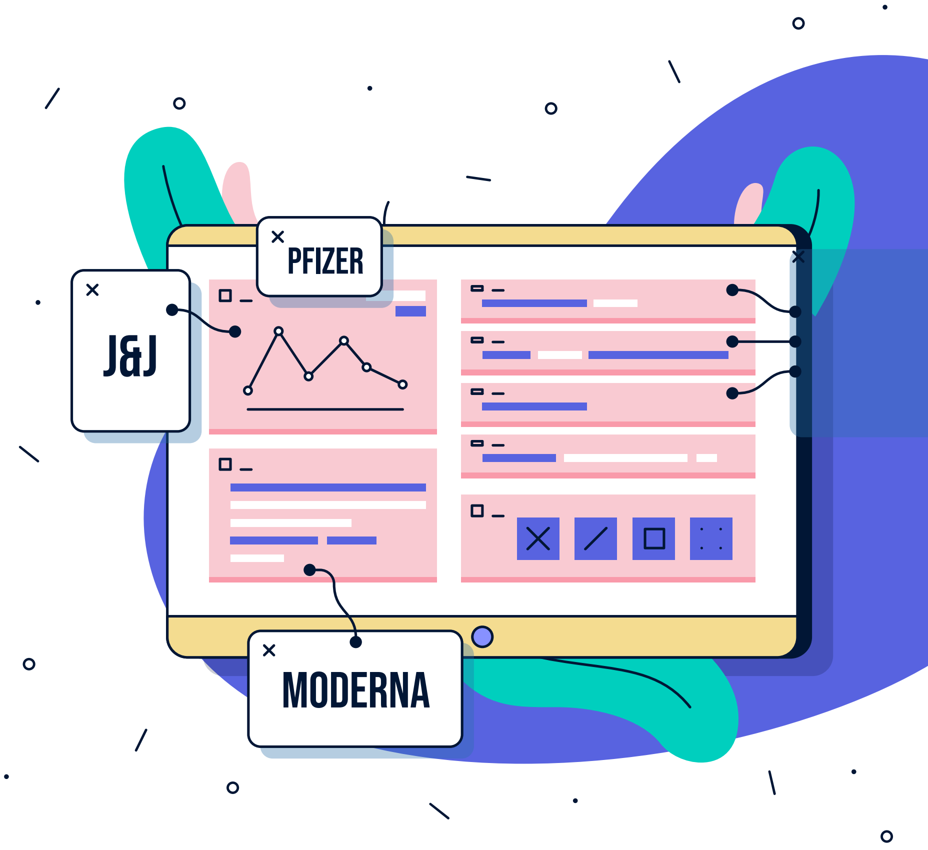
COVID VACCINE ANALYSIS

Laura Yuan
Yue Ma
Yining Ou



OBJECTIVES

PEOPLE'S PERCEPTION OF VACCINE

Sentiment Analysis. Assuming the same percentage of different vaccine recipients post their feelings on twitter.

PEOPLES REACTION ABOUT VACCINATION

World Cloud

CALIFORNIA COVID ANALYSIS

Sentiment Analysis. Evaluating COVID perception, cases, and deaths in California by county.

VACCINE BRANDS

Pfizer

Johnson & Johnson

moderna®

DATASETS

Datasets Used:

