**Predictioneer Hackathon**

**Objective:**

To develop a predictive model for estimating the number of deaths, expected cases and case fatality ratio for multiple regions, based on available data using development data and validating predictions on a separate validation dataset.

**Column Descriptions:**

* **Lat:** Latitude of the location
* **Long\_:** Longitude of the location
* **Deaths:** Number of deaths
* **Case\_fatality\_ratio:** Case fatality ratio of the location

**Data Analysis:**

Using various methods from the pandas library to get useful information from dataset

* Getting statistical insights like mean, standard deviation and checking overall size of the dataset.
* Data had a lot of missing values specially for the ‘Deaths’ column, these missing values were handled by dropping them from the dataset
* ‘Case\_fatality\_ratio’ column had some values over 100 and some values less than 0, these values were excluded from the dataset
* Using heatmap to check correlation between the features and a pair plot to explore relationships between features.

**Data Preprocessing:**

* **Handling Missing Values**: The data especially the ‘Deaths’ column had a huge number of missing values, these values were dropped from the dataset.
* Taking the logarithm values for CFR and deaths to reduces skewness, making the data closer to a normal distribution, which many machine learning models handle better.
* **Feature Engineering**: Both ‘Lat’ and ‘Long\_’ columns had weak correlations so creating ‘Lat\_Long\_Interaction’ to get a stronger correlation between the values to be predicted.
* **Scaling Features**: Used StandardScaler function to scale the features for normalization, to make sure that all features are scaled equally, and no feature is dominant for better model performance.
* **Dataset Split:** Splitting dataset into training and testing (80% training and 20% validation) model is trained on training data and its performance is evaluated on validation set.

**Model Selection: Random Forest Regressor**

The Random Forest Regressor is a popular and robust machine learning model, particularly for tabular data. Here’s why it was chosen:

How Random Forest Works:

* Random Forest is an ensemble learning method that builds multiple decision trees during training.
* Each tree is trained on a random subset of the data and uses a random subset of features for splitting.
* Predictions are made by averaging the outputs of all the trees (regression) or majority voting (classification).

Advantages:

1. Handles Nonlinearity: It captures complex relationships between features and the target variable.
2. Reduces Overfitting: Combining multiple trees reduces the risk of overfitting, as individual trees are trained on different subsets of data.
3. Works Well with Skewed Data: Can handle skewed or unbalanced data well.
4. Feature Importance: Provides insights into feature importance, helping understand the data better.

**Hyperparameter Tuning:**

To maximize the performance of the Random Forest model, Grid Search was used to tune its hyperparameters. Key parameters include:

* n\_estimators: Number of decision trees in the forest.
* max\_depth: Maximum depth of each tree (controls overfitting).

Grid Search systematically evaluates combinations of these parameters to identify the configuration that minimizes the evaluation metric (RMSE).

**Model Evaluation:**

RMSE is the square root of the average of the squared differences between predicted and actual values

**RMSE for Testing data on Deaths**: 0.819

**RMSE for Testing data on CFR**: 0.223

**Validation and Prediction:**

1. Preprocessed the validation dataset similarly to the training dataset:
   * Created ‘Lat\_Long\_Interaction’ feature for better correlations
   * Scaled the features using the StandardScaler fitted on the training data.
2. Predicted the ‘Deaths’ and ‘CFR’ using the RandomForestRegressor with the best parameters for both
3. Predicted values for ‘Deaths’ and ‘CFR’ were brought back to original scale(from logarithmic scale) using the exponential function.
4. Confirmed Cases column was calculated using the formula Confirmed cases = (Deaths/CFR)\*100
5. Saved results in csv file containing ‘Deaths’, ‘Case\_Fatality\_Ratio’ and ‘Confirmed\_Cases”

**Key Insights and Observations**

Data had a lot of missing values from the column to be predicted, which reduced the size of the dataset significantly. The features in the dataset had a very weak correlation with the values to be predicted which prevented it from achieving a better loss. With additional features that were correlated with the values to be predicted the model could’ve achieved a much better RMSE score.