DATA 101: INTRODUCTION TO DATA AND DATA PROCESSING

SPONSORED BY PURDUE WOMEN IN HPC

SARAH RODENBECK, SENIOR RESEARCH DATA SCIENTIST

WHPC WOMEN IN HIGH PERFORMANCE COMPUTING
PURDUE UNIVERSITY

Sign up for our newsletter

SCHOLARSHIPS

Present your research at conferences

CAREER DEVELOPMENT

Attend trainings and events

COMMUNITY

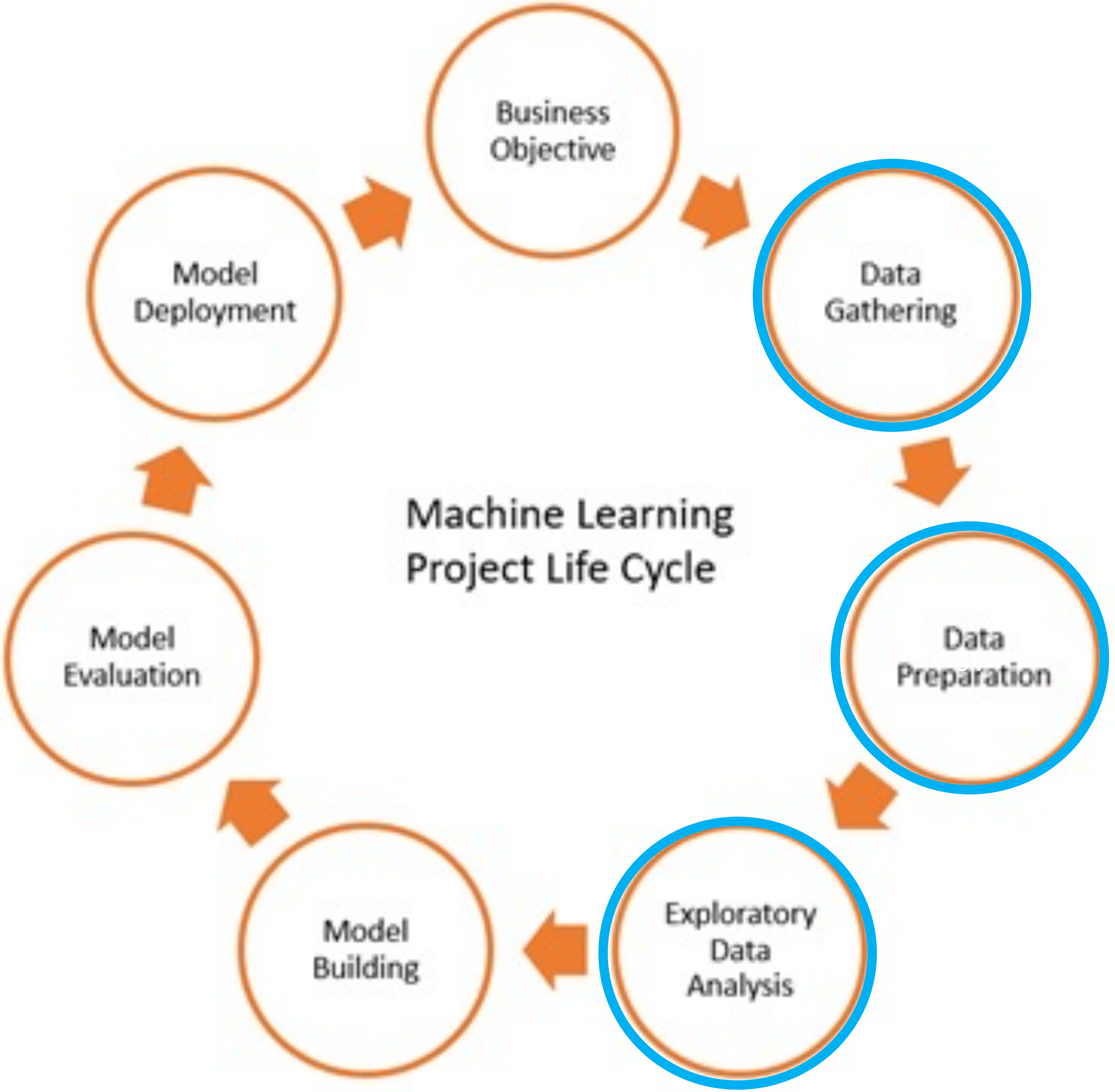
Join the conversation on Slack

OUTCOMES

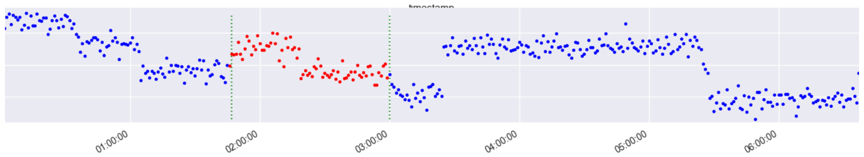
- Understand how data is used in machine learning and data science
- Data types and data collection
- Topics to consider when doing an Exploratory Data Analysis
- Basic data processing using Pandas

