What is Data?

Nicholas Mattei, *Tulane University*

CMPS3660 – Introduction to Data Science – Fall 2019

https://rebrand.ly/TUDataScience

Many Thanks
Slides based off Introduction to Data Science from John P. Dickerson - https://cmsc320.github.io/
Some examples taken from *Data Science* by John D. Kelleher and Brendan Tierney, MIT Press.

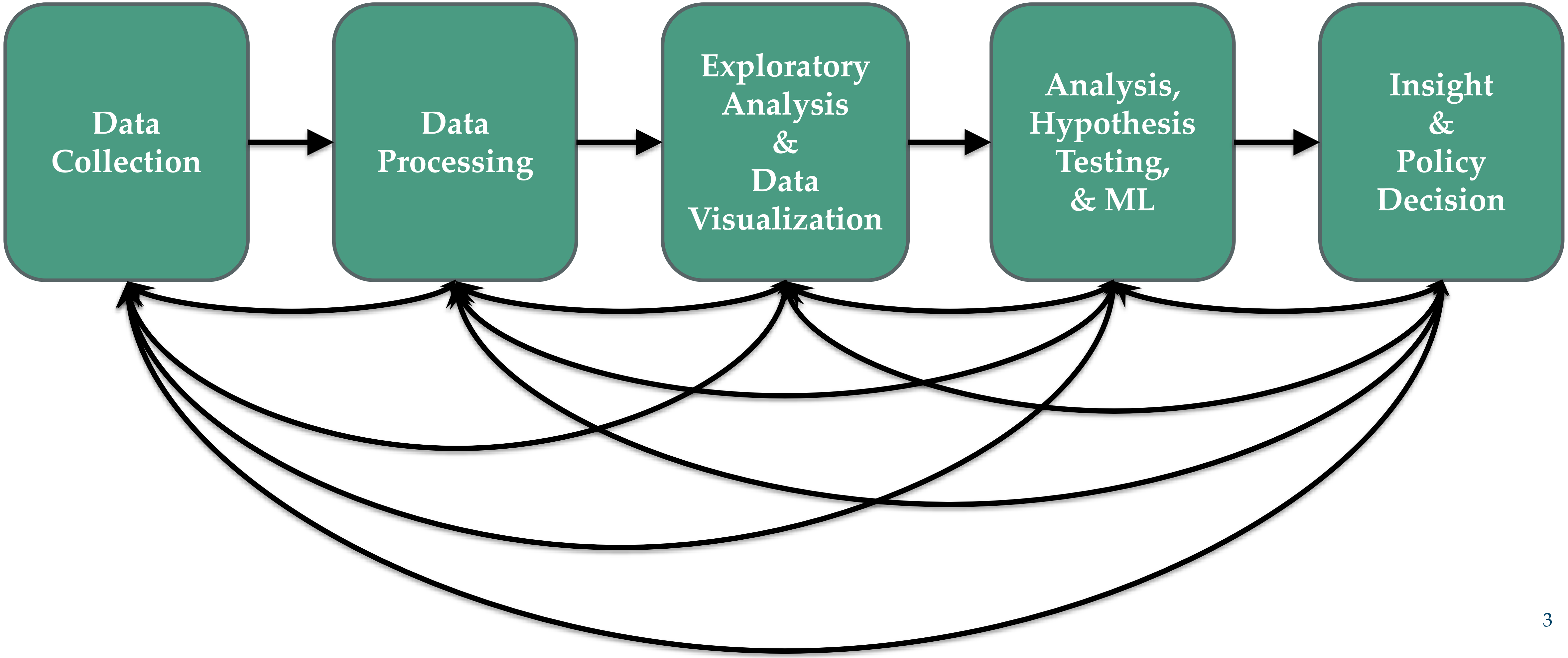
Announcements

- No office hours this Thursday!

- Lab day moved to Tuesday XXX
  - Make sure you can run Docker or Anaconda on your laptop.

