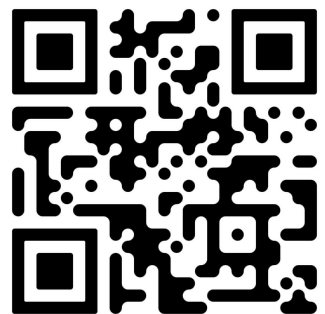


DATA 101: INTRODUCTION TO DATA AND DATA PROCESSING

SPONSORED BY PURDUE
WOMEN IN HPC

SARAH RODENBECK, SENIOR RESEARCH
DATA SCIENTIST



Sign up
for our
newsletter



SCHOLARSHIPS

Present your research
at conferences

CAREER DEVELOPMENT

Attend trainings
and events

COMMUNITY

Join the conversation
on Slack

OUTCOMES



Understand how data is used in machine learning and data science



Data types and data collection



Topics to consider when doing an Exploratory Data Analysis

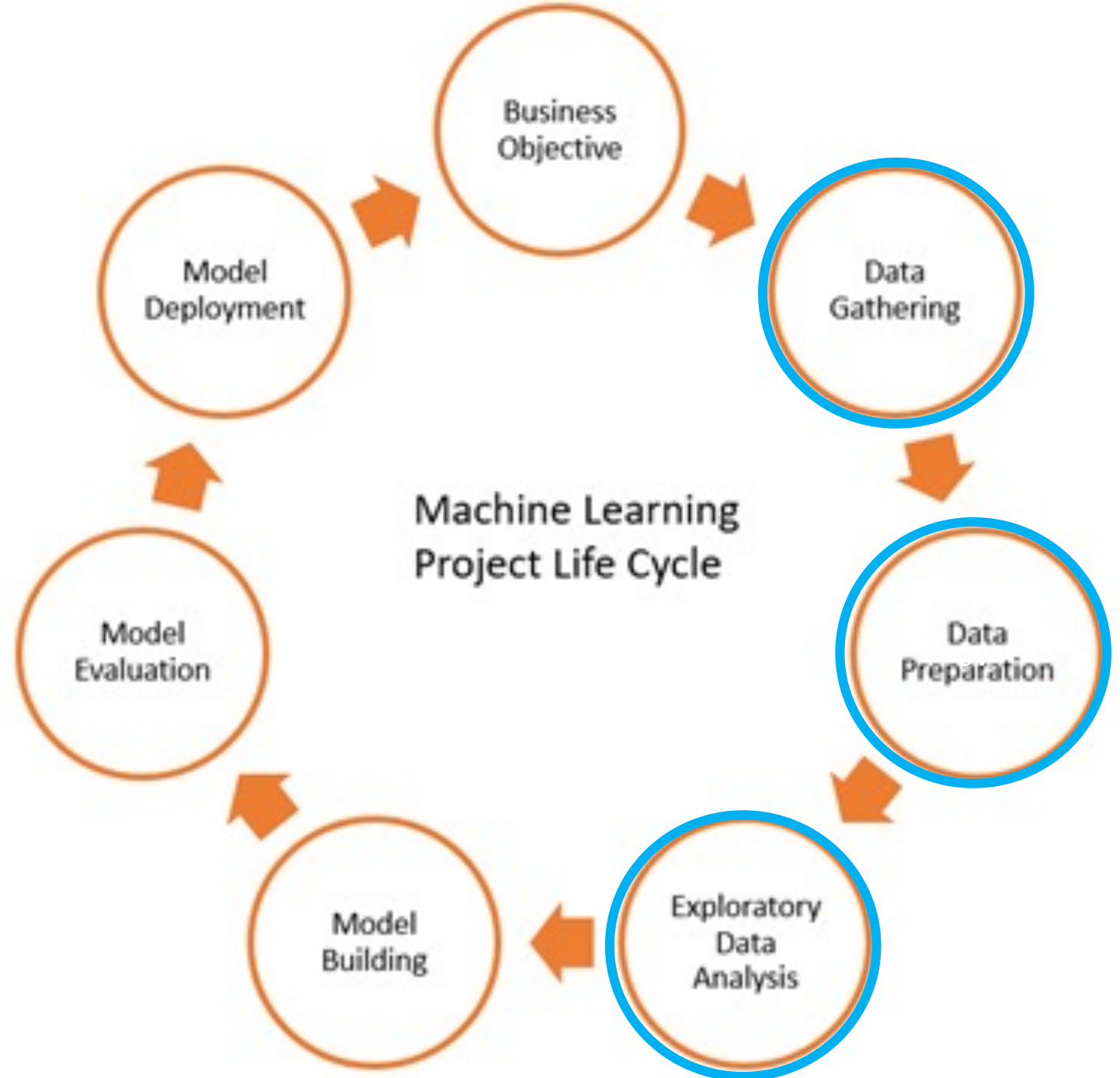


Basic data processing using Pandas

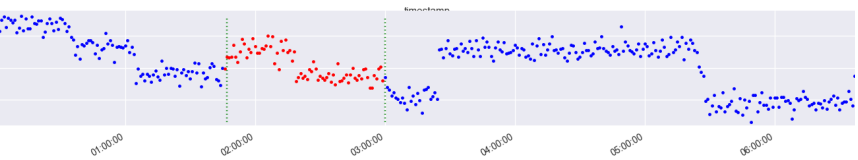


DATA BASICS

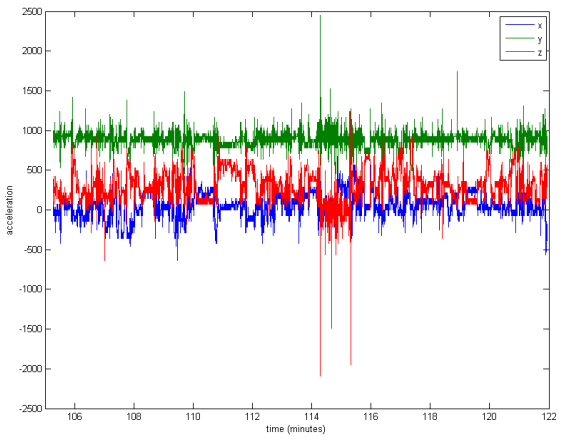
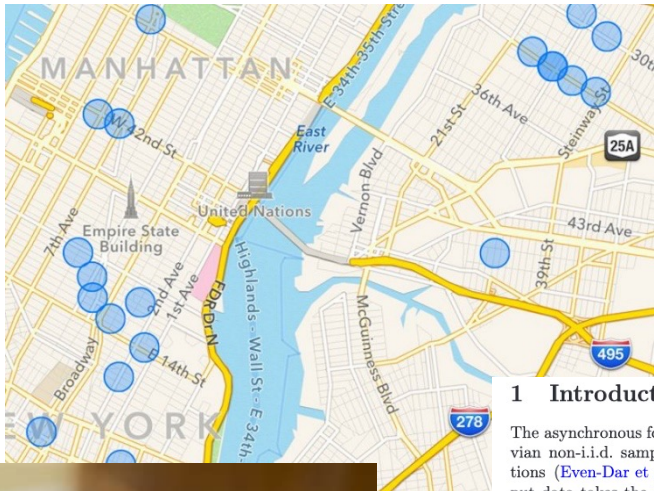
MODEL DEVELOPMENT CYCLE



WHAT IS DATA?



