Covid – 19 Cough Audio & Data Analysis

Gursmeep Singh Syan

301386570

Vaibhav Saini

301386847

Brian Michael Angelo Panjaitan

301332589

Background

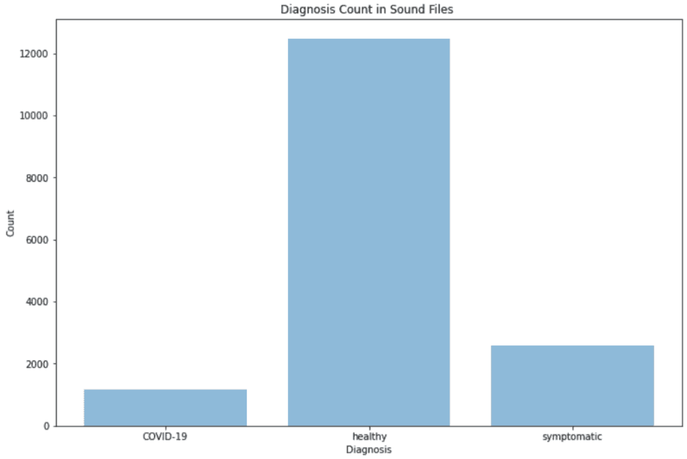
We wanted to have a project where we could tackle a real-life problem or at least get some amount of tangible result regarding that problem so we chose to analyze the dataset available at https://www.kaggle.com/andrewmvd/covid19-cough-audio-classification. This dataset deals with the Covid-19 cough audio recorded by physicians across the world and shared into a corpus for mutual progressive learning about Covid-19 and focuses on the dry cough symptom which has been reported to be an indicator of coronavirus. Covid-19 was declared a pandemic by the World Health Organization on March 11, 2020, has claimed over a million lives worldwide. During earlier periods of the pandemic when testing kits weren't as readily available as now, cough was a major factor to analyze the severity of one's contraction of this virus.

About the data

The problem that we are addressing is how well are we able to accurately predict whether or not someone has COVID-19 from their age, gender, location, and cough. The dataset that we are working with is a csv file containing all the metadata of the individuals and the details about their cough.

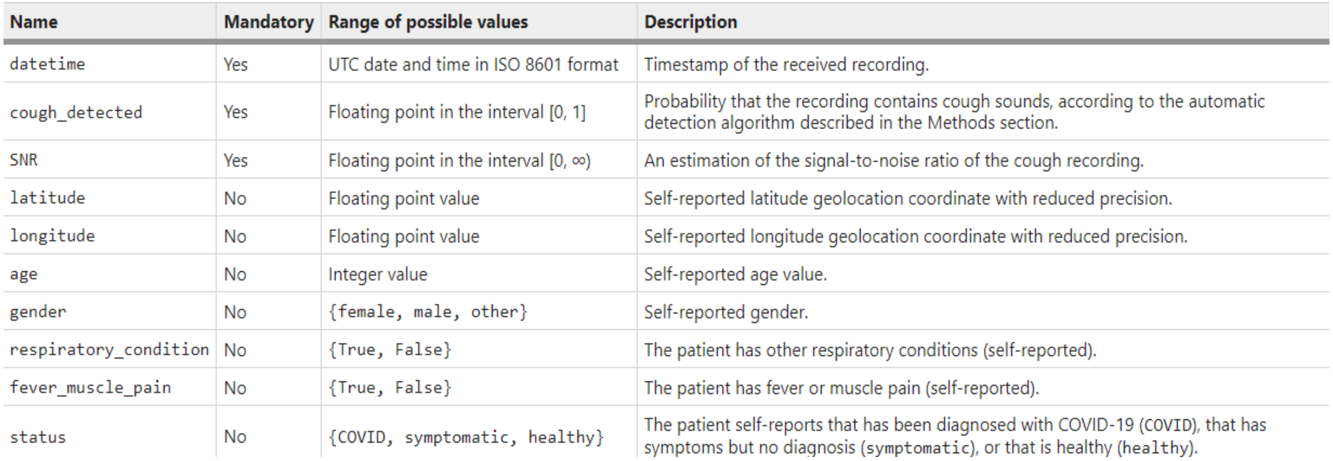
As we wanted to divide our project right from the start but still be able to access each others files we chose to work on google collab while having the data on collab. The files were approximately 12 gigabytes in size and we compiled information about the data into a metadate file.

For the files that had the actual recordings of cough samples we use feather-format objects to get valuable information out of the audio wave patterns and save processed data into multiple '.ftr' files that we can read later . COUGHVID dataset had a lot of undesirable non-cough sounds in the audiofile which we chose to exclude from our analyses.



Plotting the number of files in each diagnosis. [Covid -19 , Healthy , Symptomatic]
The data has almost 75% healthy diagnosed files.

Upon exploring the data and looking at the presented values in metadata file we were able to deduce the following:

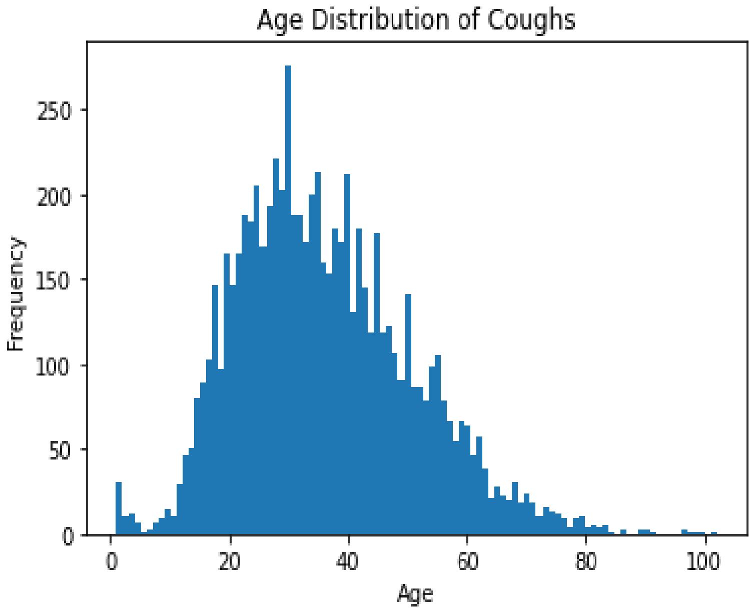


Name	Mandatory	Range of possible values	Description
datetime	Yes	UTC date and time in ISO 8601 format	Timestamp of the received recording.
cough_detected	Yes	Floating point in the interval [0, 1]	Probability that the recording contains cough sounds, according to the automatic detection algorithm described in the Methods section.
SNR	Yes	Floating point in the interval [0, ∞)	An estimation of the signal-to-noise ratio of the cough recording.
latitude	No	Floating point value	Self-reported latitude geolocation coordinate with reduced precision.
longitude	No	Floating point value	Self-reported longitude geolocation coordinate with reduced precision.
age	No	Integer value	Self-reported age value.
gender	No	{female, male, other}	Self-reported gender.
respiratory_condition	No	{True, False}	The patient has other respiratory conditions (self-reported).
fever_muscle_pain	No	{True, False}	The patient has fever or muscle pain (self-reported).
status	No	{COVID, symptomatic, healthy}	The patient self-reports that has been diagnosed with COVID-19 (COVID), that has symptoms but no diagnosis (symptomatic), or that is healthy (healthy).

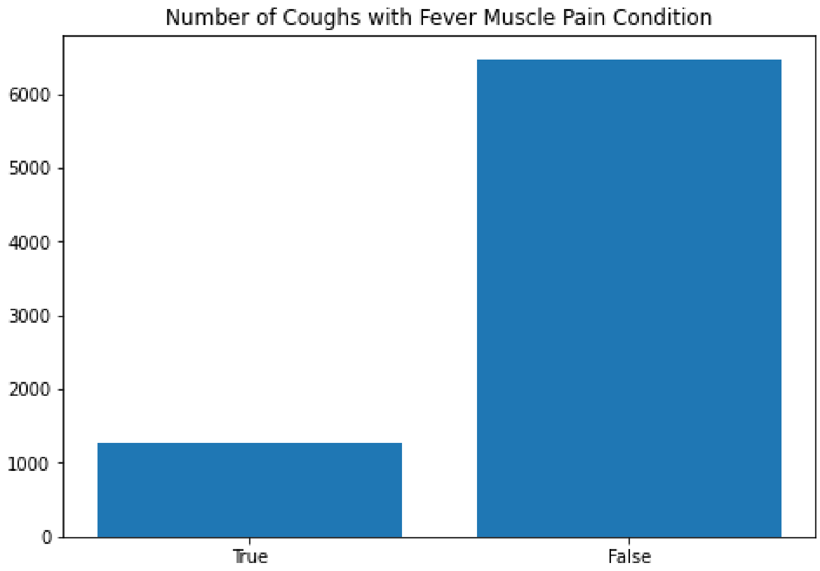
Figure 1: Referenced from https://www.nature.com/articles/s41597-021-00937-4/tables/2

Thus , the data includes personal and medical information that might help us such as respiratory conditions or fever muscle pain. The cough details were analyzed from audio recordings and written down in categories by four medical professionals. This included whether or not congestion, wheezing, choking, and other notable factors were present in the audio clip of a person's cough.

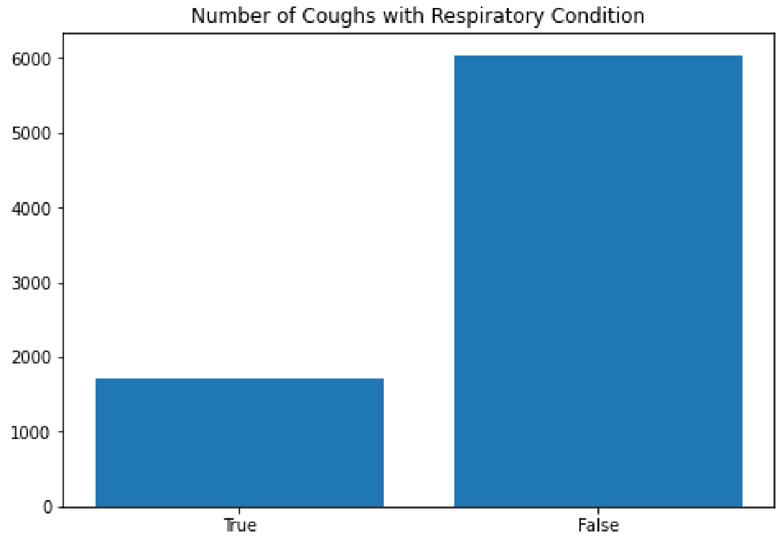
Each file had a unique ID value stored int the 'uuid' column of the dataframe we created. Performing "demographic analysis" on these different 'uuids' we were able to concur :