1. Vaccine Tweets (Kaggle)
2. Vaccines Administered by County (CA.gov)
3. Deaths from COVID-19 by County (CA.gov)
4. US Cities Database (SimpleMaps.com)

```
tweets.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 218894 entries, 0 to 218893
Data columns (total 13 columns):
 #   Column            Non-Null Count   Dtype
---  ------            --------------   -----
 0   user_name         218887 non-null  object
 1   user_location     171795 non-null  object
 2   user_description  206847 non-null  object
 3   user_created      218888 non-null  object
 4   user_followers    218887 non-null  float64
 5   user_friends      218887 non-null  object
 6   user_favourites   218887 non-null  object
 7   user_verified     218887 non-null  object
 8   date              218885 non-null  object
 9   text              218887 non-null  object
 10  hashtags          157270 non-null  object
 11  source            216489 non-null  object
 12  is_retweet        218880 non-null  object
dtypes: float64(1), object(12)
memory usage: 21.7+ MB
```

```
vacc_admin.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 9534 entries, 0 to 9533
Data columns (total 17 columns):
 #   Column                        Non-Null Count  Dtype
---  ------                        --------------  -----
 0   county                        9534 non-null   object
 1   administered_date             9534 non-null   object
 2   total_doses                   9534 non-null   int64
 3   cumulative_total_doses        9534 non-null   int64
 4   pfizer_doses                  9534 non-null   int64
 5   cumulative_pfizer_doses       9534 non-null   int64
 6   moderna_doses                 9534 non-null   int64
 7   cumulative_moderna_doses      9534 non-null   int64
 8   jj_doses                      9534 non-null   int64
 9   cumulative_jj_doses           9534 non-null   int64
 10  partially_vaccinated          9534 non-null   int64
 11  total_partially_vaccinated    9534 non-null   int64
 12  fully_vaccinated              9534 non-null   int64
 13  cumulative_fully_vaccinated   9534 non-null   int64
 14  at_least_one_dose             9534 non-null   int64
 15  cumulative_at_least_one_dose  9534 non-null   int64
 16  california_flag               9216 non-null   object
dtypes: int64(14), object(3)
memory usage: 1.2+ MB
```

```
deaths.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7605 entries, 0 to 7604
Data columns (total 8 columns):
 #   Column                   Non-Null Count  Dtype
---  ------                   --------------  -----
 0   demographic_category     7605 non-null   object
 1   demographic_value        7605 non-null   object
 2   total_cases              7605 non-null   int64
 3   percent_cases            7605 non-null   float64
 4   deaths                   7605 non-null   int64
 5   percent_deaths           7605 non-null   float64
 6   percent_of_ca_population 7585 non-null   float64
 7   report_date              7605 non-null   object
dtypes: float64(3), int64(2), object(3)
memory usage: 475.4+ KB
```

```
uscities.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 28338 entries, 0 to 28337
Data columns (total 17 columns):
 #   Column        Non-Null Count  Dtype
---  ------        --------------  -----
 0   city          28338 non-null  object
 1   city_ascii    28338 non-null  object
 2   state_id      28338 non-null  object
 3   state_name    28338 non-null  object
 4   county_fips   28338 non-null  int64
 5   county_name   28338 non-null  object
 6   lat           28338 non-null  float64
 7   lng           28338 non-null  float64
 8   population    28338 non-null  int64
 9   density       28338 non-null  int64
 10  source        28338 non-null  object
 11  military      28338 non-null  bool
 12  incorporated  28338 non-null  bool
 13  timezone      28338 non-null  object
 14  ranking       28338 non-null  int64
 15  zips          28337 non-null  object
 16  id            28338 non-null  int64
dtypes: bool(2), float64(2), int64(5), object(8)
memory usage: 3.3+ MB
```

DATA CLEANING

Data Cleaning

- Dropping Null/NaN values
- Dropping columns with irregular/inconsistent values
- Extracting user locations from tweets using **Regex** & **FlashGeoText**
- Merging datasets

	user_location	location	city	state	country
0	Assam	{'cities': {}, 'countries': {}}	None	NaN	None
1	Adelaide, South Australia	{'cities': {'Adelaide': {'count': 1}}, 'countr...	Adelaide	NaN	Australia
2	Hyderabad, India	{'cities': {'Hyderabad': {'count': 1}, 'Indija...	Hyderabad	NaN	India
3	The Great Pacific Northwest	{'cities': {}, 'countries': {}}	None	NaN	None
4	Washington, D.C., DC, United States	{'cities': {'Washington, D.C.': {'count': 1}},...	Washington, D.C.	DC	United States
5	Nashville, TN, United States	{'cities': {'Nashville': {'count': 1}}, 'count...	Nashville	TN	United States

01.

