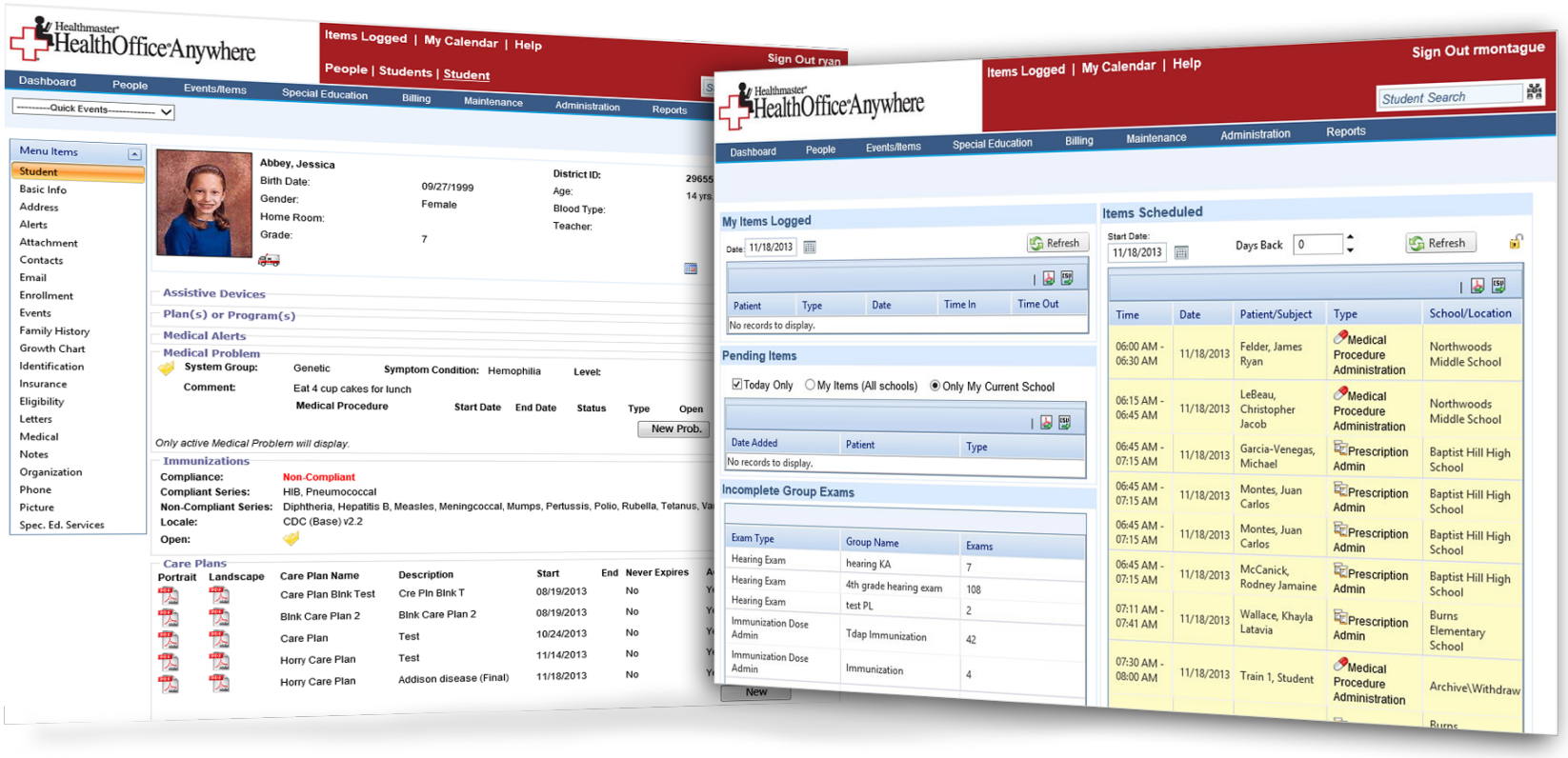
Semi-structured data (from genome sequencing). Requires parsing.

Data in the Wild: Web APIs

- Data-exchange formats: (CSV, XML, JSON, etc.).
- Usually require a key for accessing



Data in the Wild: Databases



Usually found in relational databases (MySQL, PostgreSQL, etc.). Most of the times, the person who designed the schema, did not think about your data science question!

Data Structures

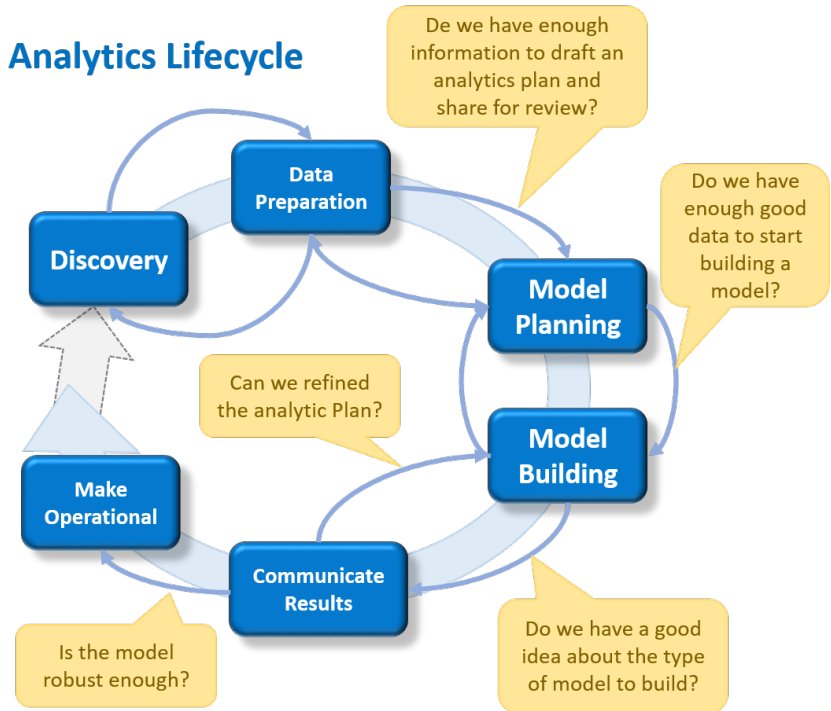
Three types of data:

1. Structured: Relational Databases (mySQL, PostgreSQL, etc.), Spreadsheets
2. Semi-structured: File-based with a certain amount of structure (CSV, XML, JSON, NoSQL).
3. Unstructured: text, multimedia.

Unstructured data is growing at the fastest rate

Steps in Data Analysis

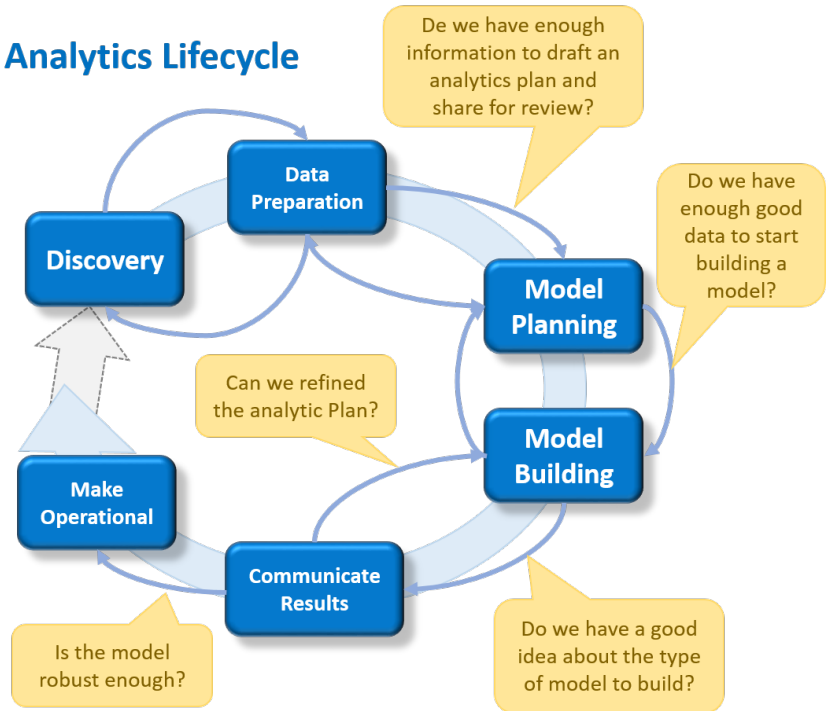
0. Define question
1. Define ideal data set



