Training a DeepAR model using Amazon SageMaker to forecast COVID-19 cases and mortalities

1. Background and Objective

Data analysis and predictions based on scenarios involving hypothetical factors have been important for devising policies to cope with the current pandemic, including measures such as national lockdowns. Being able to accurately forecast future transmission involves identifying underlying patterns and influences affecting the spread of the disease. The more that is known about what factors exactly have an impact on transmission - at both the level of unchanging characteristics of a country as well as measurable actions they have taken – the more well equipped states will be for tackling the problems they face. As this topic is having such a significant impact on the lives of such a large proportion of people around the world, I would personally like to undertake this project in an attempt to contribute to the scientific community's understanding of the crisis in any way possible.

Kaggle.com has launched the challenge of forecasting transmission, with the goal of helping to "identify factors affecting the transmission rate" by answering some of the questions posed by the World Health Organisation and the White House Office of Science and Technology Policy. These questions include the following:

- "What do we know about **non-pharmaceutical interventions**?"
- "What is known about **transmission, incubation and environmental stability**?"
- "What do we know about COVID-19 **risk factors**?"

The problem being investigated as part of this project is defined as follows;

"To accurately forecast the number of reported COVID-19 related cases and mortalities through training a DeepAR model, in order to help identify important factors affecting transmission rates."

2. Data Sets

Cases and Mortalities Time Series

The figures of the outbreak, including number of reported cases and fatalities, have been recorded daily by the John Hopkins University Centre for Systems Science and Engineering and made available for public use in the form of a daily time series. These are CSV files, constantly being updated and republished, which detail the cumulative number of cases and fatalities on each day, broken down by country and, in some cases, further by province [1]. Specific provincial breakdowns will be ignored for the purposes of this project, as data on other topics such as demographic information is only readily found at a national level.

Also included will be data sets documenting the history of the SARS outbreak of 2003 [2] and the Ebola epidemic of 2014-2016 [3], both sourced from Kaggle. These data sets, while only relating to a subset of the countries involved in the current pandemic, represent records of previous viruses spreading and subsequently being apprehended on a smaller scale. This information will be important for identifying general trends across the entire timeline of an epidemic.

It may be necessary to normalize the start dates for each time series and trim off the inactive periods at the beginning. It may also improve the model to normalize cases and mortalities by total population.

Transmission Factors

In addition to the time series, data sets will be collated from various sources as follows, in order to quantify the effect of a range of metrics on the model’s prediction. I will aggregate a dataframe of metrics for each country, from three broad categories: transmission/environmental; risk factors; and interventions. The table below shows the metrics to be collected for the three categories, its numerical value and the intended source of the data. The metrics highlighted in orange have significantly greater measurable fluctuations in time and so will be included as independent time series of incrementally changing values rather than a single static value.

Transmission/environment		
Metric	**Value**	**Source**
Border	Island (Y/N)	Google
	Landlocked (Y/N)	Google
Border activity	Annual International Tourist Arrivals	OurWorldInData
Altitude	Average elevation above sea level	Portland State University
Urbanization	Urban population percentage	World Bank
Pollution	Air pollution Index	Numbeo
Temperature	Average Celsius monthly temperature	World Bank
Population	Population area density	Wikipedia
Households	Average household size	Population Reference Bureau
Diversity	Ethnic fractionalisation	Wikipedia
	Linguistic fractionalisation	Wikipedia
	Religious fractionalisation	Wikipedia

Risk factors		
Metric	**Value**	**Source**
Wealth	GDP per Capita	International Monetary Fund
Safety of children	Infant mortality rate	Population Reference Bureau
Smoking	Annual cigarette consumption per capita	Wikipedia
Cannabis	Proportion of population using cannabis	Wikipedia
Obesity	Obesity rate	Central Intelligence Agency
Corruption	Corruption Perception Index	Transparency International
Crime	Crime rate per Capita	Numbeo
Cleanliness	Water and Sanitation Rating	Yale University

Interventions		
Metric	**Value**	**Source**
Lockdown	Proportion of population in lockdown	Wikipedia

All data sets will either be downloaded and converted to csv, or written directly into a csv manually. For the Ebola and SARS data the presence of national lockdowns will be added to simulate the stringent measures that were taken in those cases. All missing data points will be filled in with the average of all known values from adequately “similar” countries in order to avoid having to drop these countries altogether. All of the metrics will be normalised to decimal values between zero and one to make them suitable for training.