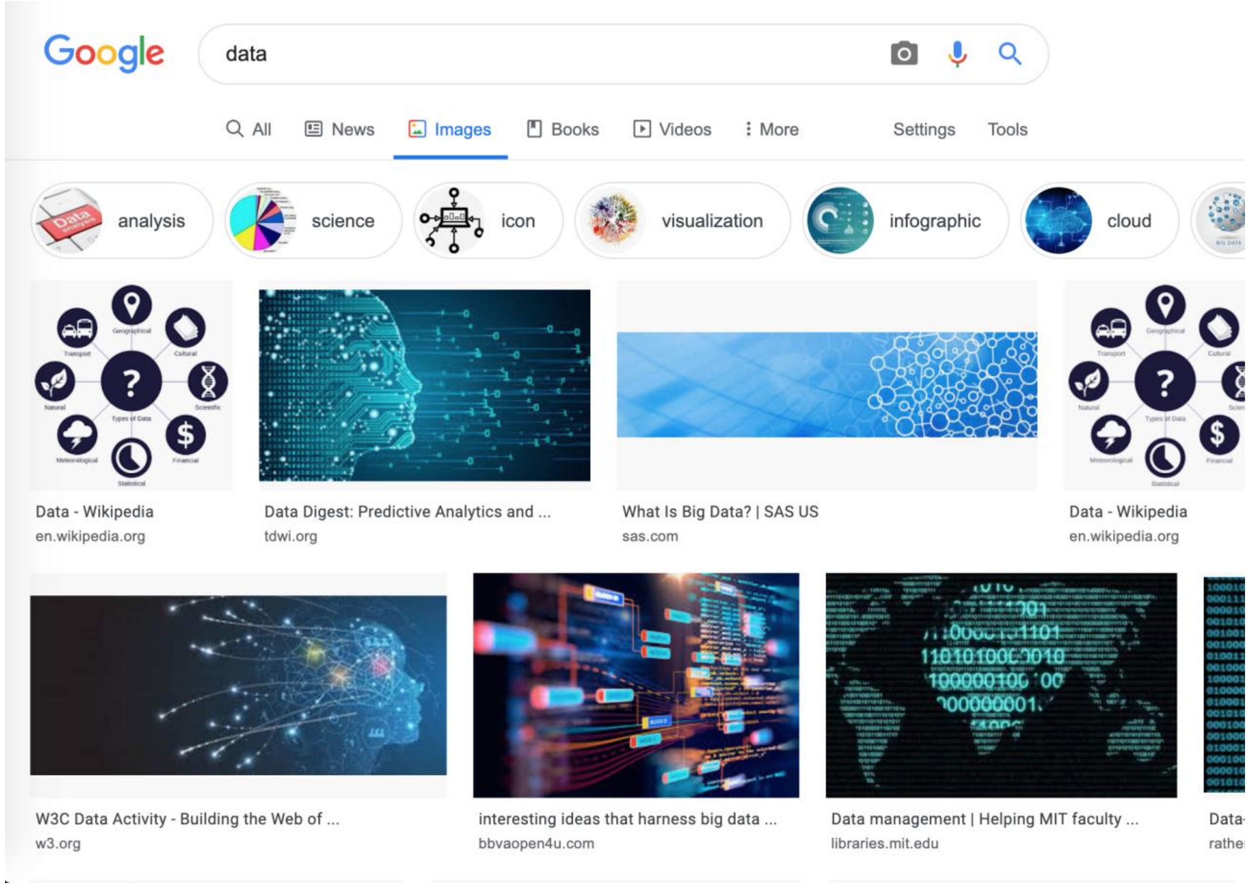
Today

The Data LifeCycle



What is Data?

- I'm going to give you a classical "statistical" overview of *tabular data*.
- As we go on I'll ask you think think more liberally about data but we gotta know our fundamentals first!



Tabular Data

- **Data is an abstraction of some real world *entity*.**
  – Sometimes also called: instance, example, record, object, case, individual.
- **Each of these entities is described by a set of features.**
  – Sometimes you'll see these called variables, features, attributes,
- **Data like this is typically processed into an *n (number of entities)* by *m (number of attributes)* matrix.**
  – Typically the result of merging and processing many different records!
  – Picking the data that goes into this table has both technical and ethical concerns (recall our examples of Target, Netflix…).

