SWAMI KESHVANAND INSTITUTE OF TECHNOLOGY,
MANAGEMENT & GRAMOTHAN
Department of Computer Science and Engineering

Project
(Covid-19 Outbreak Prediction)

Submitted To:
Mr. Ankit Kumar
Assoc. Prof.
SKIT Jaipur

Submitted By:
Ankit Saini
17ESKCS027
CS-A (G1)

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

What is Covid-19?

Coronavirus disease (COVID-19) is an inflammation disease from a new virus. The disease causes respiratory ailment with manifestations, for example, cold, cough and fever, and in progressively serious cases, the problem in breathing. Our society is facing the huge negative impacts in the field of infrastructure, finance, business, manufacturing, and several other sectors due to COVID-19 outbreak.

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

Purpose Of Covid-19 Outbreak Prediction

- This project intends to apply the machine learning models simultaneously with the forecast of expected reachability of the Covid-19 over the different regions by using the real-time data to reduce the impact of this deadly virus minimum.
- This Project can help the authorities in estimating the requirement of resources as per different regions.
- This Project can help the authorities to make better plans for lockdown and curfew to handle the pandemic better.

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

Existing Problem

The problem with the current approaches that the governments around the world are using is that either they are very aggressive or not aggressive at all. Very aggressive approaches includes completely lockdown and unnecessarily storing the medical instruments such as ventilators which makes it hard to access for the poor countries and countries that do not have manufacturing capacity. other approach is to ignore the pandemic completely. It can be easily understood by the approaches used by America and India.

India's Approach

Government of india declared the complete lockdown at 25 march 2020 without a perfect plan which resulted in complete disaster for some section of society. All the transportation facilities has been put at complete stop. It made it difficult to return back to home for migrant workers and students. Some people faced difficulty for finding two times meal. Although there are claims by expert that this saved a lot of lives but sadly it also caused some lives.

America's Approach

Even after having a largest no of cases in the world the government of america did not declared the any kind of lockdown or impose restrictions of any kind for a really long time, in opposite to that when almost every country that had less impact of covid-19 was implementing strict lockdown they keep working fully functionally for economic reasons. It resulted in america having largest no of fatalities in the world.

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

Technology Stack

Technology used in the project are following :

- Python
- Numpy
- Pandas
- Scikit-Learn
- Matplotlib
- Plotly

Covid-19 Dataset

I used Covid19 Global Forecasting dataset that was created by John Hopkins CSSE which contains information of covid cases and fatality according to different regions for a particular day. Dataset is created for the purpose of predicting the covid-19 cases as well as fatality for a particular day.This dataset includes two different csv data files. First one is train.csv which is to be used for training by machine learning model. Second is test.csv which is to be used to calculate model's performance.

Data Analysis

This Step Involves.

- Organising the Data in suitable format.
- Finding Data Type of Features in Dataset.
- Finding most relevant features from the dataset.
- Performing Statistical Analysis.
- Text Analysis.

Data Cleaning

This Step Involves :

- Converting data to format which is used by Machine Learning Algorithm.
- Handling missing Values.
- Removing Duplicate Data.
- Removing Irrelevant features from the dataset.
- Creating new features .

Data Visualization

Used Matplotlib and Plotly for ploting following

- Bar Plot
- Pie Chart
- Line Chart
- Tree Map
- Histogram

Machine Learning Models

Used various different models using Decision Tree, Random Forest and XGBoost machine learning Algorithm, Such as:

- Decision Tree Regressor with 5 leaves.
- Random Forest Regressor with 10 estimators
- Random Forest Regressor with 100 estimators
- XGBoost Regressor with 100 estimators
- XGBoost Regressor with 400 estimators

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

Conclusion

After applying different Machine Learning Algorithms and creating different machine learning models we found out that Random Forest Regressor with 10 estimators gave best result among all models with RMSE score of 73 and R2 score of 0.95 .