BREEZE Mview - [LGA0008.SCF]		File		Edit																						
Go To/Find		Window/Plot		Data Table																						
Date Range		Alt/Date/End																								
Start Date		1/1/2000																								
End Date		12/31/2000																								
Start Hour		1																								
End Hour		24																								
Data																										
ES LGA0008.SCF																										
Sensible Heat (m/s)																										
Friction Velocity																										
Conv. Vel. Scal.																										
Var. Dir. Temp.																										
Conv. Max. Hgt.																										
Moist. Max. Hgt.																										
Moist. Observed																										
Surface Rough.																										
Mean Rate																										
Albedo																										
Wind Speed (m/s)																										
Wind Dir. (deg.)																										
Wind Ref. Hgt. (m)																										
Temp. (°C)																										
Temp. Ref. Hgt.																										
Precip. Code																										
Precip. Rate (mm/hr)																										
Relative Humid.																										
Station Pressure																										
Cloud Cover (%)																										
Table Appearance																										
Background		Light																								
Bar Color		White																								
Font Color		Black																								
Font		Microsoft S.																								
Data Table																										
#	AERMOD	Yr	Mon	Day	Jul	Day	Sensib. Heat Flux	Friction Velocity	Conv. Vel. Scale	Vert. Dir. Temp. (°C)	Conv. Max. Hgt. (m)	Mech. Obs. Hgt. (m)	Moist. Obs. Hgt. (m)	Surface Rough. (m)	Bowen Ratio	Albedo	Wind Speed (m/s)	Wind Dir. (deg.)	Wind Ref. Hgt. (m)	Temp. (°C)	Temp. Ref. Hgt. (m)	Precip. Code	Precip. Rate (mm/hr)	Relative Humidity (%)	Station Pressure (mb)	Cloud Cover (tenths)
1	0	8	1	1	1	1	-44.0	0.000	0.012	0.006	3	11	4688.0	0.004	0.70	0.16	0.00	0	10.0	260.0	2.0	0	0.00	11	966	0
2	0	8	1	1	1	2	-25.5	0.218	-0.000	-0.000	-899	48	6.0	0.000	0.76	1.00	2.36	211	10.0	274.9	2.0	0	0.00	72	1019	0
3	0	8	1	1	1	3	-16.7	0.144	-0.000	-0.000	-899	127	16.2	0.078	0.76	1.00	2.86	124	10.0	279.9	2.0	0	0.00	69	1019	0
4	0	8	1	1	1	4	-5.4	0.073	-0.000	-0.000	-899	47	6.6	0.062	0.76	1.00	1.76	103	10.0	275.4	2.0	0	0.00	69	1019	0
5	0	8	1	1	1	5	-26.5	0.279	-0.000	-0.000	-899	339	69.4	0.062	0.76	1.00	3.86	113	10.0	274.9	2.0	0	0.00	75	1017	5
6	0	8	1	1	1	6	-51.4	0.510	-0.000	-0.000	-899	637	232.9	0.078	0.76	1.00	6.46	132	10.0	277.0	2.0	0	0.00	69	1019	5
7	0	8	1	1	1	7	-46.8	0.466	-0.000	-0.000	-899	734	195.9	0.078	0.76	1.00	5.96	145	10.0	276.1	2.0	0	0.00	70	1013	5
8	0	8	1	1	1	8	-36.6	0.686	-0.000	-0.000	-899	1342	725.1	0.064	0.76	0.71	8.46	153	10.0	279.9	2.0	0	0.00	72	1011	10
9	0	8	1	1	1	9	-20.7	0.826	-0.000	-0.000	-899	1720	844.9	0.071	0.76	0.39	10.05	147	10.0	280.4	2.0	0	0.00	70	1009	10
10	0	8	1	1	1	10	5.3	0.709	0.151	0.006	23	1390	4052.4	0.064	0.76	0.27	8.46	151	10.0	279.9	2.0	0	0.00	78	1008	10
11	0	8	1	1	1	11	12.4	0.633	0.246	0.009	43	1167	1946.1	0.088	0.76	0.22	7.46	164	10.0	280.4	2.0	11	3.00	82	1008	10
12	0	8	1	1	1	12	16.0	0.297	0.299	0.009	89	477	95.3	0.088	0.76	0.21	2.86	186	10.0	280.9	2.0	11	1.50	88	1006	10
13	0	8	1	1	1	13	55.4	0.412	0.544	0.005	104	608	113.1	0.009	0.76	0.21	6.96	263	10.0	282.5	2.0	0	0.00	73	1005	8
14	0	8	1	1	1	14	10.6	0.366	0.340	0.005	112	581	411.3	0.004	0.76	0.22	7.46	279	10.0	282.0	2.0	0	0.00	57	1006	10
15	0	8	1	1	1	15	17.9	0.337	0.396	0.005	124	452	191.7	0.004	0.76	0.27	6.46	292	10.0	281.4	2.0	0	0.00	55	1004	9
16	0	8	1	1	1	16	4.1	0.383	0.254	0.005	142	544	1212.3	0.004	0.76	0.39	7.46	294	10.0	281.4	2.0	0	0.00	45	1004	5
17	0	8	1	1	1	17	-44.0	0.447	-0.000	-0.000	-899	686	162.2	0.004	0.76	0.73	9.05	291	10.0	280.9	2.0	0	0.00	39	1005	5
18	0	8	1	1	1	18	-27.7	0.412	-0.000	-0.000	-899	610	226.5	0.009	0.76	1.00	7.46	257	10.0	279.9	2.0	0	0.00	39	1005	9
19	0	8	1	1	1	19	-34.3	0.446	-0.000	-0.000	-899	684	231.0	0.004	0.76	1.00	8.96	274	10.0	279.2	2.0	0	0.00	39	1005	8
20	0	8	1	1	1	20	-34.2	0.506	-0.000	-0.000	-899	827	330.7	0.009	0.76	1.00	9.05	267	10.0	278.8	2.0	0	0.00	42	1005	9
21	0	8	1	1	1	21	-27.3	0.480	-0.000	-0.000	-899	766	363.3	0.004	0.76	1.00	9.56	270	10.0	278.1	2.0	0	0.00	44	1005	10
22	0	8	1	1	1	22	-34.4	0.446	-0.000	-0.000	-899	685	230.3	0.004	0.76	1.00	8.96	272	10.0	278.1	2.0	0	0.00	44	1005	8
23	0	8	1	1	1	23	-36.3	0.535	-0.000	-0.000	-899	899	373.2	0.009	0.76	1.00	9.56	260	10.0	277.5	2.0	0	0.00	46	1004	9
24	0	8	1	1	1	24	-57.6	0.529	-0.000	-0.000	-899	884	229.7	0.009	0.76	1.00	9.56	260	10.0	277.0	2.0	0	0.00	48	1004	3
25	0	8	1	1	2	1	-47.3	0.434	-0.000	-0.000	-899	664	164.3	0.009	0.76	1.00	7.96	266	10.0	278.4	2.0	0	0.00	50	1004	3
26	0	8	1	1	2	2	-47.2	0.413	-0.000	-0.000	-899	610	133.3	0.004	0.76	1.00	8.46	272	10.0	278.4	2.0	0	0.00	52	1004	0
27	0	8	1	1	2	3	-22.7	0.396	-0.000	-0.000	-899	576	249.8	0.004	0.76	1.00	7.96	272	10.0	278.9	2.0	0	0.00	51	1004	10
28	0	8	1	1	2	4	-33.5	0.334	-0.000	-0.000	-899	446	89.6	0.004	0.76	1.00	6.96	280	10.0	278.9	2.0	0	0.00	56	1004	6
29	0	8	1	1	2	5	-29.0	0.507	-0.000	-0.000	-899	830	401.5	0.006	0.76	1.00	9.56	306	10.0	278.9	2.0	0	0.00	51	1004	10
30	0	8	1	1	2	6	-30.6	0.534	-0.000	-0.000	-899	937	444.3	0.009	0.76	1.00	10.05	317	10.0	274.9	2.0	0	0.00	51	1005	10
31	0	8	1	1	2	7	-46.1	0.586	-0.000	-0.000	-899	1031	391.7	0.006	0.76	1.00	11.05	309	10.0	273.8	2.0	0	0.00	51	1006	8
32	0	8	1	1	2	8	-21.0	0.363	-0.000	-0.000	-899	945	204.5	0.006	0.76	0.71	6.96	326	10.0	273.1	2.0	0	0.00	55	1007	10
33	0	8	1	1	2	9	-14.1	0.484	-0.000	-0.000	-899	774	225.5	0.005	0.76	0.39	9.05	320	10.0	272.5	2.0	0	0.00	53	1008	10
34	0	8	1	1	2	10	4.4	0.440	0.252	0.006	130	672	1779.1	0.007	0.76	0.27	7.96	331	10.0	272.5	2.0	0	0.00	53	1009	10
35	0	8	1	1	2	11	30.7	0.411	0.708	0.006	476	607	203.3	0.006	0.76	0.22	7.46	326	10.0	272.5	2.0	0	0.00	53	1010	9
36	0	8	1	1	2	12	37.6	0.506	0.087	0.006	603	829	311.2	0.007	0.76	0.21	6.96	331	10.0	272.5	2.0	0	0.00	46	1010	9
37	0	8	1	1	2	13	54.3	0.468	1.072	0.006	627	737	189.7	0.006	0.76	0.21	8.46	319	10.0	273.1	2.0	0	0.00	47	1010	9
38	0	8	1	1	2	14	65.0	0.511	1.164	0.006	740	839	185.0	0.007	0.76	0.22	9.05	337	10.0	273.1	2.0	0	0.00	47	1010	9
39	0	8	1	1	2	15	17.2	0.612	0.713	0.005	762	1086	1202.9	0.007	0.76	0.27	11.05	340	10.0	273.1	2.0	0	0.00	47	1010	9
40	0	8	1	1	2	16	5.2	0.676	0.773	0.006	768	1084	1261.1	0.007	0.76	0.27	11.05	340	10.0	273.1	2.0	0	0.00	47	1010	9



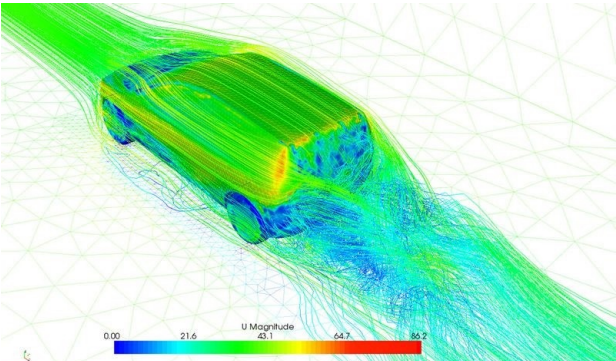
1 Introduction

The asynchronous form of Q-learning, which is a stochastic approximation paradigm that applies to Markovian non-i.i.d. samples, has found applicability in an abundance of reinforcement learning (RL) applications (Even-Dar et al., 2003; Jaakkola et al., 1994; Tsitsiklis, 1994; Watkins and Dayan, 1992). The input data takes the form of a Markovian sample trajectory induced by a policy called the *behavior policy*; in each time, asynchronous Q-learning only updates the Q-function estimate of a single state-action pair along the trajectory rather than updating all pairs at once — and hence the terminology “asynchronous” (Bertsekas and Tsitsiklis, 2003; Tsitsiklis, 1994). This classical algorithm has the virtue of being off-policy, allowing one to learn the optimal policy even when the behavior policy is suboptimal. Recent years have witnessed a resurgence of interest in understanding the performance of asynchronous Q-learning, due to a shift of attention from classical asymptotic analysis to the non-asymptotic counterpart. By and large, non-asymptotic results bear important and clear implications for the impacts of salient parameters (e.g., model capacity, horizon length) in large-dimensional RL problems.