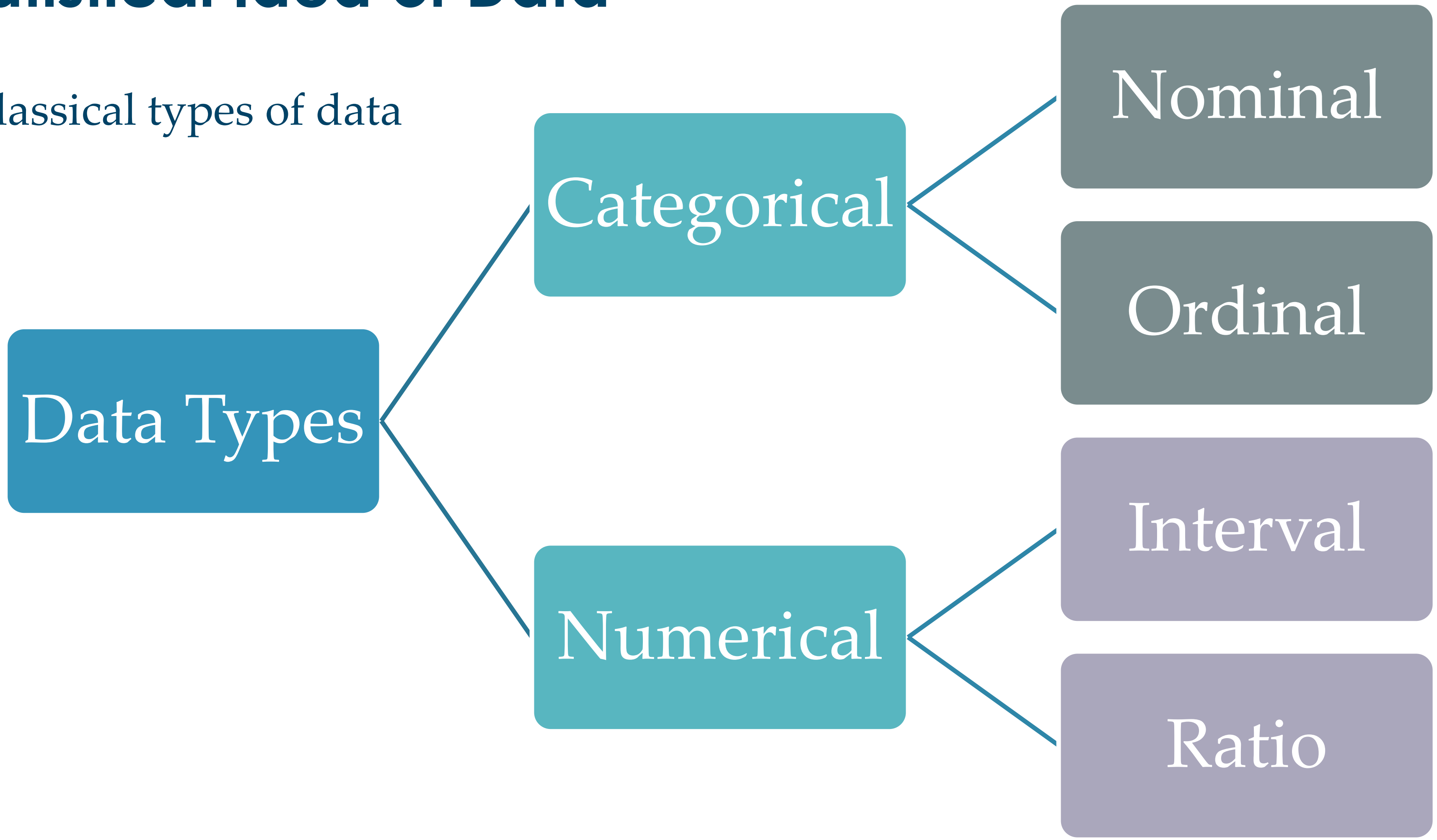
ID	Title	Author	Year	Cover	Edition	Price
1	Emma	Austen	1815	Paper	20th	$5.75
2	Dracula	Stoker	1897	Hard	15th	$12.00
3	Ivanhoe	Scott	1820	Hard	8th	$25.00
4	Kidnapped	Stevenson	1886	Paper	11th	$5.00

Data Collection

Data Processing

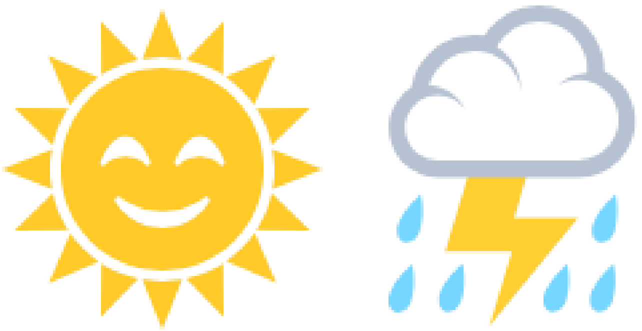
Classical Statistical Idea of Data

- There are four classical types of data



Categorical Data – Takes a Value From a Finite Set

- **Nominal (Categorical) Data.**
  - Values have *names* – describe the categories, classes, or states of things.
  - Marital status, beer type, or some binary attribute.
  - Cannot compare the values, hence we *cannot naturally order them* and we *cannot use arithmetic on them.*
- **Ordinal Data.**
  - Values have *names* – describe the categories, classes, or states of things.
  - However, there is an *ordering* over the values:
    - Strongly like, like, neutral, strongly dislike.
  - Lacks a mathematical notion of *distance between the values.*



