**Agenda :** Time series forecasting for hospitalisation based on vaccination and epidemiology

**Description :** This model will forecast the count of hospitalised patients and persons vaccinated on daily basis and in a specific location which will help in planning the requirement for hospitalisation

**Target audience :** government

* forecast of the number of patients and persons vaccinated in a particular location over a future time scale
* based on the forecast planning the requirements for the patients such as doctors count , availability of beds and others can be planned and also effect of increasing the vaccination count can be evaluated

**Model Used :**

multivariate time series forecasting – **Vector Auto Regression (VAR)**

Vector autoregression (VAR) is a statistical model used to capture the relationship between multiple quantities as they change over time. VAR is a type of stochastic process model. VAR models generalize the single-variable (univariate) autoregressive model by allowing for multivariate time series.

**Steps to be followed during implementation : -**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Process | Sub process | Optimisation | Status | Issues faced & learnings |
| Connecting with azure and importing all the datasets as dataframes | 1. Using the standard azure connection code | - | FINISHED | - |
| Defining the schema of the dataframe after analysing the datasets | 1. Collecting all the schema details and specifying manually | Takes very less time while we load the dataframe | FINISHED | Took very long time to run- after defining the schema time taken to run completely reduced |
| **DATA PREPROCESSING** |  |  |  |  |
| Cleaning the hospitalisation data frame | 1. Filling the null values 2. Formatting the date column to date format in pyspark 3. Selecting the necessary data points based on location key and date | - | FINISHED | date format issue  dataframe had repeated dates for various location keys – so filtered a particular location key |
| Cleaning vaccination data frame | 1. Selecting the required columns 2. Dropping the unwanted columns 3. Filling specific columns using forward fill feature 4. Filled the remaining null values with zeros 5. Changed to Date format in pyspark 6. Selected values based on location key | - | FINISHED | Filling null values as zeros in cumulative columns could be inefficient as the cumulative column is influenced by location key so struggled – forward fill feature fixed the issue |
| Cleaning epidemiology data frame | 1. Changed date format 2. Filled null values with zeros 3. Selected rows based on location key | - | FINISHED | - |
| Joining the data frames | 1. Joined the hospitalization and vaccination dataframe using a date – inner join 2. Joined the previous join data and epidemology dataframe using inner join | - | FINISHED | Got unexpected output with more rows – later figured out that was because of date format and fixed |
| Checking the final dataframe | Rows and columns was as expected | - | FINISHED | - |
| Converting the final data frame to pandas and exporting as csv | Exported the dataframe to csv | - | FINISHED | - |
| Dropping the location key column  Converting the final dataframe to pandas | 1. Dropping the location key as is It not an integer 2. Converting final dataframe to pandas as it is required for VAR, VARMAX model 3. Checking the datatypes of the final dataframe | - | FINISHED | - |
| Visualising the dataframe | 1. Using graphs visualising each column of the dataframe | - | FINISHED | Learned visualisation of actual dtaset |
| Checking the stationarity | 1. Performing augmented dicky fuller test for the columns to be predicted to check the stationarity | - | FINISHED | Learning : explored augmented dicky fuller test |
| Checking the corelation of the columns | 1. Performing granger causality test to check the corelation of the columns | - | FINISHED | Learning: explored granger causality test |
| Splitting the training and test data from the data frame | 1. Allocation the training data size and test data size among the dataframe | - | FINISHED | - |
| Model evaluation | 1. Fitting the training data into model and finding the minimum AIC value | - | FINISHED | - |
| Model building | 1. Fitting the data into VARMAX model and specifying the order based on previous test 2. Printing the summary of the model | - | FINISHED | Face type cast issue while prediction – fixed after casting the values to float before fitting into the model |
| Forecasting the model predictions | 1. Specifying the forecast value 2. Performing the prediction based on forecast value | - | FINISHED | - |
| Visualizing the predictions | 1. Visualising predictions using graph | - | FINISHED | - |
| Findinig the mean square error of the predictions and test data | 1. Finding root mean square error to evaluate the correctness of the prediction | - | FINISHED | - |
| Model enhancement and accuracy |  |  | ON PROCESS\* |  |
| Adding additional features |  |  | ON PROCESS\* |  |