Steps in Data Analysis

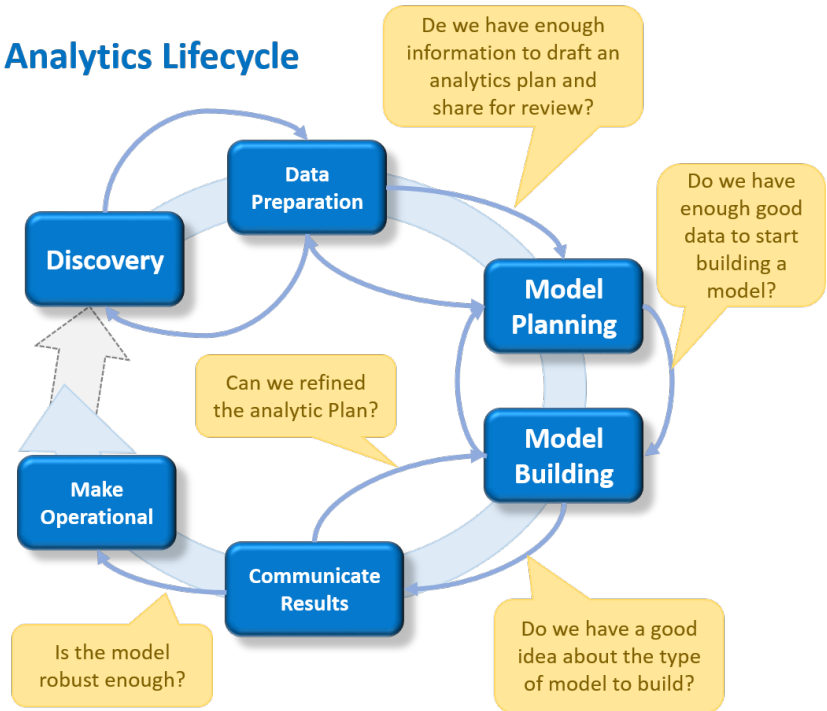
0. Define question
1. Define ideal data set

2. Find what data you can access
3. Obtain the data
4. Clean the data



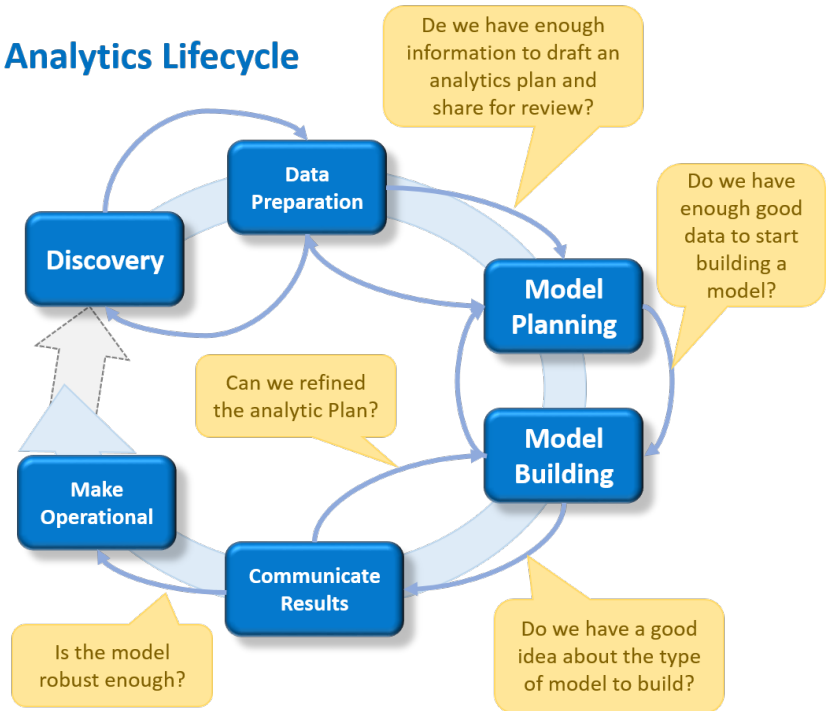
Steps in Data Analysis

0. Define question
1. Define ideal data set
2. Find what data you can access
3. Obtain the data
4. Clean the data
5. Exploratory analysis
6. Modeling (statistical, ML, etc.)



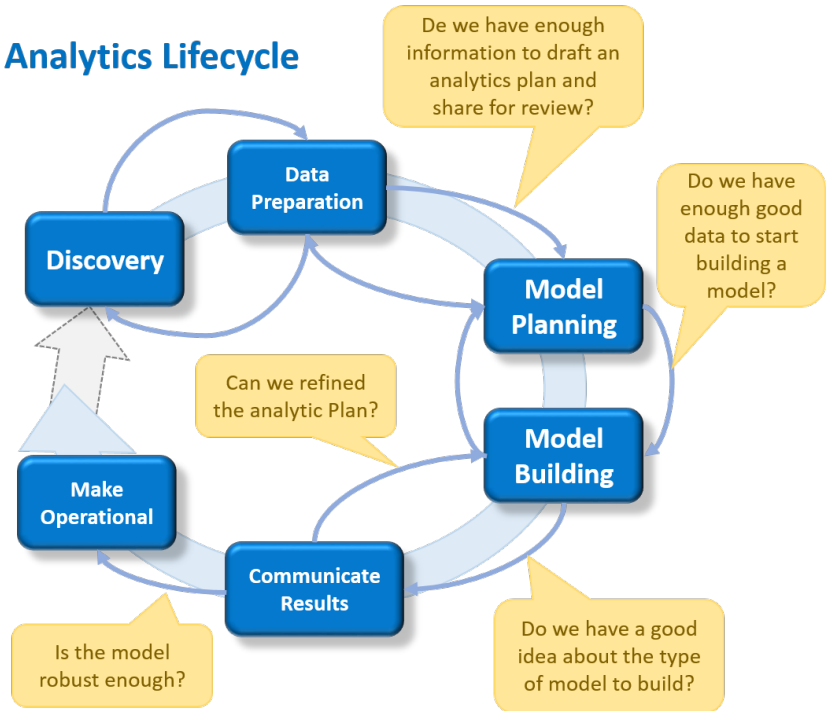
Steps in Data Analysis

0. Define question
1. Define ideal data set
2. Find what data you can access
3. Obtain the data
4. Clean the data
5. Exploratory analysis
6. Modeling (statistical, ML, etc.)
7. Interpret results
8. Threats to validity
9. Write up results



Steps in Data Analysis

0. Define question
1. Define ideal data set
2. Find what data you can access
3. Obtain the data
4. Clean the data
5. Exploratory analysis
6. Modeling (statistical, ML, etc.)
7. Interpret results
8. Threats to validity
9. Write up results
10. Ensure reproducibility (Deploy)



Reproducibility

"If you do it once, great.
If you do it twice, frown.
If you do it three times, automate it."

