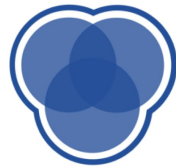# INFO 1998: Introduction to Machine Learning

CDS Education

We explore, learn, and educate big minds.

Lecture 2: Data Manipulation

INFO 1998: Introduction to Machine Learning

CDS Education

We explore, learn, and educate big minds.

Agenda

1. Define Good Question + Get Raw Data
2. Data Manipulation Techniques
3. Data Imputation
4. Other Techniques
5. Demo + Summary

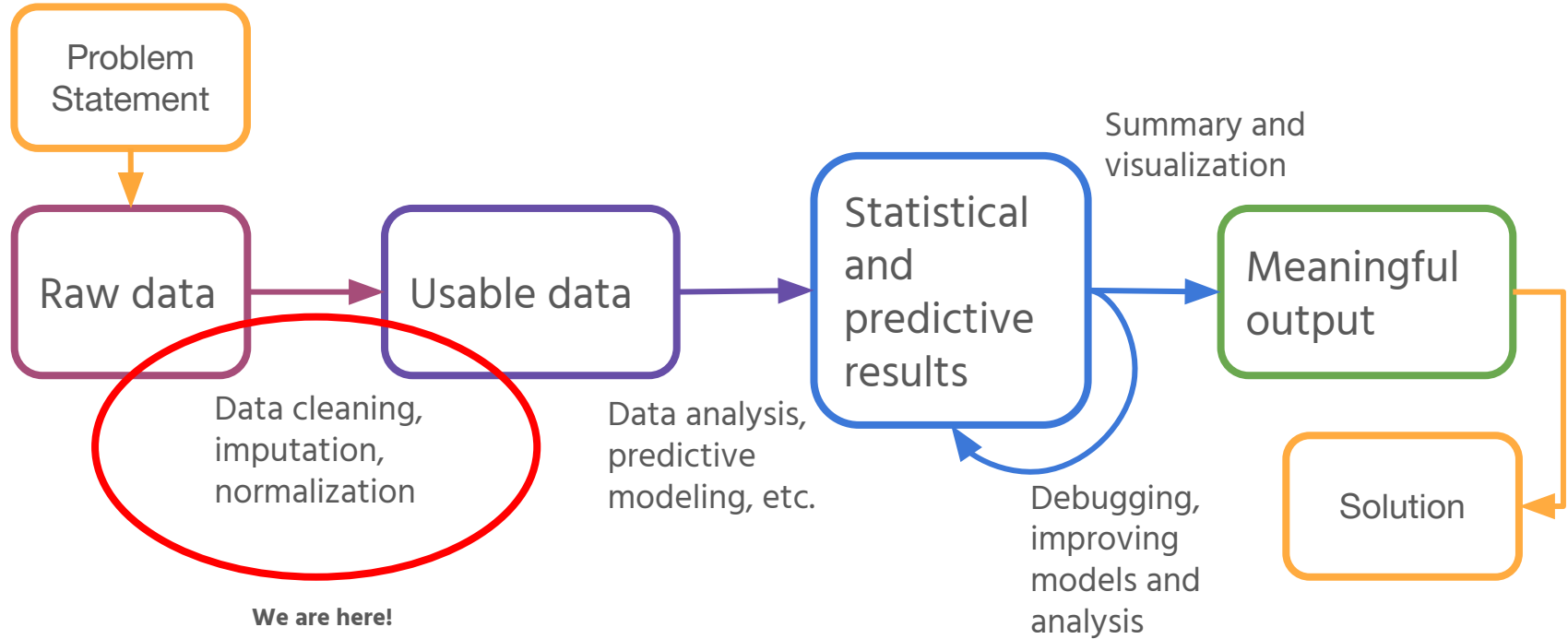# Survey Time

Creating A Good Question

Good Examples:
- What work and lifestyle conditions greatly impact mental health, and in what way?
- Based on this data, what factors can be used to predict a candidate's success within a Canadian election?
- What features best predict the amount of solar radiation the Earth gets based on data collected by NASA?

Poor Examples:
- What can the data tell me about mental health?
- Is there a relationship between the data and a candidate's success in a Canadian election?
- Can we predict amount of solar radiation the earth gets?

The Data Pipeline

Problem Statement

Raw data

Usable data

Statistical and predictive results

Meaningful output

Summary and visualization

Data cleaning, imputation, normalization

Data analysis, predictive modeling, etc.

Debugging, improving models and analysis

Solution

We are here!

Acquiring data

- **Option 1:** Web scraping directly from web with tools like BeautifulSoup
- **Option 2:** Querying from databases
- **Option 3:** Downloading data directly (ex. from Kaggle/Inter-governmental organizations/Govt./Corporate websites)

...and more!