Data Processing

The training data will be taken from four csv files, all attached. Three of the files contain the covid, sars and ebola daily time series data for cases and mortalities, and the other file contains the transmission factors and time series. Time series for each country's cases and mortalities will be created from lists taken directly from the appropriately labelled columns where the country name matches, from the first three files. This will be done for the covid, sars and ebola data sets, with the results stored in dictionaries (example below):

```
print(covid_cases_dict)
{ Afghanistan: {“start”: “2020-01-22 00:00:00”, “target”: [0, 0, 0, 0, …..
   ……………….
```

Some linear interpolation will have to be done for the sars and ebola time series as they have some dates missing.

Time series will also be taken from the right hand side of the transmission_factors.csv file as follows. The weather information will be taken from the columns labelled like “Avg {month} temp (C)”. From these columns a time series of temperatures for each country will be created for the relevant month's temperature taking the value on any given day. For example, each time series will contain 31 consecutive values of the average temperature for that country in March, followed by copies of April's value and so on.

Similarly, time series will be made for lockdowns from the columns located at [‘Lockdown 1 Start’: ’Pop. Proportion 3] for each country as follows. Each element in the time series will denote the proportion of the population in lockdown at that particular date. A country will only have values for ‘Lockdown 2 Start’, for example, if it has locked down a new area since its initial lockdown.

There will be a simple time series made from the column headed “Asian response”, to identify the planned measures put in place by a handful of South East Asian countries, with a simple binary time series indicating whether the measures are in place or not for each date. There will be a similar binary time series made for the presence of an international travel ban from the “TB Start”, “TB End” columns.

The static transmission factors will be taken from the left hand side of this file in the range [‘Island (Y/N)’: ‘Water Sanitation Rating’]. These will be normalised to values between zero and one, before a PCA is conducted. Once the PCA has been carried out, the new principal component values for each country will be converted to a daily time series of non-changing values.

All the time series created here will be used as arrays in the dynamic_feat field of the training input.

3. Solution

The project will be carried out in a SageMaker Jupyter notebook workspace, working in Python 3.8 and using an estimator object to train the model. Once the data has been properly preprocessed, and the features have been engineered as intended, the appropriate records will be stored in an S3 bucket. Visualisations will be done using Seaborn for PCA and otherwise Matplotlib.

PCA

Using the static metrics gathered, a Principal Component Analysis model will be trained in order to reduce the number of features and avoid including multiple metrics closely correlated with each other. Testing will need to be carried out at this stage in order to determine a desirable number of principal components: the components should capture 90% of the original data variance. It is expected that at least one principal component should be made using only metrics from the 'transmission/environment' subsection, and at least another from only the 'risk factors' data. Once values of the principal components have been determined for each country, these values will be transformed into a time series of constant values to be used for training.

Training the DeepAR

The principal components, as well as the lockdown and weather features will be included as dynamic feature arrays for training, and a json object will then be created for each national virus outbreak. Two models will be trained, one for cases and the other for mortalities. Each training input will be a single epidemic – e.g. covid_Italy or ebola_Liberia. The DeepAR will be trained on each of the records, and with the hyperparameters epochs, prediction length, and context length being experimented with.

Testing

In order to test the quality of the model, generated predictions for both infections and fatalities will be visualised alongside target values. The quality of the model could be quantified as the proportion of the target data points which lie within a certain confidence interval of the prediction.

The accompanying dynamic factors will also be removed, one by one, and the predictions rerun as a comparison to try to gain insight into the type/size of impact these factors have in determining the quality of the model, as a potential indicator of their influence in the pandemic. These impacts will also be visualised with rerun predictions and quantified by calculating the change in the proportion of target data points captured within the confidence interval.

4. Benchmark and Evaluation

The model trained as part of this project will be tested against the DeepAR model trained in the Jupyter Notebook that was part of the lesson on energy consumption forecasting. This is an example of a reasonably accurate DeepAR model which largely follows the patterns shown by the target for cyclical data, and is a good compromise in the absence of a lack of published examples of this model trained on this particular data set.

The main evaluation metrics will be the proportion of target data points captured within an 80% and 50% confidence interval of the model's predicted values for both infections and mortalities. The general calculation of this metric is outlined below, given the upper and lower confidence bound series, and the target series:

$$accuracy = \frac{x}{l}$$

x: element-wise sum of bool($C_{0.1}$ <= t <= $C_{0.9}$) along 1-D target and confidence bound arrays
where $C_{0.1}$ = 0.1 confidence bound, $C_{0.9}$ = 0.9 confidence bound, t = target
l: length of the target array

Python code representation:

```
ctr = 0

for target, upper, lower in zip(target_series, upper_series, lower_series):
    if (target >= lower) & (target <= upper):
        ctr += 1

accuracy = ctr / len(target_series)
```