`http://wiki.c2.com/?AutomationIsOurFriend`

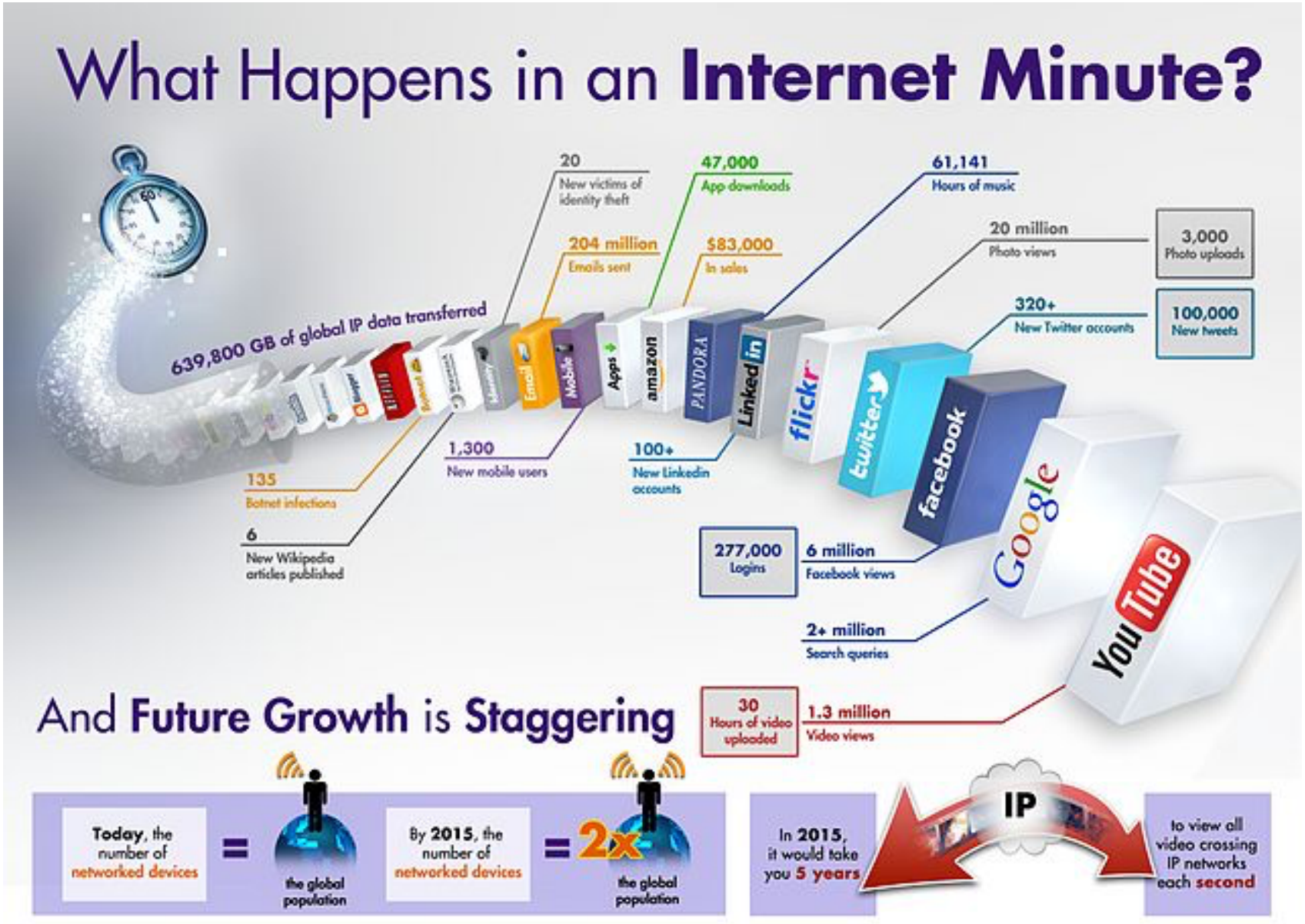
Your friends in this context are:

- Scripting languages
- Version control systems (e.g., git)
- Social coding sites (e.g., GitHub, Figshare)
- Markdown, LaTeX, RMarkdown, Jupyter Notebooks