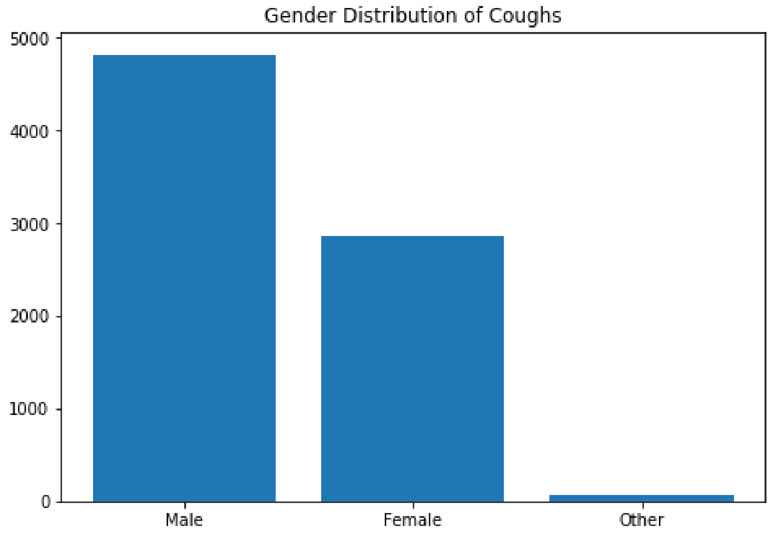
1. Number of data entries according to 'feature' age of patients



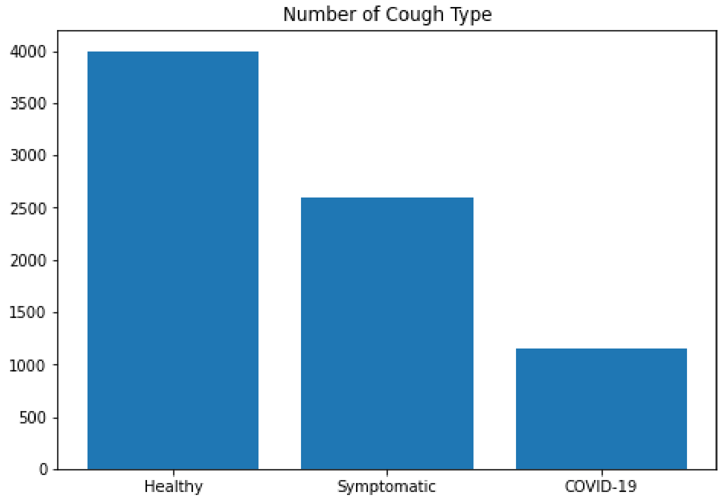
2. Number of data entries according to 'feature' fever muscle pain reported by patients



3. Number of data entries according to ‘feature’ respiratory condition of patients recorded



4. Number of data entries according to ‘feature’ gender of patients



5. Number of data entries according to ‘cough type ’ as reported by physicians patients
   NOTE: This is different than previous graph

Now we take a deeper look at the data we have and found that there were more features which may not look so effective to establish more information about the data but nonetheless we decided to evaluate them as well. These features are result of diagnosis of cough of patients by physicians and some refer to the audio files :

- Quality - Audio quality
- Cough type - Type of cough
- Dyspnea - Shortness of breath
- Wheezing – If cough leaves the patient wheezing
- Stridor - Noisy or high-pitched sound when breathing
- Choking - If patient is choking
- Congestion - If patient has congestion
- Nothing - If there is no cough
- Diagnosis - The verdict of the physician
- Severity - Severity of coughing

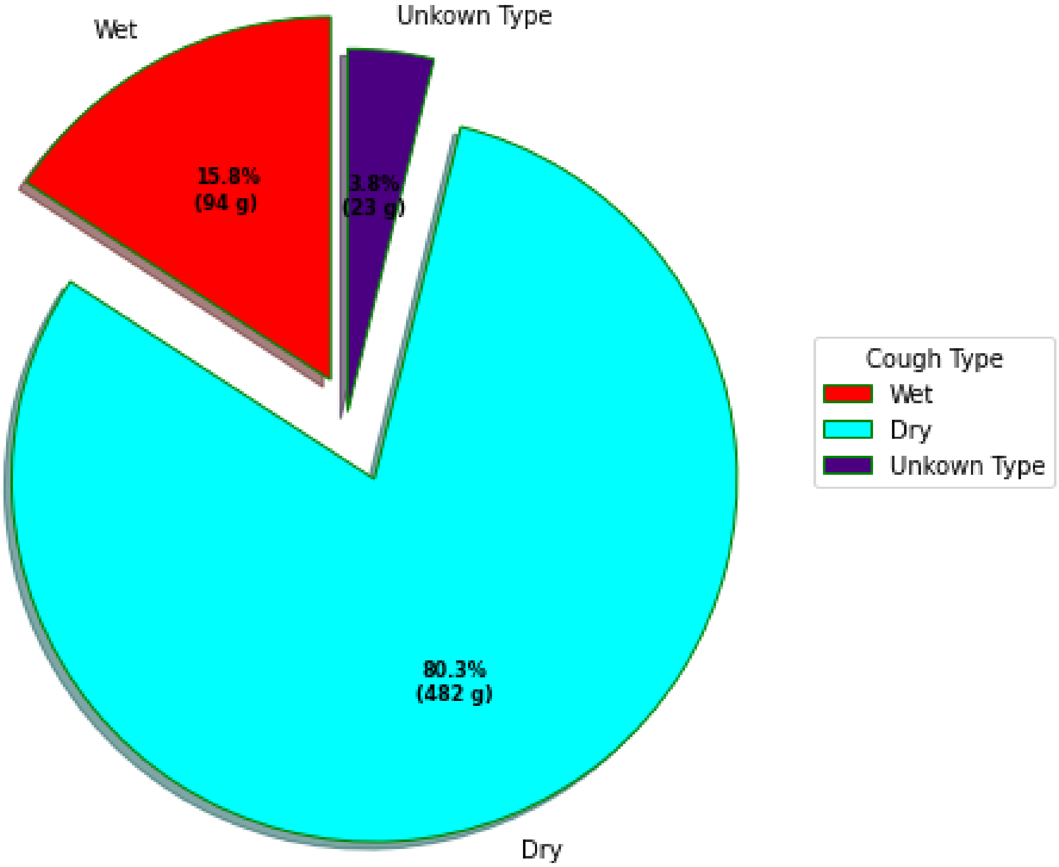
All of the coughs in the public database that were labeled as COVID-19 among the four experts were subsequently pooled together and analyzed for trends in the attributes of the cough recordings.