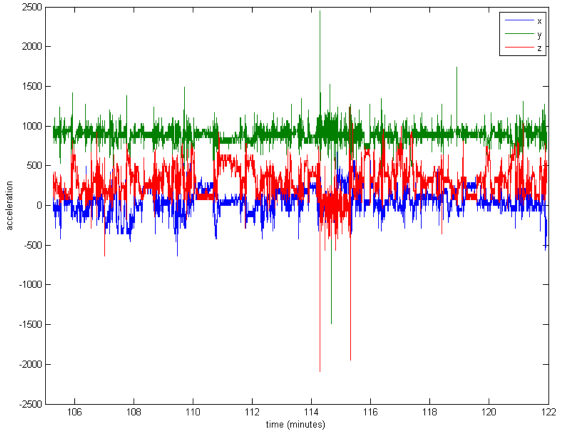
DATA BASICS

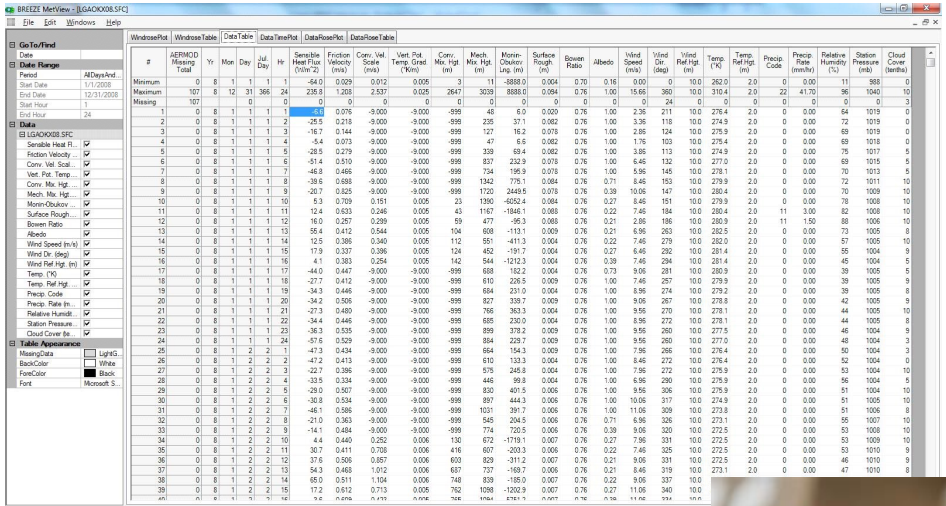
MODEL DEVELOPMENT CYCLE

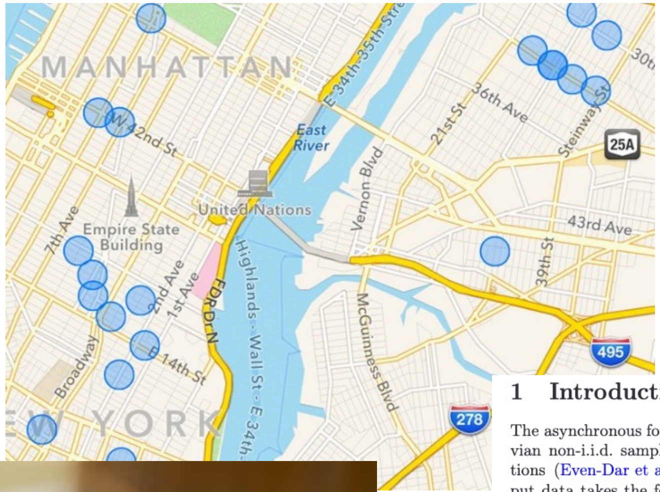


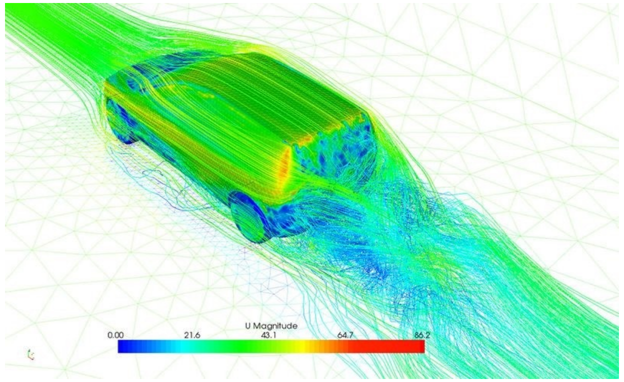
WHAT IS DATA?











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 of[1] (Li et al., 2021a)

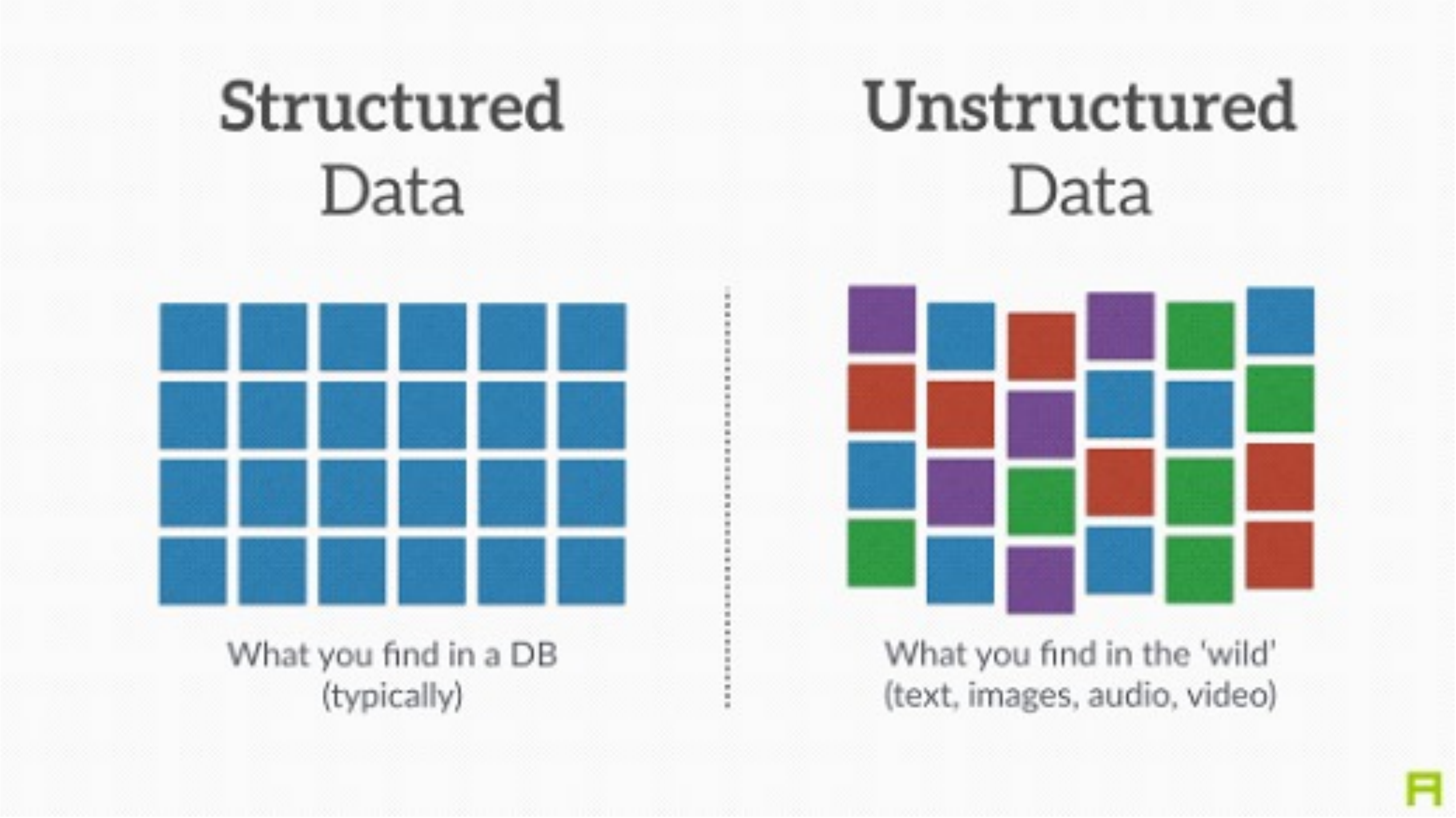
$$\frac{1}{\mu_{\min}(1-\gamma)^4\varepsilon^2} + o\left(\frac{1}{\varepsilon^2}\right) \qquad (1.1)$$

WHY DOES THIS MATTER?

Intended model dictates data needs

Available data dictates possible models

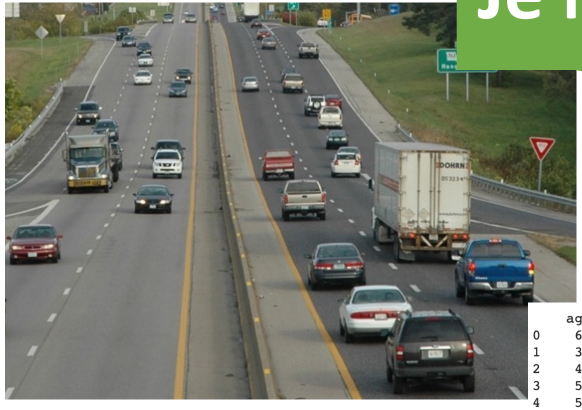
STRUCTURED VS UNSTRUCTURED DATA



LABELLED VS UNLABELLED DATA

Unlabeled Data

Je m’appelle Sarah.