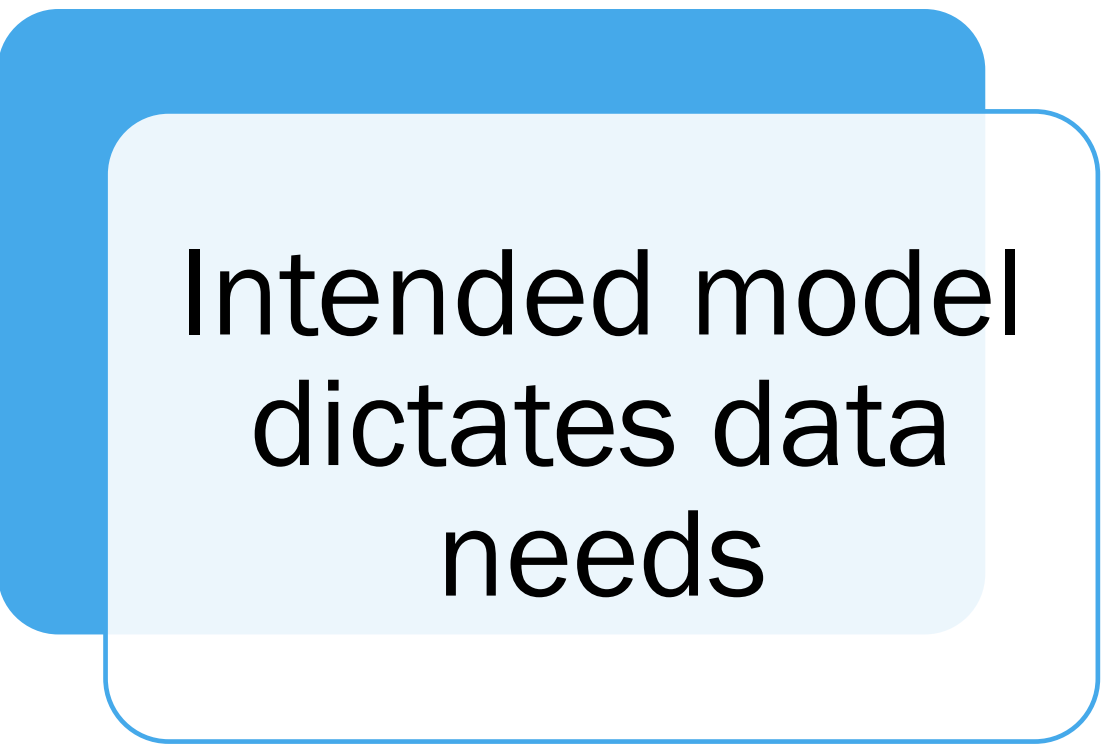
1.1 Motivation

A central consideration in modern RL applications is data efficiency: the limited availability of data samples places increasing demands on sample-efficient RL solutions, and in turn, calls for reexamining classical algorithms like Q-learning. When it comes to asynchronous Q-learning, recent theoretical advances have led to sharpened sample complexity analyses (Li et al., 2021a,c; Qu and Wierman, 2020). For concreteness, consider a γ -discounted infinite-horizon Markov decision process (MDP) and a stationary behavior policy: asynchronous Q-learning provably yields ϵ -accuracy as soon as the sample size exceeds the order of¹ (Li et al., 2021a)

$$\frac{1}{\mu_{\min}(1-\gamma)^4\epsilon^2} + o\left(\frac{1}{\epsilon^2}\right) \quad (1.1)$$

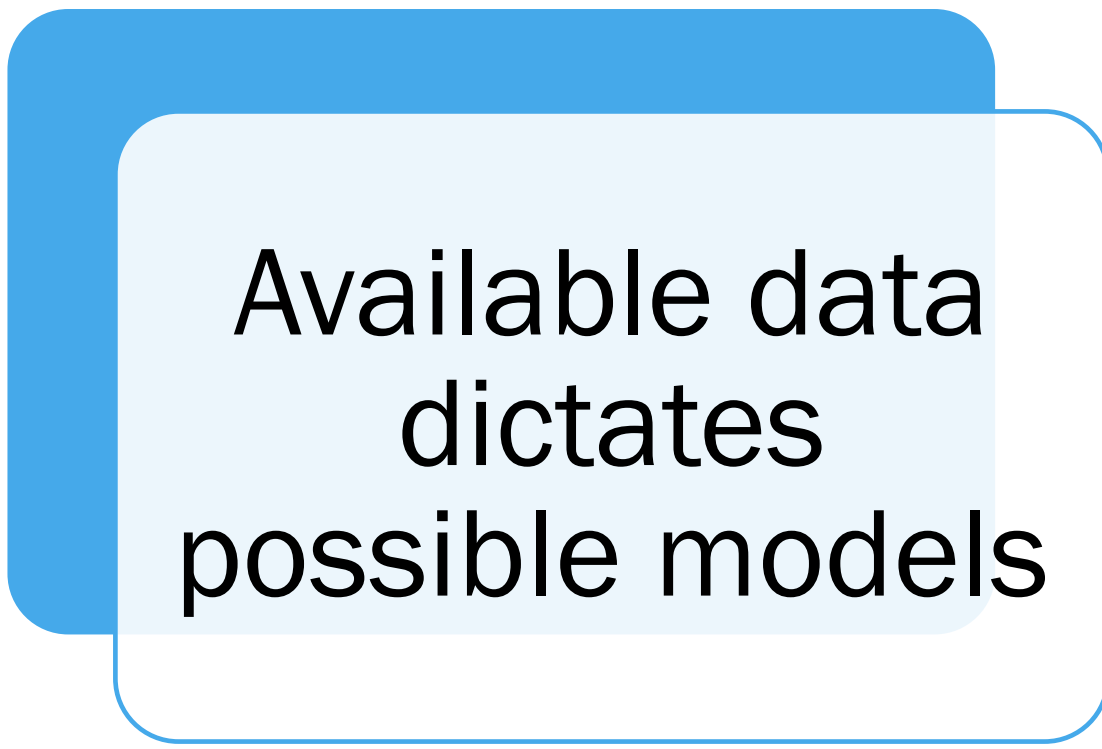


WHY DOES THIS MATTER?