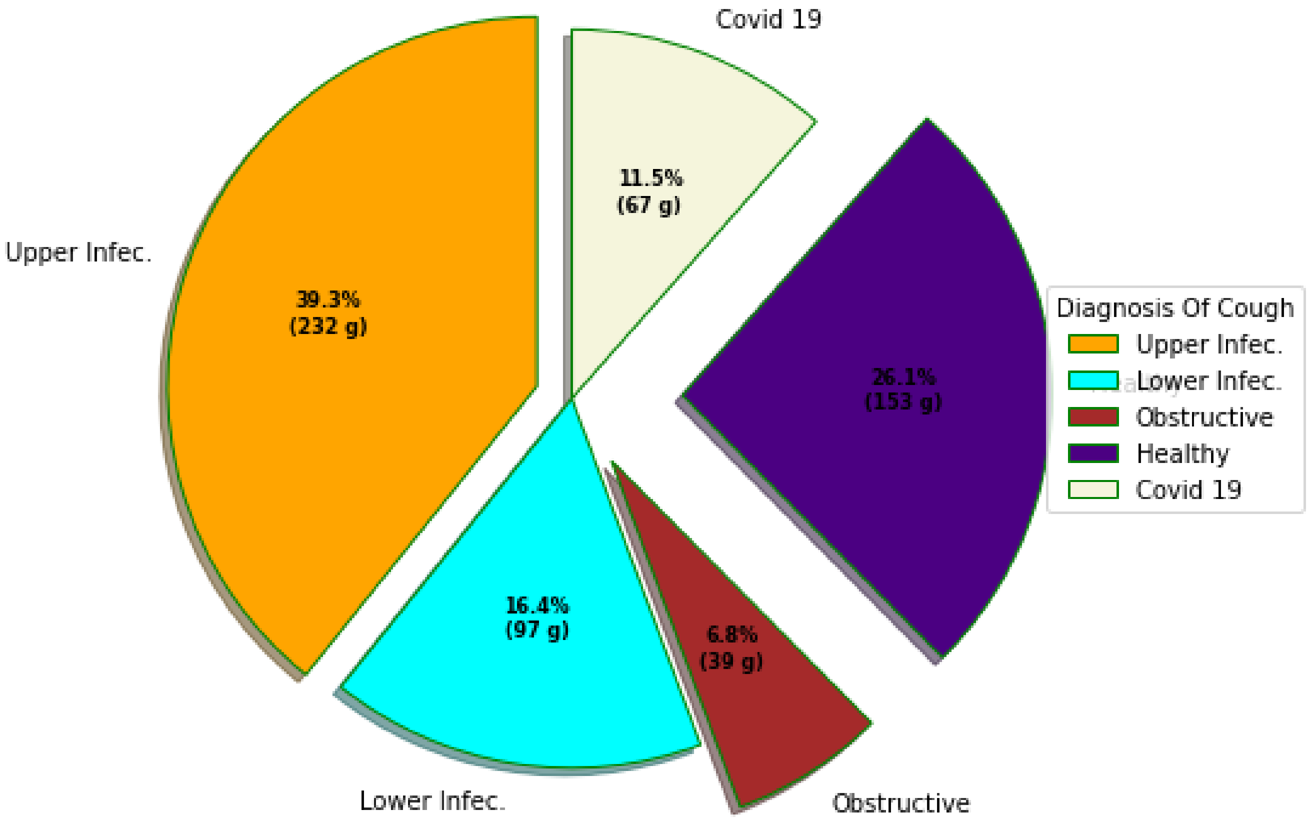
For the project, data was continuously cleaned and filtered so we were able to build accurate models for predicting. The original dataset was heavily imbalanced, having a 11:1 ratio in favour of the individual being “healthy”. To mitigate this bias, we randomly selected and removed “healthy” entries so our dataset would be a little bit more balanced. Additionally, any of the null entries for the “cough_detected” column were removed. The vast majority of coughs did not exhibit audible dyspnea (93.72%), wheezing (92.43%), stridor (98.71%), choking (99.20%), or nasal congestion (99.03%), so our group had to make an assumption that any null entries for these categorical and boolean columns were categorized as either “False” or “None” so we could make predictions