Finding a Relevant Dataset

Questions to Ask Yourself…

- Does the data measure what you care about?

- Is your data connected/related?

- Do you have a lot of data?

https://www.kaggle.com/datasets

How does input data usually look?

```
Timestamp,Class Year:,Major:,"On a scale 1 to 5 (1=unfamiliar, 5=proficient) , how well do you know Python?",How did you hear about this class?,"We will hold some
optional workshops to dive deeper into industry applications of advanced analytics, and any other topics that might be of interest to you (eg. Data Scraping).
What are some workshops you would like to attend? Anything goes.",What is a data problem that interests you the most?
2/9/20 0:26,2020,MBA,1,Referral by Friend,Tensorflow,A/B testing and setting up experiments
2/10/20 16:33,2023,Computer Science,1,In-class advertisement,"Website Analytics, Sentiment Analysis, Cleaning Data",How can we design efficient metrics to gauge
performance of any type of data?
2/11/20 8:26,2022,MechE,1,In-class advertisement,,I would like to know more about how computational methods are used in engineering or physics researchs.
2/11/20 22:43,2023,ILR,1,Referral by Friend,,The ethics behind data sharing and privacy laws online
2/12/20 17:41,2023,Food Science,1,Referral by Friend,"artificial intelligence
human behavior
```

	Timestamp	Class Year:	Major:	On a scale 1 to 5 (1=unfamiliar, 5=proficient) , how well do you know Python?	How did you hear about this class?	We will hold some optional workshops to dive deeper into industry applications of advanced analytics, and any other topics that might be of interest to you (eg. Data Scraping). What are some workshops you would like to attend? Anything goes.	What is a data problem that interests you the most?
0	2/9/20 0:26	2020	MBA	1	Referral by Friend	Tensorflow	A/B testing and setting up experiments
1	2/10/20 16:33	2023	Computer Science	1	In-class advertisement	Website Analytics, Sentiment Analysis, Cleanin...	How can we design efficient metrics to gauge p...
2	2/11/20 8:26	2022	MechE	1	In-class advertisement	NaN	I would like to know more about how computatio...
3	2/11/20 22:43	2023	ILR	1	Referral by Friend	NaN	The ethics behind data sharing and privacy law...
4	2/12/20 17:41	2023	Food Science	1	Referral by Friend	artificial intelligence \nhuman behavior\necon...	how to predict human behavior using internet d...
...	...	...	...	...	...	...	...

- Usually saved as .csv or .tsv files
- Known as **flat text files**, require parsers to load into code

However...

Most datasets are **messy**.

Datasets can be **huge**.

Datasets **may not make sense**.

Question

What are some ways in which data can be "*messy*"?
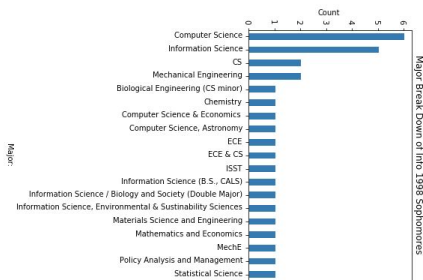
Examples of Drunk Data

From the onboarding form!

Example 1: Let's find CS majors in INFO 1998.

Different cases:

- Computer Science
- CS
- Cs
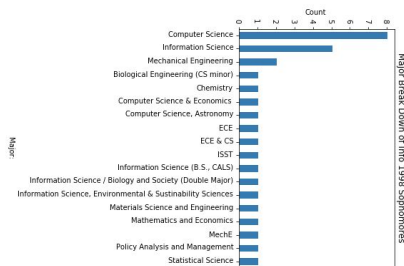- computer science
- CS and Math
- OR/CS

...goes on

Example 2: From INFO 1998 (Fall '18)

Answers for 'What Year Are You?'

- 1999
- 1st Master
- Junor
- INFO SCI

...goes on

Why we manipulate data?

Ease of Use	Prevent calculation errors	Improve memory efficiency

DataFrames!

- **Pandas** (a Python library) offers **DataFrame** objects to help manage data in an orderly way
- Similar to Excel spreadsheets or SQL tables
- DataFrames provides functions for selecting and manipulating data

import pandas as pd

Data Manipulation Techniques

- Filtering & Subsetting

- Concatenating

- Joining

- *Bonus*: Summarizing

Filtering vs. Subsetting

- Filters **rows**
- Focusing on data entries

- Subsets **columns**
- Focusing on characteristics

Name	Year	Major
Jerry	2023	CS
Kevin	2022	ORIE
Varun	2024	CS & ORIE
Sam	2022	ECE

Filtering

Name	Year	Major
Jerry	2023	CS
Kevin	2022	ORIE
Varun	2024	CS & ORIE
Sam	2022	ECE

Subsetting

Joining

Joins together two data frames on any specified key (fills in NaN otherwise). The index is the key here.