- This distinction can be blurry…
  - Is there an ordering over: sunny, overcast, rainy?

Numerical Data - Measured Using Integers or Real Quantities.

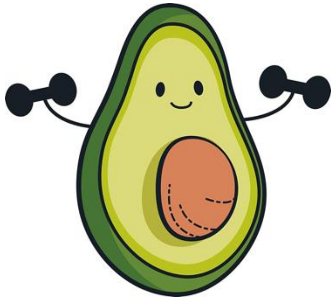
- **Interval Scale.**
  - Scale with fixed but arbitrary interval (e.g., dates).
  - The difference between two values is meaningful:
    - Difference between 9/1/2019 and 10/1/2019 is the same as the difference between 9/1/2018 and 9/1/2019.
  - **However:** we cannot compute ratios or scales: what unit is 9/1/2019 * 9/1/2018?

- **Ratio Scale**
  - All of the same properties as an inverval scale data, but the scale of measurement possesses a **true-zero origin.**
  - Can look at the *ratio* between two quantities (unlike an interval scale).
  - E.g., *zero money* is an absolute, one money is half as much as two money… etc.

Numerical Data - Examples

- **Temperatures:**
  - Celsius / Fahrenheit: interval or ratio scale ?????
  - **Interval:** 0C or 0F is not 0 heat but rather an arbitrary fixed point.
  - Hence, we can’t say that 30F is twice as warm as 15F.
- **Kelvin (K):** interval or ratio scale ????
  - **Ratio:** 0k is defined as zero heat (no molecular motion) hence a true fixed point.

- **Weight:**
  - **Grams:** interval or ratio scale ?????
  - **Ratio:** 0g serves as a fixed point, 4g is 2x 2g.



General Rules

OK to compute....	Nominal	Ordinal	Interval	Ratio
frequency distribution	?	?	?	?

Thanks to GraphPad

General Rules

OK to compute....	Nominal	Ordinal	Interval	Ratio
frequency distribution	Yes	Yes	Yes	Yes
median and percentiles	?	?	?	?

General Rules

OK to compute....	Nominal	Ordinal	Interval	Ratio
frequency distribution	Yes	Yes	Yes	Yes
median and percentiles	No	Yes	Yes	Yes
addition or subtraction	?	?	?	?

General Rules

OK to compute....	Nominal	Ordinal	Interval	Ratio
frequency distribution	Yes	Yes	Yes	Yes
median and percentiles	No	Yes	Yes	Yes
addition or subtraction	No	No	Yes	Yes
mean or standard deviation	?	?	?	?

General Rules

OK to compute....	Nominal	Ordinal	Interval	Ratio
frequency distribution	Yes	Yes	Yes	Yes
median and percentiles	No	Yes	Yes	Yes
addition or subtraction	No	No	Yes	Yes
mean or standard deviation	No	No	Yes	Yes
ratio, or coefficient of variation	?	?	?	?

Tulane University

General Rules

OK to compute....	Nominal	Ordinal	Interval	Ratio
frequency distribution	Yes	Yes	Yes	Yes
median and percentiles	No	Yes	Yes	Yes
addition or subtraction	No	No	Yes	Yes
mean or standard deviation	No	No	Yes	Yes
ratio, or coefficient of variation	No	No	No	Yes

Lots of fine grained rules…

- Nominal Data:
  - Frequencies, proportions, and percentages.. Histograms… etc.
- Continuous Data:
  - Also get to use standard deviation, mean, etc.
- Nice Summary: https://en.wikipedia.org/wiki/Statistical_data_type
- We'll discuss more as we get into the data!

Simple data types [edit]

The following table classifies the various simple data types, associated distributions, permissible operations, etc. Regardless of the logical possible values, all of these data types are generally coded using real numbers, because the theory of random variables often explicitly assumes that they hold real numbers.

