Intersecting Social Media, Mental Health, and COVID-19: Predictive Analysis from Twitter Data

-- A study about predicting COVID-19 cases with social media and mental health data

Authors

Yunhua Su/ Jinmeng Zhang/ Yuqin Jiao

Abstract

In this research, we aim to predict daily increases in COVID-19 confirmed and death cases using data from Tweets' sentiments and mental health statistics. We use logistic and linear regression models for prediction. Logistic regression model is for creating a binary classification model determining whether the daily increase in confirmed/death cases is high or low(compared with the median daily increase number). After preprocessing and merging data, we trained our model, achieving a training accuracy of 75.55% and testing accuracy of 67.24%. Linear regression model is for predicting the daily increase case number. Each model used features such as sentiment scores and tweet counts related to COVID-19 and vaccines. Moreover, we use regularization and feature engineering for better model performance. Then we evaluate and discuss our model performance, and identify areas for model improvement, such as dataset extension and more feature inclusion. We also discuss the societal impacts and ethical concerns related to our study, such as user privacy and potential bias. Our research offers insights into public health policy and communication strategies during a pandemic and the implementation of social media data to understand public sentiment.

Keyword

Tweet, Social Media, Covid, vaccine, mental health, nlp, sentiment

1. Introduction, motivation, and presentation of project questions

1.1 Research question, motivation and introduction

Social media has become an integral part of our daily lives, and its impact on health care has been significant, especially during the COVID-19 pandemic. Our study explores the relationship between tweets data, mental health issue data and COVID-19 case growth to predict the daily increasing number from confirmed cases and deaths using datasets "time_series_covid19_confirmed_US.csv," "time_series_covid19_deaths_US.csv," from provided DatasetA, "nchs_covid_indicators_of_anxiety_depression.csv" from provided DatasetB, "COVID-19 Tweets Dataset"[1], and "Covid Vaccine Tweets"[2]. Therefore, our

research question is “can we use the sentiments from tweets and mental health data to predict the case number of COVID-19 during the pandemic”?

In order to achieve our goal, we dig deep inside Tweets datasets with sentiment analysis through NLP tool sets to learn about what people care about and how public sentiment changes through different periods of the pandemic. We also examine correlations between social media activity and pandemic progression.

Finally, we build a classification and predictive model for the number of daily increasing confirmed cases, which is an important parameter for medical institutions and relative governments departments to manage resources and develop policies during the pandemic. Our research offers insights on public sentiment, epidemic situation, and social media's impact on public health, informing policy and communication strategies during the pandemic.

1.2 Research gap

On the one hand, some previous work has examined the role of social media and mental health separately in the context of COVID-19, however, few have investigated their intersection. This is a critical gap, given the complicated relationship of these elements, especially during a pandemic.

On the other hand, although some studies have had sentiment analysis on social media data to predict COVID-19 case trends, most of these have not incorporated mental health data in their models. This is an important consideration given the considerable impact of the pandemic on mental health. Moreover, the role of sentiment towards the COVID-19 vaccination and its impact on case rates remains underexplored.

In this case, our study aims to provide a comprehensive understanding of the relationship between social media, mental health, and COVID-19 case rates, and to develop a more accurate and granular predictive model that accounts for these multiple aspects.

2. Exploratory data analysis and data processing methods

2.1 Basic dataset

To find the COVID-19 infection and death numbers in the US, we use two reports from DatasetA: "time_series_covid19_confirmed_US.csv" and "time_series_covid19_deaths_US.csv". These datasets provide daily confirmed cases or death numbers for US counties from 1/22/2020 to 3/28/2022. In these two datasets, each row represents a county, including attributes like country, state, latitude, longitude, combined key, and population. We aggregate confirmed cases from "time_series_covid19_confirmed_US.csv" to calculate the total US confirmed number from known counties from 1/22/2020 to 3/28/2022.

And then use n day's confirmed number minus the n-1 day's confirmed number to get the daily increasing number named "diff", shown in Table1. A similar process is applied to "time_series_covid19_deaths_US.csv" for death numbers.

Date	confirmed_number	yesterday_confirmed	diff
03/24/2022	79333998	79291983	42015
03/25/2022	79379719	79333998	45721

Table1. Daily increase number in US based on "time_series_covid19_confirmed_US.csv". This dataframe has a daily confirmed case number and uses "yesterday_confirmed" minus "confirmed_number " get "diff" column.

We also explore the "nchs_covid_indicators_of_anxiety_depression.csv" data set in provided DatasetB. This dataset contains the prevalence of depressive/anxiety disorder symptoms in the United States, segmented by various demographic factors such as age, sex, race/ethnicity, education, and state, during the time period of April 23,2020 to March 14, 2022. To aggregate the reported depressive or anxiety disorder symptoms in the US, we fill everyday's mental issue report rate from the weekly data, and then pivot the dataset to a dataframe showing in Table2.

Date	average	median
04/23/2020	34.567308	34.7
04/24/2020	34.567308	34.7

Table2. Daily reported mental issue(depressive or anxiety disorder symptoms) rate(average and median in all states) in the US based on "nchs_covid_indicators_of_anxiety_depression.csv". This dataframe has a daily reported average and median mental health issue rate.

2.2 Tweets Datasets

2.2.1 General information of the tweets datasets

For our other prediction features, we find two additional datasets: "COVID-19 Tweets Dataset"[1] and "Covid Vaccine Tweets"[2]. The "COVID-19 Tweets Dataset"(Table3) presents daily sentiment data (negative, neutral, positive) with respective counts("Count") and average percentage per sentiment category("Avg_Per"), from January 29, 2020, to January 7, 2022.

Date	Sentiment	Count	Avg_Per
01/07/2022	negative	492500	0.873165
01/07/2022	neutral	56138	0.098453
01/07/2022	positive	15951	0.028382

Table3. Daily tweets' sentiment, numbers of negative/neutral/positive tweets and percentage respectively from "COVID-19 Tweets Dataset". This dataframe shows how many tweets in a day are negative/neutral/positive.

We do both EDA and sentiment analysis based on NLP toolkits from nltk, huggingface on the other additional dataset "Covid Vaccine Tweets". This dataset(Table4) tracks daily tweets related to COVID-19 vaccines, including user details, tweet content and tweets' source. We first clean the data, picking all tweets from the US, then we apply the nature language processing model to the cleaned dataframe.

date	user_location	user_name	text	hashtags	source
18-08-2020 12:55	Assam	MyNewsNE	"Australia to Manufacture Covid-19 Vaccine and give it to the Citizens for free of cost: AFP quotes Prime Minister #CovidVaccine"	['CovidVaccine']	Twitter Web App
17-08-2020 21:10	New York	Public Seminar	"A vaccine can protect us from an infectious disease. It cannot, however, provide immunity from the multiple personâ€¦ https://t.co/DjpYWZFp9O	NA	Twitter Web App

Table4. "Covidvacccine.csv" tweets about covid vaccine all over the world within nearly 2 years. To make the report clearer, we created this table by dropping some columns. This table shows what the raw data in "Covidvacccine.csv" looks like.

2.2.2 EDA in Mental Health Dataset

For the dataset "nchs_covid_indicators_of_anxiety_depression.csv" in datasetB, we focus on the ratio that the responders report "Symptoms of Anxiety Disorder or Depressive Disorder" mental health issue from 04/23/2020 to 03/14/2022 from different states.

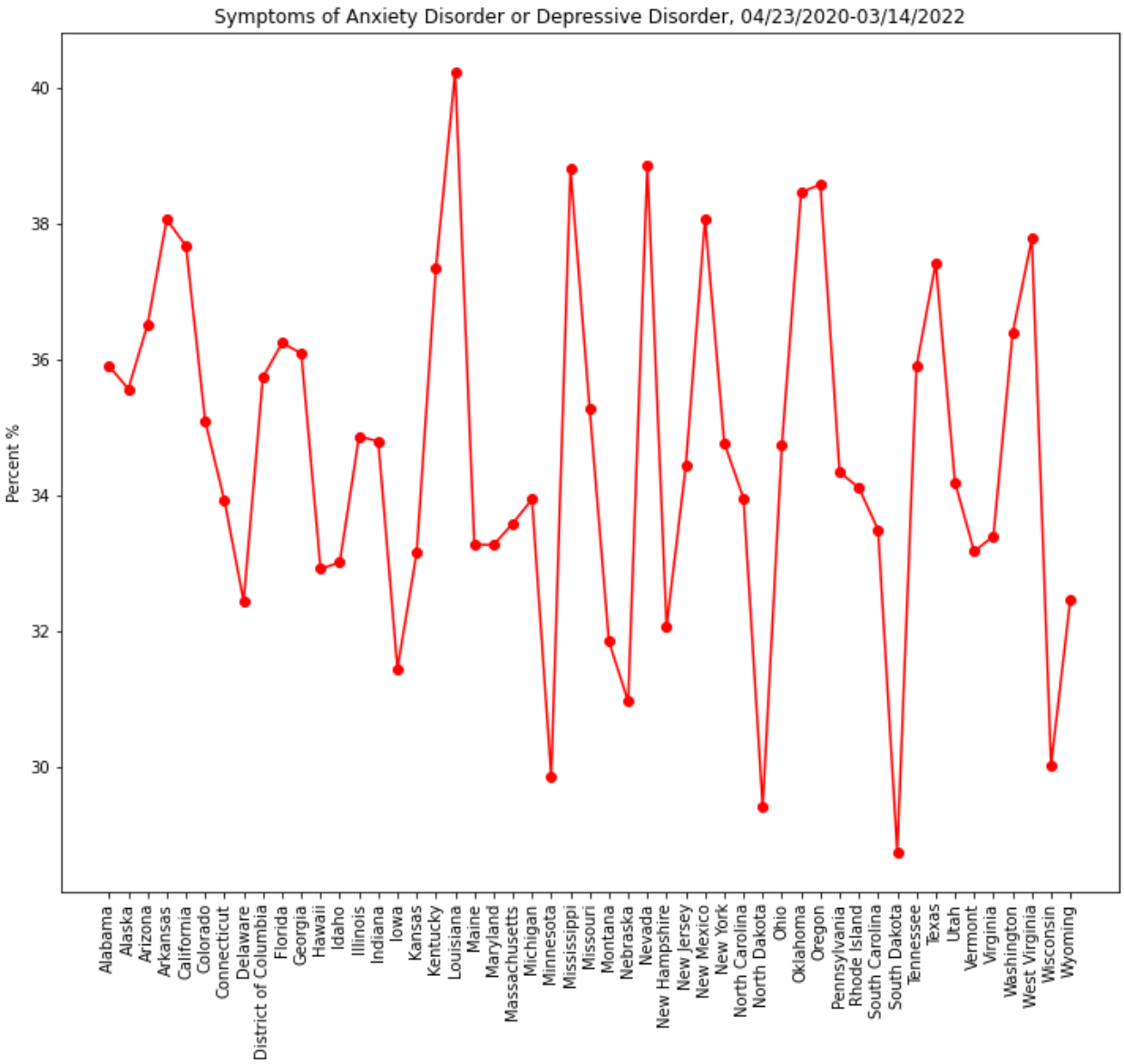

Figure1. Symptoms of Anxiety Disorder or Depressive Disorder, 04/23/2020-03/14/2022, in US. This figure shows that responders in Louisiana have the highest percentage(40.2%) of reporting "Symptoms of Anxiety Disorder or Depressive Disorder" mental health issues. Also, responders in South Dakota(28.7%) have the lowest percentage of reporting "Symptoms of Anxiety Disorder or Depressive Disorder" mental health issues. The mean value is 34.6%, which shows about 34.6% of responders have had symptoms of anxiety disorder or depressive disorder from 04/23/2020 to 03/14/2022 in the US.

This result showed that responders in Louisiana have the highest percentage(40.2%) of reporting "Symptoms of Anxiety Disorder or Depressive Disorder" mental health issues. Also, responders in South Dakota(28.7%) have the lowest percentage of reporting "Symptoms of Anxiety Disorder or Depressive Disorder" mental health issues. The mean value is 34.6%, which

shows about 34.6% of responders have had symptoms of anxiety disorder or depressive disorder from 04/23/2020 to 03/14/2022 in the US.