	Name
0	Jerry
1	Kevin
2	Varun
3	Vivian
4	Vincent

	Age	Major
0	20	CS
1	21	ORIE
2	19	CS & ORIE

	Name	Age	Major
0	Jerry	20	CS
1	Kevin	21	ORIE
2	Varun	19	CS & ORIE
3	Vivian	NaN	NaN
4	Vincent	NaN	NaN

```
DataFrame.join(other, on=None, how='left', lsuffix='', rsuffix='', sort=False)
```

Types of Joins

Concatenating

Combines together two data frames, either row-wise or column-wise

Name	Age	Major
Jerry	20	CS
Kevin	21	ORIE

Name	Age	Major
Varun	19	CS & ORIE
Sam	21	ECE

→ concat!

Name	Age	Major
Jerry	20	CS
Kevin	21	ORIE
Varun	19	CS & ORIE
Sam	21	ECE

```
pandas.concat(objs, axis=0, join='outer', ignore_index=False, keys=None,
levels=None, names=None, verify_integrity=False, sort=False, copy=True)
```

Bonus: Summarizing

- Gives a quantitative overview of the dataset
- Useful for understanding and exploring the dataset!

```
>>> s = pd.Series([1, 2, 3])
>>> s.describe()
count    3.0
mean     2.0
std      1.0
min      1.0
25%      1.5
50%      2.0
75%      2.5
max      3.0
dtype: float64
```
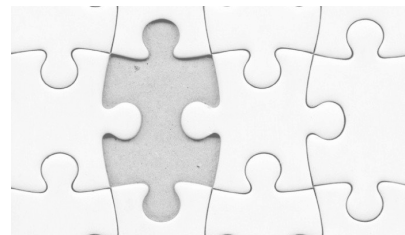
```
>>> s = pd.Series(['a', 'a', 'b', 'c'])
>>> s.describe()
count     4
unique    3
top       a
freq      2
dtype: object
```

Above: stats made easy

Dealing with missing data

Datasets are usually incomplete. We can solve this by:

Leaving out samples with missing data

Data imputation

Randomly Replacing NaNs

Using summary statistics

Using predictive models

1: Leaving out samples with missing values

- Option: Remove NaN values by removing specific samples or features
- **Beware** not to remove too many samples or features!
  - Information about the dataset is lost each time you do this

Question: How much is too much?

2: Data Imputation

3 main techniques to impute data:

1. Randomly replacing NaNs
2. Using summary statistics
3. Using regression, clustering, and other advanced techniques

2.1: Randomly replacing NaNs

- **This is not good - don't do it**

- Replacing NaNs with random values adds unwanted and unstructured noise

2.2: Using summary statistics

non-categorical data

- Works well with small datasets
- Fast and simple
- Does not account for correlations & uncertainties
- Usually does **not** work on categorical features

categorical data

- Using mode works with categorical data (only theoretical)
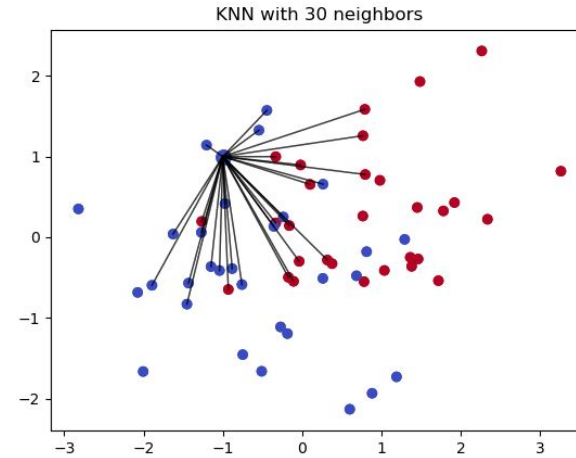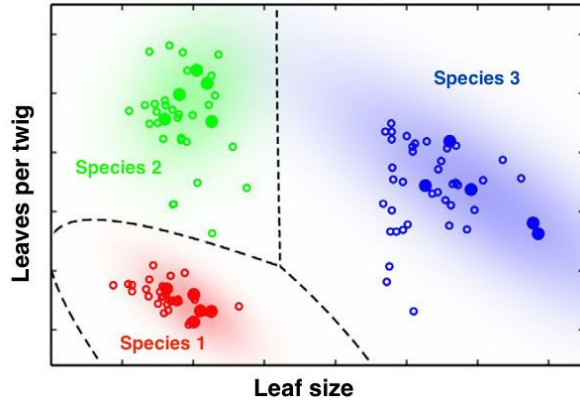- But it introduces **bias** in the dataset

```
>> an_array.mean(axis=1) # computes means for each row
```

```
>> an_array.median() # default is axis=0
```