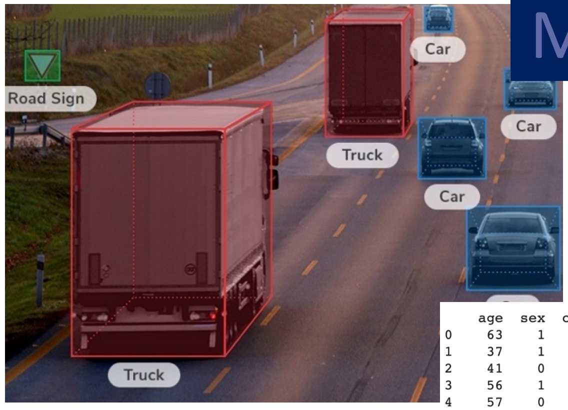


	age	sex	cp	trestbps	chol	fbs	...	exang	oldpeak	slope	ca	thal
0	63	1	3	145	233	1	...	0	2.3	0	0	1
1	37	1	2	130	250	0	...	0	3.5	0	0	2
2	41	0	1	130	204	0	...	0	1.4	2	0	2
3	56	1	1	120	236	0	...	0	0.8	2	0	2
4	57	0	0	120	354	0	...	1	0.6	2	0	2
5	57	1	0	140	192	0	...	0	0.4	1	0	1
6	56	0	1	140	294	0	...	0	1.3	1	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”**

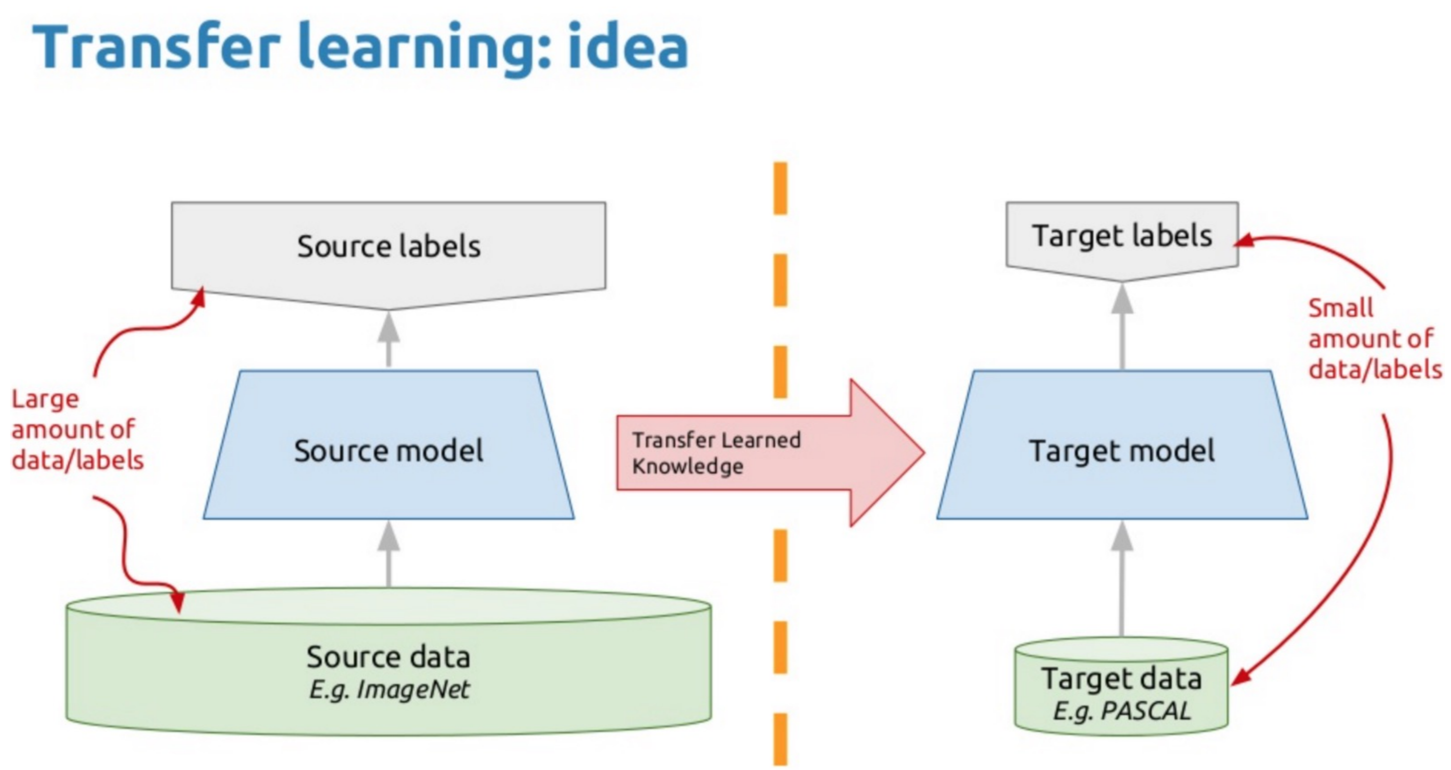
- Data preparation and limited models
  - Clustering for anomaly detection
  - Dimensionality reduction
  - Association and Recommender systems
- **Does not require labelled data**