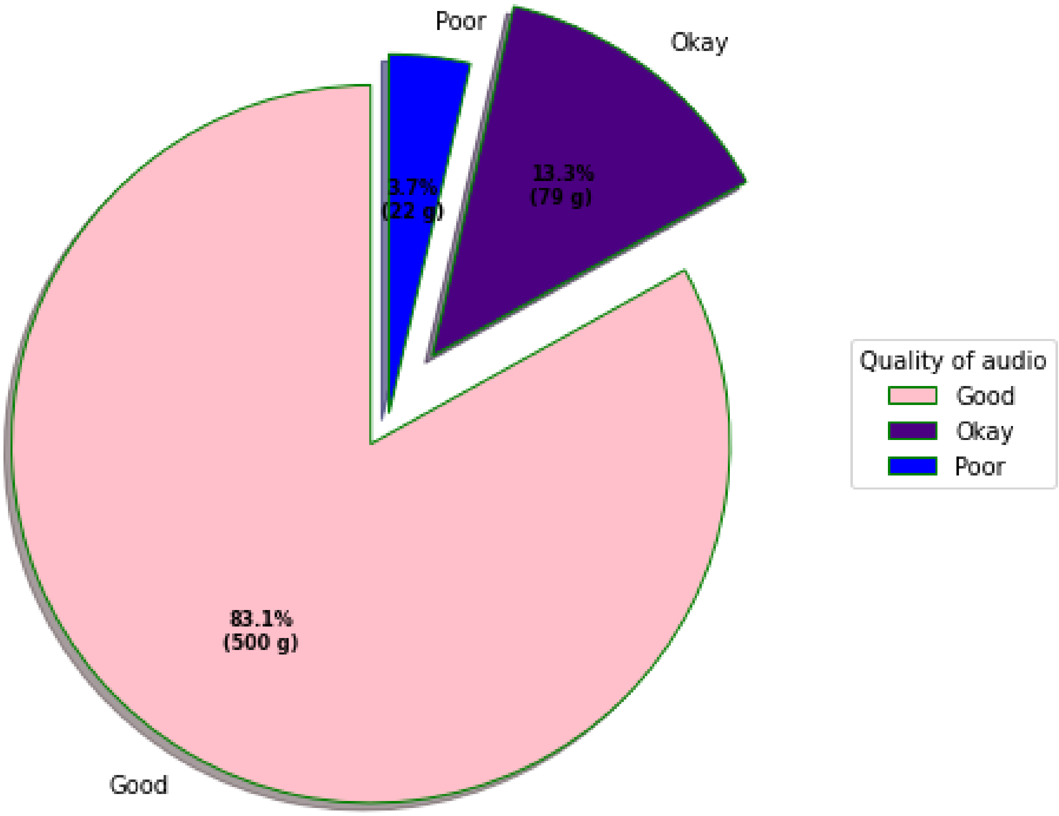
Additionally, 86.96% of COVID-19-labeled coughs are annotated as dry, which is consistent with literature stating that a dry cough is a common COVID-19 symptoms.

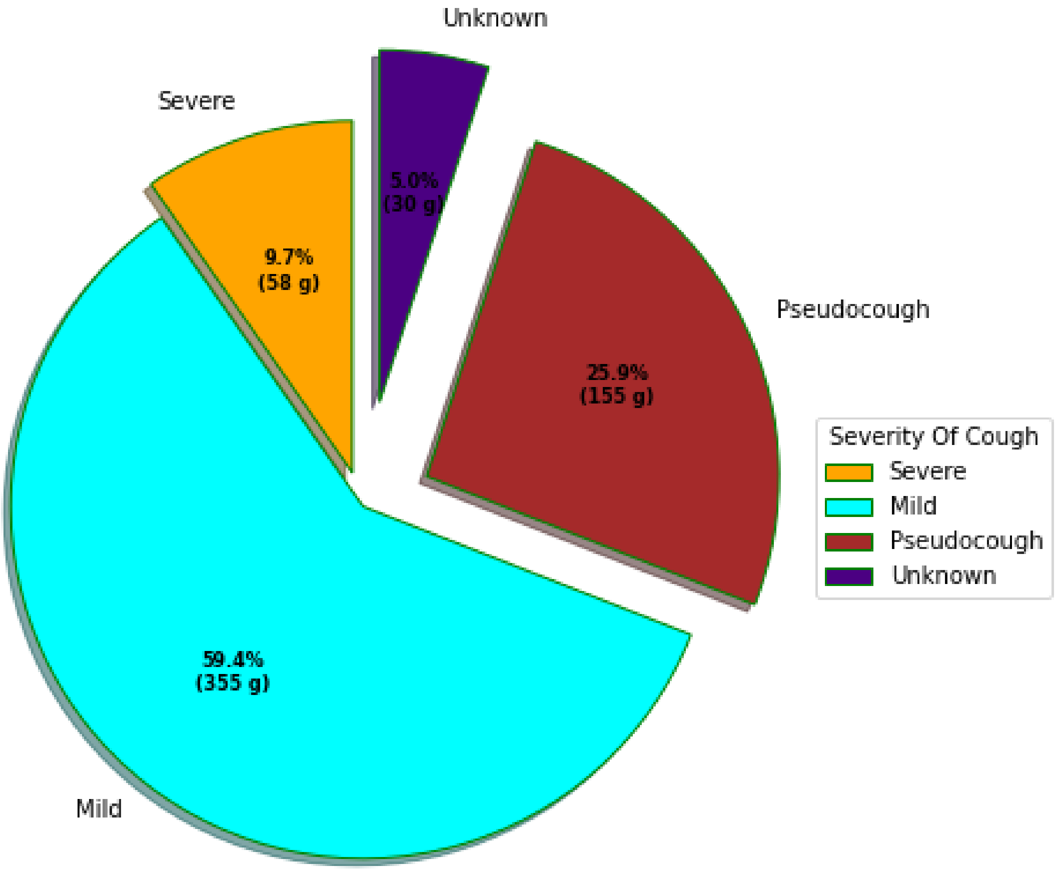
Before we began modelling, four of our feature columns were categorical while seven of them were boolean. In order for us to build our models, we needed all of our values to be numeric. For categorical variables, a technique named as one-hot encoding with dummy variables was employed. For example, instead of having a "severity" column that took in the values mild, severe, pseudo cough, unknown, or none, the column was replaced with all the possible values and was either assigned 0 or 1 for being present or not. For our boolean columns, we mapped the values true and false to both 1 and 0 respectively.