Intended model
dictates data
needs

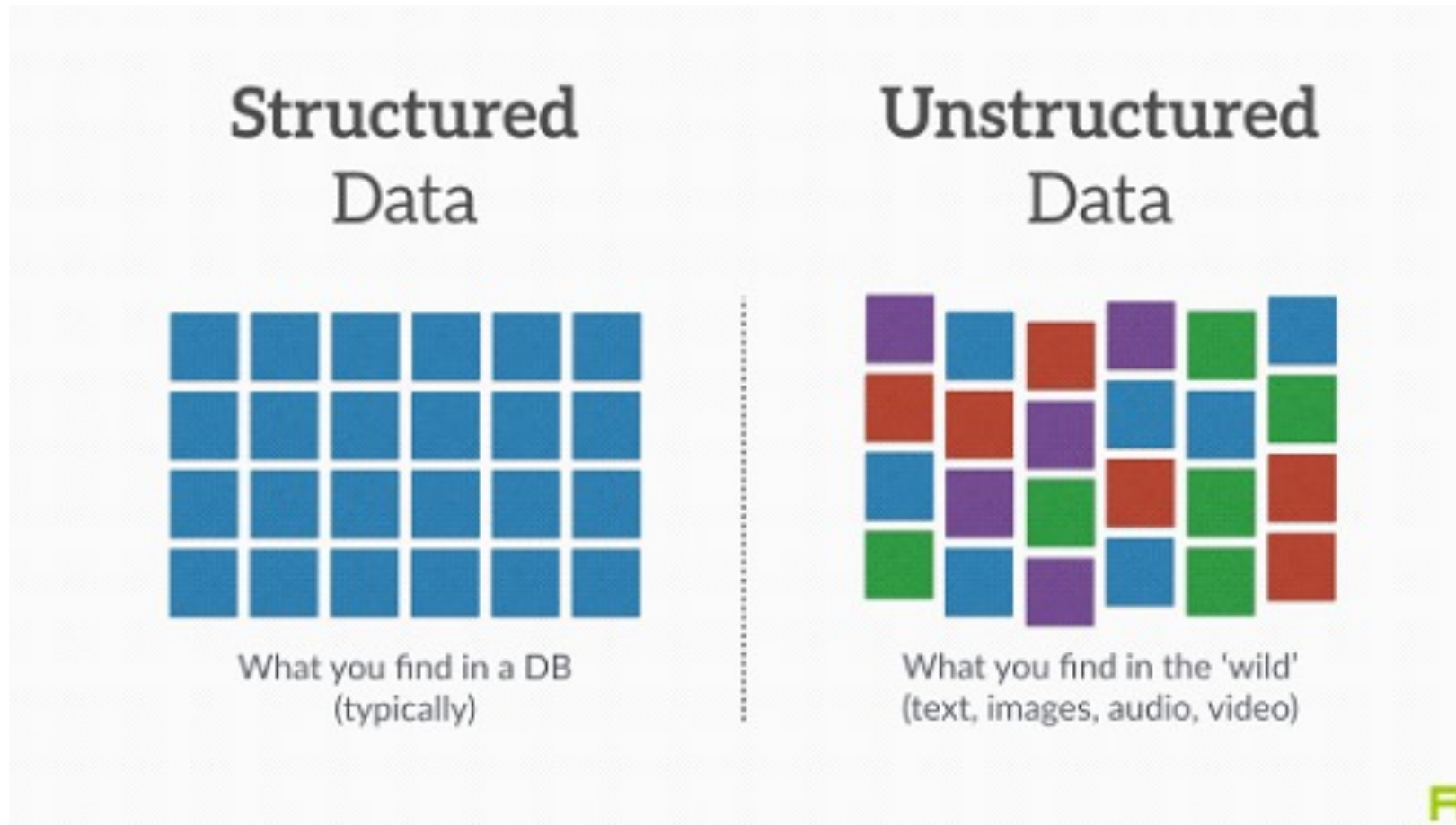
The diagram consists of two nested rounded rectangles. The outer rectangle is blue, and the inner rectangle is light blue. The text is centered within the inner rectangle.



Available data
dictates
possible models

The diagram consists of two nested rounded rectangles. The outer rectangle is blue, and the inner rectangle is light blue. The text is centered within the inner rectangle.

STRUCTURED VS UNSTRUCTURED DATA



LABELLED VS UNLABELLED DATA

Unlabeled Data

Je m'appelle Sarah.

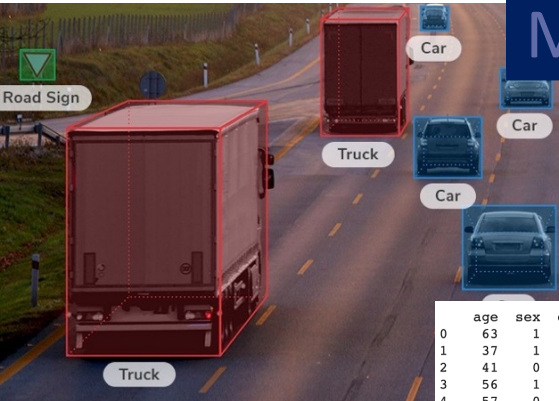


	age	sex	cp	trestbps	chol	fbs	...	exang	oldpeak	slope	ca	thal
0	63	1	3	145	233	1	...	0	2.3	0	0	1
1	37	1	2	130	250	0	...	0	3.5	0	0	2
2	41	0	1	130	204	0	...	0	1.4	2	0	2
3	56	1	1	120	236	0	...	0	0.8	2	0	2
4	57	0	0	120	354	0	...	1	0.6	2	0	2
5	57	1	0	140	192	0	...	0	0.4	1	0	1
6	56	0	1	140	294	0	...	0	1.3	1	0	2
7	44	1	1	120	263	0	...	0	0.0	2	0	3
8	52	1	2	172	199	1	...	0	0.5	2	0	3
9	57	1	2	150	168	0	...	0	1.6	2	0	2
10	54	1	0	140	239	0	...	0	1.2	2	0	2
11	48	0	2	130	275	0	...	0	0.2	2	0	2
12	49	1	1	130	266	0	...	0	0.6	2	0	2
13	64	1	3	110	211	0	...	1	1.8	1	0	2
14	58	0	3	150	283	1	...	0	1.0	2	0	2
15	50	0	2	120	219	0	...	0	1.6	1	0	2
16	58	0	2	120	340	0	...	0	0.0	2	0	2
17	66	0	3	150	226	0	...	0	2.6	0	0	2
18	43	1	0	150	247	0	...	0	1.5	2	0	2
19	69	0	3	140	239	0	...	0	1.8	2	2	2

[20 rows x 14 columns]

Labeled Data

Je m'appelle Sarah =
My name is Sarah



	age	sex	cp	trestbps	chol	fbs	...	exang	oldpeak	slope	ca	thal	target
0	63	1	3	145	233	1	...	0	2.3	0	0	1	1
1	37	1	2	130	250	0	...	0	3.5	0	0	2	1
2	41	0	1	130	204	0	...	0	1.4	2	0	2	1
3	56	1	1	120	236	0	...	0	0.8	2	0	2	1
4	57	0	0	120	354	0	...	1	0.6	2	0	2	1
5	57	1	0	140	192	0	...	0	0.4	1	0	1	1
6	56	0	1	140	294	0	...	0	1.3	1	0	2	1
7	44	1	1	120	263	0	...	0	0.0	2	0	3	1
8	52	1	2	172	199	1	...	0	0.5	2	0	3	1
9	57	1	2	150	168	0	...	0	1.6	2	0	2	1
10	54	1	0	140	239	0	...	0	1.2	2	0	2	1
11	48	0	2	130	275	0	...	0	0.2	2	0	2	1
12	49	1	1	130	266	0	...	0	0.6	2	0	2	1
13	64	1	3	110	211	0	...	1	1.8	1	0	2	1
14	58	0	3	150	283	1	...	0	1.0	2	0	2	1
15	50	0	2	120	219	0	...	0	1.6	1	0	2	1
16	58	0	2	120	340	0	...	0	0.0	2	0	2	1
17	66	0	3	150	226	0	...	0	2.6	0	0	2	1
18	43	1	0	150	247	0	...	0	1.5	2	0	2	1
19	69	0	3	140	239	0	...	0	1.8	2	2	2	1

[20 rows x 14 columns]

SUPERVISED VS UNSUPERVISED LEARNING

- Most Machine Learning and Statistical Models
 - Image Recognition
 - Neural Machine Translation
 - Loan Default Prediction
 - **Relies on Labelled Data as the "ground truth"**
- Data preparation and limited models
 - Clustering for anomaly detection
 - Dimensionality reduction
 - Association and Recommender systems
 - **Does not require labelled data**

SUPERVISED LEARNING

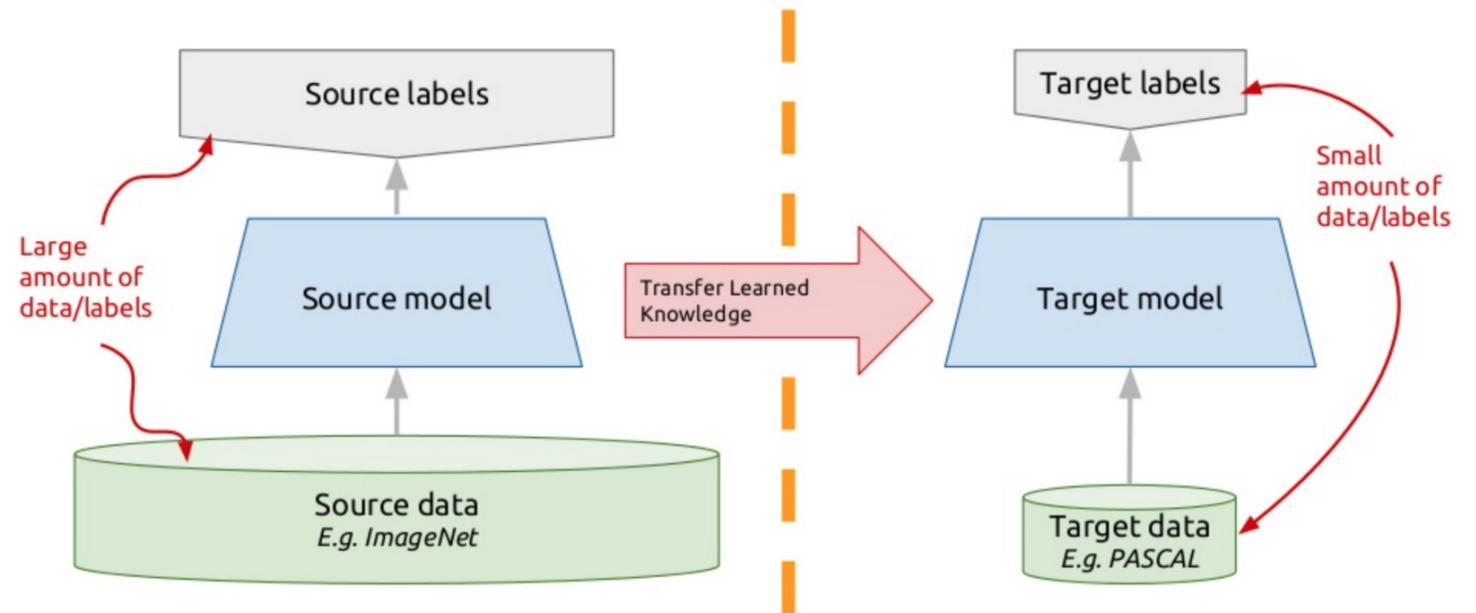
- More data preparation
- More performant
- More types of models