2.2.3 Interesting Findings in Tweets EDA

Note that we only can get the Tweets related to vaccines as the number of them is relatively small – the whole dataset of Tweets number related to COVID included more than 2 billion in just two years. Besides, there is a high charge of twitter API now, so we can only get free data from kaggle. Although the dataset of Vaccine tweets are very limited (have only 472 days effective data within 2 years) and have many faults, such as the datetime format is not unified, Data misalignment and the location information is not IP address but the one that users can customized to something like ‘My bed’ or ‘earth’, resulting in nearly 50,000 types of location names.

One of the most largest problems in this tweets dataset is that with a rough analysis of the distribution of the tweets location, we can see that mostly, approximately more than 50% are from India, which shall be useless in our US-based study. The US tweets might only take a proportion of 25%[Figure2]. Plus the problem of tens of thousands of user-customized locations, it is necessary for us to use Regex to find out all US tweets by capturing some patterns behind them. Besides, those tweets without any location information takes up 21.15% [Figure3]of the whole dataset. In order to get more US tweets, we should build a good filter to loop over all 399588 rows of the data.

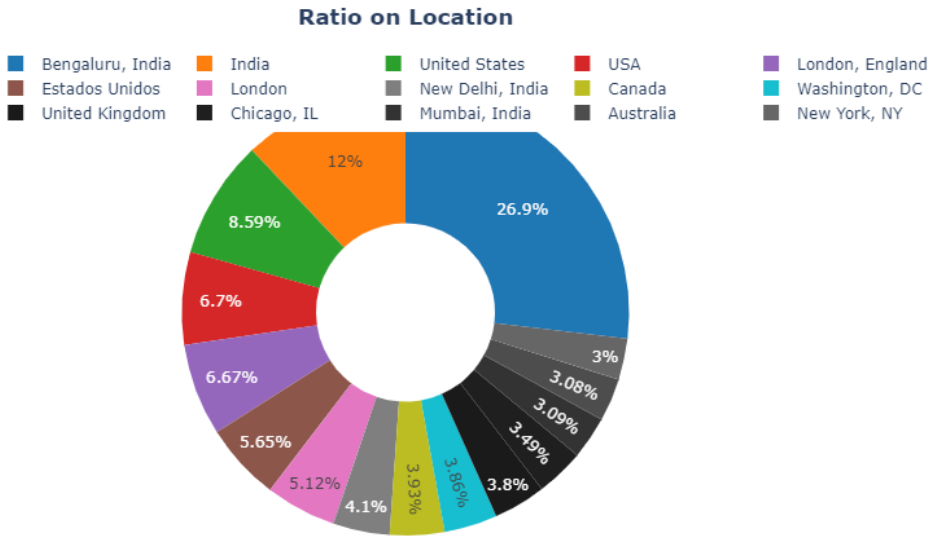

Figure2. Location distribution of original vaccine tweets. It’s interesting that tweets from India are the most(26.9%). And people in the US sent the second most number of tweets.

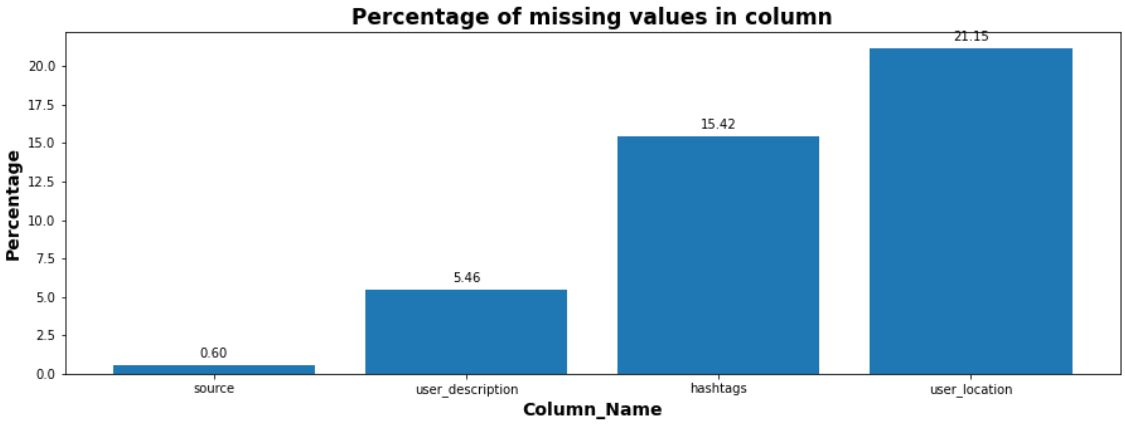

Figure3. Missing value in original vaccine tweets dataset. This plot shows that 21.15% tweets about 'vaccine' have no data about user location, and 15.42% tweets about 'vaccine' have no hashtags.

We build 3 lists to filter them, including US cities and different ways to write their names, includes all capitals and large cities

us_cities = ['New York City', 'Los Angeles', 'Chicago', 'Houston', 'Phoenix', 'Philadelphia', 'San Antonio', 'San Diego', 'Dallas', 'San Jose', 'Austin', 'Jacksonville', 'Fort Worth', 'Columbus', 'San Francisco',...]
us_ways = ['United States','USA','U.S.A']
us_states_abbrev = ['AL', 'AK', 'AZ', 'AR', 'CA', 'CO', 'CT', 'DE', 'FL', 'GA', 'HI', 'ID', 'IL', 'IN', 'IA', 'KS', 'KY', 'LA', 'ME', 'MD', 'MA', 'MI', 'MN', 'MS',...]

In this way, we capture all location names that haven't been customized by the users.

After filtering, we get nearly 100,000 rows of tweets, which takes up nearly 25% of the original dataset, that's a great result!

2.2.4 Sentiment Analysis of Vaccine Tweets

2.2.4.1 EDA before start

Before we started our sentiment analysis, we made some word clouds and word frequency plots to find out what people cared most in Covid and Vaccine. We first find a 1 month (24, Jul, 2020 - 30, Aug, 2020) dataset COVID Tweet dataset[3] to do some EDA and analysis [Figure4] [Figure5].

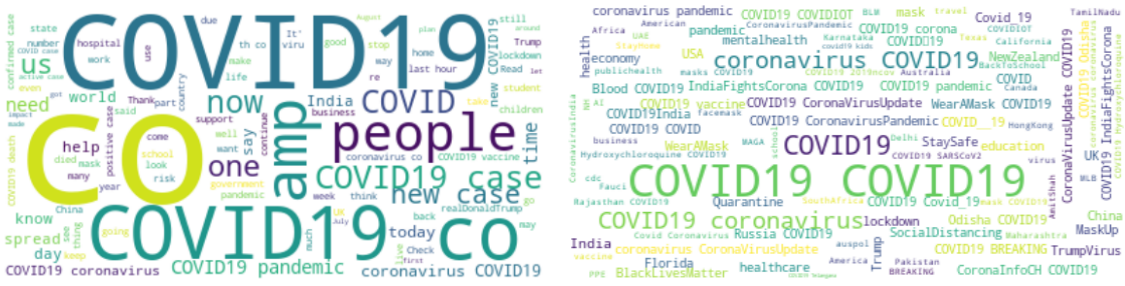

Figure4(a,b) 1 month COVID Tweet dataset's word cloud, a: from texts; b: from hashtags. In the figures, we find that in tweets related to COVID-19, words such as "amp", "one", "coronavirus" are also popular on twitter.

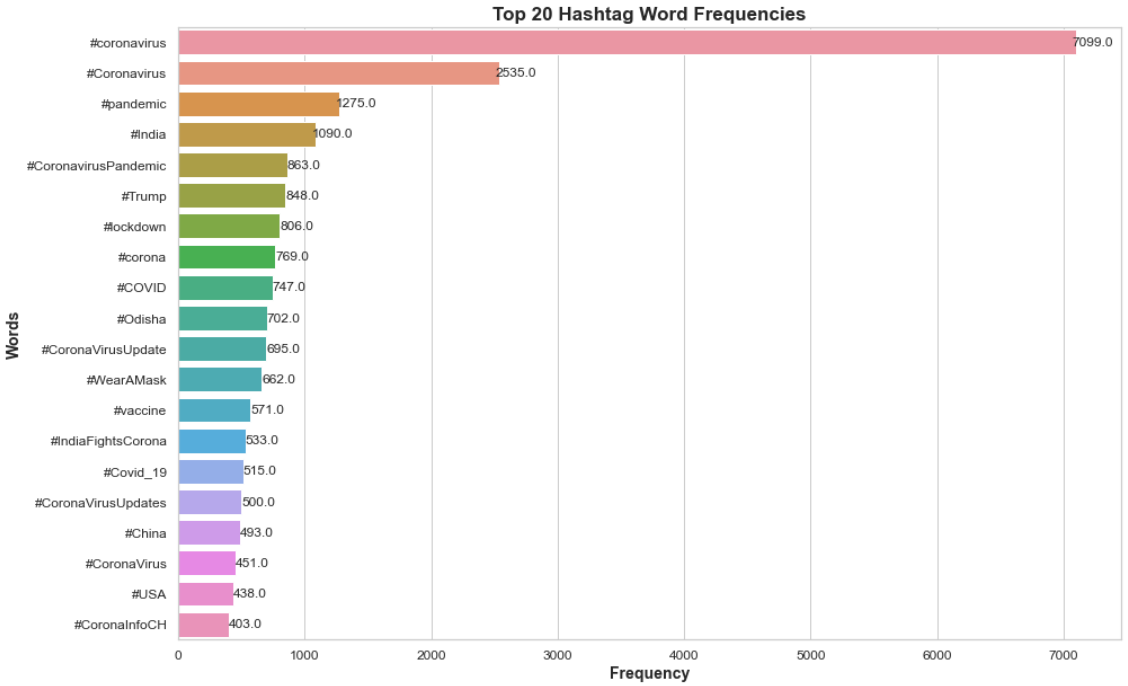

Figure5. One month COVID Tweet dataset's Top 20 Hashtag Word Frequencies. Similar with the figure4, there are some interesting words in the hashtags' word cloud: 'lockdown', 'Trump', 'WearAMask', 'mentalhealth', 'Vaccine', 'economy' and many names of locations.

There are some interesting words in the hashtags' word cloud: 'lockdown', 'Trump', 'WearAMask', 'mentalhealth', 'Vaccine', 'economy' and 'education' and many names of locations. It gave us some clues to further explore.

To make sure this data set's result is also meaningful towards our US-based study, we took a look at the location distribution too [Figure6].

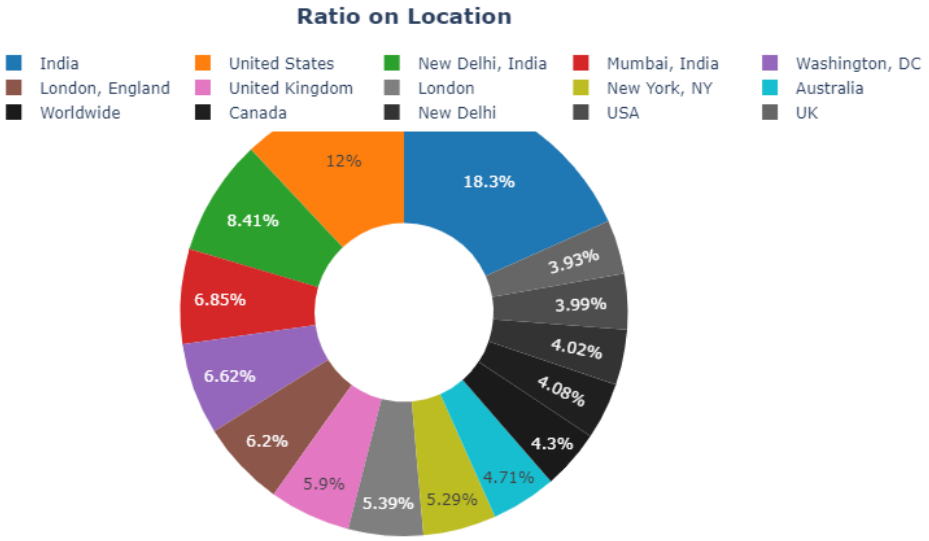

Figure6. One month COVID Tweet dataset's location distribution. This plot displays that there are many tweets related to "vaccine" from India and the United States.