These charts will show more information about these features which we took from only the non-null values we had:




Covid 19
11.5%
(67 g)
Upper Infec.
39.3%
(232 g)
26.1%
(153 g)
16.4%
(97 g)
6.8%
(39 g)
Lower Infec.
Obstructive
Diagnosis Of Cough
Upper Infec.
Lower Infec.
Obstructive
Healthy
Covid 19


Poor
Okay
3.7%
(22 g)
13.3%
(79 g)
83.1%
(500 g)
Good
Quality of audio
Good
Okay
Poor

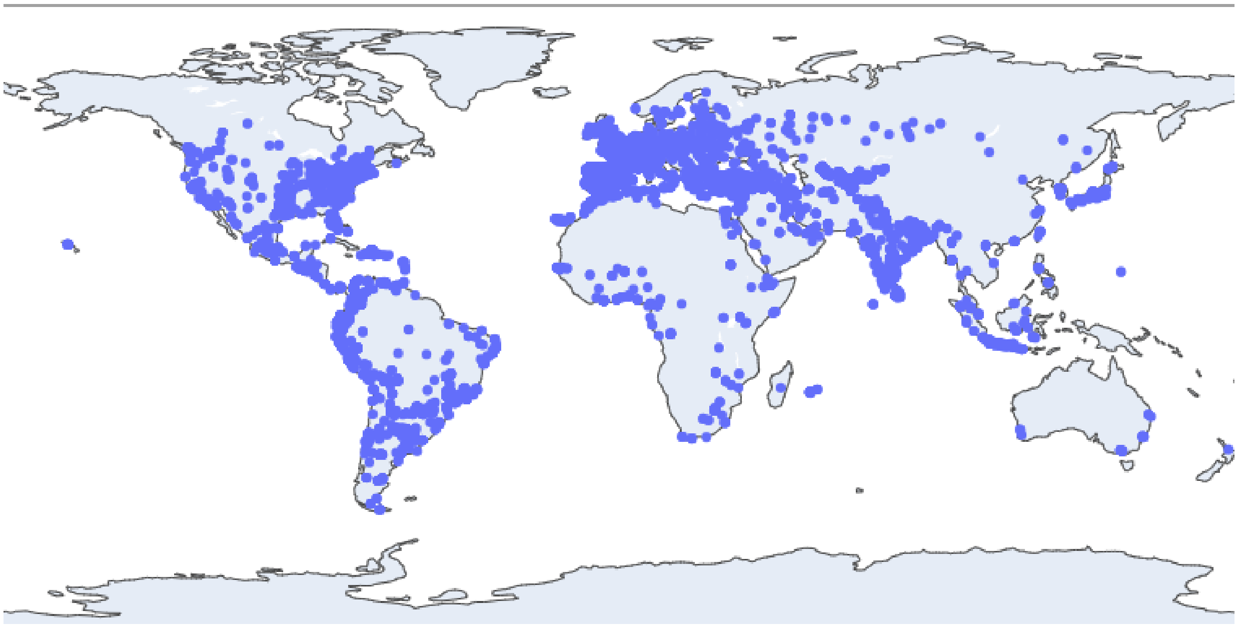


Another important feature of the dataset is the latitude and longitude values of the recorded Coughvid data which shall further shed light on whether the data is collected on a large scale and spread uniformly because only then we'd be able to make a 'global prediction' about our findings.

Using plotly , geopandas and descartes libraries in python we were able to generate the following map with our coughvid data.

It is also important to note that the map below does not indicate anything about the covid cases in the countries with data points plotted in them and shall not be considered as a source of determining how densely the cases are reported in such countries. It merely represents the data we have in accordance with their latitudinal and longitudinal features.

Showing the data on World map



Modelling

Basic Modelling on MetaData

The algorithms that we used to predict whether or not someone had COVID-19 was Linear Regression, Neural Network, Decision Tree, Random Forest, Naive Bayes, SVM with feature scaling, KNeighbours with feature scaling, and a Voting Ensemble. Our group also used two different X sets to predict y. The first set were only features that were from the cough details. This excluded the SNR (Signal-to-Noise Ratio) and the cough detected number. For the second set included, we included all the features in our X.

Using the train test split, we trained our models with 'X_train, y_train' and made predictions for 'y_test' using 'X_test'. The y_test array contained 400 healthy entries, 278 symptomatic ones, and 97 individuals with COVID-19.

•With the first set, the models score and predictions:

Linear Regression

- Healthy/Symptomatic: 772
- Covid: 3

Neural Network

- Score: 0.6051612903225806
- Healthy: 487
- Symptomatic: 219
- COVID-19: 69

Decision Tree

- Score: 0.6335483870967742
- Healthy: 486
- Symptomatic: 225
- COVID-19: 64

Random Forest

- Score: 0.5987096774193549
- Healthy: 588
- Symptomatic: 187
- COVID-19: 0

Naive Bayes

- Score: 0.5406451612903226
- Healthy: 711
- Symptomatic: 7
- COVID-19: 57

SVM (Feature Scaling)

- Score: 0.6425806451612903
- Healthy: 485
- Symptomatic: 283
- COVID-19: 7

KNN (Feature Scaling)

- Score: 0.6361290322580645
- Healthy: 484
- Symptomatic: 234
- COVID-19: 57

Voting Ensemble

- Score: 0.5883870967741935
- Healthy: 572
- Symptomatic: 142
- COVID-19: 61

From the first set, the Decision Tree, SVM, and KNN algorithms produced the highest score while the Neural Network model had the most COVID-19 results at 69.