ON LABELED DATA

- You need a lot of labeled data for most ML systems!
 - Typically on the scale of thousands or tens of thousands of data points
- **Transfer learning** reduces data needs but you still need enough to fine tune the model for your specific task

Transfer learning: idea



DATA ANNOTATION

- Some times labelled data is naturally collected (e.g. engineers marking if a test worked or not)
- Other times you can use pre-existing labelled data (e.g. Kaggle dataset)
- **Most of the time you will need to go through a data annotation process**

The screenshot shows a data annotation interface. At the top, there is a toolbar with icons for close, filter, view, comment, edit, delete, and a menu. To the right of the toolbar, it says "2 of 5493" with navigation arrows. Below the toolbar, there are four colored buttons: "Not an ADR" (grey, 1), "Failure of Therapy" (green, 2), "Misuse" (blue, 3), and "Side Effect" (purple, 4). The main content area displays a tweet: "I am now part of the group of people for which gabapentin is the devil. 600mg was fine but useless. 900mg: I am now the useless one. 🙄". Below the tweet, there is a "Progress" section with a table showing "Total" as 5493 and "Complete" as 0, with a green progress bar at 0%. At the bottom, there is a table with two columns: "Key" and "Value".

Key	Value
drug	Neurontin
TweetID	6229

DATA ANNOTATION DIFFICULTIES

- Clear and consistent guidelines are key
 - Inconsistent labelling can confuse the model's training
 - It can be helpful to think about whether false positives or false negatives are more acceptable for your model and advise your annotators to err on that side
- Annotation can be a tedious and error prone process but is one of the most important
- **Garbage in, garbage out!**

Coming off Zoloft is \$%&@*#! weird

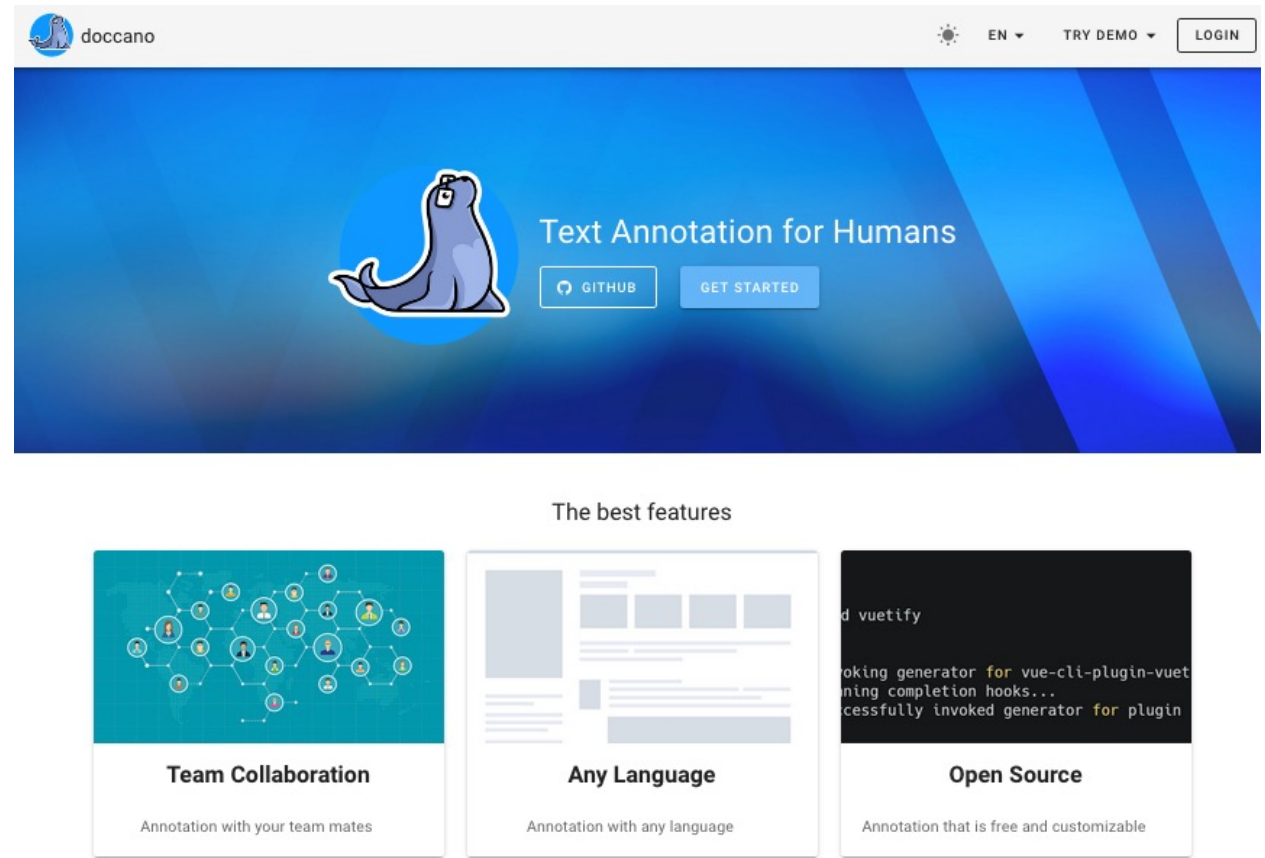
Does "weird" count as experiencing a side-effect?



Does this count as a car?

BEST PRACTICES FOR DATA ANNOTATION

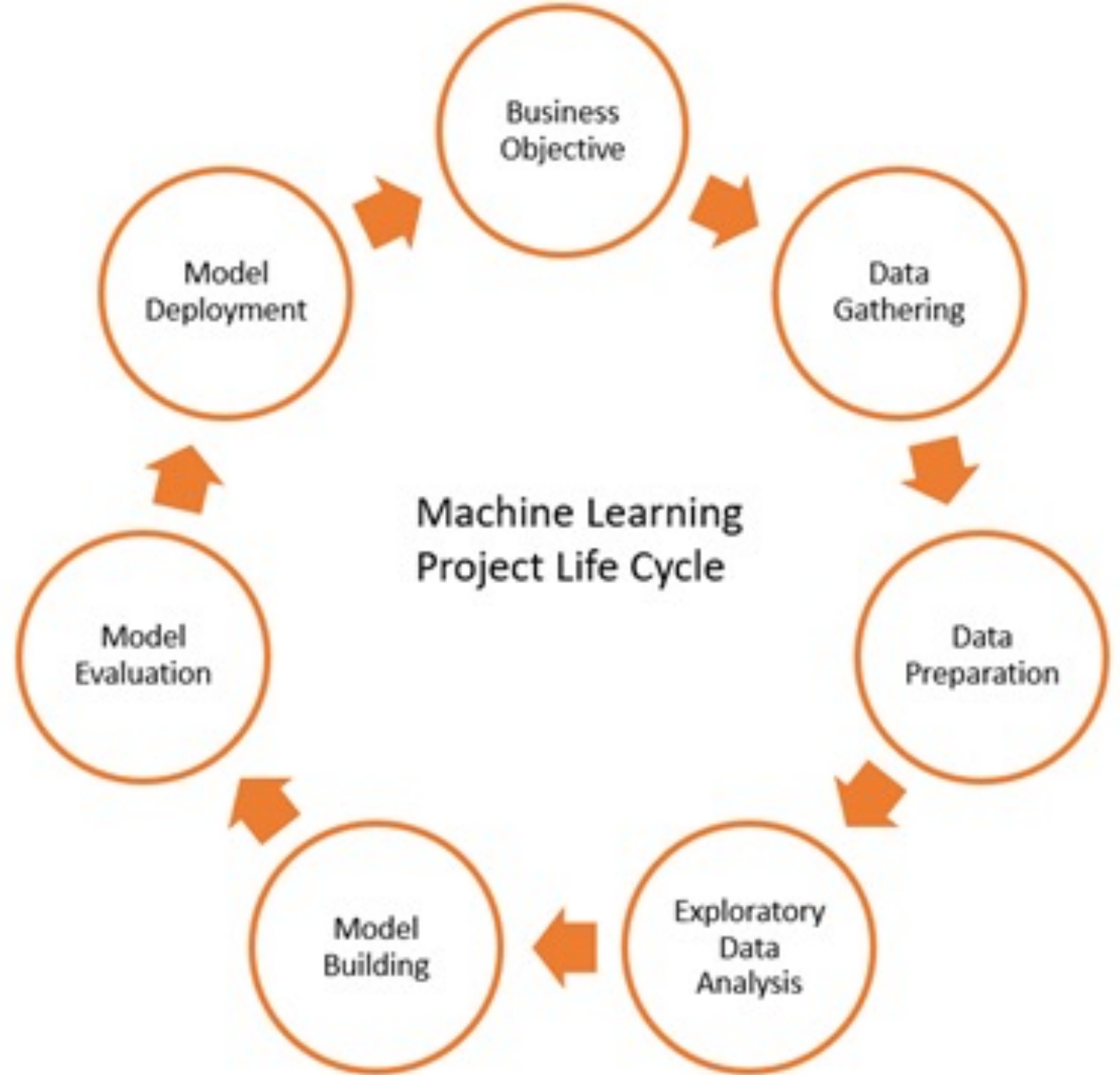
- If multiple annotators:
 - Create common guidelines
 - Hold an alignment meeting to go through examples together and calibrate
 - Have everyone complete a pre-test by annotating a limited number of examples and assess inter-annotator accuracy and analyze common misconceptions
- Tools like Doccano (open-source) can make data annotation easier