Based on the above analysis, we believe that vaccine, mental health should be meaningful to dig deeper. So we found the Vaccine tweets dataset and did a similar EDA： Word cloud [Figure7] and words' frequency plot [Figure8].

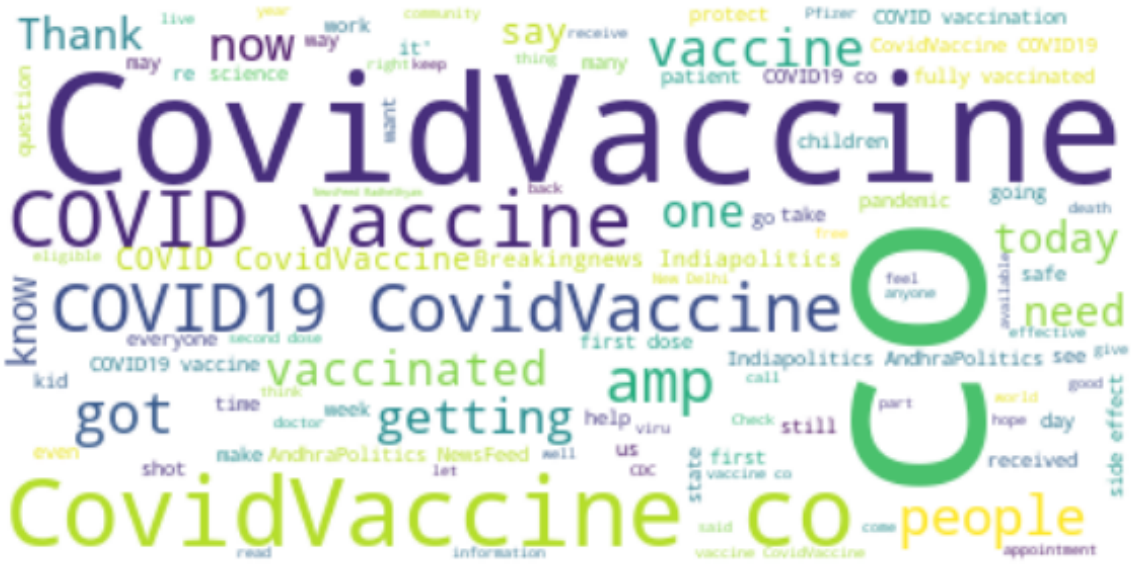

Figure7. COVID Vaccine Tweet dataset's word cloud. Like the figure4 shows, this word cloud from vaccine tweets also have popular words: "CovidVaccine", "CO", "amp".

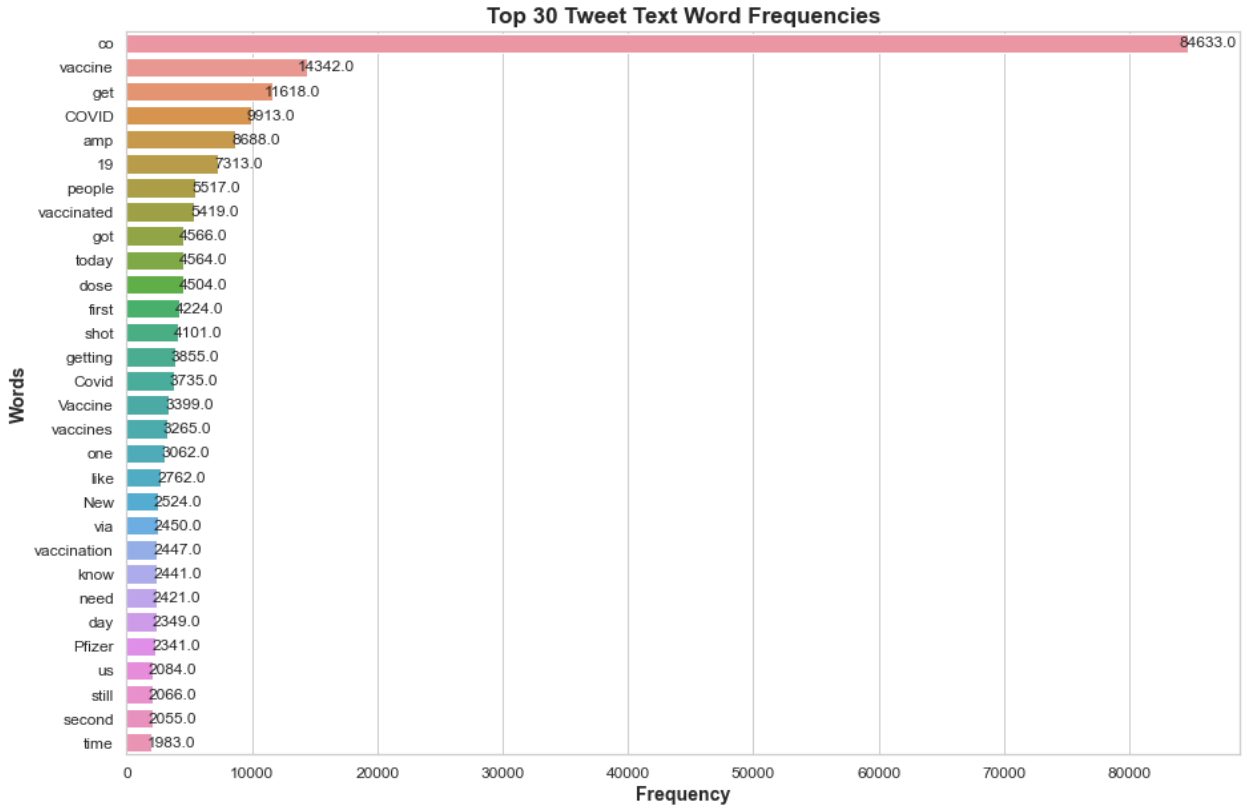

Figure8. COVID Vaccine Tweet dataset's words' frequency plot. In this plot, The most frequent word is "CO". The second most common word is "vaccine". Also words such as "get", "COVID", "19" also have high frequency.

2.2.4.2 Sentiment Analysis with NLP toolkits

We use three different NLP models to deal with the dataset. Initially, we utilize the VADER sentiment intensity analyzer to calculate sentiment scores for each tweet. However, about 33 percent tweets received a neutral score of 0. To solve this issue, we use the TextBlob library to reevaluate the sentiment scores for neutral tweets. Although this step improves our analysis, there are still 20 percent tweets with a sentiment score of 0.

To find a solution to this issue, we try Face's DistilBERT model, which is a transformer-based model fine-tuned for sentiment analysis. We recalculate sentiment scores for the remaining neutral tweets using the hugging face sentiment analysis pipeline. As a result, all tweets are assigned a sentiment score, and we then add a column named "sentiment_score" to the original dataset. After that, we use the date as indexes, to calculate daily "sum_sentiment_score","avg_sentiment_score", and "tweet_count(count the number of tweets related to covid vaccine in the US everyday)". We get Table5 for the following model training.

date	sum_sentiment_score	avg_sentiment_score	tweet_count

2022-09-11	-8.115179	-0.279834	29
2022-09-12	5.571819	0.214301	26
2022-09-13	-4.304051	-0.113264	38

Table5. Daily tweets(related to vaccines in the US) sentiment information. In this table, please note that 'Avg_sentiment_score' times 'tweet_count' equals to 'sum_sentiment_score'.

3. Modeling or inference techniques

We use two models: logistic regression and linear regression to predict the daily increasing COVID-19 confirmed population number.

3.1 Logistic Regression to build Binary Classification of whether the number of daily increasing confirmed & death cases is High or Low

After completing the EDA of datasets mentioned above, we merge these datasets from Table1, Table2, Table3 and Table4 by 'Date' columns[Figure9]. The merged dataframe 'total_clear' contains 287 rows and 14 columns and it contains the following three types of data:

- tweets sentiment analysis data, including sentiments toward covid and vaccine mental health statistical data, including the average and the median
- covid-19 daily confirmation data
- the 'fast' column is our 'Y' for classification, which means if the daily increasing number "diff" is more than the median of "diff", then "fast" equals to 1, if "diff" is less than the median of "diff", then "fast" equals to 0.

	covid_positive_count	covid_positive_percent	Date	covid_neutral_count	covid_neutral_percent	covid_negative_count	covid_negative_percent	vaccine_senti_sum	vaccine_senti_avg	vaccine_tweet_count	mh_avg	mh_medi	yesterday_confirmed_us	fast
0	22807	0.063837	05/09/2020	44074	0.122688	294714	0.813475	2.607551	0.096576	27	33.819231	33.4	1287687.0	0
1	40052	0.059227	06/09/2020	72884	0.102723	588829	0.838050	0.822600	0.091400	9	34.269231	34.7	1952876.0	0
2	33533	0.060656	07/09/2020	65598	0.114939	467824	0.824405	-1.763827	-0.073493	24	38.492308	38.3	3058150.0	1
3	36082	0.059620	08/09/2020	64415	0.106229	506614	0.834152	-4.102150	-0.066164	62	40.236538	40.5	4980997.0	0
4	49039	0.062066	08/13/2020	79596	0.098827	673123	0.839107	-6.671170	-0.230040	29	40.236538	40.5	5197225.0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
282	17127	0.042370	12/28/2021	45843	0.111037	347072	0.846593	-14.954344	-0.138466	108	31.396154	31.1	52602307.0	1
283	18815	0.036914	12/29/2021	63788	0.121816	426338	0.841270	2.198731	0.020549	107	32.221154	32.2	52962064.0	1
284	17004	0.031906	12/30/2021	59681	0.111451	461424	0.856644	-19.440019	-0.144000	135	32.221154	32.2	53455538.0	1
285	16482	0.026760	01/06/2022	56573	0.090361	543672	0.882880	-1.550094	-0.029247	53	32.221154	32.2	57480135.0	1
286	15951	0.028382	01/07/2022	56138	0.098453	492500	0.873165	-8.081796	-0.074831	108	32.221154	32.2	58299363.0	1

Figure9. The merged dataframe 'total_clear' with columns from COVID-19 confirmed number, mental health issue rate, tweets sentiments, and tweets vaccine sentiments. In this dataframe, column "fast" is our Y, other columns are features.

Following the data preprocessing and merging, we first focus on the logistic regression model for our analysis. To train and test our model, we first split the 'total_clear' dataset into a train set (80%, 229 rows) and a test set (20%, 58 rows). After that, in order to select the features for the logistic regression model, we plot a heatmap to visualize the pairwise correlation between each combination of columns[Figure10]. Then we choose the top 9 related parameters as features in our model: 'covid_positive_count', 'covid_positive_percent', 'covid_neutral_percent',

'covid_negative_percent', 'vaccine_senti_sum', 'vaccine_tweet_count', 'mh_avg', 'mh_medi', 'yesterday_confirmed_us'.

