Zhi Heng:

Hi prof. Quick check, how do we know if a date is of interest in the COVID-19 Pandemic?

To start, we need to define what is date of interest and we define it as days during the pandemic that there are either surges or plunges in case numbers. In other words, anomalies in the data.

So, we intend to implement a monitoring system to detect anomalies in the COVID-19 pandemic and you might ask how it can be useful?

The project will be useful for authorities in contact tracing and identifying potential events that caused the anomaly. And..

Possibly preventing a meltdown in healthcare systems.

The data that we will be using is collected by the COVID Tracking Project, though they have stopped collecting new data since 7 March 2021. While this would impair our model's accuracy in the future. The existing data still proves to be very useful.

To prepare and clean the data, we dropped all columns deemed unnecessary, as they do not provide any additional insights. For example, deprecated variables. Whilst, preparing the National dataset, we found that not many states were reporting their case numbers accurately and correctly.

This would impact our model's accuracy, and we decided to not apply it on the National dataset. As we are not using National Data.

For our project, we decided to use New York's data. According to the Centers for Disease Control and Prevention (CDC), it was one of the early epicenters of the COVID-19 pandemic in the US. (Thompson et al., 2020)

An article on Financial Times covered how New York's missteps let Covid-19 overwhelm the US. (Kuchler & Edgecliffe-Johnson, 2020) Thus, making it a suitable choice for study.

In the Exploratory Data Analysis phase, often we are more concerned with the growing number of new cases, and not the total number of cases.

With this in mind, we first identified what could this variable be in the dataset and the PositiveIncrease variable is what we are looking for.

We then used pair plots and correlation matrix to learn more about the distribution and relationship of PositiveIncrease to other variables. With that, I will pass the time to Zhi Hua.

Zhi Hua:

Thanks Zhi Heng.

Hi, my name is Zhi Hua and now, I would like to talk about Local Outlier Factor, which is one of the methods we tried to detect anomalies in the COVID-19 dataset.

Local Outlier Factor will first generate a score for each data point based on the density and distance to its K nearest neighbours.

And data points with high outlier scores are then identified as anomalies.

Local Outlier Factor is usually used for complete data sets and since we are trying to detect anomalies on a day-to-day basis,

We need to ensure that this method can be used in streaming data environment.

So our hypothesis is that,
feeding the last K days will result in reliable anomaly detection but we are unable to confirm it without testing.

Thus, how do we test our hypothesis?

We generate a graph daily to observe if the anomaly detection for the last day is reliable.

The inputs we used for the test are the new-york dataset, k = 7 and contamination factor at default.

So to quickly explain what it will look like, on this particular date, we only have 2 days worth of data, so only 2 data points.
Since 2 is lower than K which is 7, we will use all the data available.

It will remain this way until when the amount of data points are higher than K, then we will start using only the latest 7 data points.

So with a testing method formulated we wrote a script and many simulated days later… … this is what we get!

So at one look, things seem fine.

Until we realize anomalies are sometimes not where we expect them to be.

Let’s look at the both of them in detail shall we?

As the daily positive surge, the program will initially detect the latest date as anomalies until it starts to become an upwards trend.

For the days which are unexpected, because of how much the data fluctuate, anomalies are mostly not detected.

So can we conclude that the Local Outlier Factor is somewhat reliable to detect anomalies?

Not quite, we need to do more testing.

So for our other tests,
We played around with

different K values,

data sets

and contamination factor.
...and after more testing...

We found that the dates marked as outliers do not always reflect anomalies of the trend

and for the algorithm to be accurate, we need to manually set the contamination factor and K values for each environment.

Even with those adjustments, we are still not confident with the output of the program.

Therefore, we need something else!
I’ll let my friend Chang Han talk about the other algorithm we tried to detect anomalies.

Chang Han:

Thank you Zhi Hua, I will now be explaining the ARIMA model.

What is the ARIMA model?

The ARIMA model is an acronym that stands for: Auto Regressive Integrated Moving Average

It is a class of models that 'explains' a given time series.

The model has uses, mostly involving a time series, but we will focus on using it to

1. Identify trends
2. Forecast future values, based on past values in the existing time series

We will also be slightly repurposing the ARIMA model to suit our problem of detecting anomalies.

The ARIMA model relies on 3 values, named p, d, and q.

P stands for the number of lags to be used as predictors.

If we give P a value of 2, a forecasted value will be a function of the previous 2 days of the time series

D stands for the number of times we have to differentiate the graph.

If the graph has seasonalities or gradients that causes the mean to fluctuate too much, we can differentiate it

Q is the Number of lagged forecast errors we want to include in our forecasting.

With this in mind, we will use the values 4,1,1 for our ARIMA model.

So here is how we will apply our model:

First, we partition the data into train and test. We will use 66% of our data for train, and 34% for test.

We then use the get forecast function to get the next day's forecast

For every one day we forecast using train data, we will compare predicted values with actual test values

we will then add the actual test values back into the train data to model the data again for the next day

Confidence intervals shows us a range where we are fairly sure our true value lies in, based on forecasted values

For this diagram, we use a confidence interval of 85%

As we exhaust our test samples, we generate forecasted values, as well as its confidence intervals

Through this plot, any actual test value that exceeds the confidence interval is identified.

Here are some anomalies that we can identify!

Through this, a list of dates can be compiled, acting as outliers predicted by our model

Our model can now be used as a tool to monitor anomalies

However, it should be noted that if we want to forecast more than a day ahead, our range of confidence interval will significantly widen.

Back to you, Zhi Hua.

Zhi Hua

Thanks Chang Han!
Now that we've tried two different approaches that will detect anomalies, which algorithm should we use?

Obviously, we will be using ARIMA due to Local Outlier Factors' unreliability, but before that, let's talk about two important things we've learnt from this project.

Local Outlier Factor is unreliable in streaming data environment, even when we are only using

the last K days and result of the final day, as it is designed for static data.

On the other hand, ARIMA can understand trends and is significantly more accurate at detecting anomalies, as it is based on a time series.

Since we want to build a program that will alert authorities to investigate suspicious surges and/or plunges in the increment of positive cases

And we can detect anomalies using the ARIMA method…

…ARIMA is the way to go, to accomplish what we initially set out to do.

Before I end the presentation, let me briefly go through each of our contributions.

Zhi Heng did most of the Data Exploration, such as data set and variables selection

I did most of the Local Outlier Factor testing

Chang Han did most of the ARIMA testing

As for the other stuff like problem formulation, project direction and conclusion, we did all of them together with equal contributions.
Thank you!