A close-up, blue-tinted photograph of a document. In the foreground, a pen is positioned over a line graph. The graph shows a jagged line with several peaks and valleys. A dotted line extends from a peak on the left towards the pen. To the right of the pen, the number '2,47' is visible. On the left side of the graph, the number '2,5' is visible. The background is slightly blurred, showing more of the document and the graph line.

DATA PRE-PROCESSING

MODEL DEVELOPMENT CYCLE



■ DATA CLEANING AND DATA PREPARATION ACCOUNTS FOR AS MUCH AS 80% OF A DATA SCIENTIST'S TIME



DATAFRAMES

- An extremely common tool used for working with data are dataframes
 - Contains both data and metadata
- Dataframes are structures that organizes data into a two-dimensional data
- Doesn't have size limitations like Excel!

*All provided commands in this presentation are for use with the Pandas DataFrames package in Python

	TweetID	text	Created_at	Retweet_count	Like_count	drug	annotator1	annotator2
0	60175	I haven't been able to take my regular dose of...	2021-04-27 03:43:14+00:00	0	0	Zoloft	1	3
1	68886	Xanax is expensive but I'm worth it.	2021-09-28 02:45:44+00:00	0	0	Xanax	1	3
2	58836	there is no tarantino film that can frighten m...	2021-01-26 16:33:45+00:00	0	0	Zoloft	1	3
3	30204	I just applied at my first bar job! I think it...	2021-08-19 04:03:53+00:00	1	1	Ambien	1	3
4	25322	Welp its official - I've payed over \$5000.00 ...	2021-11-22 22:16:51+00:00	0	8	Polymox	1	3
...
5688	34709	I forgot a dose of prednisone and now I'm too ...	2021-06-30 21:19:46+00:00	0	0	Rayos	25	3
5689	39646	Can I get albuterol without having to wait a m...	2021-05-09 21:07:02+00:00	0	0	ProAir	25	3
5690	46250	goin as a container of hydrocortisone as half ...	2021-10-20 20:05:48+00:00	0	2	Westcort	25	3
5691	2187	found out my body metabolizes anesthesia way t...	2021-06-21 23:06:03+00:00	0	2	Asperflex	25	3
5692	34373	#MedTwitter tweeps- is there any evidence that...	2021-05-24 09:21:46+00:00	1	4	Rayos	25	3

5693 rows x 8 columns

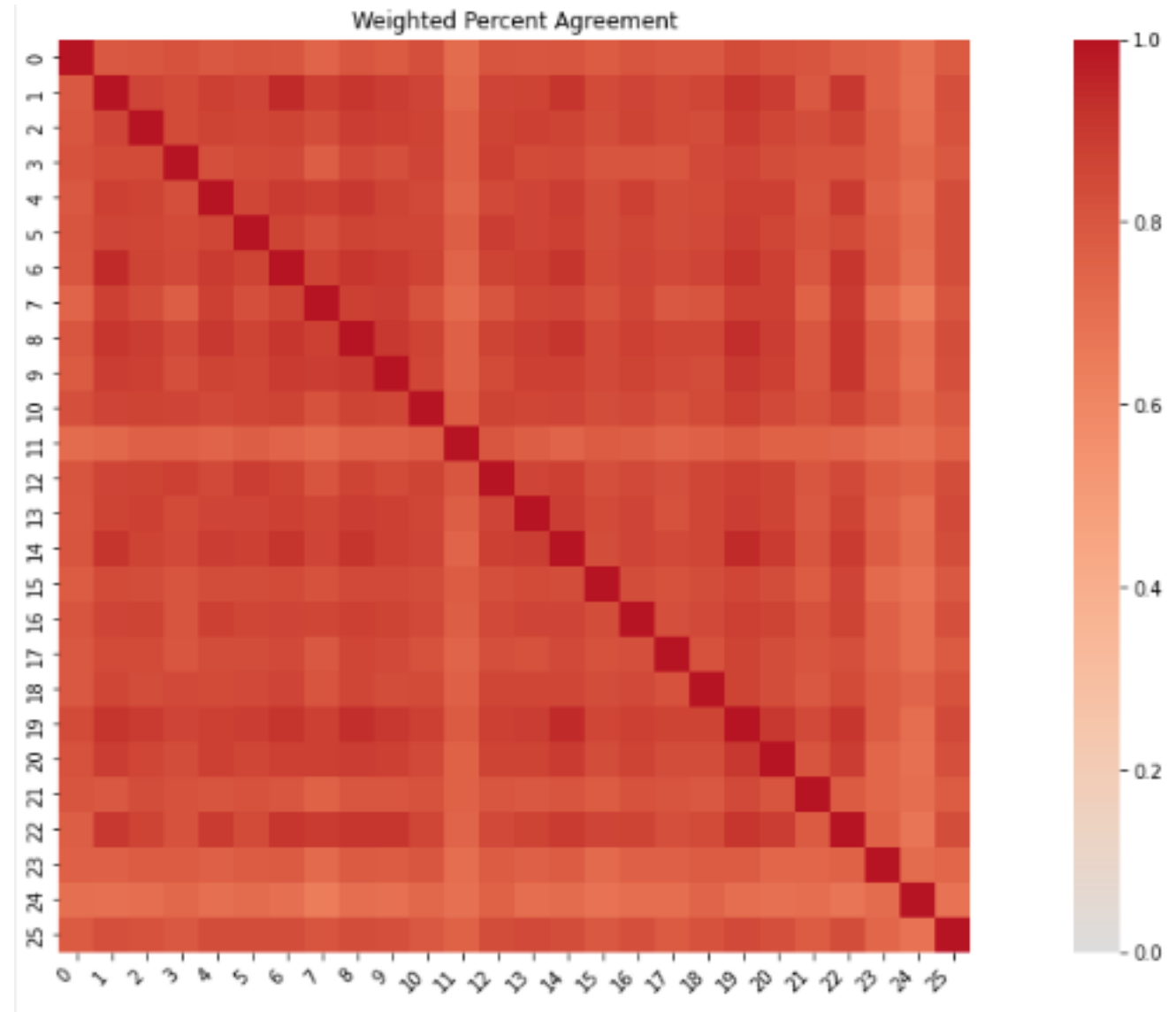
MISSING DATA

- What you do with missing outliers depends on your use case and the columns
- If you are missing data in the label column you may be best to delete the row
- For other columns you may be able to interpolate data
 - e.g. in a time series analysis you could average the previous and next data point