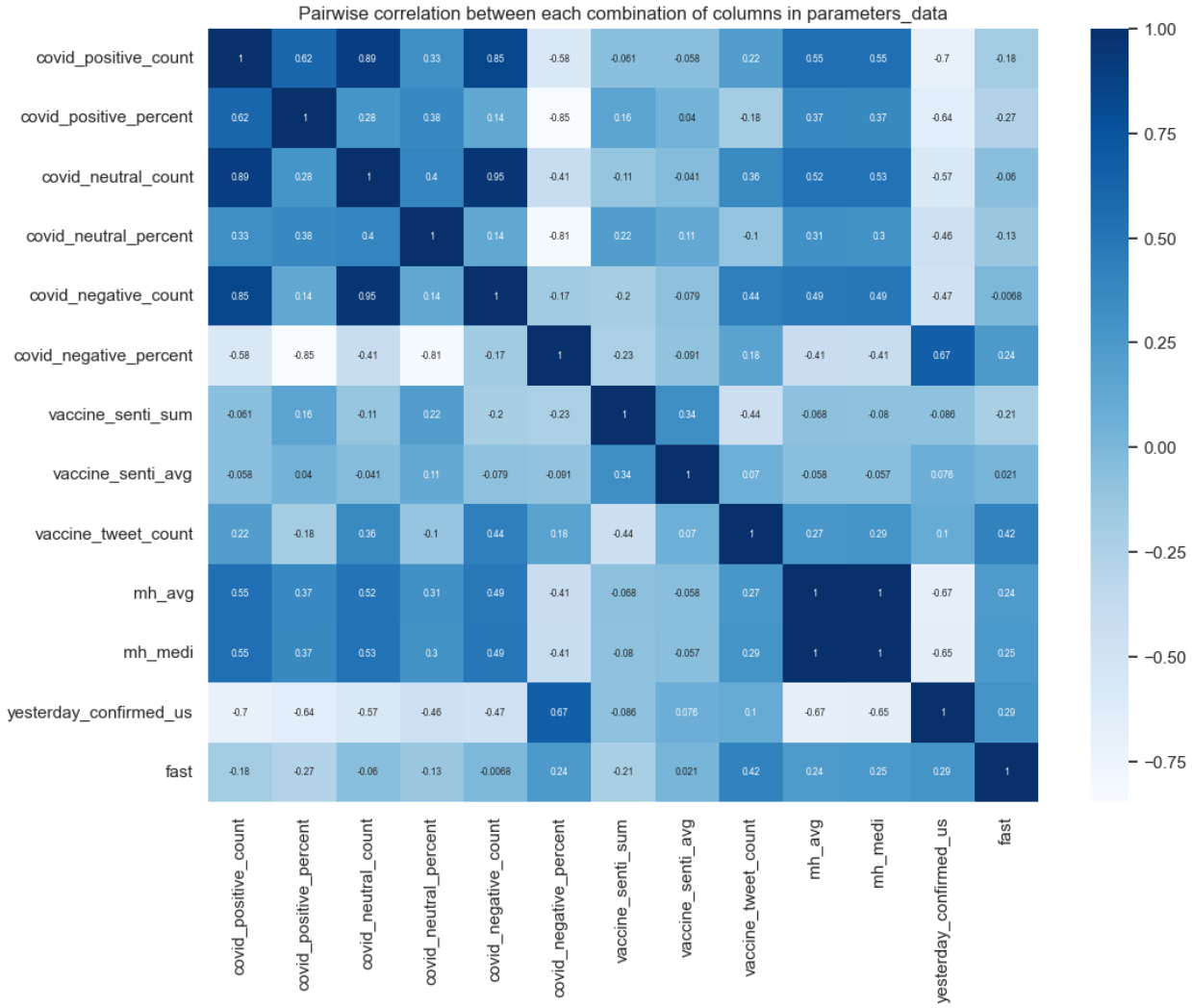

Figure10. Pairwise correlation coefficients in logistic regression model. In this correlation heatmap, we find that "covid_positive_count", "covid_neutral_count", and "covid_negative_count" have the most correlation with Y.

With the features selected, we proceed to train the logistic regression model on the train set. After training the model, we achieve a training accuracy of approximately 75.55%.

Next, we apply the trained model to the test set. By evaluating the model's performance on the test set, we obtain a testing accuracy of about 67.24%. Therefore, we can use this classification model to predict whether the next day's COVID-19 confirmed number's growth speed will be bigger than the median growth speed, which means, to predict the COVID-19 confirmed number will grow faster or slower.

Similarly, we apply the same method to the death dataset, to predict whether the death case growth is accelerating or decelerating. We also apply regularization[figure11] and feature

engineering(cube, square, division)[figure12] to refine this model. Finally we get training and test accuracy as 61% and 62% respectively.

```
from sklearn.linear_model import LogisticRegression
X_train_d_df = X_train_d.loc[:, ['covid_positive_percent', 'covid_negative_count',
                                 'covid_negative_percent', 'vaccine_senti_sum',
                                 'vaccine_tweet_count', 'reported_mh_average',
                                 'reported_mh_median', 'covid_positive_percent_sqrt',
                                 'covid_negative_percent_sqrt', 'vaccine_senti_sum_sqrt',
                                 'vaccine_tweet_count_sqrt', 'reported_mh_average_sqrt',
                                 'reported_mh_median_sqrt', 'covid_positive_percent_cube',
                                 'covid_negative_percent_cube', 'vaccine_senti_sum_cube',
                                 'vaccine_tweet_count_cube', 'reported_mh_average_cube',
                                 'reported_mh_median_cube', 'covid_p_n_rate']]

# print(X_train_d_df.isnull().sum())
X_train_d_df = X_train_d_df.fillna('0')
# print(X_train_d_df.isnull().sum())

X_Train_d = X_train_d_df.to_numpy()
Y_Train_d = np.array(y_train_d)

# display(X_Train_d)

model_d_f = LogisticRegression(penalty='12', C=0.000001).fit(X_Train_d, Y_Train_d)
training_accuracy_d = model_d_f.score(X_Train_d, Y_Train_d)
print("Training Accuracy: ", training_accuracy_d)
```

Figure11. Regularization in logistic regression model. We choose penalty='12', C=0.000001 as the hyperparameter to refine the logistic regression model.

```
total_d_clear["covid_positive_percent_sqrt"] = np.sqrt(total_d_clear["covid_positive_percent"])
total_d_clear["covid_negative_percent_sqrt"] = np.sqrt(total_d_clear["covid_negative_percent"])
total_d_clear["vaccine_senti_sum_sqrt"] = np.sqrt(total_d_clear["vaccine_senti_sum"])
total_d_clear["vaccine_tweet_count_sqrt"] = np.sqrt(total_d_clear["vaccine_tweet_count"])
total_d_clear["reported_mh_average_sqrt"] = np.sqrt(total_d_clear["reported_mh_average"])
total_d_clear["reported_mh_median_sqrt"] = np.sqrt(total_d_clear["reported_mh_median"])

total_d_clear["covid_positive_percent_cube"] = total_d_clear["covid_positive_percent"] ** 3
total_d_clear["covid_negative_percent_cube"] = total_d_clear["covid_negative_percent"] ** 3
total_d_clear["vaccine_senti_sum_cube"] = total_d_clear["vaccine_senti_sum"] ** 3
total_d_clear["vaccine_tweet_count_cube"] = total_d_clear["vaccine_tweet_count"] ** 3
total_d_clear["reported_mh_average_cube"] = total_d_clear["reported_mh_average"] ** 3
total_d_clear["reported_mh_median_cube"] = total_d_clear["reported_mh_median"] ** 3

total_d_clear["covid_p_n_rate"] = total_d_clear["covid_positive_percent"] / total_d_clear["covid_negative_percent"]
```

Figure12. Feature engineering(cube, square, division) for features in logistic regression model. After applying the feature engineering, the model accuracy increases about 10 percent.

3.2 Linear Regression to build Predictive Model of the number of daily increasing confirmed & Death cases

After we complete the EDA of datasets mentioned above, we merge these datasets from Table1, Table2, Table3 and Table4 by 'Date' columns. The merged dataframe 'df' contains 290 rows and 17 columns and it contains the following three types of data:

- tweets sentiment analysis data, including sentiments toward covid and vaccine
- covid-19 daily confirmation data
- covid-19 daily deaths data
- The ‘death_diff’ and ‘confirm_diff’ columns are our ‘Y’ for predicting, which means daily increasing death and confirmed cases of covid-19 total number.

Before building our model, we first use IQR to remove the outliers and then calculate the log of yesterday_confirmed_us, and yesterday_deaths_us which means the last day’s confirmed cases and death cases number that would be used as one of the features in the model afterwards, because the number of them are very large and the Stander Scaler may be not sufficient enough to standardize them.

Then we selected: 'avg_sent_vaccine_Tweets', 'tweet_count_vaccine_Tweets', 'Count_all_tweets_neutral', ‘Count_all_tweets_positive', 'Avg_Per_all_tweets_negative', 'Count_all_tweets_negative' as our features for the number of daily increasing confirmed cases at the very beginning.

Similarly, we selected: 'avg_sent_vaccine_Tweets', 'tweet_count_vaccine_Tweets', 'Count_all_tweets_neutral', ‘Count_all_tweets_positive', 'Avg_Per_all_tweets_negative', 'Count_all_tweets_negative' as our features for the number of daily increasing death cases[figure13] at the very beginning.

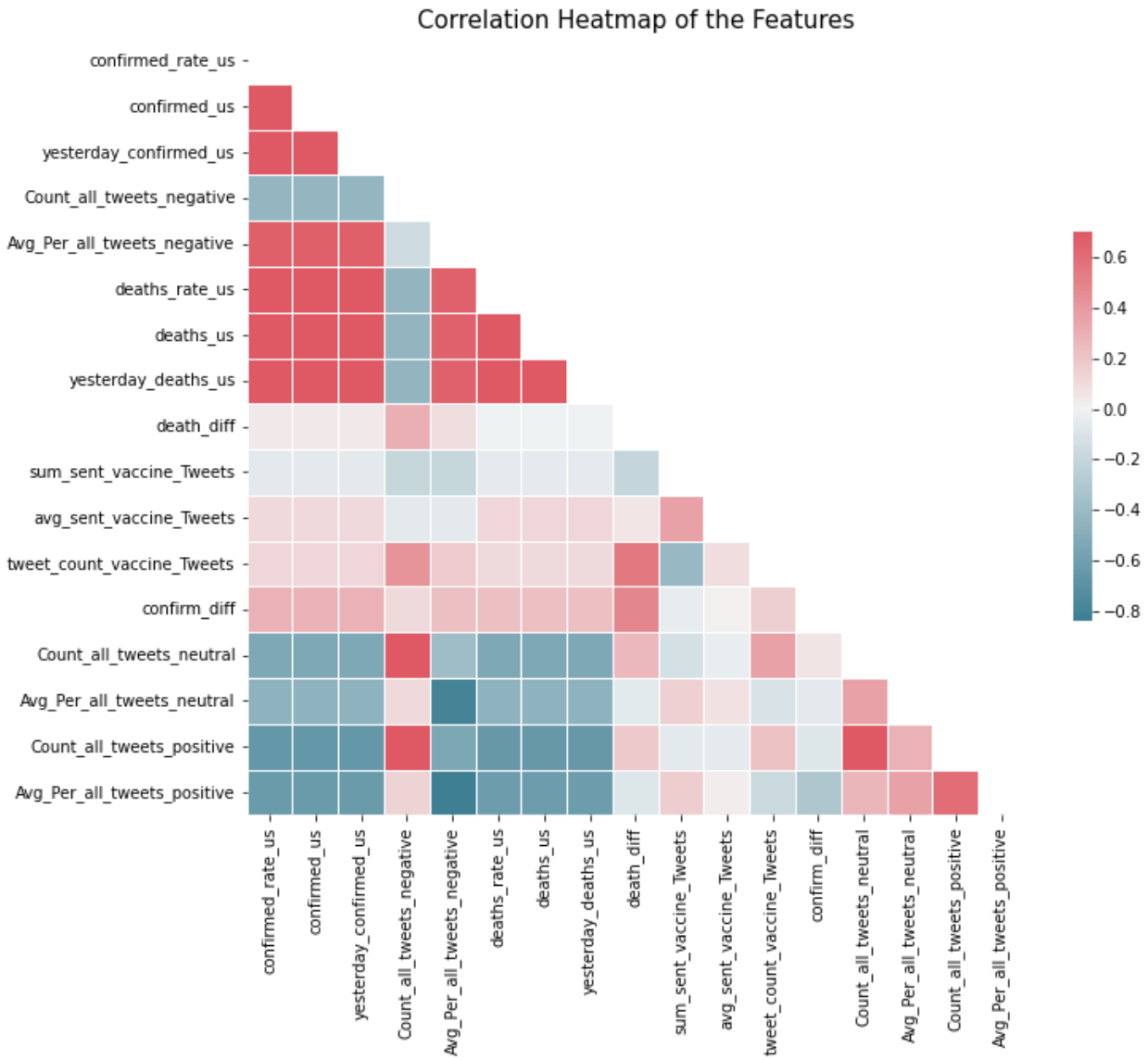

Figure13. Pairwise correlation coefficients in linear regression model. *In this correlation heatmap, we find that the count of tweets have the most correlation with Y.*

With our features selected, we proceed to train the Linear Regression models on the train set. However, simply using the features we collected results in an unsatisfactory model.