•With the second set, the models predicted:

Linear Regression

- Healthy/Symptomatic: 762
- Covid: 13

Neural Network

- 0.6051612903225806
- Healthy: 486
- Symptomatic: 233
- COVID-19: 56

Decision Tree

- Score: 0.6606451612903226
- Healthy: 484
- Symptomatic: 258
- COVID-19: 33

Random Forest

- Score: 0.6296774193548387
- Healthy: 583
- Symptomatic: 192
- COVID-19: 0

Naive Bayes

- Score: 0.5277419354838709
- Healthy: 711
- Symptomatic: 4
- COVID-19: 60

SVM (Feature Scaling)

- Score: 0.6154838709677419
- Healthy: 492
- Symptomatic: 258
- COVID-19: 25

KNN (Feature Scaling)

- Score: 0.6464516129032258
- Healthy: 496
- Symptomatic: 218
- COVID-19: 61

Voting Ensemble

- Score: 0.6167741935483871
- Healthy: 593
- Symptomatic: 126
- COVID-19: 56

From the second set, we can see that the KNN and Decision Tree models still performed the best in terms of score while the SVM model dropped by 3%.

After close analysis of our results, we noticed that the two different X_test sets didn't make a major difference. If anything, we felt like using the second set which included all the features performed slightly worse and gave a lower score than the first set in terms of accurately predicting an individual's status.

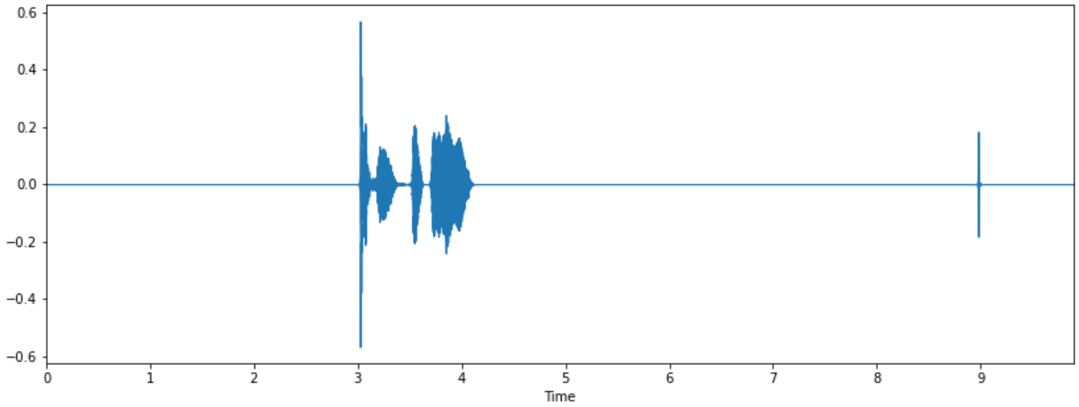
We felt like a big reason as to why the scores and predictions were relatively low is due to the dataset being inherently flawed. We also felt like a lot of the symptoms from an ordinary cough and from a COVID-19 cough overlapped significantly. There weren't many factors that really distinguished the two since one of the most common COVID-19 symptoms is a dry cough. Another thing to note is that all of the coughs that were analyzed by medical professionals were done through audio files. **This means that the doctors were not able to conduct any further analysis on the patients themselves or how they coughed which in turn can affect how they recorded the data.**

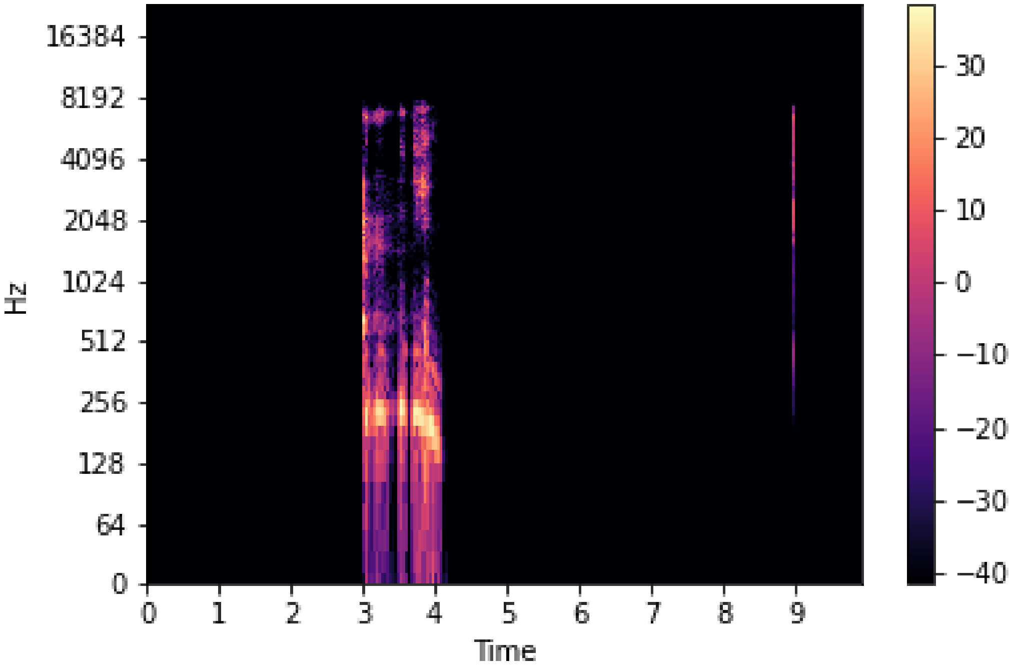
Ultimately, it is very difficult to determine whether or not someone has COVID-19 from just their cough and personal information alone. We do believe improvements could be made though. Because the data was so limited, a lot of assumptions in the empty entries had to be made which in turn affected how our models predicted. If we had more time, we believe we could have also weighted certain columns/features over others when we were mapping and hot encoding. We felt like some factors such as severity and respiratory conditions were more important than wheezing or choking. Because we did not employ this, all binary values were treated equally which may have affected how our model is trained.