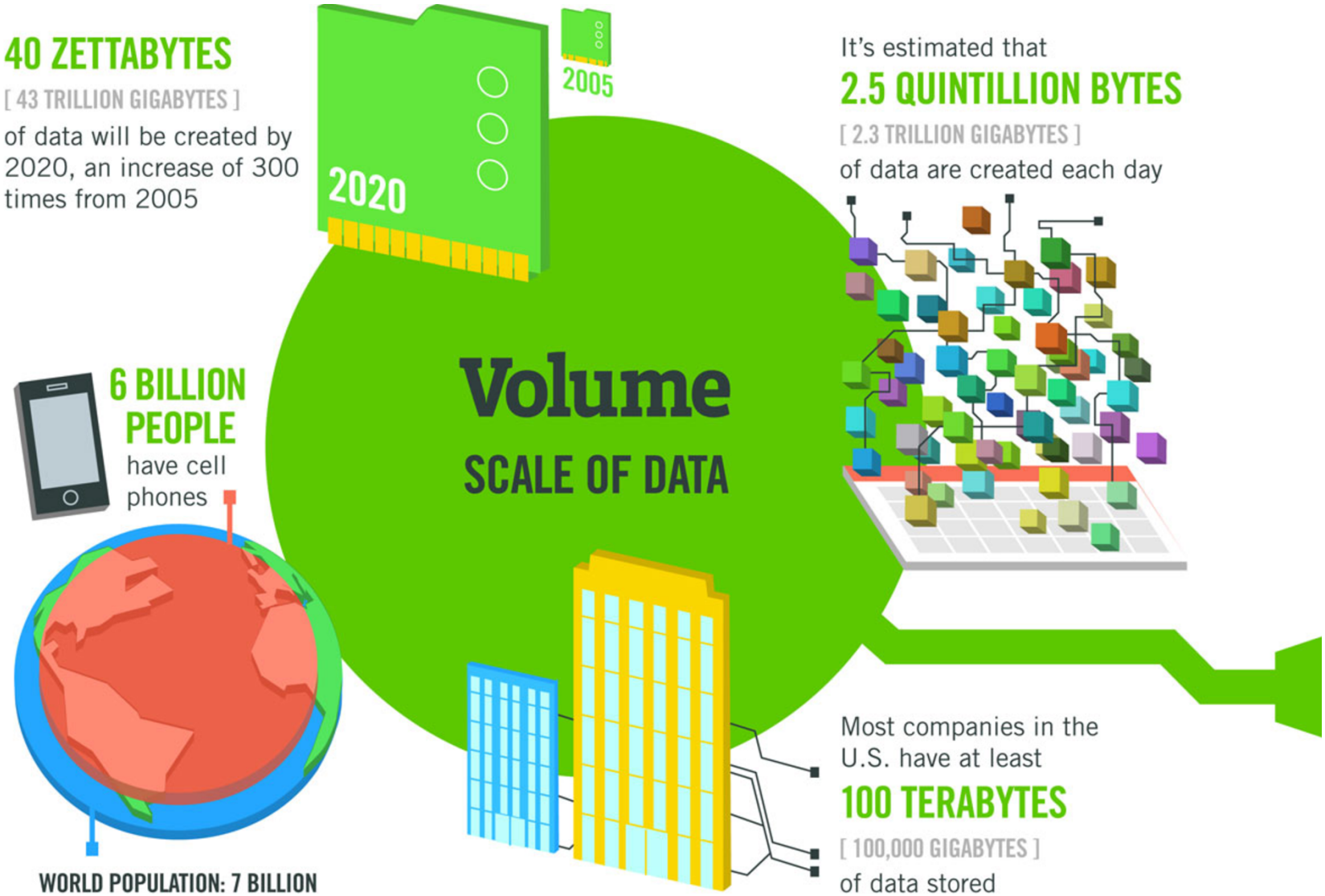
Big Data

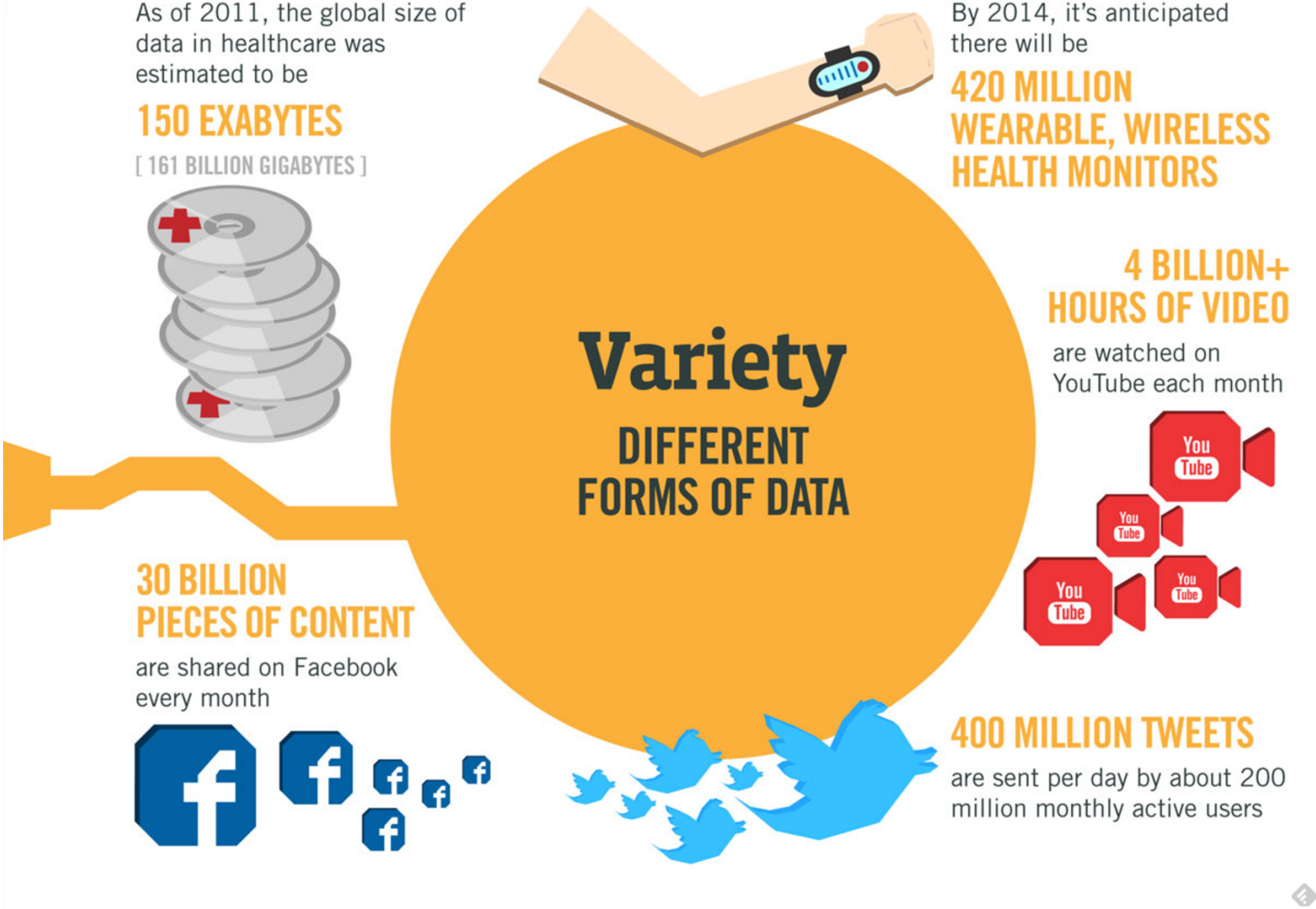
Data is created constantly, at ever increasing rates.

Examples?


What Happens in an Internet Minute?
639,800 GB of global IP data transferred
20
New victims of identity theft
204 million
Emails sent
47,000
App downloads
$83,000
In sales
61,141
Hours of music
20 million
Photo views
3,000
Photo uploads
320+
New Twitter accounts
100,000
New tweets
1,300
New mobile users
100+
New LinkedIn accounts
135
Botnet infections
6
New Wikipedia articles published
277,000
Logins
6 million
Facebook views
2+ million
Search queries
30
Hours of video uploaded
1.3 million
Video views
Email
Mobile
Apps
amazon
PANDORA
LinkedIn
flickr
twitter
facebook
Google
YouTube
And Future Growth is Staggering
Today, the number of networked devices
=
the global population
By 2015, the number of networked devices
= 2x
the global population
In 2015, it would take you 5 years
IP
to view all video crossing IP networks each second

cca. 2013

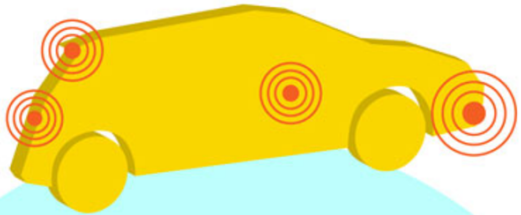

40 ZETTABYTES
[43 TRILLION GIGABYTES]
of data will be created by 2020, an increase of 300 times from 2005
2020
2005
It's estimated that
2.5 QUINTILLION BYTES
[2.3 TRILLION GIGABYTES]
of data are created each day
Volume
SCALE OF DATA
6 BILLION PEOPLE
have cell phones
WORLD POPULATION: 7 BILLION
Most companies in the U.S. have at least
100 TERABYTES
[100,000 GIGABYTES]
of data stored


As of 2011, the global size of data in healthcare was estimated to be
150 EXABYTES
[161 BILLION GIGABYTES]
By 2014, it's anticipated there will be
420 MILLION WEARABLE, WIRELESS HEALTH MONITORS
Variety
DIFFERENT FORMS OF DATA
4 BILLION+ HOURS OF VIDEO
are watched on YouTube each month
30 BILLION PIECES OF CONTENT
are shared on Facebook every month
400 MILLION TWEETS
are sent per day by about 200 million monthly active users