SUPERVISED LEARNING

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

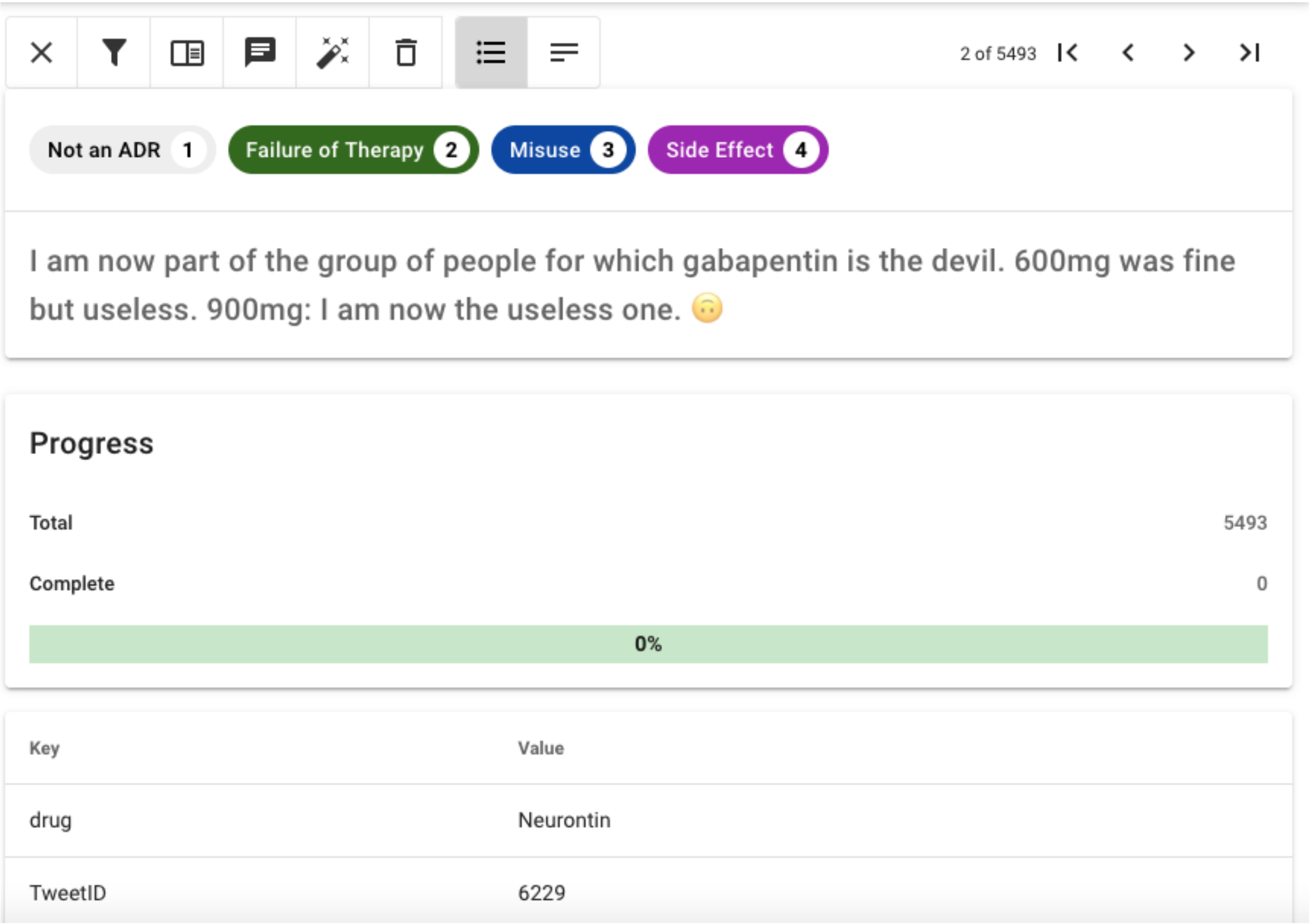
ON LABELED DATA

- 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



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**



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!**

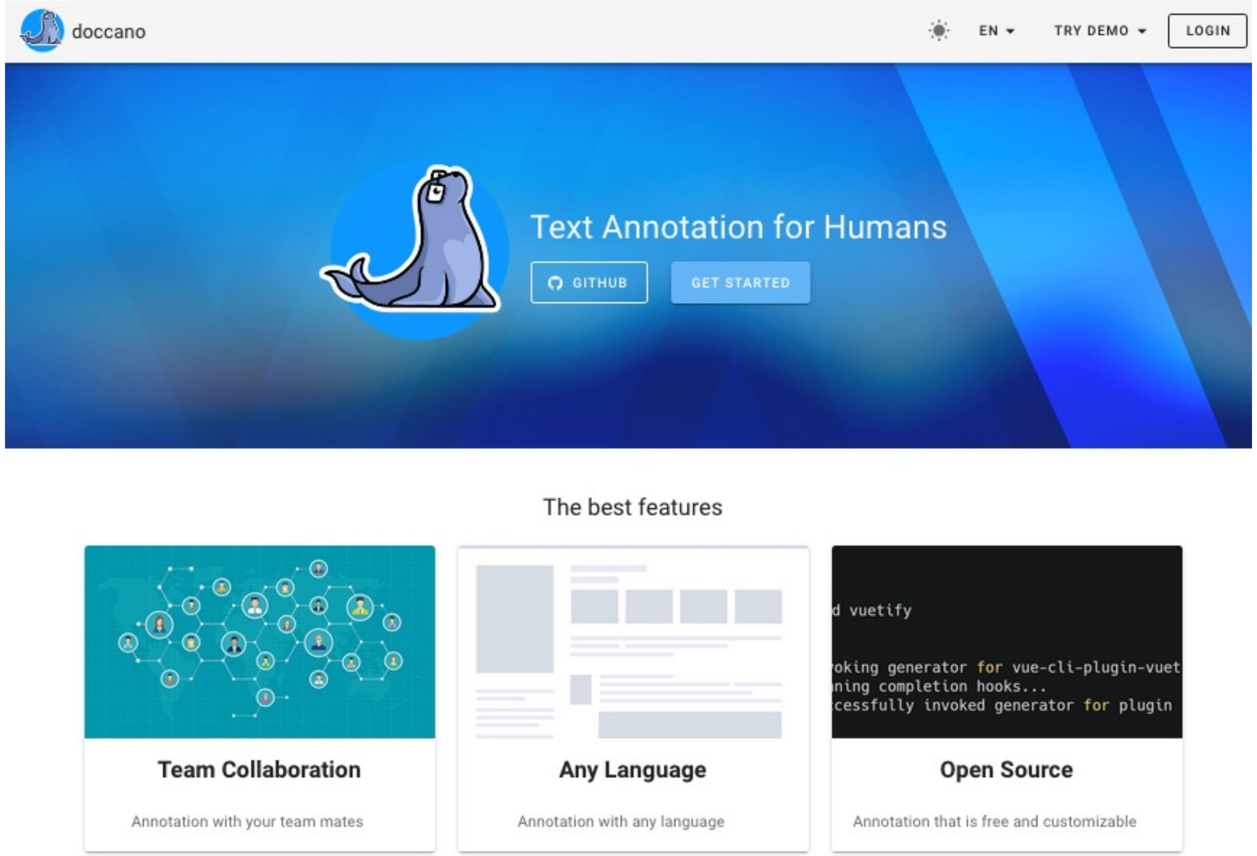
Does "weird" count as experiencing a side-effect?

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

Does this count as a car?