To figure out the problem, we tried the following approaches:

1. L2 regularization (Ridge regularization)
2. Feature Engineering
3. Using IQR to remove the outliers.

First, we added more features to the linear model, by intersecting different features, or adding polynomial features and ratio features to capture more complex relationships within the data.

The following code shows the features we added, after feature engineering, we finally get 12 reliable features[Figure14]:

```
X = df_clean[['avg_sent_vaccine_Tweets',#average of sentiment score of vaccine tweets
              'tweet_count_vaccine_Tweets',#count of vaccine tweets
                'Count_all_tweets_neutral',#count of All covid neutral tweets
                'Count_all_tweets_positive',#count of All covid positive tweets
                'Avg_Per_all_tweets_negative',#average percent of All covid negative tweets
              'Count_all_tweets_neu&pos&neg-avg',#average count of All covid 3 kind of tweets
                'Count_all_tweets_negative',#count of All covid negative tweets
                'squared_avg_sent_vaccine_Tweets',#squared average of sentiment score of vaccine tweets
              'cubed_tweet_count_vaccine_Tweets',#cubed average of sentiment score of vaccine tweets
              'interaction_vaccine_tweets_sent',#interaction between the count of vaccine tweets and the sentiment scores
             'ratio_negative_neutral_tweets',
              'ratio_positive_negative_tweets',


             ]]
y = df_clean['confirm_diff']
```

Figure14. Features selected for Linear model. These features are used both in linear regression model for daily increased confirmed cases, and model for daily increased Death cases.

Then we create a Ridge Regression model for daily increased confirmed cases[Figure15]:

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a Ridge regression model with L2 regularization
# Here we're setting the regularization strength to 1.0, you can adjust this value
model = Ridge(alpha=500)
# Train the model
model.fit(X_train, y_train)
# Use the model to make predictions on the testing data
y_pred = model.predict(X_test)
```

Figure15. Linear model with L2 Regularization code. After trying the regularization parameters α from 0.01-1000, we choose α=500 to reduce the model complexity.

After training the model, we evaluate the model with the test set:
For number of daily increased confirmed cases:
Root Mean Squared Error(RMSE): 53906.94569261472
Mean Absolute Error (MAE): 38691.54126783042

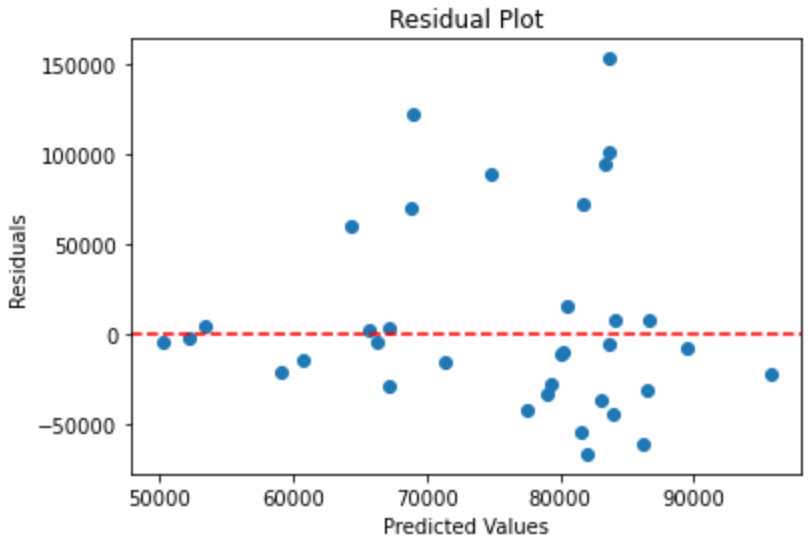

Figure16. Residual plot for Linear model predicting daily confirmed cases. Points with residual < 0 are more close to the zero-line, points with residual > 0 are more far from the zero-line.

Using same features, here is our model performance for predicting the number of daily increased death cases:
Root Mean Squared Error: 618.9691212478901
Mean Absolute Error (MAE): 469.20966935594925

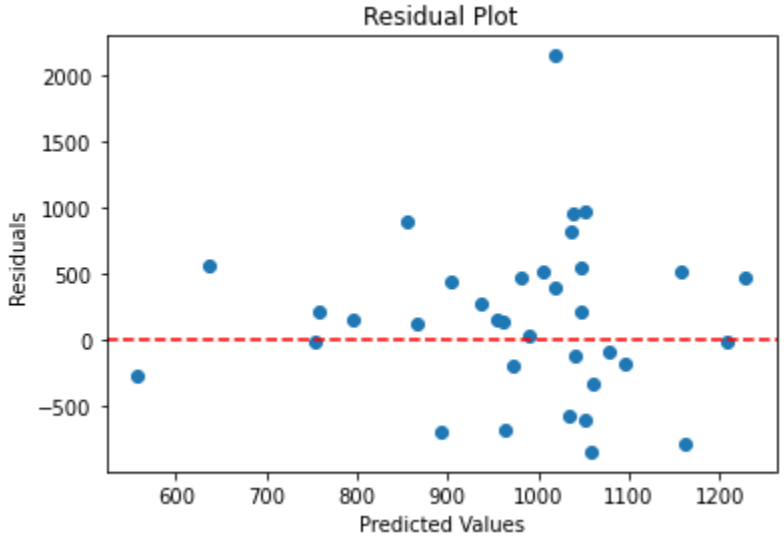

Figure17. Residual plot for Linear model predicting daily death cases. Points with residual < 0 are more close to the zero-line, points with residual > 0 are more far from the zero-line. The residual plot for this death cases model looks better than that for the confirmed cases model.

Comparing the predictive model of daily confirmed cases and death cases, we can learn that the RMSE of them are different by orders of magnitude. That's because confirmed cases total number and death number inherently have 10^2 differences by orders of magnitude.

3.3 Comparison of Linear Regression Model and Logistic Regression model

We use the multiple linear regression model to predict the daily increase of confirmed/death case number, and use the logistic regression model to determine whether the number of daily increasing confirmed/death cases is high or low. These two models help us to know the daily increasing number and find out whether the case number growth is accelerating or decelerating. According to the peer review feedback, we use Table6 to compare these two regression models.

	Linear Regression Model		**Logistic Regression Model**	
	Confirmed	**Death**	**Confirmed**	**Death**
RMSE	**53906.9**	**618.97**	**-**	**-**
MAE	**38691.5**	**469.2**	**-**	**-**
Mean Accuracy	**-**	**-**	**0.672414**	**0.706897**

Table6. Comparison of performance of Linear Regression Model and Logistic Regression model. The Logistic Regression Model has mean accuracy, the Linear Regression Model has RMSE and MAE as the evaluation parameters.

3.4 Interesting Findings in Model Training Process

When training the linear regression model, we find the most related feature to Y is the counts of tweets(related to COVID-19), instead of other features(sentiments). This finding has been shown in [Figure13] Pairwise correlation coefficients in the linear regression model.

4. Analysis of results and discussion

In this project, our main goal is to predict the increasing number of covid confirmed cases and death cases based on the sentiment data from tweets and the mental health value from CDC in DatasetB(used in logistic regression model not in linear regression model). The logistic regression model achieves a training accuracy of 75.55% and a testing accuracy of 67.24%, an acceptable result to predict the next day's COVID-19 confirmed number's growth speed.

However, we can improve this logistic regression model in the following ways. First, we will extend our dataset. To be specific, we only have 287 valid data. After we split it into a train set and a test set, the number of data in each set is small. Second, we will find more features that are related to the COVID-19 confirmed number's growth speed, for example, finding tweets with "mask", "flu" keywords. Third, the mental health data[Table2] from DatasetB is very sparse, we only have weekly or even biweekly data.

Likewise, we get RMSE=53907 in confirmed cases, RMSE=619 in death cases from our linear regression model. After changing the regularization parameters α from 0.01-1000, we choose

α=500 to reduce the model complexity according to the peer review comments. Also, to refine our linear model, extending our dataset is our primary mission. Also, we will add more features(tweets with “mask”, “flu” keywords/hashtags) in the training process. Furthermore, we may try to change how to divide the test and train dataset, using K-fold cross validation, also we can change other hyperparameters in the linear regression process.

In conclusion, both the logistic regression model and linear regression model demonstrate a reasonable ability to predict whether the daily increase in COVID-19 confirmed and death cases is high or low based on the selected features. We will improve our models later.

5. Discussion of potential societal impacts and ethical concerns

About the potential societal impacts, on the one hand, our study offers guidelines about public health policy and communication strategies during a pandemic, such as COVID-19. On the other hand, with implementation of social media data (such as Twitter), the research can help understand the public sentiment and mental health status, leading to more effective public health resources and resource allocation.

However, there are several ethical concerns. Firstly, there's the concern of privacy. Although tweets are public and used anonymously in our research, the use of this data for research purposes without explicit consent from users might be considered as an invasion of privacy. Secondly, the trained model may have a concern of bias. Twitter users cannot represent the entire population, and the demographics of Twitter users are skewed towards certain groups, potentially leading to biased conclusions.

In conclusion, though we can use the model precisely to predict the growth of case number using the social media and mental health data for public health, it must be done responsibly, with considerations with ethical concerns, to ensure the benefits outweigh the potential issues.

6. Appendix

According to the introduction on eds, we put figures in context for more readability, we also aggregate all figures we used in the context here for an overview.

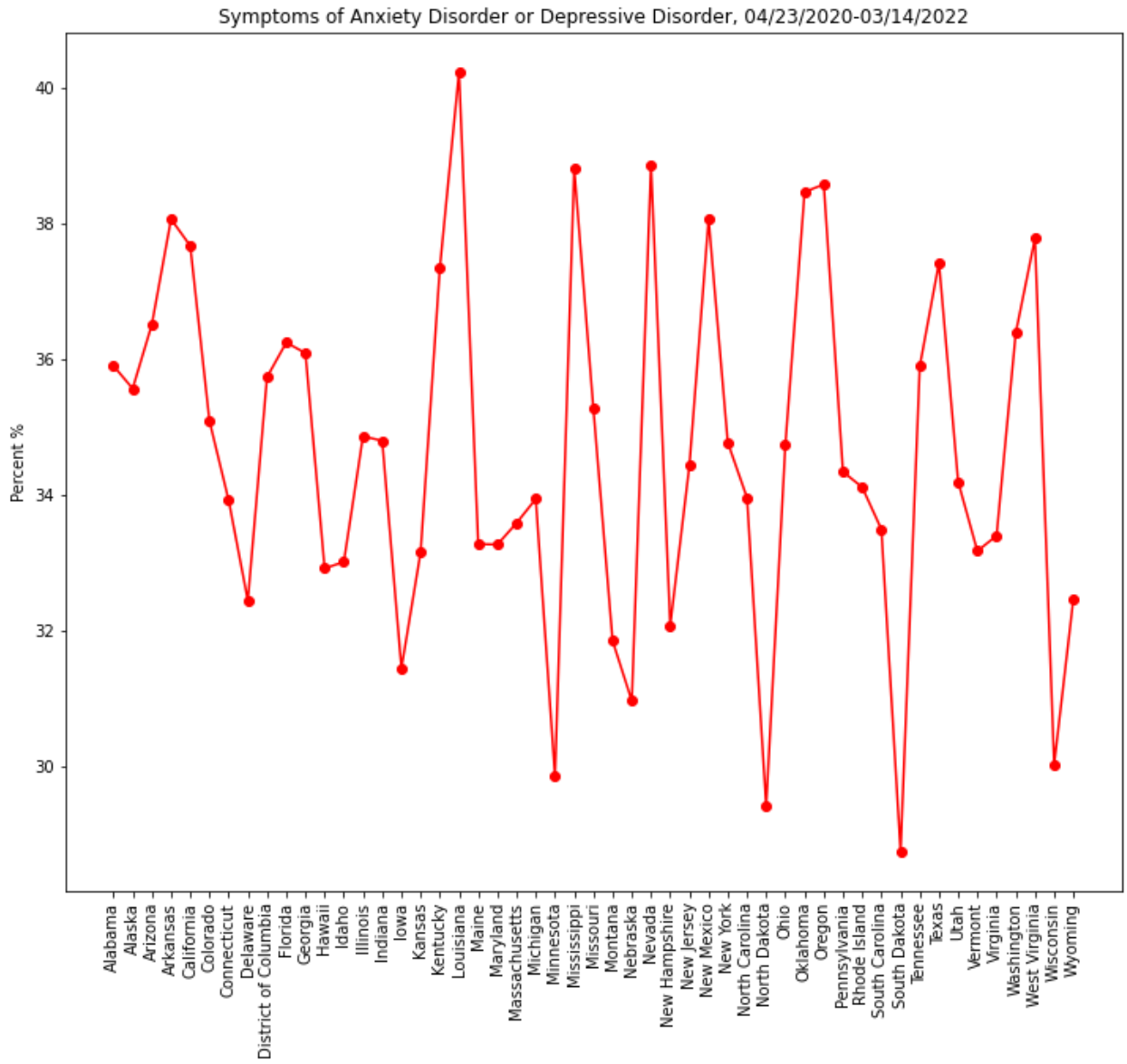

Figure1. Symptoms of Anxiety Disorder or Depressive Disorder, 04/23/2020-03/14/2022, in US. This figure shows that responders in Louisiana have the highest percentage(40.2%) of reporting "Symptoms of Anxiety Disorder or Depressive Disorder" mental health issues. Also, responders in South Dakota(28.7%) have the lowest percentage of reporting "Symptoms of Anxiety Disorder or Depressive Disorder" mental health issues. The mean value is 34.6%, which shows about 34.6% of responders have had symptoms of anxiety disorder or depressive disorder from 04/23/2020 to 03/14/2022 in the US.

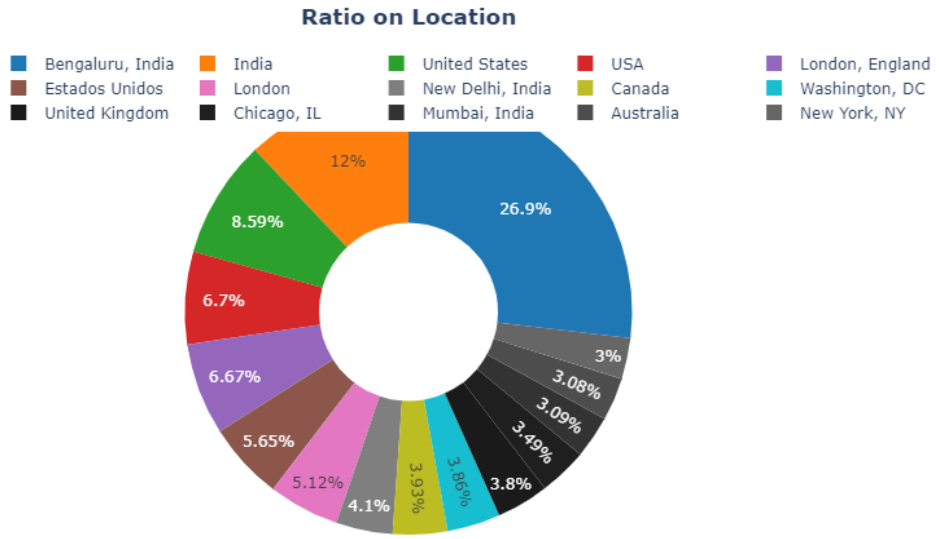

Figure2. Location distribution of original vaccine tweets. It's interesting that tweets from India are the most(26.9%). And people in the US sent the second most number of tweets.

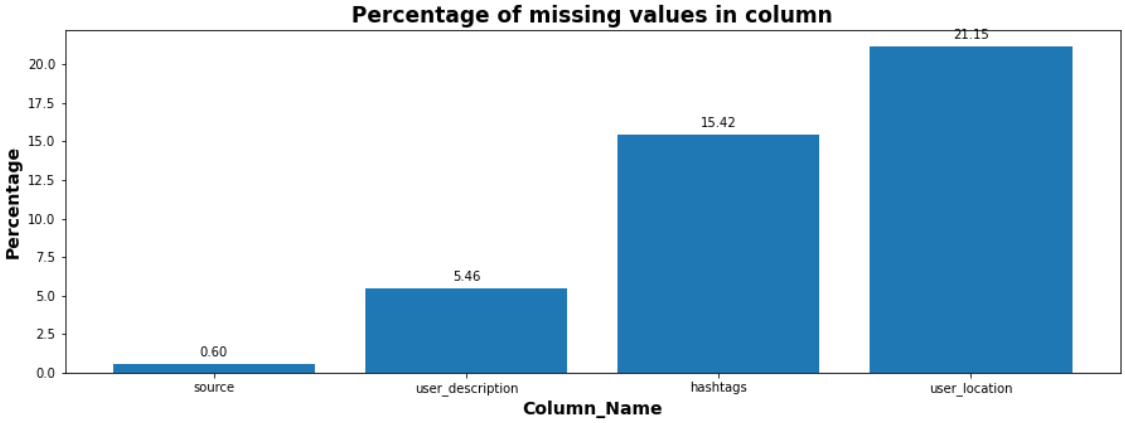

Figure3. Missing value in original vaccine tweets dataset. This plot shows that 21.15% tweets about 'vaccine' have no data about user location, and 15.42% tweets about 'vaccine' have no hashtags.

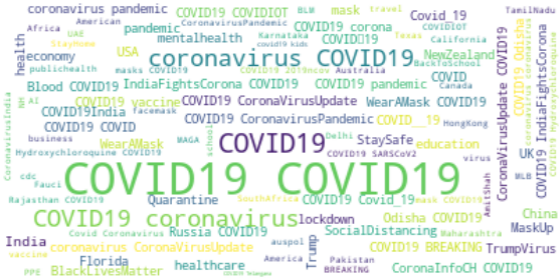

Figure4(a,b) 1 month COVID Tweet dataset's word cloud, a: from texts; b: from hashtags. In the figures, we find that in tweets related to COVID-19, words such as "amp", "one", "coronavirus" are also popular on twitter.

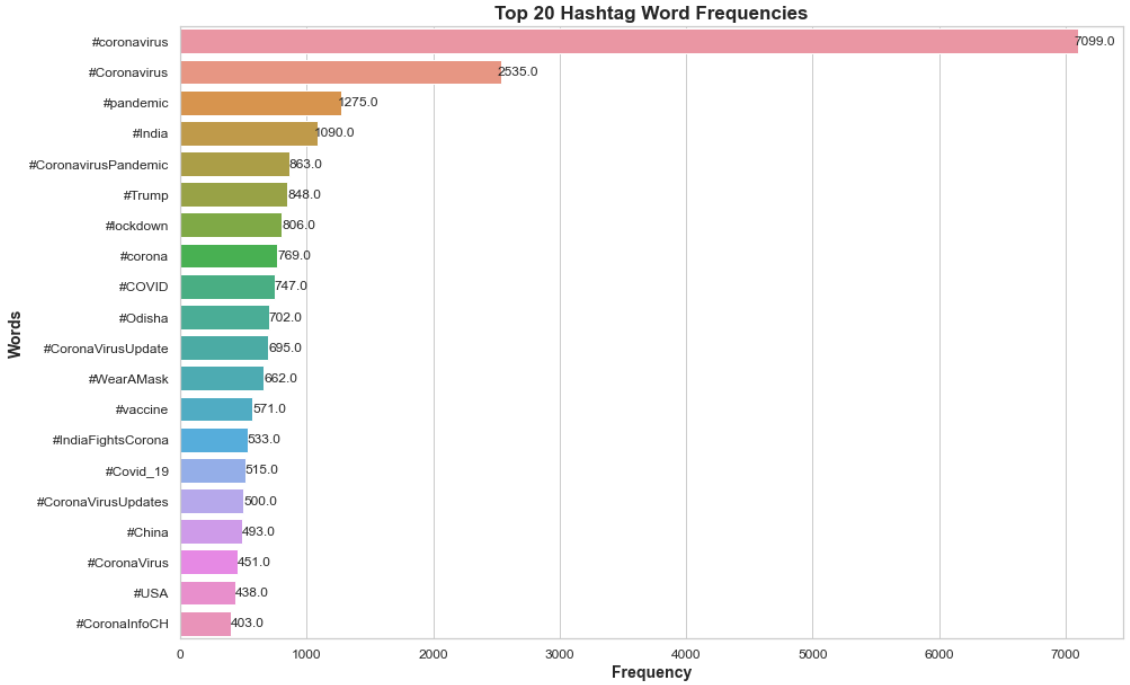

Figure5. One month COVID Tweet dataset's Top 20 Hashtag Word Frequencies. Similar with the figure4, there are some interesting words in the hashtags' word cloud: 'lockdown', 'Trump', 'WearAMask', 'mentalhealth', 'Vaccine', 'economy' and many names of locations.

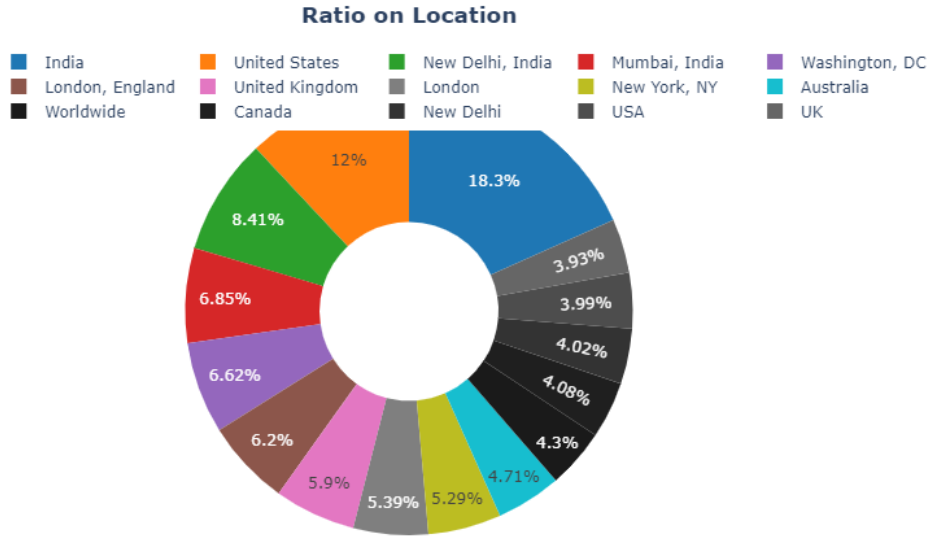

Figure6. One month COVID Tweet dataset's location distribution. This plot displays that there are many tweets related to "vaccine" from India and the United States.

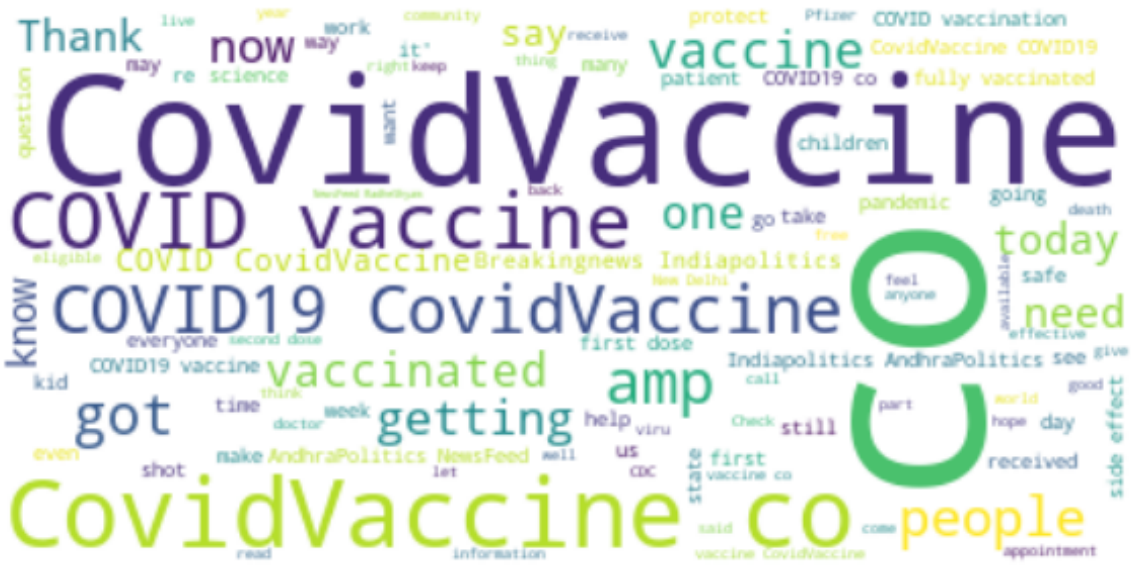

Figure7. COVID Vaccine Tweet dataset's word cloud. Like the figure4 shows, this word cloud from vaccine tweets also have popular words: "CovidVaccine", "CO", "amp".

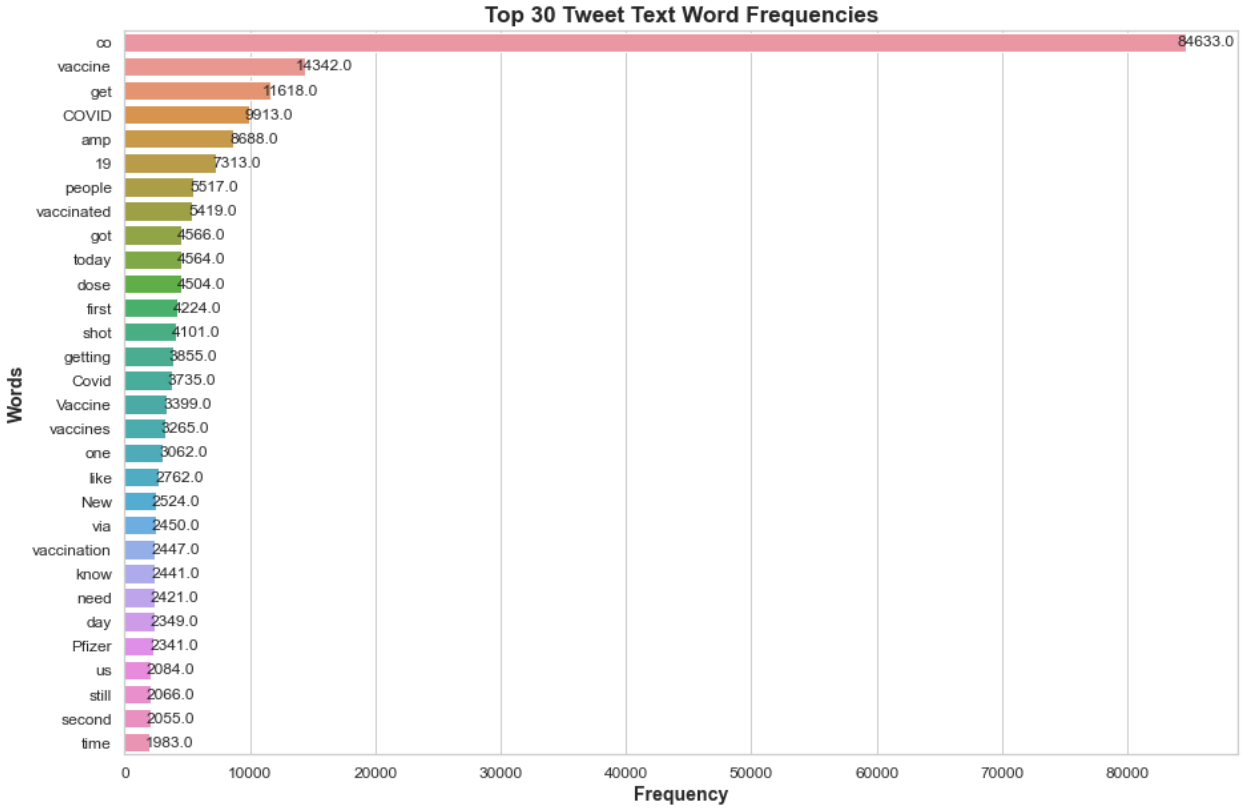

Figure8. COVID Vaccine Tweet dataset's words' frequency plot. In this plot, The most frequent word is "CO". The second most common word is "vaccine". Also words such as "get", "COVID", "19" also have high frequency.

	covid_positive_count	covid_positive_percent	Date	covid_neutral_count	covid_neutral_percent	covid_negative_count	covid_negative_percent	vaccine_senti_sum	vaccine_senti_avg	vaccine_tweet_count	mh_avg	mh_medi	yesterday_confirmed_us	fast
0	22807	0.063837	05/09/2020	44074	0.122688	294714	0.813475	2.607551	0.096576	27	33.819231	33.4	1287687.0	0
1	40052	0.059227	06/09/2020	72884	0.102723	588829	0.838050	0.822600	0.091400	9	34.269231	34.7	1952876.0	0
2	33533	0.060656	07/09/2020	65598	0.114939	467824	0.824405	-1.763827	-0.073493	24	38.492308	38.3	3058150.0	1
3	36082	0.059620	08/09/2020	64415	0.106229	506614	0.834152	-4.102150	-0.066164	62	40.236538	40.5	4980997.0	0
4	49039	0.062066	08/13/2020	79596	0.098827	673123	0.839107	-6.671170	-0.230040	29	40.236538	40.5	5197225.0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
282	17127	0.042370	12/28/2021	45843	0.111037	347072	0.846593	-14.954344	-0.138466	108	31.396154	31.1	52602307.0	1
283	18815	0.036914	12/29/2021	63788	0.121816	426338	0.841270	2.198731	0.020549	107	32.221154	32.2	52962064.0	1
284	17004	0.031906	12/30/2021	59681	0.111451	461424	0.856644	-19.440019	-0.144000	135	32.221154	32.2	53455538.0	1
285	16482	0.026760	01/06/2022	56573	0.090361	543672	0.882880	-1.550094	-0.029247	53	32.221154	32.2	57480135.0	1
286	15951	0.028382	01/07/2022	56138	0.098453	492500	0.873165	-8.081796	-0.074831	108	32.221154	32.2	58299363.0	1

Figure9. The merged dataframe 'total_clear' with columns from COVID-19 confirmed number, mental health issue rate, tweets sentiments, and tweets vaccine sentiments. In this dataframe, column "fast" is our Y, other columns are features.

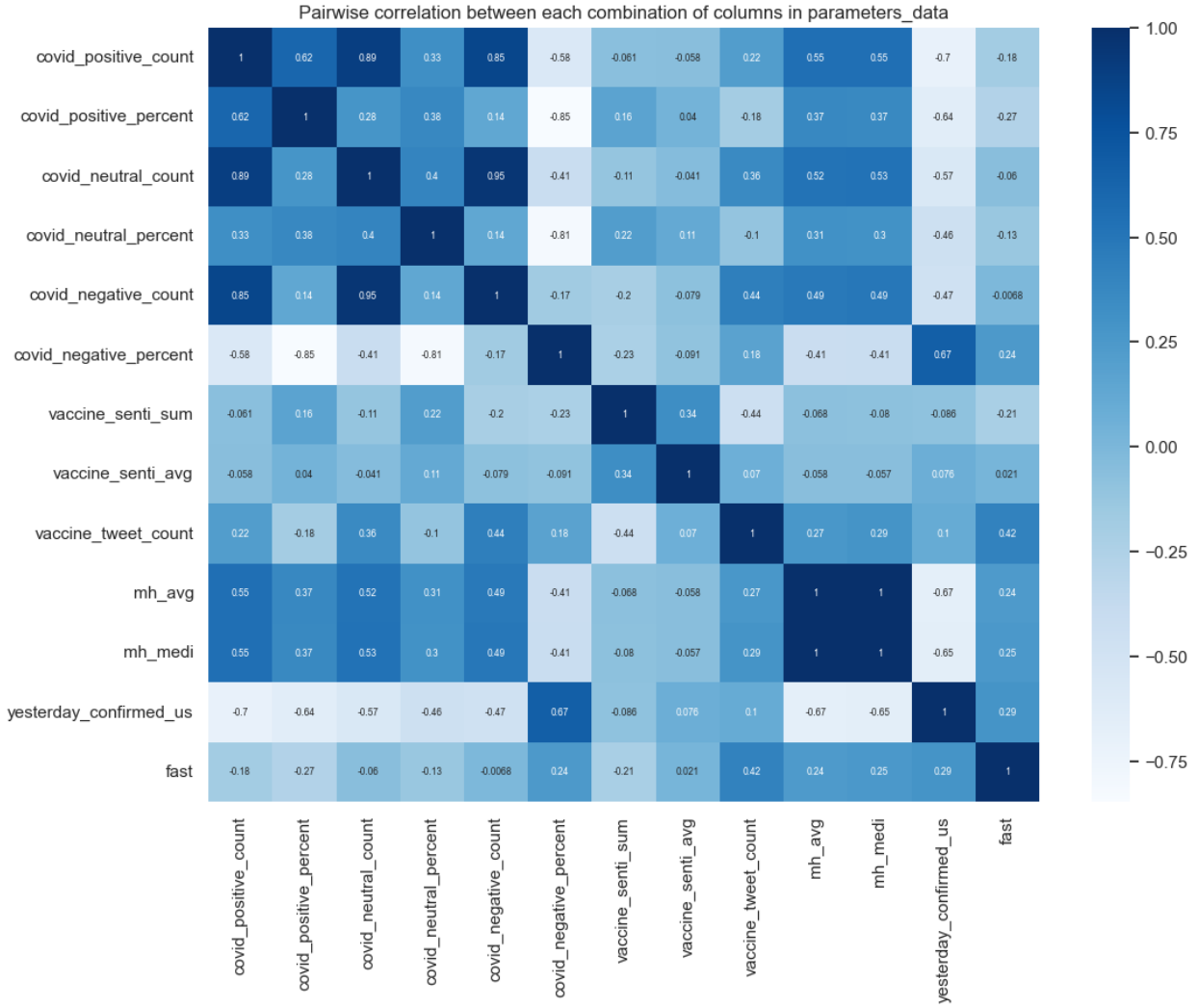

Figure10. Pairwise correlation coefficients in logistic regression model. In this correlation heatmap, we find that "covid_positive_count", "covid_neutral_count", and "covid_negative_count" have the most correlation with Y.