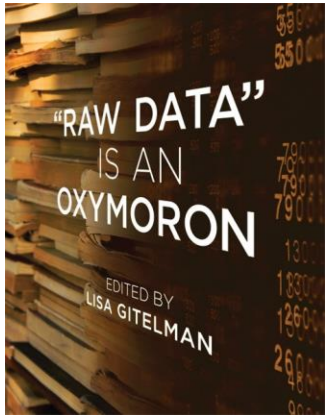
Data Type	Possible values	Example usage	Level of measurement	Distribution	Scale of relative differences	Permissible statistics	Regression analysis
binary	0, 1 (arbitrary labels)	binary outcome ("yes/no", "true/false", "success/failure", etc.)	nominal scale	Bernoulli	incomparable	mode, Chi-squared	logistic, probit
categorical	1, 2, ..., K (arbitrary labels)	categorical outcome (specific blood type, political party, word, etc.)		categorical			multinomial logit, multinomial probit
ordinal	integer or real number (arbitrary scale)	relative score, significant only for creating a ranking	ordinal scale	categorical	relative comparison		ordinal regression (ordered logit, ordered probit)
						mean,	

A Take Home Point About Data Ethics

- Always remember that data is generated by abstraction.
  - Someone picked what data to look at, how to count things, and what not to count.
  - How could this lead to problems?
  - What about when it changes?

- However, we can still do useful things
  - “A map is not the territory that it represents, but, if correct, it has a similar structure to the territory which accounts for its usefulness” -- Alfred Korzybski, *Science and Sanity*

- Never forget that there is no such thing as “raw data” and that data is never a purely objective description of the world.

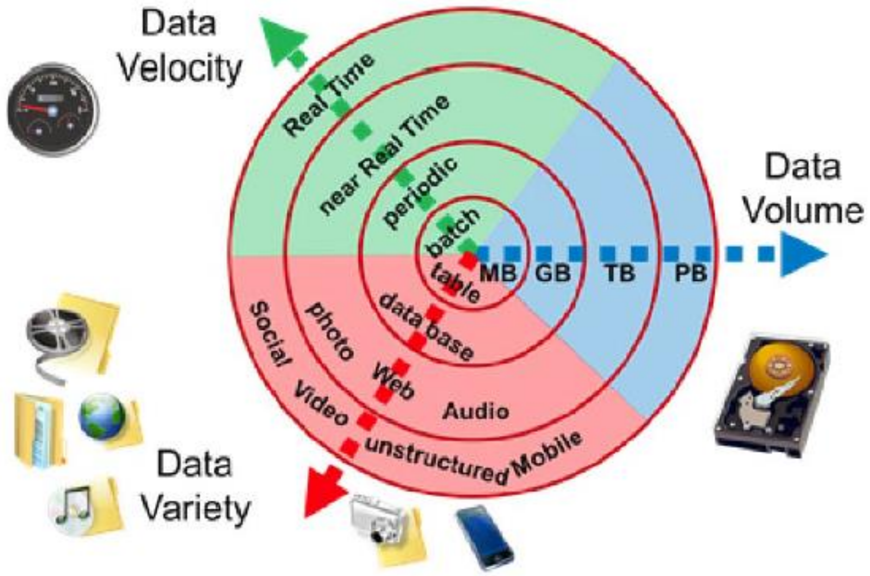


Data Manipulation and Computation

- Data Science == manipulating and computing on data
  - Large to very large, but somewhat "structured" data

- Wait, what about **BIG DATA?**
  - **Volume:** The quantity of generated and stored data. The size determines the value and potential insight.
  - **Variety:** The type and nature of the data. Big data draws from text, images, audio, video; plus it completes missing pieces through data fusion.
  - **Velocity:** The speed at which the data is generated and processed. Big data is often available in real-time.
    - Compared to small data, big data are produced more continually. Two kinds of velocity related to big data are the frequency of generation and the frequency of handling, recording, and publishing.

- Sometimes Veracity but we won't consider this..