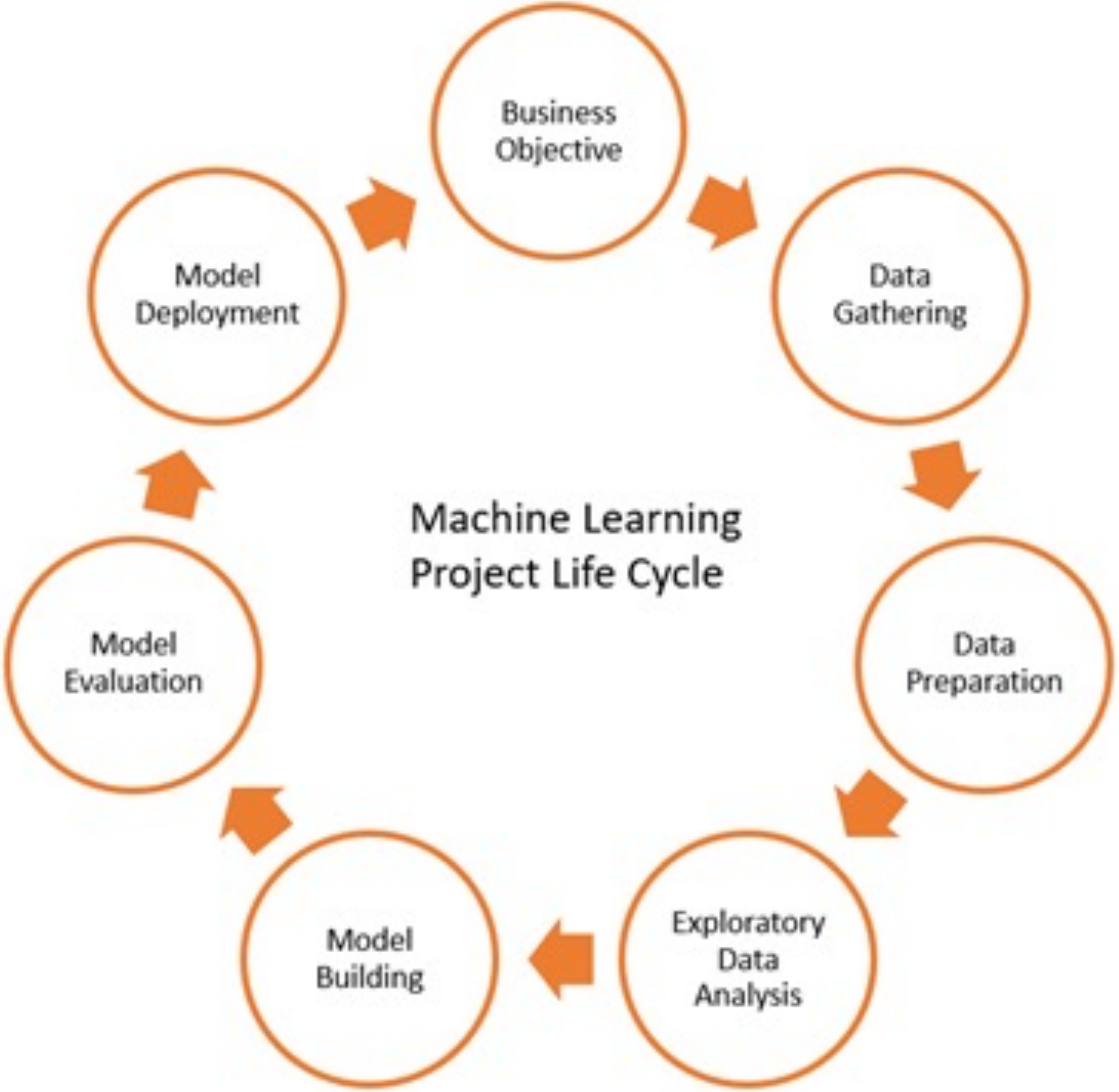
BEST PRACTICES FOR DATA ANNOTATION

- 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



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!

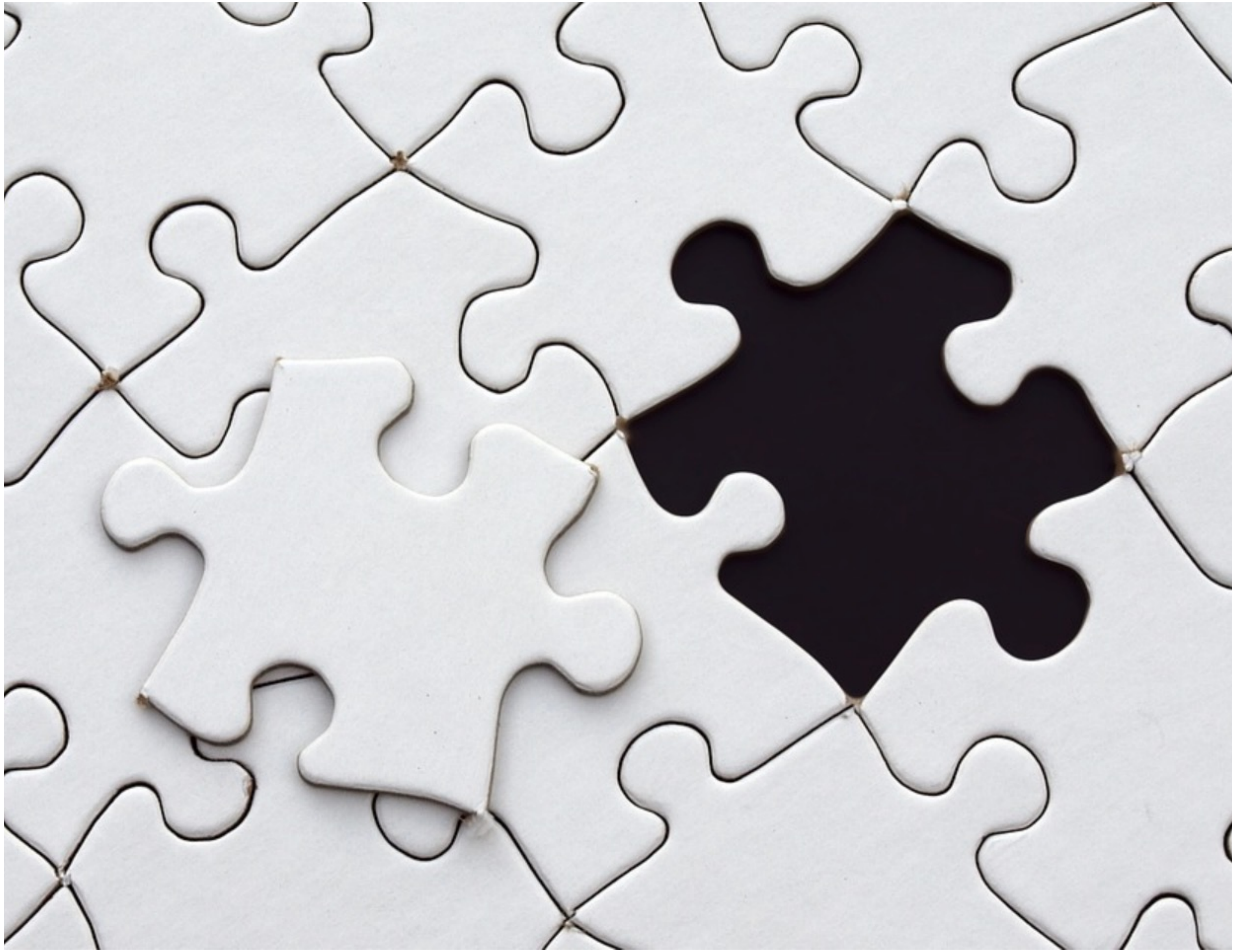
*All provided commands in this presentation are for use with the Pandas DataFrames package in Python