2.3: Using Regression / Clustering

- Use other variables to predict the missing values
  - Through regression, clustering, KNN…
- Doesn't include an error term, so it's not clear how confident the prediction is
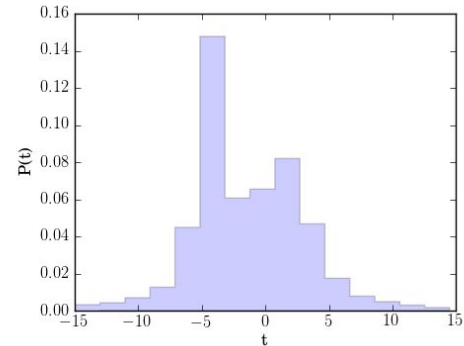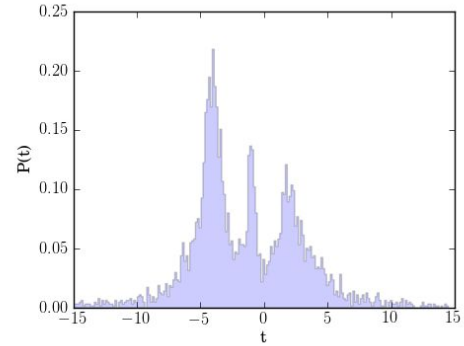

KNN with 30 neighbors

Technique 1: Binning

What?
Makes continuous data categorical by lumping ranges of data into discrete "levels"

Why?
Applicable to problems like (third-degree) price discrimination

Technique 2: Normalizing

What?	Turns the data into values between 0 and 1

Why?	Easy comparison between different features that may have different scales

Technique 3: Standardizing

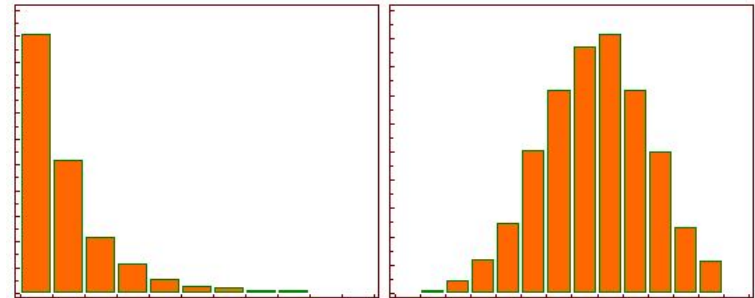| **What?**	Turns the data into a normal distribution with mean = 0 and SD = 1
Why?	Meet model assumptions of normal data; act as a benchmark since the majority of data is normal; wreck GPAs

Standardizing
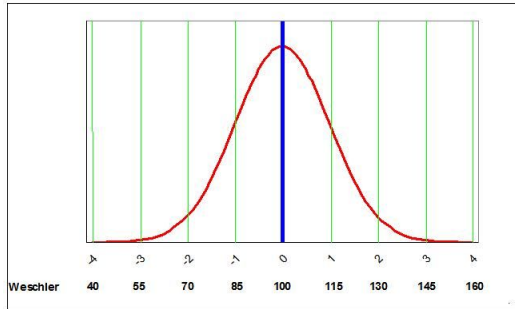
Log transformation

Others include square root, cubic root, reciprocal, square, cube…

Technique 4: Ordering

What?	Why?	Example
Converts categorical data that is inherently ordered into a numerical scale	Numerical inputs often facilitate analysis	January → 1 February → 2 March → 3 …

Technique 5: Dummy Variables

What?	Creates a binary variable for each category in a categorical variable

plant	is a tree
aspen	1
poison ivy	0
grass	0
oak	1
corn	0

Technique 6: Feature Engineering

What?	Generates new features which may provide additional information to the user and to the model

Why?	You may add new columns of your own design using the assign function in pandas

ID	Num
0001	2
0002	4
0003	6
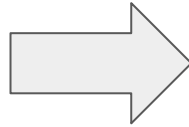
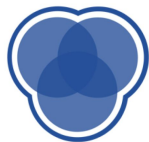| ID	Num	Half	SQ
0001	2	1	4
0002	4	2	16
0003	6	3	36

Summary

Demo

Coming Up

- **Assignment 2**: Due at 11:59pm on September 28th, 2022
- **Next Lecture**: Data Visualization

CDS Education

We explore, learn, and educate big minds.