The New York Stock Exchange captures

1 TB OF TRADE INFORMATION

during each trading session



Modern cars have close to

100 SENSORS

that monitor items such as fuel level and tire pressure

Velocity

ANALYSIS OF STREAMING DATA

By 2016, it is projected there will be

18.9 BILLION NETWORK CONNECTIONS

– almost 2.5 connections per person on earth

Big Data - Definition

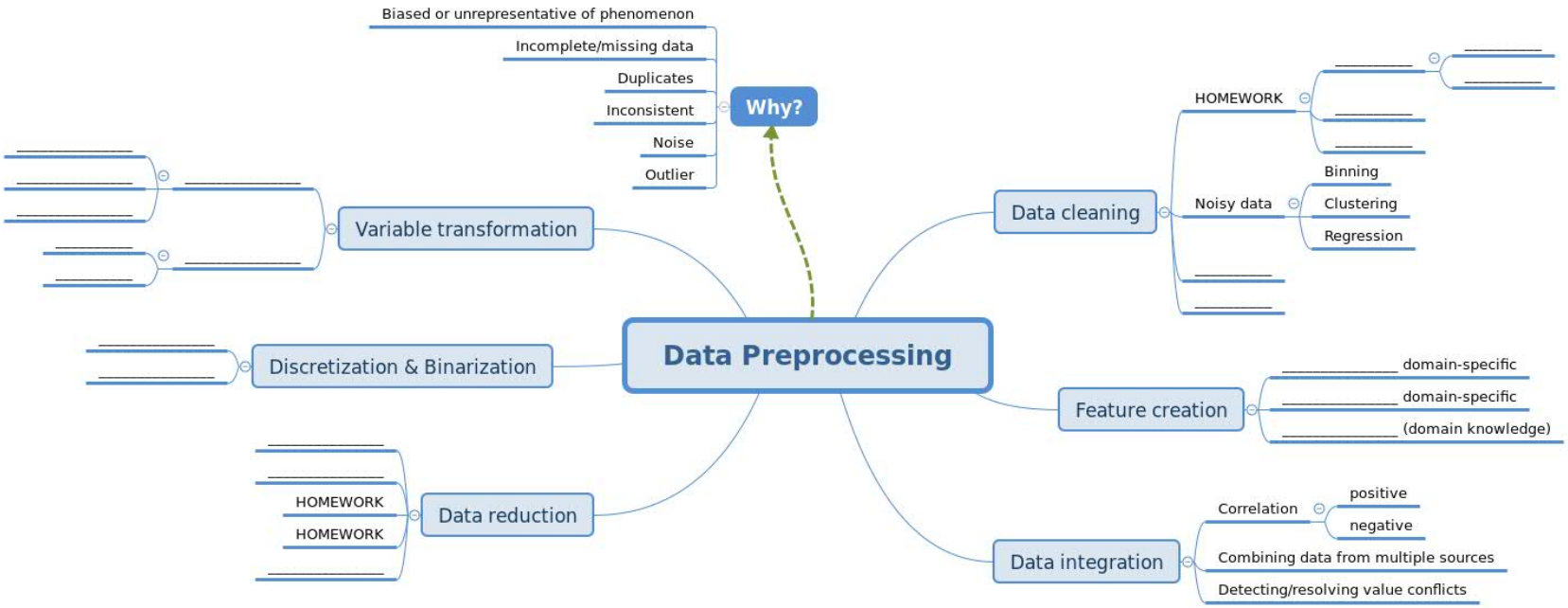
"Big Data is data whose **scale**, distribution, **diversity**, and/or **timeliness** require the use of **new technical architectures** and analytics to enable insights that unlock new sources of business value."

(McKinsey & Co.; Big Data: The Next Frontier for Innovation, Competition, and Productivity)

http://www.rug.nl/ocasys/fwn/vak/show?code=WMCS16003

Preprocessing I

Preprocessing

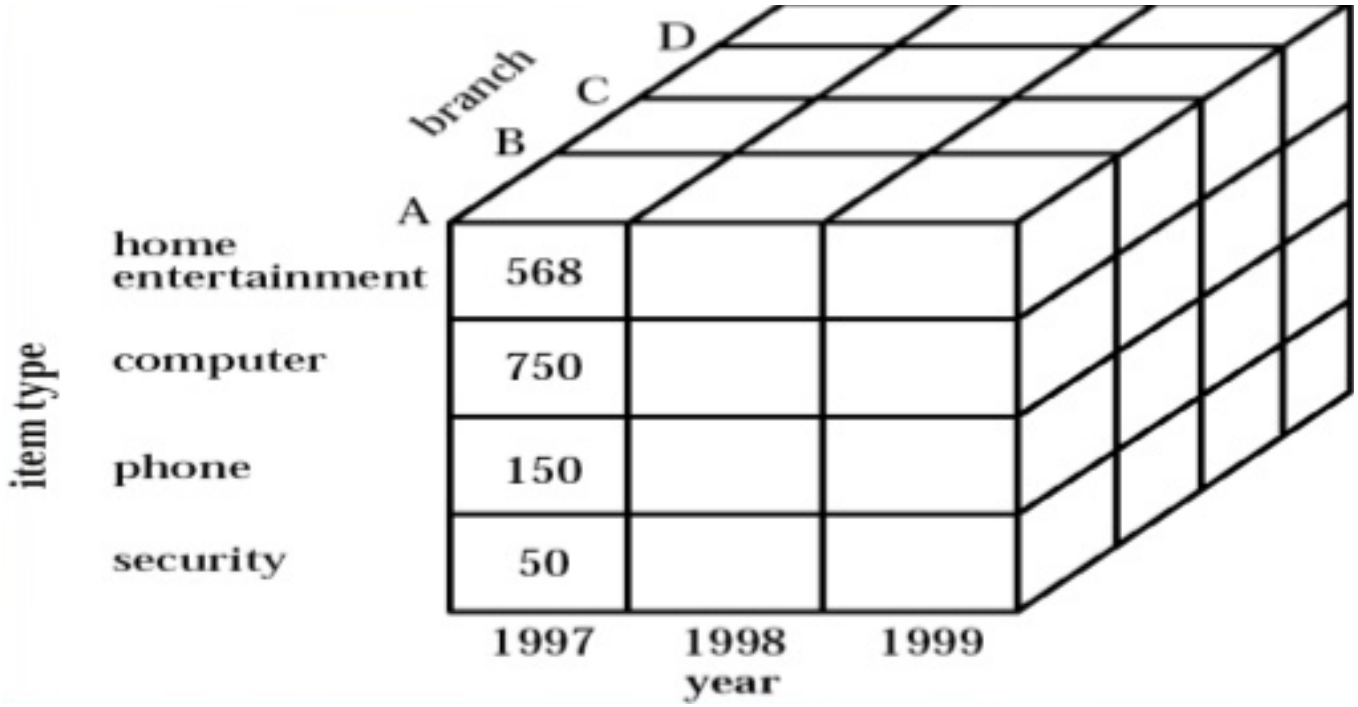


Preprocessing

- Data Cube Aggregation (combining objects into single), high-level vs. low-level, monthly vs. daily