PEOPLE'S PERCEPTIONS OF VACCINES

SENTIMENTAL ANALYSIS

Steps:

1. Tokenize each tweet
2. Remove punctuations, stop words
3. Access Harvard Inquirer Dictionary for positive and negative words
4. Calculate statistics/counts of positive and negative words

```
In [49]: tweets['pos_words']= tweets.tokens.apply(lambda x: get_pos(x))
         tweets['neg_words']= tweets.tokens.apply(lambda x: get_neg(x))

In [50]: tweets['total_count'] = tweets.tokens.apply(lambda x: len(x))
         tweets['pos_count'] = tweets.pos_words.apply(lambda x: len(x))
         tweets['neg_count'] = tweets.neg_words.apply(lambda x: len(x))

In [51]: tweets.head(2)
```

Out[51]:

	user_name	date	text	verified	city	state	country	tokens	total_count	pos_words	neg_words	pos_count	neg_count
0	MyNewsNE	2020-08-18	Australia to Manufacture Covid-19 Vaccine and ...	0	None	NaN	None	[australia, manufactur, covid-19, vaccin, give...	13	[give, free, prime, minist]	[cost]	4	1
1	Ann-Maree O'Connor	2020-08-18	@michellegrattan @ConversationEDU This is what...	0	Adelaide	NaN	Australia	[michellegrattan, conversationedu, pass, leade...	10	[pass]	[pass]	1	1

SENTIMENTAL ANALYSIS

```python
from tqdm.auto import tqdm
from tqdm import tqdm
tqdm.pandas()
```

```python
tweets['sentiment'] = tweets.text.progress_apply(lambda x: scoreSentiment(x))
```

```
100%|██████████| 171782/171782 [4:14:18<00:00, 11.26it/s]
```

```python
tweets['sentiment_label'] = tweets.sentiment.apply(lambda x: x[0]['label'])
tweets.head(2)
```

state	country	tokens	pos_words	neg_words	total_count	pos_count	neg_count	sentiment	sentiment_label
NaN	None	[australia, to, manufactur, covid-19, vaccin, ...	[give, free, prime, minist]	[cost]	20	4	1	[{'label': 'POSITIVE', 'score': 0.563022434711...	POSITIVE
NaN	Australia	[michellegrattan, conversationedu, thi, is, wh...	[pass, our]	[pass]	19	2	1	[{'label': 'POSITIVE', 'score': 0.814165651798...	POSITIVE

SENTIMENTAL ANALYSIS

```
tweets.groupby('sentiment_label')['sentiment_label'].count()
```

```
sentiment_label
NEGATIVE    132883
POSITIVE     38899
Name: sentiment_label, dtype: int64
```

SENTIMENTAL ANALYSIS

```
total_pos = {}
def count_total_pos(lst):
    for pos in lst:
        if pos not in total_pos:
            total_pos[pos] = 1
        else:
            total_pos[pos] += 1
```

```
tweets.pos_words.progress_apply(lambda x: count_total_pos(x))
```

```
df_pos = pd.DataFrame(total_pos,index=[0]).T
df_pos.rename(columns = {0:'counts'}, inplace = True)
df_pos.sort_values(by = 'counts', ascending = False).head(10)
```

SENTIMENTAL ANALYSIS

```
tweets_sentiment_by_date = tweets[['date','sentiment_label']]
```

```
tweets_sentiment_by_date.set_index('date',inplace = True)
```

```
tweets_sentiment_by_date.head()
```

	sentiment_label
date	
2020-08-18	POSITIVE
2020-08-18	POSITIVE
2020-08-18	NEGATIVE
2020-08-18	NEGATIVE
2020-08-18	NEGATIVE

SENTIMENTAL ANALYSIS

date	negative	positive	total	neg_rate	pos_rate
2020-01-09	66.0	17.0	83.0	0.795181	0.204819
2020-02-09	97.0	15.0	112.0	0.866071	0.133929
2020-03-09	143.0	12.0	155.0	0.922581	0.077419
2020-04-09	116.0	24.0	140.0	0.828571	0.171429
2020-05-09	59.0	9.0	68.0	0.867647	0.132353