	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 × 8 columns

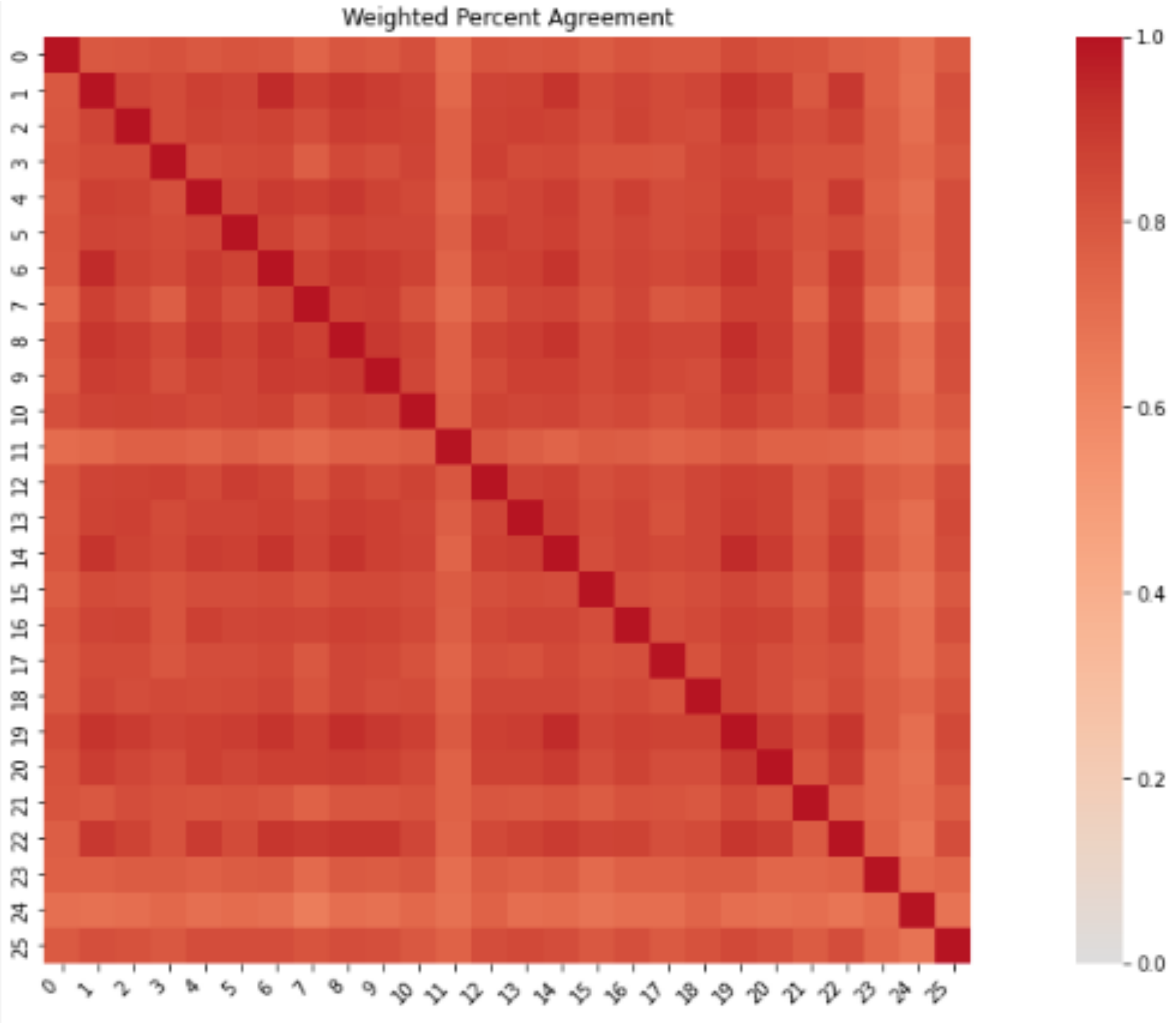
MISSING DATA

- 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

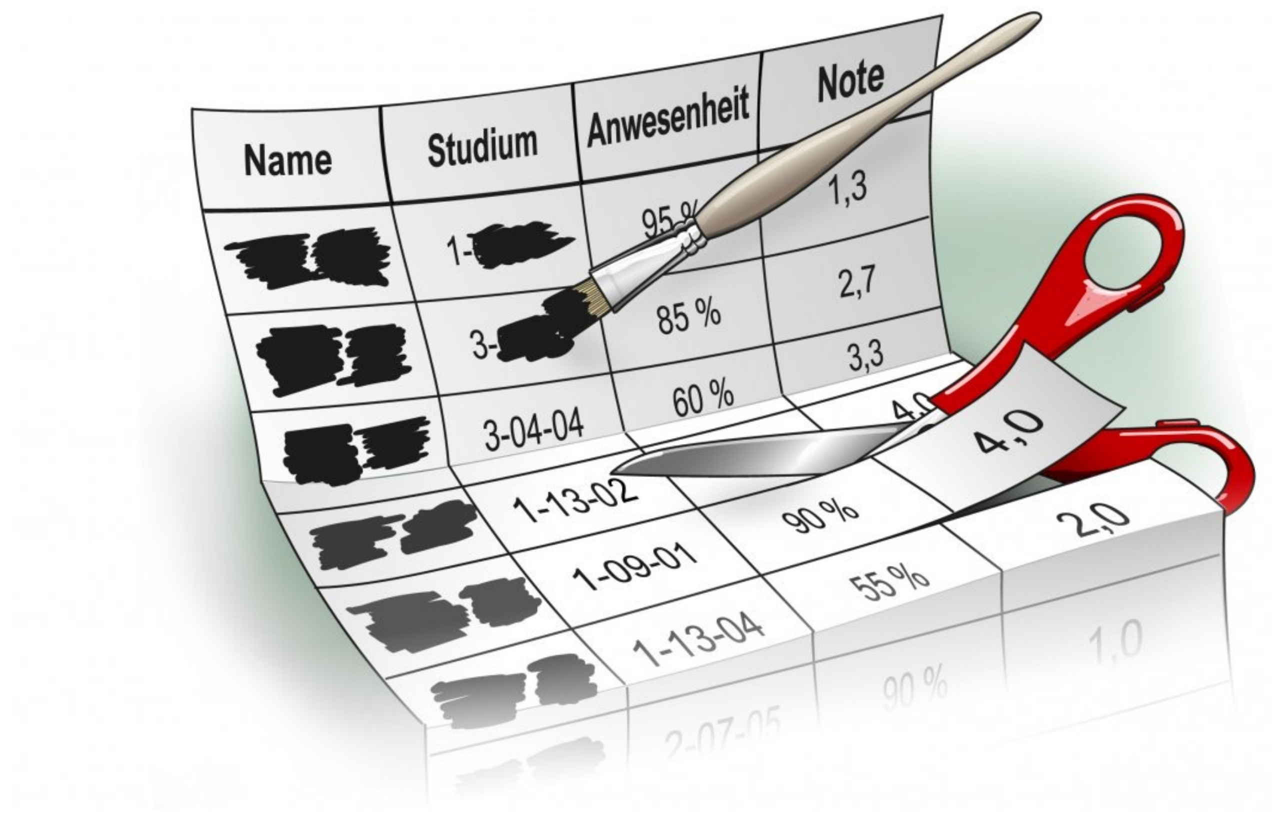
- **What you do with outliers depends on your use case**
- Are outliers important in your data (e.g. anomaly detection) or are they representative of someone not labelling the data well?
  - With the annotation data visualized here, I might want to not use annotator 11's data for model training