ACCOUNT FOR OUTLIERS

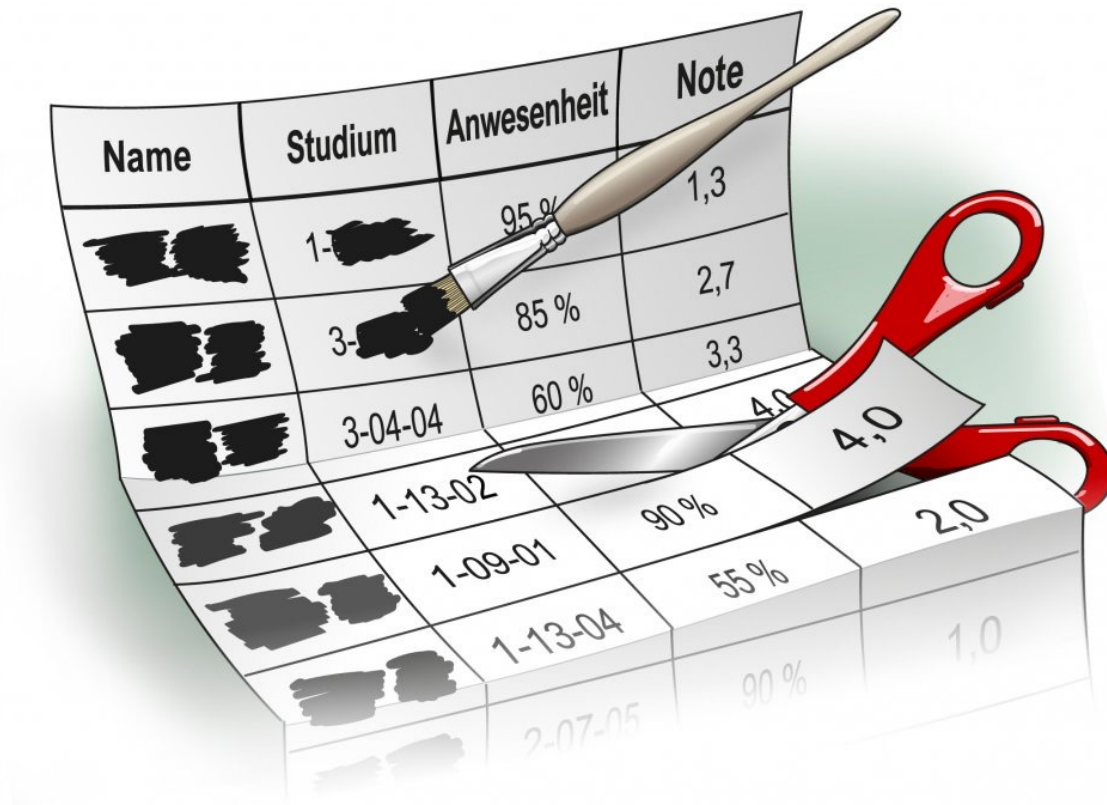
- What you do with outliers depends on your use case
- Are outliers important in your data (e.g. anomaly detection) or are they representative of someone not labelling the data well?
 - With the annotation data visualized here, I might want to not use annotator 11's data for model training



ANONYMIZE DATA

If you have a column with the subject name or identifiable information you can:

- Delete the column
- Substitute an anonymized value instead

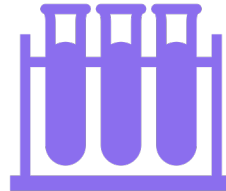


DATA SETS



Training Data (70%)

Data used by the model for training



Validation Data (20%)

Used to assess model performance during training



Test Data (10%)

Data used after training is complete for evaluation metrics

OTHER POTENTIAL STEPS IN PRE-PROCESSING

- Ensure consistent descriptors/categories (e.g. 'female' and 'woman' should be the same)
- Rename columns
- Data encoding
 - Data must be in a specific format for training. For example, with text data you will need to get the word embeddings
- Feature engineering
 - Using domain knowledge to manipulate raw data into a format that better captures key characteristics of the data
- Binning data for easier analysis
- Feature Selection/Dimensionality reduction
 - Reduces the data used to help the model identify what's most important
- Etc...



EXPLORATORY DATA ANALYSIS



EXPLORATORY DATA ANALYSIS

- Understand and summarize the main characteristics of a data set prior to model training

WHY PERFORM AN EDA?

Better understand data and data patterns

Can uncover data issues

Help select the right model for your data

COMMON ASPECTS OF AN EDA

- Average, median, high and low values for each column
- Relationships between columns (correlations)
- Explore data quality trends
- Identify unnecessary columns
- Null and outlier analysis
- Visualize data
- Identify data biases






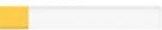












VISUALIZATIONS

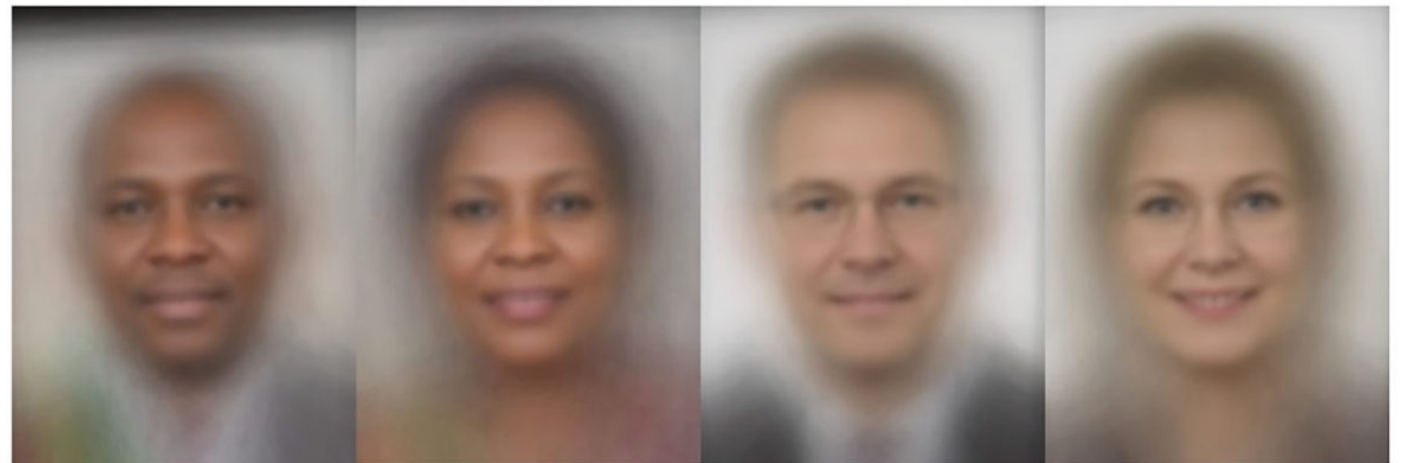


Recommended
Introductory Python
Visualization Libraries:

- matplotlib
- seaborn

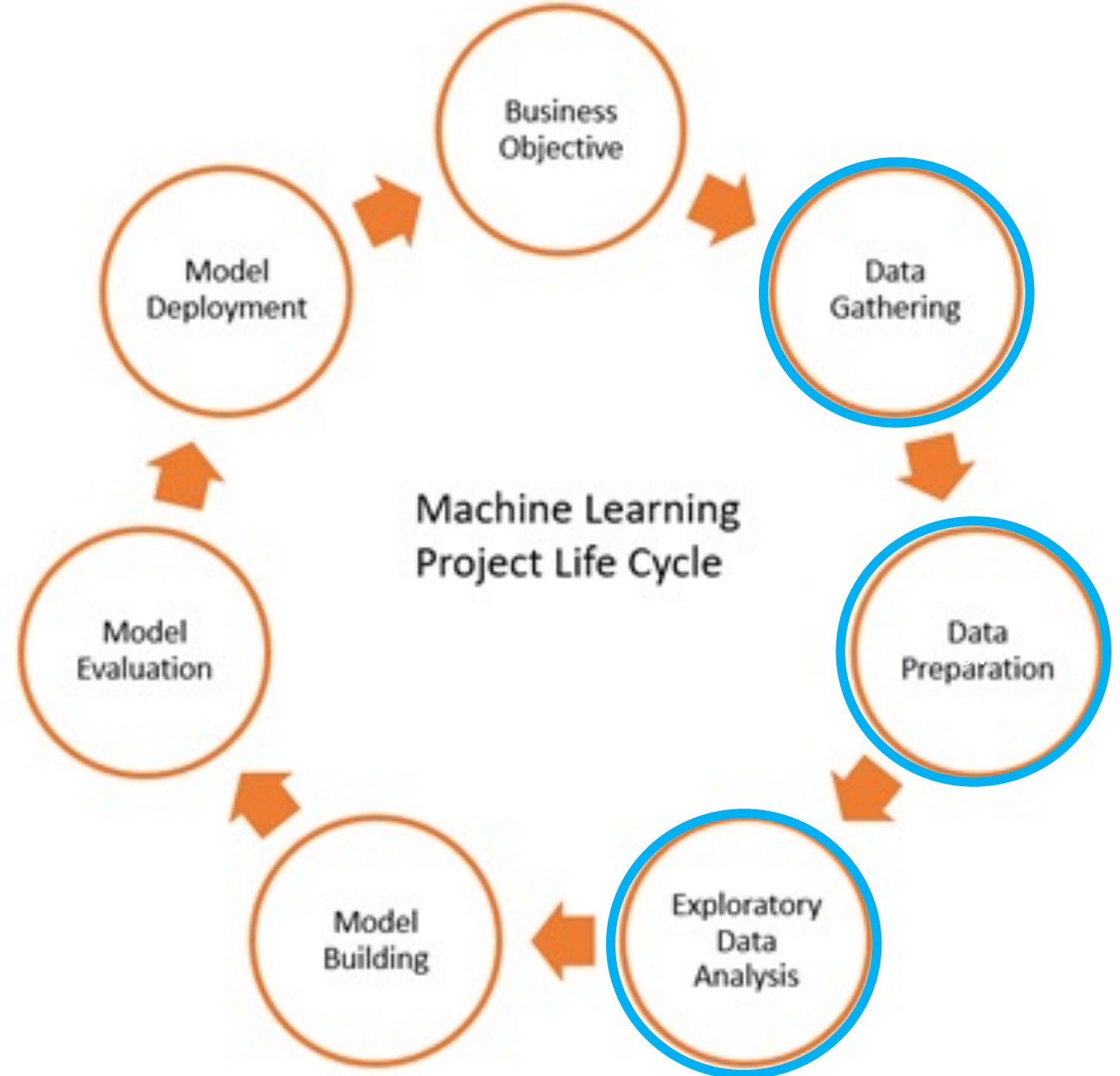
DATA BIAS

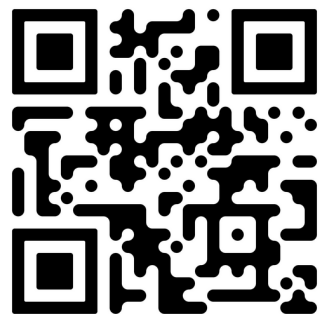
Gender Classifier	Darker Male	Darker Female	Lighter Male	Lighter Female	Largest Gap
 Microsoft	94.0% 	79.2% 	100% 	98.3% 	20.8% 
 FACE++	99.3% 	65.5% 	99.2% 	94.0% 	33.8% 
 IBM	88.0% 	65.3% 	99.7% 	92.9% 	34.4% 



CONCLUSION

- Preparation for building a model often takes longer than building the model itself
- Steps are not strictly linear
- The quality of each of these preparation steps directly impacts the quality of your outcome





Sign up
for our
newsletter



SCHOLARSHIPS

Present your research
at conferences

CAREER DEVELOPMENT

Attend trainings
and events

COMMUNITY

Join the conversation
on Slack

A close-up, blue-tinted photograph of a financial chart. A silver pen is positioned in the upper right corner, pointing towards a data point on the chart. The chart features a jagged line graph with several peaks and troughs. Faint numerical values are visible on the left and right sides of the chart, including '2,5' and '2,47'. The word 'APPENDIX' is centered in the middle of the image, underlined.

APPENDIX

GETTING STARTED WITH PANDAS DATAFRAMES

Standard import statement




```
import pandas as pd  
pd.read_csv("Data_to_Annotate_25/3_of_25_project_data.csv")
```




Reads the specified CSV file
into a DataFrame

COMBINE DATASETS AND REMOVE DUPLICATES

This concatenates the given dataframes (essentially stacks them on top of one another)



```
df = pd.concat([df1, df2, df3], ignore_index=True)
df.drop_duplicates(subset=None, keep='first', inplace=False)
```



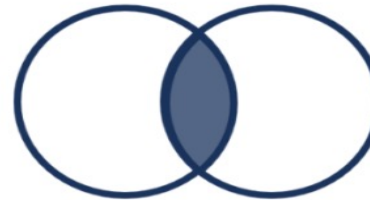
Deletes duplicates and keeps only the first value of each duplicate

SOMETIMES YOU WANT TO MERGE DATAFRAMES ON CERTAIN VALUES AS WELL

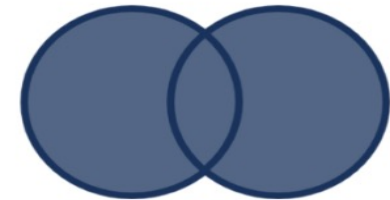
```
merged = pd.merge(left=df1, right=df2, left_on='ID', right_on='ID', how='outer')
```

Merges df1 and df2 on the ID column in each and keeps all values

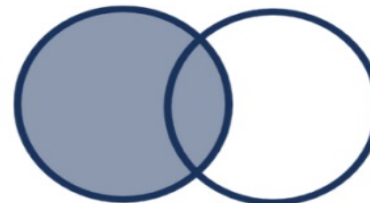
INNER JOIN



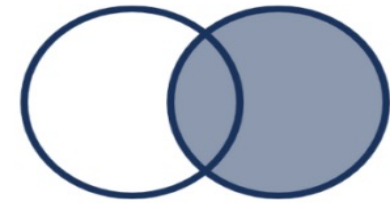
OUTER JOIN



LEFT JOIN



RIGHT JOIN



INFORMATION


Gives summary information
about the dataframe

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 2919 entries, 0 to 2918
Data columns (total 81 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Id                     2919 non-null   int64
1   MSSubClass             2919 non-null   int64
2   MSZoning               2915 non-null   object
3   LotFrontage           2433 non-null   float64
4   LotArea               2919 non-null   int64
5   Street                2919 non-null   object
6   Alley                 198 non-null    object
7   LotShape              2919 non-null   object
8   LandContour           2919 non-null   object
9   Utilities             2917 non-null   object
10  LotConfig             2919 non-null   object
11  LandSlope             2919 non-null   object
12  Neighborhood          2919 non-null   object
13  Condition1            2919 non-null   object
14  Condition2            2919 non-null   object
15  BldgType              2919 non-null   object
16  HouseStyle            2919 non-null   object
17  OverallQual           2919 non-null   int64
18  OverallCond           2919 non-null   int64
19  YearBuilt             2919 non-null   int64
20  YearRemodAdd          2919 non-null   int64
21  RoofStyle             2919 non-null   object
22  RoofMatl             2919 non-null   object
23  Exterior1st           2918 non-null   object
24  Exterior2nd           2918 non-null   object
25  MasVnrType            2895 non-null   object
```

FILTERING DATA


Shows only data matching the
specified condition



```
df[df['PoolQC'].isna() == False]
```

DROP OR FILL NULL VALUES

Deletes rows where there is a
null value in the 'Utilities'
column



```
df = df.dropna(subset=['Utilities'])  
df = df.fillna({'Functional':0})  
|
```



Fills null values with zeroes

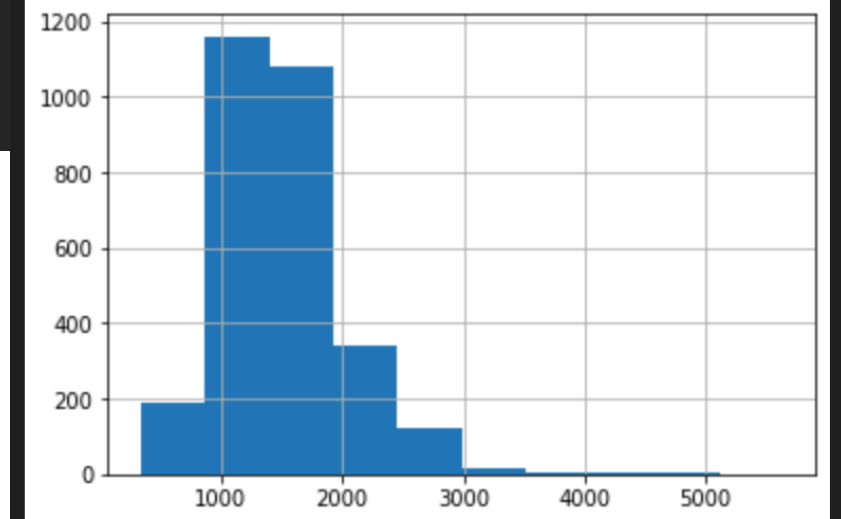
INFORMATION ABOUT THE DATA

Output a statistical description
of the column

```
print(df['GrLivArea'].describe())  
df['GrLivArea'].hist()
```


Histogram of the distribution of
the column

```
count    2919.000000  
mean     1500.759849  
std       506.051045  
min       334.000000  
25%      1126.000000  
50%      1444.000000  
75%      1743.500000  
max       5642.000000  
Name: GrLivArea, dtype: float64
```




STANDARDIZE/NORMALIZE DATA

Standardize data to have mean of 0 and standard deviation of 1



```
df['number_tweets'] = (df['number_tweets']-df['number_tweets'].mean()) / df['number_tweets'].std()
```

```
df['number_tweets'] = df['number_tweets']/df['number_tweets'].abs().max()
```



Normalize data so all values are between 0 and 1

TRAIN-VAL-TEST SPLIT

Common package for working
with arrays

```
import numpy as np
train, validate, test = np.split(df.sample(frac=1), [int(.7*len(df)), int(.9*len(df))])
```

Splits the data into
three data sets

Randomizes data
before splitting

Specifies where the splits should
happen (i.e. between train and val at
the 70% and between val and test at
90%)