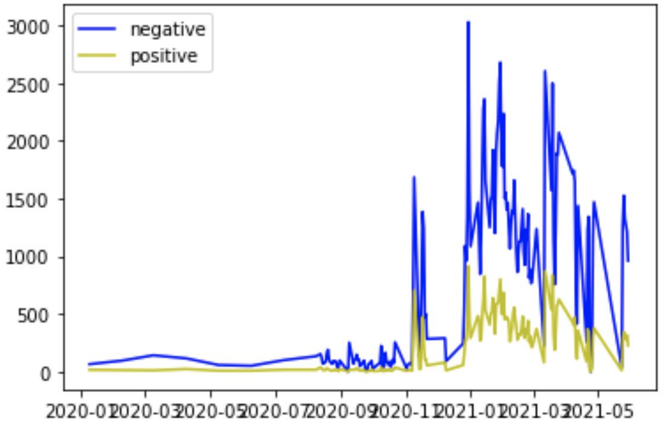
SENTIMENTAL ANALYSIS

```
plt.plot(tweets_sentiment_count_by_date.negative, 'b')
plt.plot(tweets_sentiment_count_by_date.positive, 'y')
plt.legend(['negative', 'positive'])
```

```
[<matplotlib.lines.Line2D at 0x7face0a08b70>]

[<matplotlib.lines.Line2D at 0x7face0a55c88>]

<matplotlib.legend.Legend at 0x7face09c39b0>
```

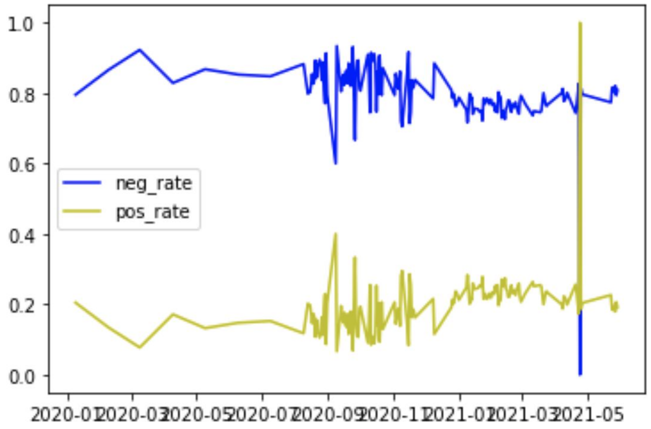



```
plt.plot(tweets_sentiment_count_by_date.neg_rate, 'b')
plt.plot(tweets_sentiment_count_by_date.pos_rate, 'y')
plt.legend(['neg_rate', 'pos_rate'])
```

```
[<matplotlib.lines.Line2D at 0x7face0599518>]

[<matplotlib.lines.Line2D at 0x7face0593470>]

<matplotlib.legend.Legend at 0x7face05494a8>
```



VACCINE SENTIMENT FOR ALL COUNTRIES

```
On average, the percent of positive words count for Pfizer is 0.0909262797833859
On average, the percent of positive words count for Moderna is 0.0718102494947933
On average, the percent of positive words count for J&J is 0.0673076923076923
On average, the percent of positive words count COVID vaccines is 0.10358571678180302
```

02.