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…

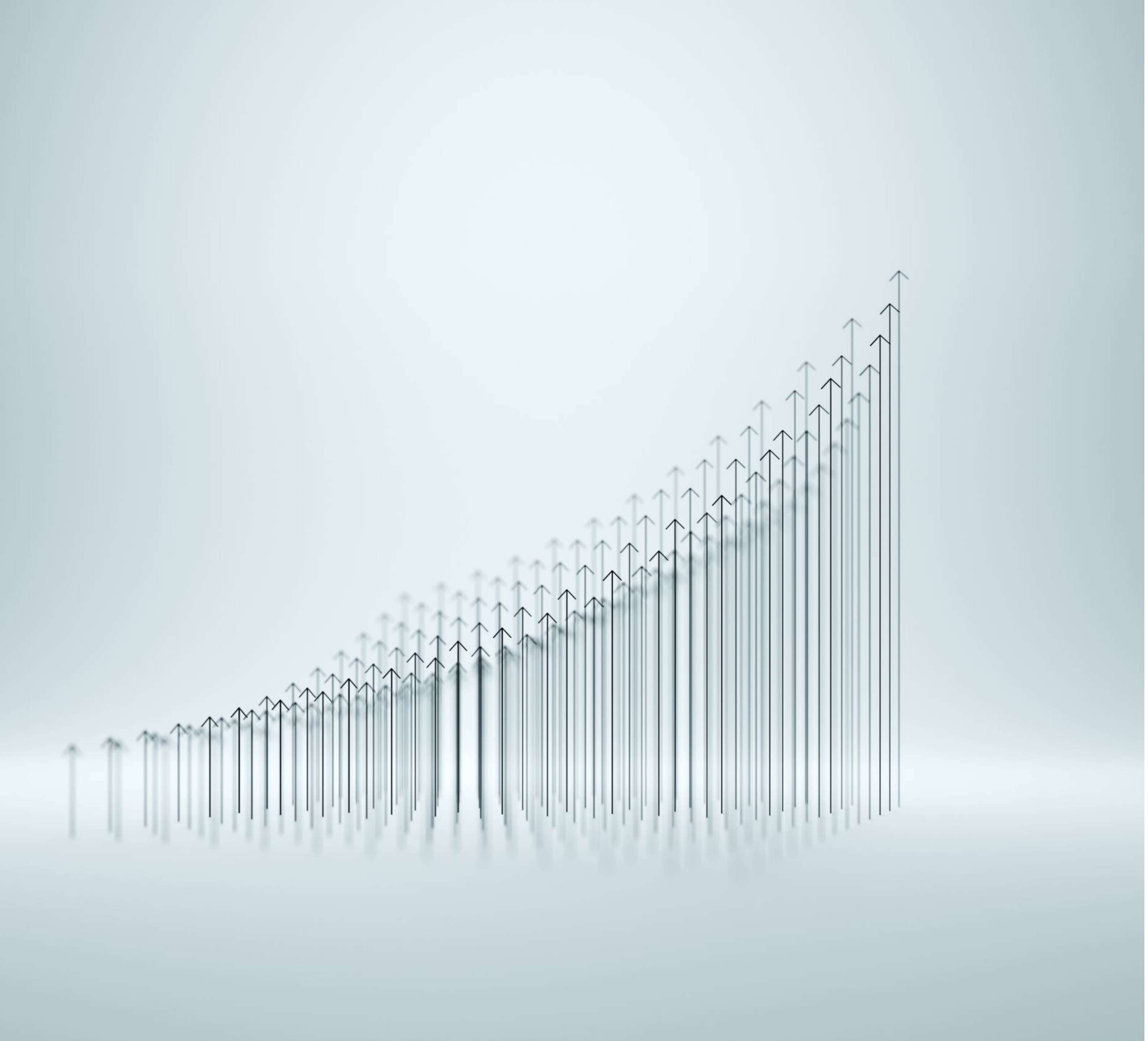
EXPLORATORY DATA ANALYSIS

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

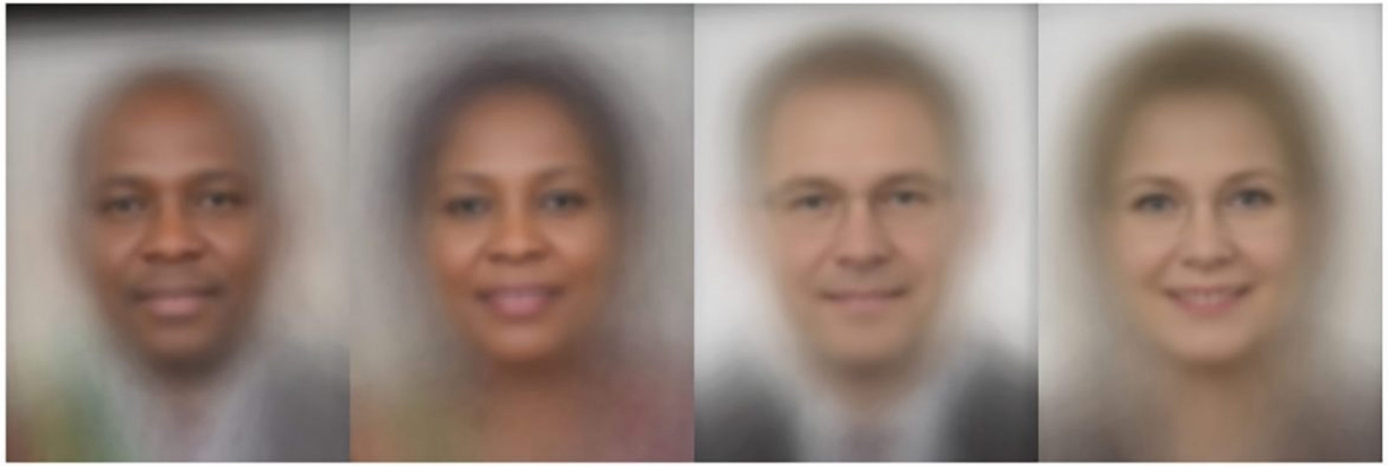
VISUALIZATIONS

Recommended Introductory Python Visualization Libraries:

- matplotlib
- seaborn

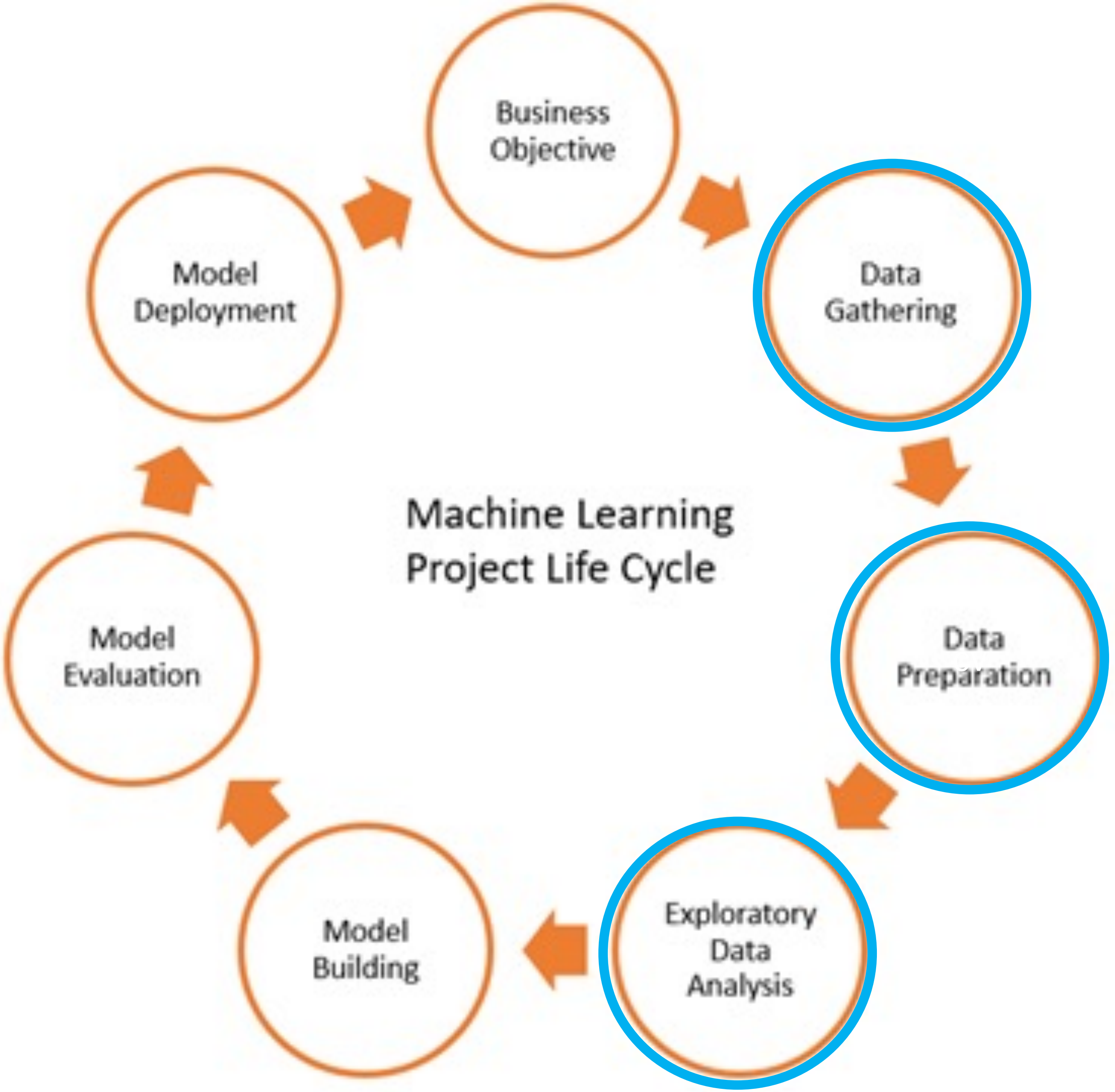
DATA BIAS

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



WHPC WOMEN IN HIGH PERFORMANCE COMPUTING
PURDUE UNIVERSITY

Sign up for our newsletter

SCHOLARSHIPS

Present your research at conferences

CAREER DEVELOPMENT

Attend trainings and events

COMMUNITY

Join the conversation on Slack

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)

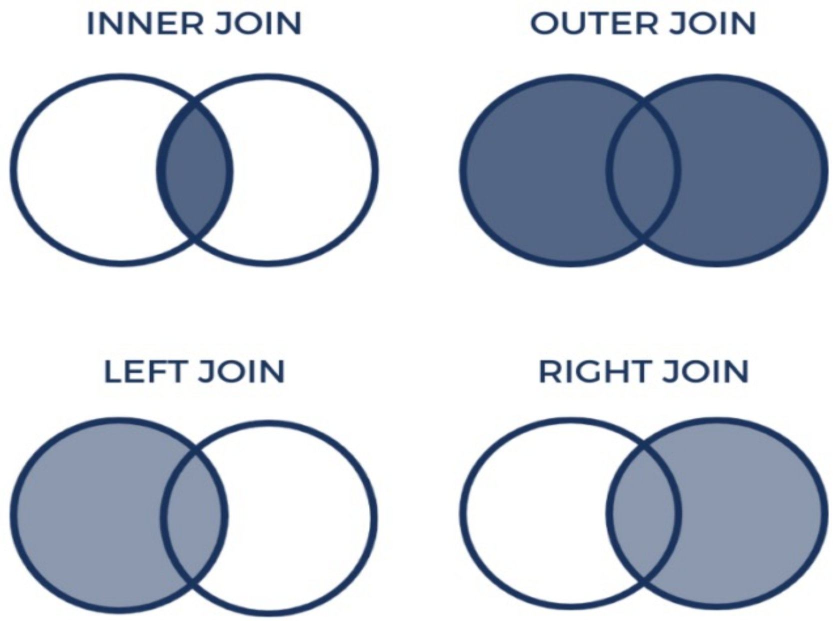
```
df = pd.concat([df1, df2, df3], ignore_index=True)
df.drop_duplicates(subset=None, keep='first', inplace=False)
```

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



INFORMATION

Gives summary information about the dataframe

```python
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

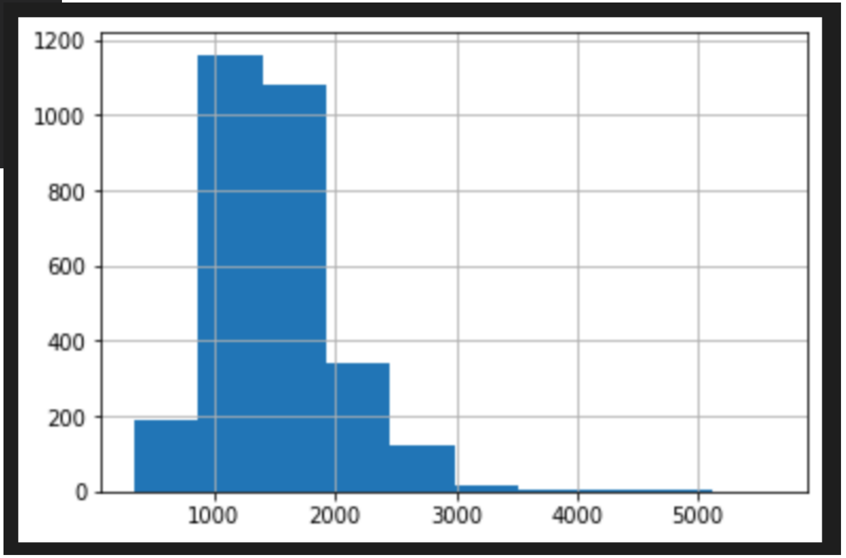
INFORMATION ABOUT THE DATA

```
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
```

Output a statistical description of the column

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

Histogram of the distribution of the column



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%)