Data Manipulation and Computation

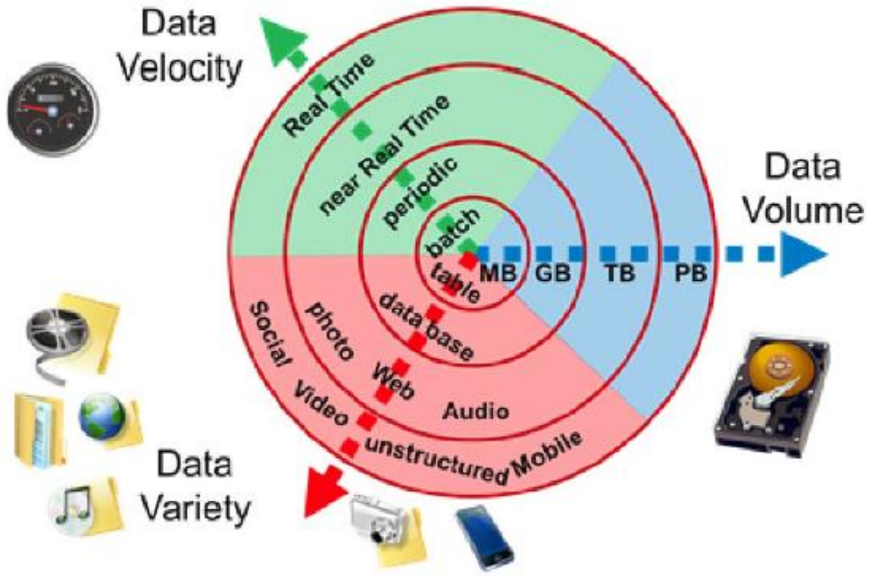
- Data Science == manipulating and computing on data
  - Large to very large, but somewhat “structured” data

- We will see several tools for doing that this semester
  - Thousands more out there that we won’t cover

- Need to learn to shift thinking from:
  - *Imperative code to manipulate data structures*

to:
  - *Sequences/pipelines of operations on data*

- Should still know how to implement the operations themselves, especially for debugging performance (covered in classes like Machine Learning), but we won’t cover that much formal algorithmic treatment.



Data Manipulation and Computation

1. **Data Representation ==** what is the natural way to think about given data

One-dimensional Arrays, Vectors

0.1	2	3.2	6.5	3.4	4.1

“data”	”representation”	”i.e.”

Indexing
Slicing/subsetting
Filter
‘map’ → apply a function to every element
’reduce/aggregate’ → combine values to get a single scalar (e.g., sum, median)

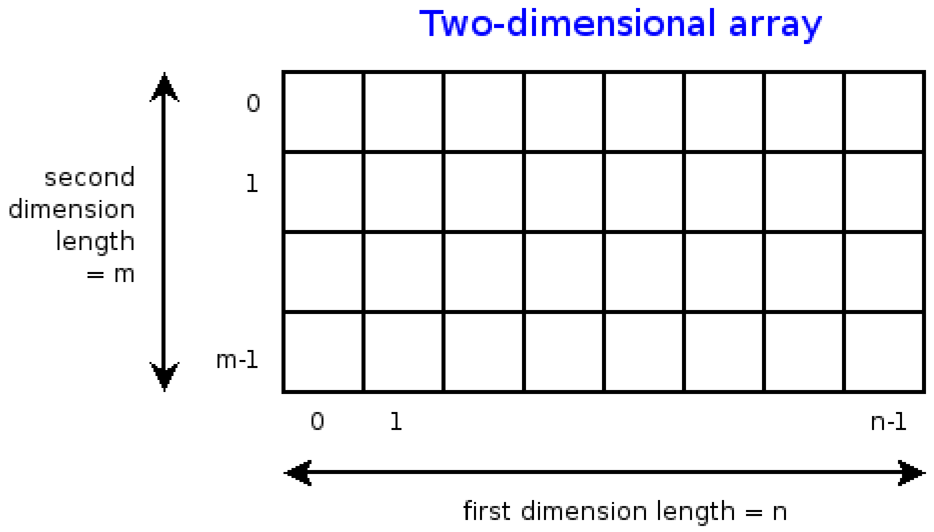
Given two vectors: **Dot and cross products**

2. **Data Processing Operations ==** take one or more datasets as input and produce one or more datasets as output

Data Manipulation and Computation

1. **Data Representation ==** what is the natural way to think about given data

n-dimensional arrays



Indexing
Slicing/subsetting
Filter
'map' → apply a function to every element
'reduce/aggregate' → combine values across a row or a column (e.g., sum, average, median etc..)