```
from sklearn.linear_model import LogisticRegression
X_train_d_df = X_train_d.loc[:, ['covid_positive_percent', 'covid_negative_count',
                                 'covid_negative_percent', 'vaccine_senti_sum',
                                 'vaccine_tweet_count', 'reported_mh_average',
                                 'reported_mh_median', 'covid_positive_percent_sqrt',
                                 'covid_negative_percent_sqrt', 'vaccine_senti_sum_sqrt',
                                 'vaccine_tweet_count_sqrt', 'reported_mh_average_sqrt',
                                 'reported_mh_median_sqrt', 'covid_positive_percent_cube',
                                 'covid_negative_percent_cube', 'vaccine_senti_sum_cube',
                                 'vaccine_tweet_count_cube', 'reported_mh_average_cube',
                                 'reported_mh_median_cube', 'covid_p_n_rate']]

# print(X_train_d_df.isnull().sum())
X_train_d_df = X_train_d_df.fillna('0')
# print(X_train_d_df.isnull().sum())

X_Train_d = X_train_d_df.to_numpy()
Y_Train_d = np.array(y_train_d)

# display(X_Train_d)

model_d_f = LogisticRegression(penalty='l2', C=0.000001).fit(X_Train_d, Y_Train_d)
training_accuracy_d = model_d_f.score(X_Train_d, Y_Train_d)
print("Training Accuracy: ", training_accuracy_d)
```

Figure11. Regularization in logistic regression model. We choose penalty='12', C=0.000001 as the hyperparameter to refine the logistic regression model.