PEOPLES REACTION ABOUT VACCINATION

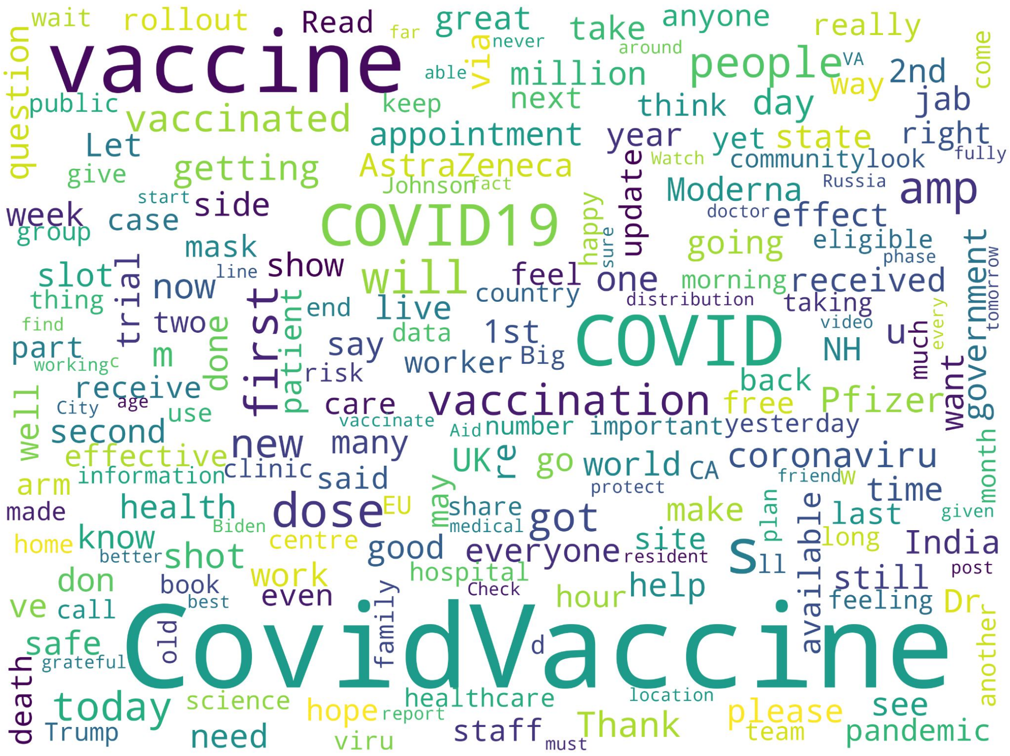
WORDCLOUD

Creating the Word Cloud:

- Import tool library
- Read data
- Clean data
- Count the word frequency
- Draw a word cloud diagram

Findings

- Brand name
- Government officials
- Number of inoculations
- Side effects of the vaccine



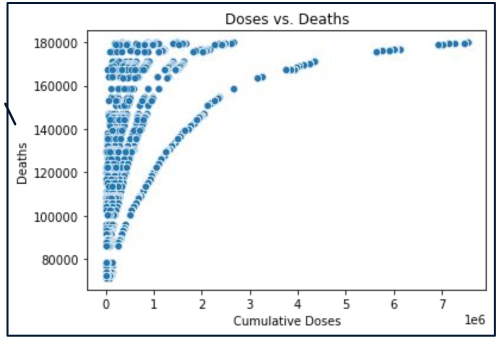
03.

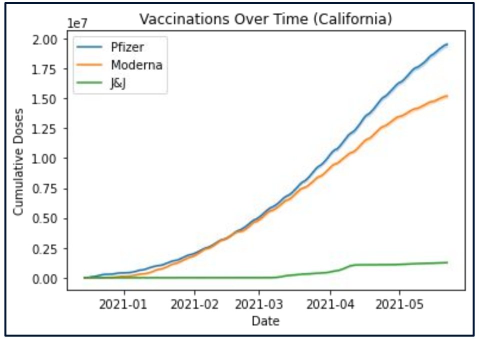
CALIFORNIA COVID ANALYSIS

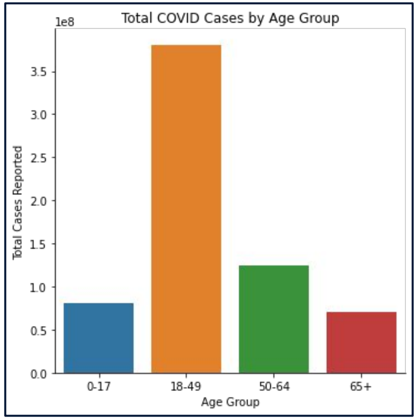
```
cali_data.info()
```

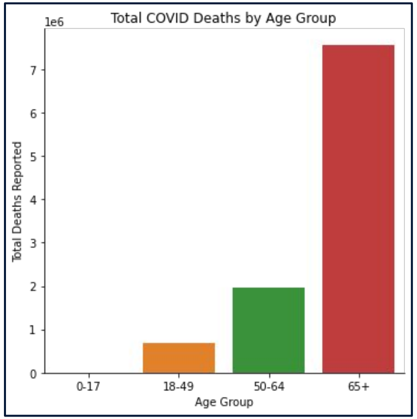
```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 4587 entries, 0 to 4586
Data columns (total 13 columns):
 #   Column                  Non-Null Count  Dtype
---  ------                  --------------  -----
 0   user_name               4587 non-null   object
 1   date                    4587 non-null   datetime64[ns]
 2   text                    4587 non-null   object
 3   city                    4587 non-null   object
 4   state                   4587 non-null   object
 5   country                 4562 non-null   object
 6   county                  4587 non-null   object
 7   cumulative_total_doses  4587 non-null   int64
 8   deaths                  4587 non-null   int64
 9   pos_count               4587 non-null   int64
 10  neg_count               4587 non-null   int64
 11  percent_pos             4038 non-null   float64
 12  percent_neg             4038 non-null   float64
dtypes: datetime64[ns](1), float64(2), int64(4), object(6)
memory usage: 501.7+ KB
```