2. **Data Processing Operations ==** take one or more datasets as input and produce one or more datasets as output

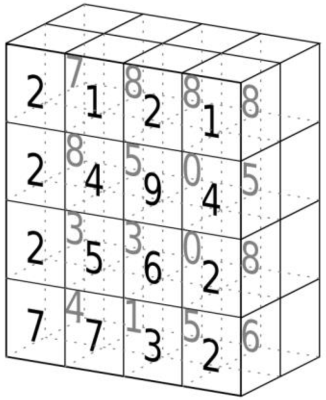
Data Manipulation and Computation

1. **Data Representation ==** what is the natural way to think about given data

Matrices, Tensors

3	1	4	1
5	9	2	6
5	3	5	8
9	7	9	3
2	3	8	4
6	2	6	4

tensor of dimensions [6,4]
(matrix 6 by 4)



tensor of dimensions [4,4,2]

n-dimensional array operations
+
Linear Algebra
- **Matrix/tensor multiplication**
- **Transpose**
- **Matrix-vector multiplication**
- **Matrix factorization**

2. **Data Processing Operations ==** take one or more datasets as input and produce one or more datasets as output

Data Manipulation and Computation

1. **Data Representation ==** what is the natural way to think about given data

Sets: of Objects

Sets: of (Key, Value Pairs)

(cs@tulane,(email1, email2,…))

(nsmattei@tulane.edu,(email3, email4,…))

Filter
Map
Union

Reduce/Aggregate

Given two sets, **Combine/Join** using "keys"

Group and then aggregate

2. **Data Processing Operations ==** take one or more datasets as input and produce one or more datasets as output

Data Manipulation and Computation

1. **Data Representation ==** what is the natural way to think about given data

Tables/Relations == Sets of Tuples

company	division	sector	tryint
00nil_Combined_Company	00nil_Combined_Division	00nil_Combined_Sector	14625
apple	00nil_Combined_Division	00nil_Combined_Sector	10125
apple	hardware	00nil_Combined_Sector	4500
apple	hardware	business	1350
apple	hardware	consumer	3150
apple	software	00nil_Combined_Sector	5625
apple	software	business	4950
apple	software	consumer	675
microsoft	00nil_Combined_Division	00nil_Combined_Sector	4500
microsoft	hardware	00nil_Combined_Sector	1890
microsoft	hardware	business	855
microsoft	hardware	consumer	1035
microsoft	software	00nil_Combined_Sector	2610
microsoft	software	business	1215
microsoft	software	consumer	1395

Filter rows or columns

"Join" two or more relations

"Group" and "aggregate" them

Relational Algebra formalizes some of them

Structured Query Language (SQL)

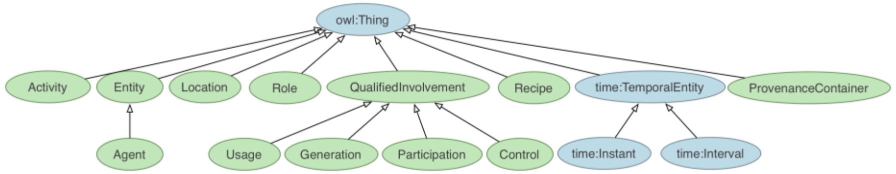
Many other languages and constructs, that look very similar

2. **Data Processing Operations ==** take one or more datasets as input and produce one or more datasets as output

Data Manipulation and Computation

1. **Data Representation ==** what is the natural way to think about given data

Hierarchies/Trees/Graphs



"Path" queries

Graph Algorithms and Transformations

Network Science

Somewhat more ad hoc and special-purpose

Changing in recent years

2. **Data Processing Operations ==** take one or more datasets as input and produce one or more datasets as output

Data Manipulation and Computation

1. **Data Representation ==** what is the natural way to think about given data.
2. **Data Processing Operations ==** take one or more datasets as input and produce one or more datasets as output.

- Why?
  - Allows one to think at a higher level of abstraction, leading to simpler and easier-to-understand scripts.
  - Provides "independence" between the abstract operations and concrete implementation.
  - Can switch from one implementation to another easily.

- For performance debugging, useful to know how they are implemented and rough characteristics

Next Few Classes

1. NumPy: Python Library for Manipulating nD Arrays
   Multidimensional Arrays, and a variety of operations including Linear Algebra

2. Pandas: Python Library for Manipulating Tabular Data
   Series, Tables (also called **DataFrames**)
   Many operations to manipulate and combine tables/series

3. Relational Databases
   Tables/Relations, and SQL (similar to Pandas operations)

   Other tools like Git and Docker!