```
total_d_clear["covid_positive_percent_sqrt"] = np.sqrt(total_d_clear["covid_positive_percent"])
total_d_clear["covid_negative_percent_sqrt"] = np.sqrt(total_d_clear["covid_negative_percent"])
total_d_clear["vaccine_senti_sum_sqrt"] = np.sqrt(total_d_clear["vaccine_senti_sum"])
total_d_clear["vaccine_tweet_count_sqrt"] = np.sqrt(total_d_clear["vaccine_tweet_count"])
total_d_clear["reported_mh_average_sqrt"] = np.sqrt(total_d_clear["reported_mh_average"])
total_d_clear["reported_mh_median_sqrt"] = np.sqrt(total_d_clear["reported_mh_median"])

total_d_clear["covid_positive_percent_cube"] = total_d_clear["covid_positive_percent"] ** 3
total_d_clear["covid_negative_percent_cube"] = total_d_clear["covid_negative_percent"] ** 3
total_d_clear["vaccine_senti_sum_cube"] = total_d_clear["vaccine_senti_sum"] ** 3
total_d_clear["vaccine_tweet_count_cube"] = total_d_clear["vaccine_tweet_count"] ** 3
total_d_clear["reported_mh_average_cube"] = total_d_clear["reported_mh_average"] ** 3
total_d_clear["reported_mh_median_cube"] = total_d_clear["reported_mh_median"] ** 3

total_d_clear["covid_p_n_rate"] = total_d_clear["covid_positive_percent"] / total_d_clear["covid_negative_percent"]
```

Figure12. Feature engineering(cube, square, division) for features in logistic regression model. After applying the feature engineering, the model accuracy increases about 10 percent.

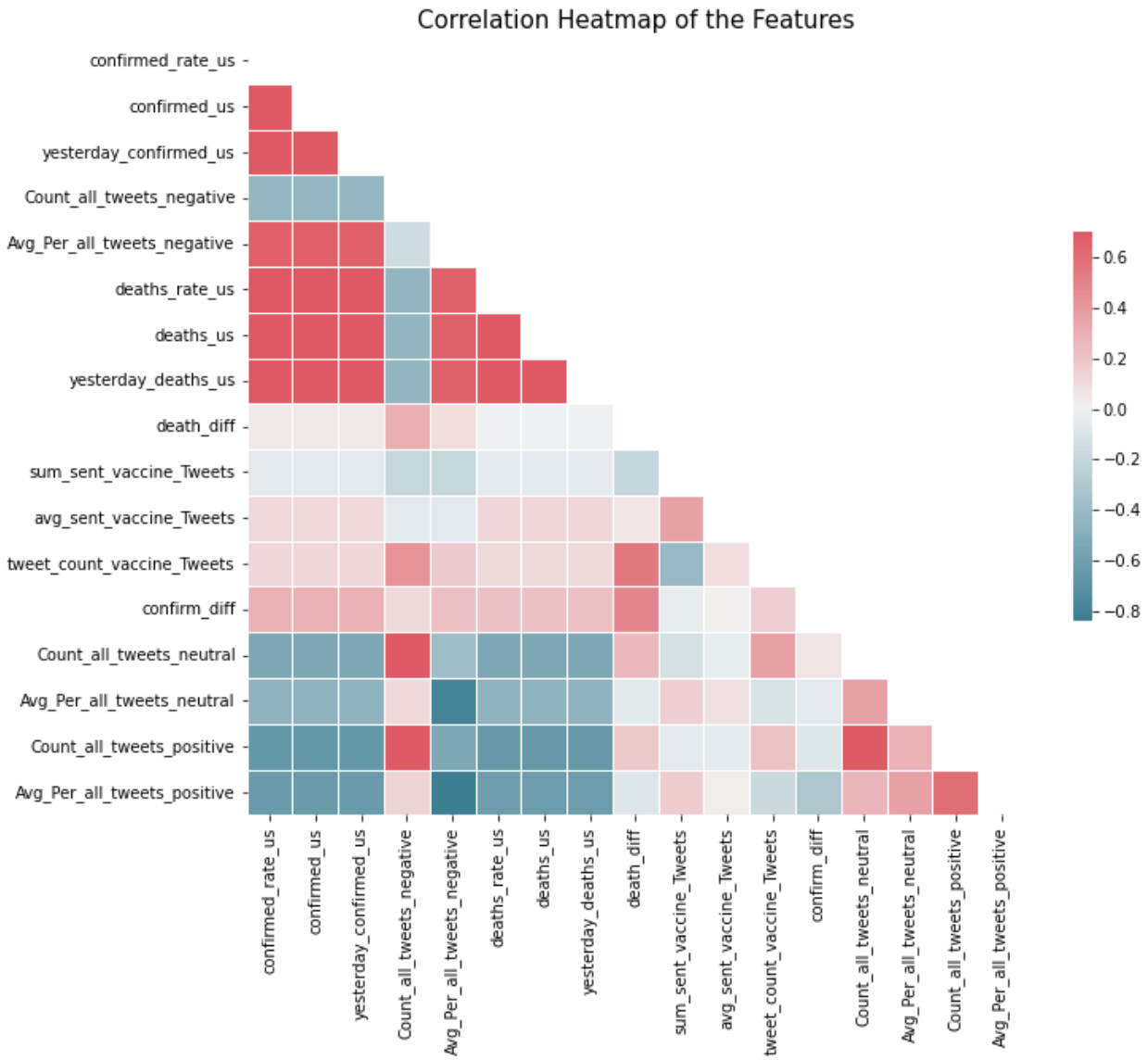

Figure13. Pairwise correlation coefficients in linear regression model. *In this correlation heatmap, we find that the count of tweets have the most correlation with Y.*

```
X = df_clean[['avg_sent_vaccine_Tweets',#average of sentiment score of vaccine tweets
              'tweet_count_vaccine_Tweets',#count of vaccine tweets
                'Count_all_tweets_neutral',#count of All covid neutral tweets
                'Count_all_tweets_positive',#count of All covid positive tweets
                'Avg_Per_all_tweets_negative',#average percent of All covid negative tweets
              'Count_all_tweets_neu&pos&neg-avg',#average count of All covid 3 kind of tweets
                'Count_all_tweets_negative',#count of All covid negative tweets
                'squared_avg_sent_vaccine_Tweets',#squared average of sentiment score of vaccine tweets
              'cubed_tweet_count_vaccine_Tweets',#cubed average of sentiment score of vaccine tweets
              'interaction_vaccine_tweets_sent',#interaction between the count of vaccine tweets and the sentiment scores
             'ratio_negative_neutral_tweets',
              'ratio_positive_negative_tweets',


             ]]
y = df_clean['confirm_diff']
```

Figure14. Features selected for Linear model. These features are used both in linear regression model for daily increased confirmed cases, and model for daily increased Death cases.

```
# Split the data into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=42)

# Create a Ridge regression model with L2 regularization
# Here we're setting the regularization strength to 1.0, you can adjust this value
model = Ridge(alpha=500)
# Train the model
model.fit(X_train, y_train)
# Use the model to make predictions on the testing data
y_pred = model.predict(X_test)
```

Figure15. Linear model with L2 Regularization code. After trying the regularization parameters α from 0.01-1000, we choose α=500 to reduce the model complexity.

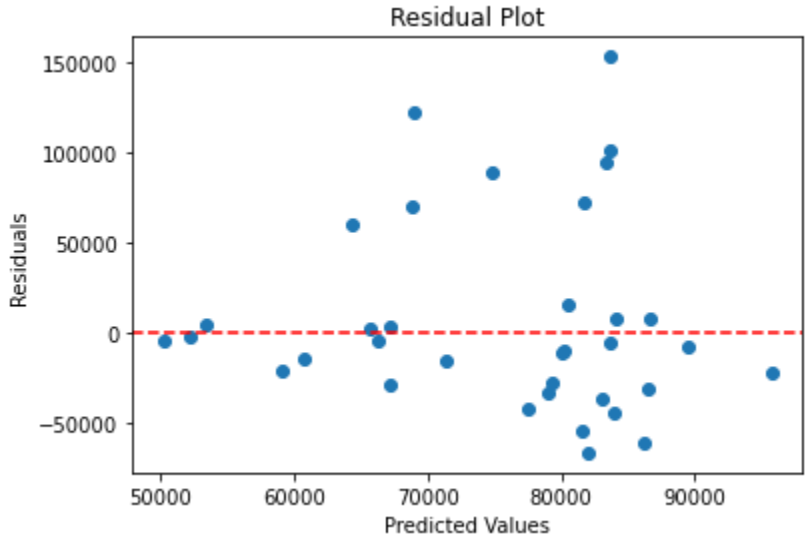

Figure16. Residual plot for Linear model predicting daily confirmed cases. Points with residual < 0 are more close to the zero-line, points with residual > 0 are more far from the zero-line.

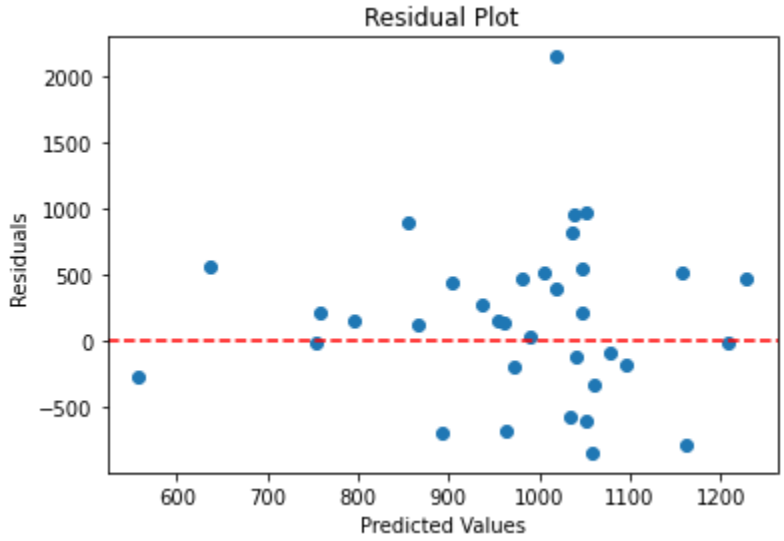

Figure17. Residual plot for Linear model predicting daily death cases. Points with residual < 0 are more close to the zero-line, points with residual > 0 are more far from the zero-line. The residual plot for this death cases model looks better than that for the confirmed cases model.

References:
[1] This dataset “COVID-19 Tweets Dataset” is from kaggle provided by CHRISTIAN LOPEZ, updated a year ago. Link: https://www.kaggle.com/datasets/lopezbec/covid19-tweets-dataset
[2] This dataset “Covid Vaccine Tweets” is from kaggle provided by KASH, updated 7 months ago.Link: https://www.kaggle.com/datasets/kaushiksuresh147/covidvaccine-tweets
[3] This dataset “COVID19 Tweets” is from kaggle provided by GABRIEL PREDA, updated 3 years ago. Link: https://www.kaggle.com/datasets/gpreda/covid19-tweets