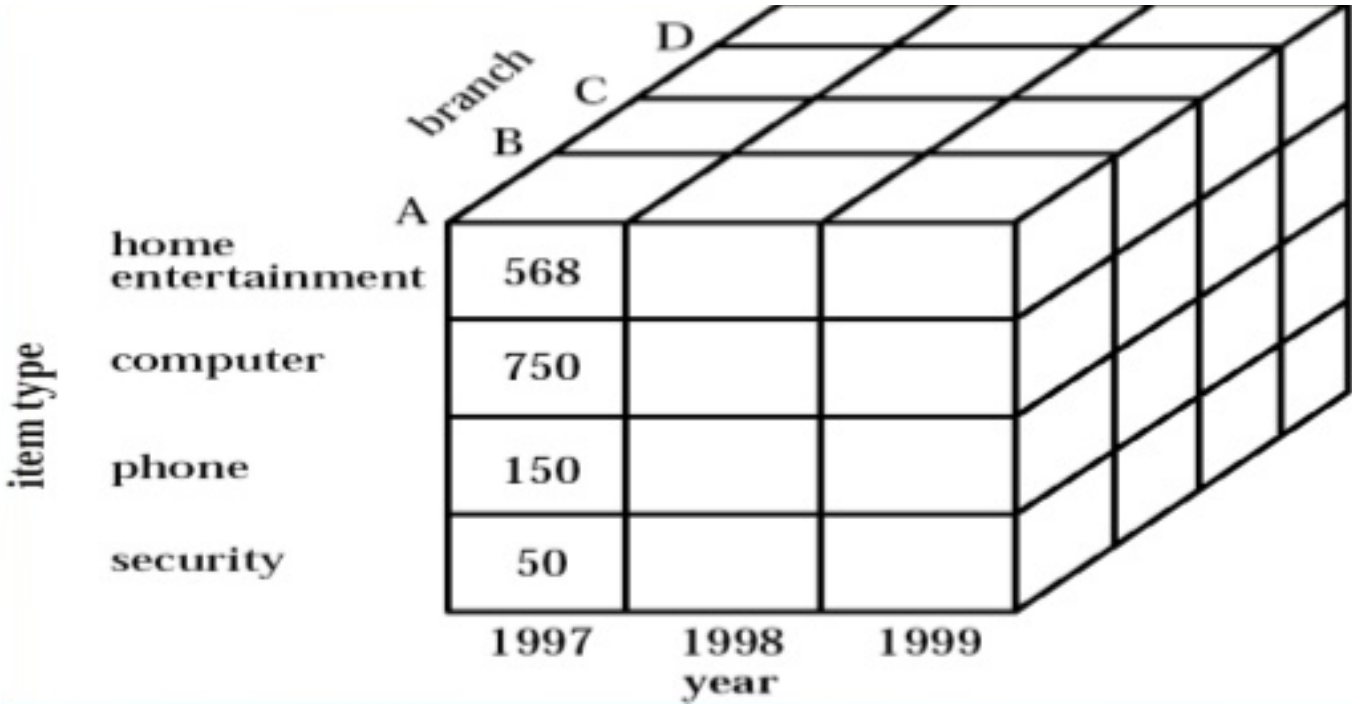
Preprocessing

- Data Cube Aggregation (combining objects into single), high-level vs. low-level, monthly vs. daily



Preprocessing

- Data Cube Aggregation (combining objects into single), high-level vs. low-level, monthly vs. daily



- Missing values (part of the assignment)

Preprocessing: Feature Creation

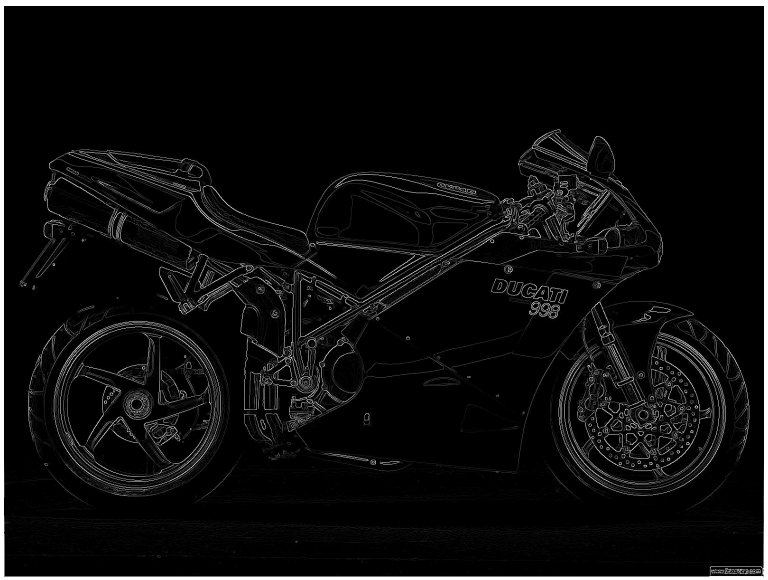
- Feature extraction[3]:
- Feature mapping:
- Feature construction:

[3] More about that in "Pattern Recognition", semester I b

Preprocessing: Feature Creation

- Feature extraction[3]:

new feature set extracted from raw (higher-level,e.g. for image filters); domain-specific



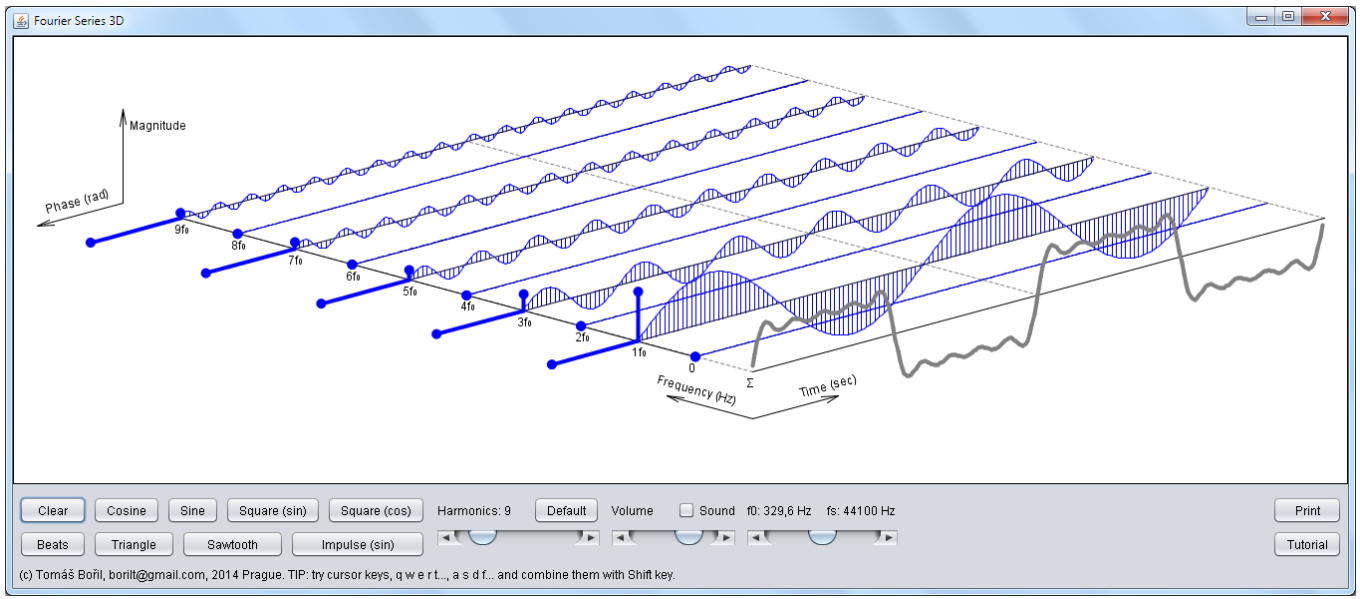
- Feature mapping:
- Feature construction:

[3]More about that in "Pattern Recognition", semester I b

Preprocessing: Feature Creation

- Feature extraction[3]:
- Feature mapping:

different view of the data (e.g. Fourier transform for time series to represent frequency information); domain-specific



- Feature construction:

[3] More about that in "Pattern Recognition", semester I b

Preprocessing: Feature Creation

- Feature extraction[3]:
- Feature mapping:
- Feature construction:

information in form not suitable for further analysis
(new features more useful,
different studies $\Rightarrow$ assumed knowledge)

knowComp	knowArti	knowAstro	knowMath
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
FALSE	TRUE	FALSE	FALSE
FALSE	FALSE	FALSE	TRUE
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
FALSE	FALSE	TRUE	FALSE
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
FALSE	TRUE	FALSE	FALSE
FALSE	TRUE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE
TRUE	FALSE	FALSE	FALSE

[3] More about that in "Pattern Recognition", semester I b

Preprocessing

Measurement scale:

- Discrete: finate or countably infinite set of values (categorical, numeric, or counts, $\mathbb{N}$)
- Continuous: real numbers (only limited by precision measureable, $\mathbb{R}$)

Theoretically: every measure scale can combine with any data type

Preprocessing

Measurement scale:

- Discrete: finate or countably infinite set of values (categorical, numeric, or counts, $\mathbb{N}$)
- Continuous: real numbers (only limited by precision measureable, $\mathbb{R}$)

Theoretically: every measure scale can combine with any data type

$\Rightarrow$ though binary continuous not much sense

$\Rightarrow$ interval/ratio usually continuous (exception: counting attributes)

Preprocessing

Discretization and Binaryzation (best approach: the one producing the best result in subsequent data analysis)

Categoricals:

- into integer $[0, m-1]$

Categorical Value	Integer Value
awful	0
poor	1
OK	2
good	3
great	4

Preprocessing

Discretization and Binaryzation (best approach: the one producing the best result in subsequent data analysis)

Categoricals:

- into integer $[0, m-1]$
- m integers to binary $n = \lceil \log_2(m) \rceil$

Categorical Value	x_1	x_2	x_3
awful	0	0	0
poor	0	0	1
OK	0	1	0
good	0	1	1
great	1	0	0

Preprocessing

Discretization and Binaryzation (best approach: the one producing the best result in subsequent data analysis)

Categorical Value	x_1	x_2	x_3	x_4	x_5
awful	1	0	0	0	0
poor	0	1	0	0	0
OK	0	0	1	0	0
good	0	0	0	1	0
great	0	0	0	0	1

Categoricals:

- into integer $[0, m-1]$
- m integers to binary $n = \lceil \log_2(m) \rceil$
- or into asymmetric binaries, presence of 1 is important (M/F?)

Preprocessing

Discretization and Binaryzation (best approach: the one producing the best result in subsequent data analysis)

Continuous attribute:

- n intervals $\Rightarrow$ $n-1$ split points (how many and where place them?)
  $\rightarrow$ set of intervals $\{(x_0, x_1], (x_1, x_2], \ldots, (x_{n-1}, x_n)\}$ (R:cut)
  or set of inequalities $x_0 < x \leq x_1, \ldots, x_{n-1} < x < x_n$

Preprocessing

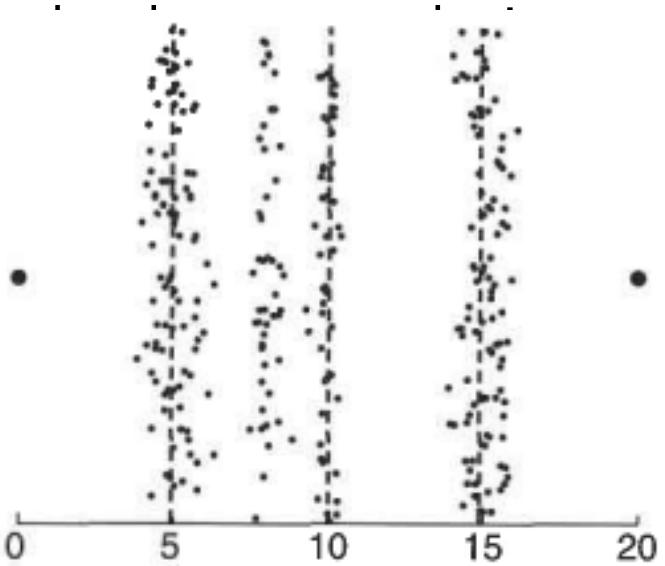
Discretization and Binaryzation (best approach: the one producing the best result in subsequent data analysis)

Continuous attribute:

- n intervals $\Rightarrow n-1$ split points (how many and where place them?)
  $\rightarrow$ set of intervals $\{(x_0, x_1], (x_1, x_2], \ldots, (x_{n-1}, x_n)\}$ (R:cut)
  or set of inequalities $x_0 < x \leq x_1, \ldots, x_{n-1} < x < x_n$
- unsupervised: (simple approaches common)

Preprocessing

Discretization and Binaryzation (best appro
the best result in subsequent data analysis)

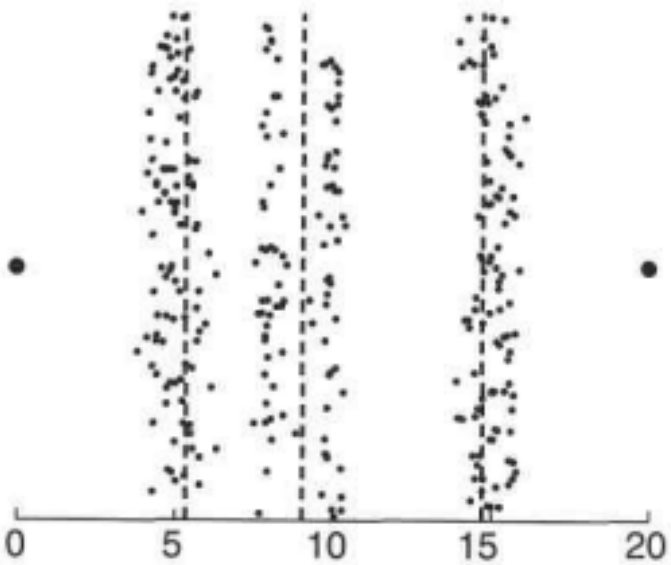


Continuous attribute:

- n intervals $\Rightarrow n-1$ split points (how ma
  $\rightarrow$ set of intervals $\{(x_0, x_1], (x_1, x_2], \ldots$
  or set of inequalities $x_0 < x \leq x_1, \ldots, x$
- unsupervised: (simple approaches common)
  - equal width: specified no. same width intervals (badly affected by outliers)

Preprocessing

Discretization and Binaryzation (best appro
the best result in subsequent data analysis)

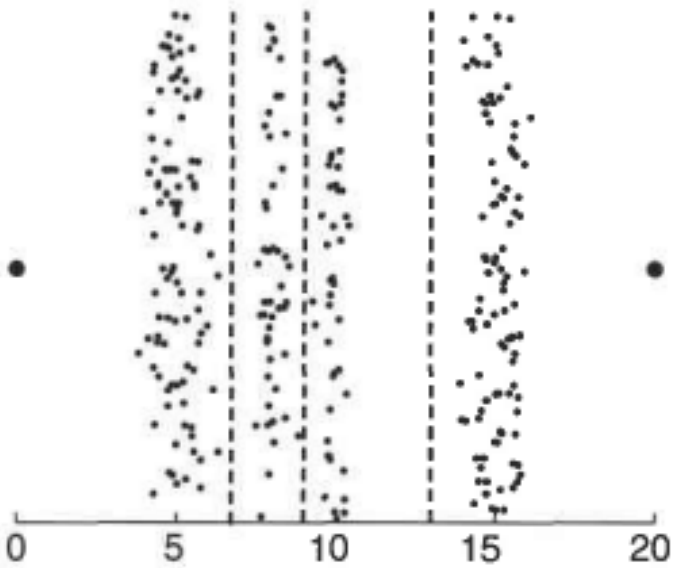


Continuous attribute:

- n intervals $\Rightarrow$ $n-1$ split points (how ma
  $\rightarrow$ set of intervals $\{(x_0, x_1], (x_1, x_2], \ldots$
  or set of inequalities $x_0 < x \leq x_1, \ldots, x$
- unsupervised: (simple approaches common)
  - equal width: specified no. same width intervals (badly affected by outliers)
  - equal frequency: samve no. objects into each interval

Preprocessing

Discretization and Binaryzation (best appro
the best result in subsequent data analysis)



Continuous attribute:

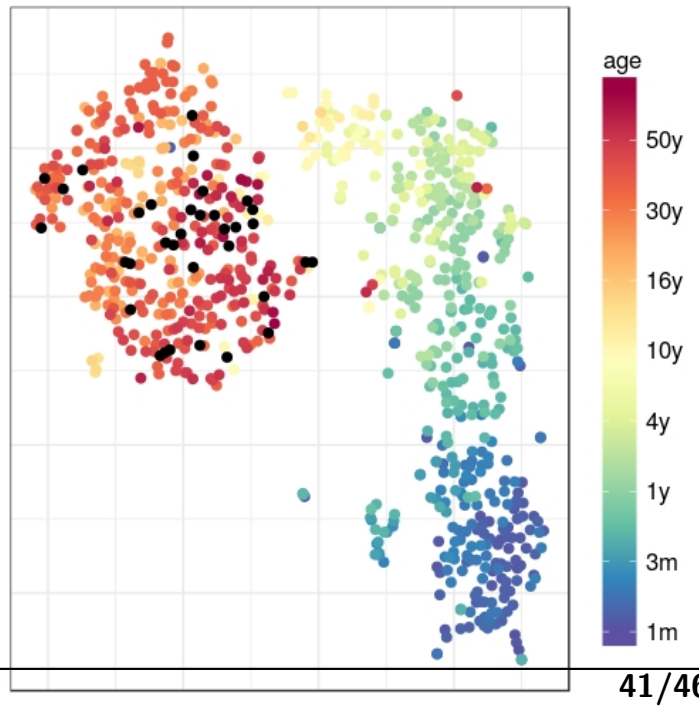
- n intervals $\Rightarrow$ $n-1$ split points (how ma
  $\rightarrow$ set of intervals $\{(x_0, x_1], (x_1, x_2], \ldots$
  or set of inequalities $x_0 < x \leq x_1, \ldots, x$
- unsupervised: (simple approaches common)
  - equal width: specified no. same width intervals (badly affected by outliers)
  - equal frequency: samve no. objects into each interval
  - clustering

Preprocessing

Discretization and Binaryzation (best approach: the one producing the best result in subsequent data analysis)

Continuous attribute:

- n intervals $\Rightarrow$ $n-1$ split points (how many and where place them?)
  $\rightarrow$ set of intervals $\{(x_0, x_1], (x_1, x_2], \ldots, (x_{n-1}, x_n)\}$ (R:cut)
  or set of inequalities $x_0 < x \leq x_1, \ldots, x_{n-1} < x < x_n$
- supervised: (use label information)
  - maximize purity of intervals:



Preprocessing

Discretization and Binaryzation (best approach: the one producing the best result in subsequent data analysis)

Continuous attribute:

- n intervals $\Rightarrow n-1$ split points (how many and where place them?)
  $\rightarrow$ set of intervals $\{(x_0, x_1], (x_1, x_2], \ldots, (x_{n-1}, x_n)\}$ (R:cut)
  or set of inequalities $x_0 < x \leq x_1, \ldots, x_{n-1} < x < x_n$
- supervised: (use label information)
  - maximize purity of intervals:
  - Entropy based: $p_{ij} = \frac{m_{ij}}{m_i}$ prob. of class j in interval i

    $e_i = \sum_{i=1}^{k} p_{ij} \log_2 p_{ij} \begin{cases} 0 & \text{only values from one class} \\ \max & \text{every class equally often} \end{cases}$

  strategy: bisecting such that interval give min entropy, continue until stopping criteria

Preprocessing

Normalization or Standardization

(often used interchangeably, don't confuse with making a variable normal!)

$\rightarrow$ make entire set of values having particular property

Preprocessing

Normalization or Standardization

(often used interchangeably, don't confuse with making a variable normal!)

$\rightarrow$ make entire set of values having particular property

- variable centering: extract mean $\overline{x}$

Preprocessing

Normalization or Standardization

(often used interchangeably, don't confuse with making a variable normal!)

$\rightarrow$ make entire set of values having particular property

- variable centering: extract mean $\overline{x}$
- z-score: $x' = (x - \overline{x})/s_x \rightarrow$ 0 mean unit standard deviation

  avoid variable with large values dominating results of calculation

Preprocessing

Normalization or Standardization

(often used interchangeably, don't confuse with making a variable normal!)

$\rightarrow$ make entire set of values having particular property

- variable centering: extract mean $\overline{x}$
- z-score: $x' = (x - \overline{x})/s_x \rightarrow$ 0 mean unit standard deviation
  avoid variable with large values dominating results of calculation
- affected by outliers: mean replaced by median and std by absolute std $\sigma_A = \sum_{i=1}^{m} |x_i - \mu|$

Preprocessing

Variable transformation (applied to all values of variable)
Functional transformation and normalization

- e.g. if only magnitude is important:

Preprocessing

Variable transformation (applied to all values of variable)
Functional transformation and normalization

- e.g. if only magnitude is important: absolute value $|x|$

Preprocessing

Variable transformation (applied to all values of variable)
Functional transformation and normalization

- e.g. if only magnitude is important: absolute value $|x|$
- simple function: $\frac{1}{x}, x^k, \log x, e^x, \sqrt{x}, \sin x$, etc.

Preprocessing

Variable transformation (applied to all values of variable)
Functional transformation and normalization

- e.g. if only magnitude is important: absolute value $|x|$
- simple function: $\frac{1}{x}, x^k, \log x, e^x, \sqrt{x}, \sin x$, etc.
- $\log x, \sqrt{x}, \frac{1}{x}$ often used to transform non-Gaussian data to look more Gaussian (although this is controversal)

Preprocessing

Variable transformation (applied to all values of variable)
Functional transformation and normalization

- e.g. if only magnitude is important: absolute value $|x|$
- simple function: $\frac{1}{x}, x^k, \log x, e^x, \sqrt{x}, \sin x$, etc.
- $\log x, \sqrt{x}, \frac{1}{x}$ often used to transform non-Gaussian data to look more Gaussian (although this is controversal)
- compress large ranges, e.g. log_{10} for byte ranges

Preprocessing

Variable transformation (applied to all values of variable)
Functional transformation and normalization

- e.g. if only magnitude is important: absolute value $|x|$
- simple function: $\frac{1}{x}, x^k, \log x, e^x, \sqrt{x}, \sin x$, etc.
- $\log x, \sqrt{x}, \frac{1}{x}$ often used to transform non-Gaussian data to look more Gaussian (although this is controversal)
- compress large ranges, e.g. log_{10} for byte ranges
- Caution: changes nature of data! e.g. $\frac{1}{x}$ reduces magnitude of values ≥ 1, but increases magnitude of values $\in [0, 1]$
  ($\{1, 2, 3\} \Rightarrow \{1, \frac{1}{2}, \frac{1}{3}\}$) and $\{1, \frac{1}{2}, \frac{1}{3}\} \Rightarrow \{1, 2, 3\} \rightarrow$ order reversed!

Preprocessing

- Variable Selection (student assignments)

Preprocessing

- Sampling (student assignments)

Preprocessing

- Compression: typical signal processing task, encode information using fewer bits than original (lossless or lossy), e.g.

Preprocessing

- Compression: typical signal processing task, encode information using fewer bits than original (lossless or lossy), e.g.
  - ZIP, gzip
  - machine learning (dictionary learning, vector quantization)
  - and for multimedia (images, audio, video)

Image Credits:

- http://drewconway.com/zia/2013/3/26/the-data-science-venn-diagram
- http://www.ibmbigdatahub.com/infographic/four-vs-big-data

Content based on:

- Experience
- Introduction to Data Mining, Tan et al.
- Data Science and Big Data Analytics, EMC Education Services
- Jeff Leek, Data Analysis Slides (https://github.com/jtleek/dataanalysis)
- http://stattrek.com/statistics/measurement-scales.aspx?Tutorial=AP