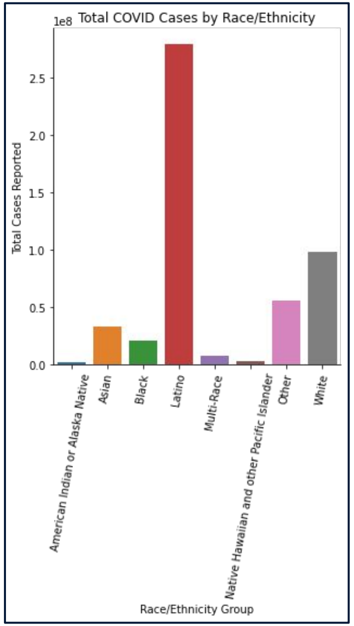
DATA VISUALIZATION

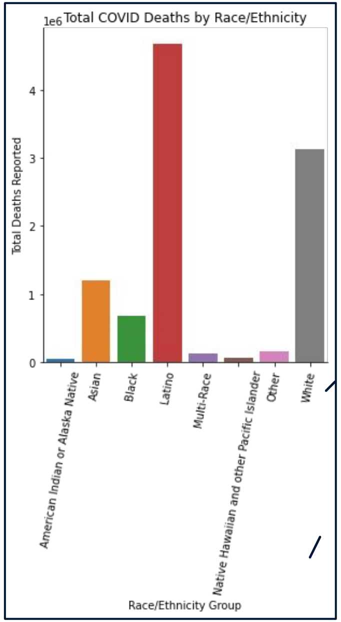












FINDINGS

NLP Applications:

- NLTK (Stop Words)
- Harvard Inquirer Dictionary (Sentiment Analysis)

Summary of Findings:

1. Highest correlation is identified between **number of doses administered** and **deaths** due to COVID
2. Most optimistic residents are in **San Benito** county
3. Most pessimistic residents are in **Santa Cruz** county
4. Most deaths are reported in **Los Angeles** county

```
Top Correlations between Variables:
cumulative_total_doses  deaths          0.603767
deaths                  percent_pos     0.039459
cumulative_total_doses  percent_pos     0.012439
```

county	cumulative_total_doses	deaths	percent_pos
San Benito	3749	119888	1.000000
Imperial	2794	86377	1.000000
Merced	8902	105948	1.000000
Tulare	400534	561220	0.833333
Marin	613149	837798	0.760000

county	cumulative_total_doses	deaths	percent_pos
Santa Cruz	234354	251680	0.000000
Kings	32072	167347	0.000000
Humboldt	142591	659293	0.300000
Monterey	766014	1379553	0.408333
Kern	1640959	1452939	0.435185

county	cumulative_total_doses	deaths	percent_pos
Los Angeles	4059770460	253284083	0.569917
San Francisco	189902708	106149896	0.596262
San Diego	337688973	58223011	0.587036
Orange	203490112	45284675	0.522773
Alameda	96835749	32813881	0.598625

04.

CONCLUSION

FINDINGS/LIMITATIONS

Findings:

- People actually pay attention to the brand of vaccines
- In California, the number of people vaccinated is increasing day by day
- For some fixed collocations, such as side effect and Johnson & Johnson, NLP recognizes them as two words. This may be the reason why the recognition results are not accurate enough.

Limitations:

1. Lack of Tweets data
2. Lack of international data on COVID cases and deaths
3. Tweets are not always a good indicator of sentiment
4. There is a problem with the existing analysis tools, vaccine have side effects, but it doesn't mean people have a negative attitude towards it.
5. Omitted variables that are unaccounted for

THANK YOU!

Group 7

Laura Yuan

Yue Ma

Yining Ou