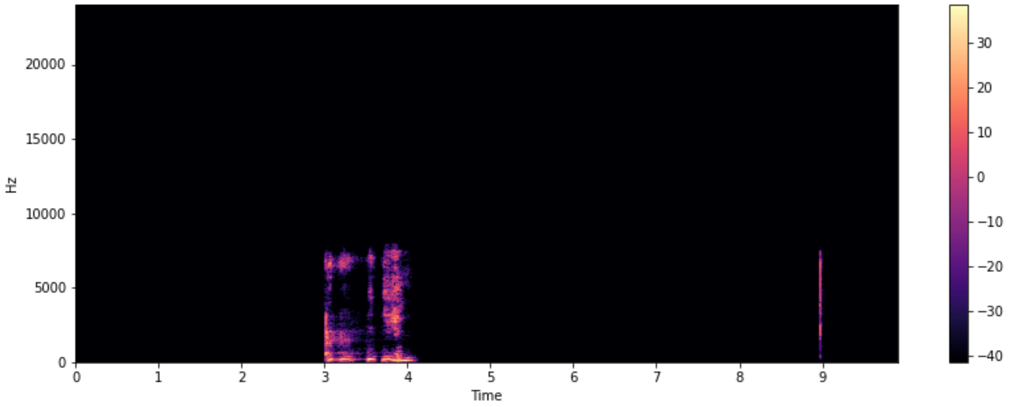
Modelling on Processed Audio Data

To begin modelling our audio file inputs we first need to convert the audio data we had into a numerical form. To do this, the best results were shown by using the 'librosa' package which we used to convert all of the data of a spectral audio wave into a numerical set of values representing that wave which is a numpy array. Also , we suppressed the data from these audio values into a pickle-type object as converting large chunks of data present in audio files into a csv may lead to data loss. This loss can

completely ruin our feature extraction process hence this change in format from what we have usually done in this course is a necessary step for our modelling.
Here are examples of one spectral graph and 2 audio file spectral images of cough data which we converted to numpy array using librosa and stored with pickle.







A major step is to decide what to do with the files containing audio data on the symptomatic patients. We chose to ignore this data and remove it from our training and testing data as it fails the purpose of creating a model which can predict either a cough sample is indicator whether a person has covid or not. Including the 'symptomatic' data wont help as it isn't truly useful to us.

Feature extraction is the essential part of extracting any Machine Learning process. It helps us speed up the training , reduce the risk of overfitting , reduce number of variables involved and create a much more explainable model. The features were:

- mfccs : Mel-Frequency Cepstral Coefficients
- mel : Mel-scaled spectrogram
- cstft : Chroma feature
- rms : Root Mean Square
- spec_cent : Spectral Centroid
- spec_bw : spectral bandwidth
- rolloff : rolloff frequency
- zcr : zero crossing rate

Since they were large arrays with multiple sets of these features , we extracted only the mean value features for each audiofile.
We then split the data we had into training and testing subsets and perform Scaling and PCA on these features. Next using the GridSearchCV model we were able to see

the best result from a KNN type algorithm which is executed by GridSearchCV along the set of neighbours ranging from 3 to 21 and delivering the one giving highest accuracy.
It is important to note that using a GridSearchCV model is essentially much better than a traditional model because it takes away the hit & trial usage of a KNN with a specified number of neighbors and always delivers the best possible result.

Our model then produced the following results for prediction and classification on the test set with the model giving a total **<u>Accuracy of 0.77</u>** which is much higher than any of the models we saw before.

	precision	recall	f1-score	support
healthy	0.78	0.99	0.87	940
COVID-19	0.35	0.03	0.05	268
accuracy			0.77	1208
macro avg	0.57	0.51	0.46	1208
weighted avg	0.68	0.77	0.69	1208

Also given below is a Confusion Matrix for our results.

927	13
261	7

This confusion matrix tells us that there are only 13 scenarios when a covid audio has been classified as a healthy audio (TYPE 1 ERROR) which is very impressive as compared to the scenario when there are 261 healthy audios reported as covid (TYPE 2 ERROR). This trade-off is essentially not harmful to real life situation when we can recheck the files multiple times to be sure of our results.

Another interesting pointer here is that this model works so well because the KNN model we saw before which gave results on metadata inputs was one of the highest performing models and we were able to boost its accuracy upto 77% because the audio in naturally linked with the features of metadata.

Project Experience

Brain :

- Plotted and analysed the notable characteristics and features of our data and organized entries into categories.
- Conducted through various ETL tasks
- Task was completed to analyse the imbalance and flaws in the dataset so adjustments and assumptions could be safely made.
- Created and trained a plethora of models to compare and analyse results. Models were used to determine whether we could accurately predict someone having COVID-19 by their basic information and the details of their cough.
- Employed different data cleaning/filtering techniques such as one-hot encoding with dummy variables and mapping for boolean/categorical variables.
- Also performed data visualization to plot and compare the predictions and test set of each model.

Gursmeep :

- Used ETL and filtering methods on the dataset.
- Analysis of given features
- Create Plots and Models for said features along with their scores
- Used ML on datasets
- Learned about descartes, plotly, pickel, feather, librosa and other python modules and also how to use Google Colab
- Created a report for the project
- Helped in productive group functioning and timely planning

Vaibhav:

- Used Librosa to convert audio data into numpy array

- Saw how the spectral data is stored into the array and saved files in pickle and feather formats for ease of use
- Analysed and extracted useful features
- Trained Models for special extracted features along with their scores
- Then also created a confusion matrix to show errors
- Was able to create a model of 77% accuracy
- Learned how to use Colab and Gitlab