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

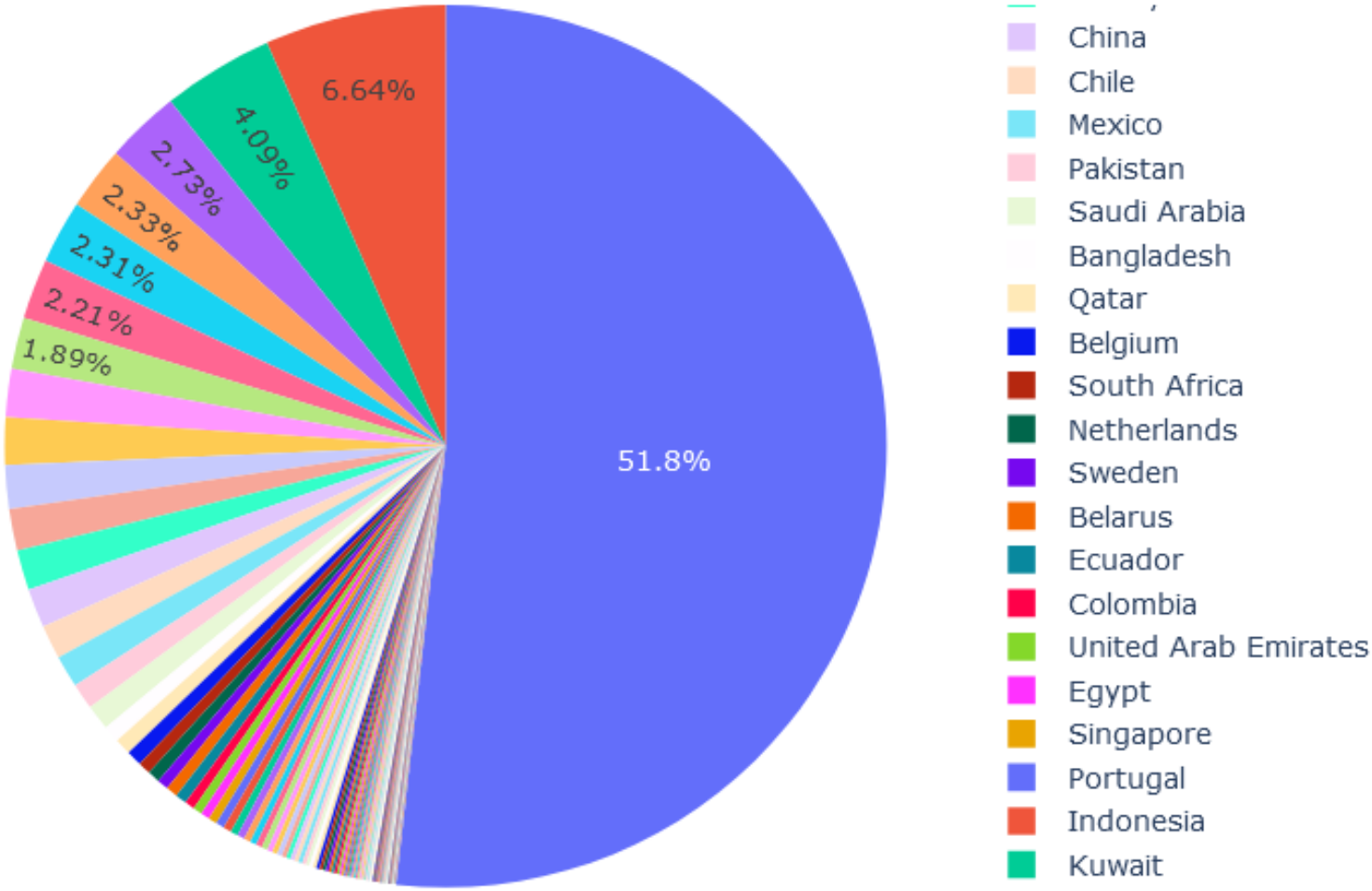
Importing Libraries

```
#visualiation data
import matplotlib.pyplot as plt
%matplotlib inline
import seaborn as sns
import matplotlib
import plotly.graph_objects as go
import plotly.express as px
import plotly.graph_objects as go
from plotly.offline import init_notebook_mode, iplot

#default theme
sns.set(context='notebook', style='darkgrid', palette='Spectral', font='sans-serif', font_scale=1, rc=None)
matplotlib.rcParams['figure.figsize'] =[8,8]
matplotlib.rcParams.update({'font.size': 15})
matplotlib.rcParams['font.family'] = 'sans-serif'

# dataprep library
from dataprep.eda import *
from dataprep.datasets import load_dataset
from dataprep.eda import create_report
```

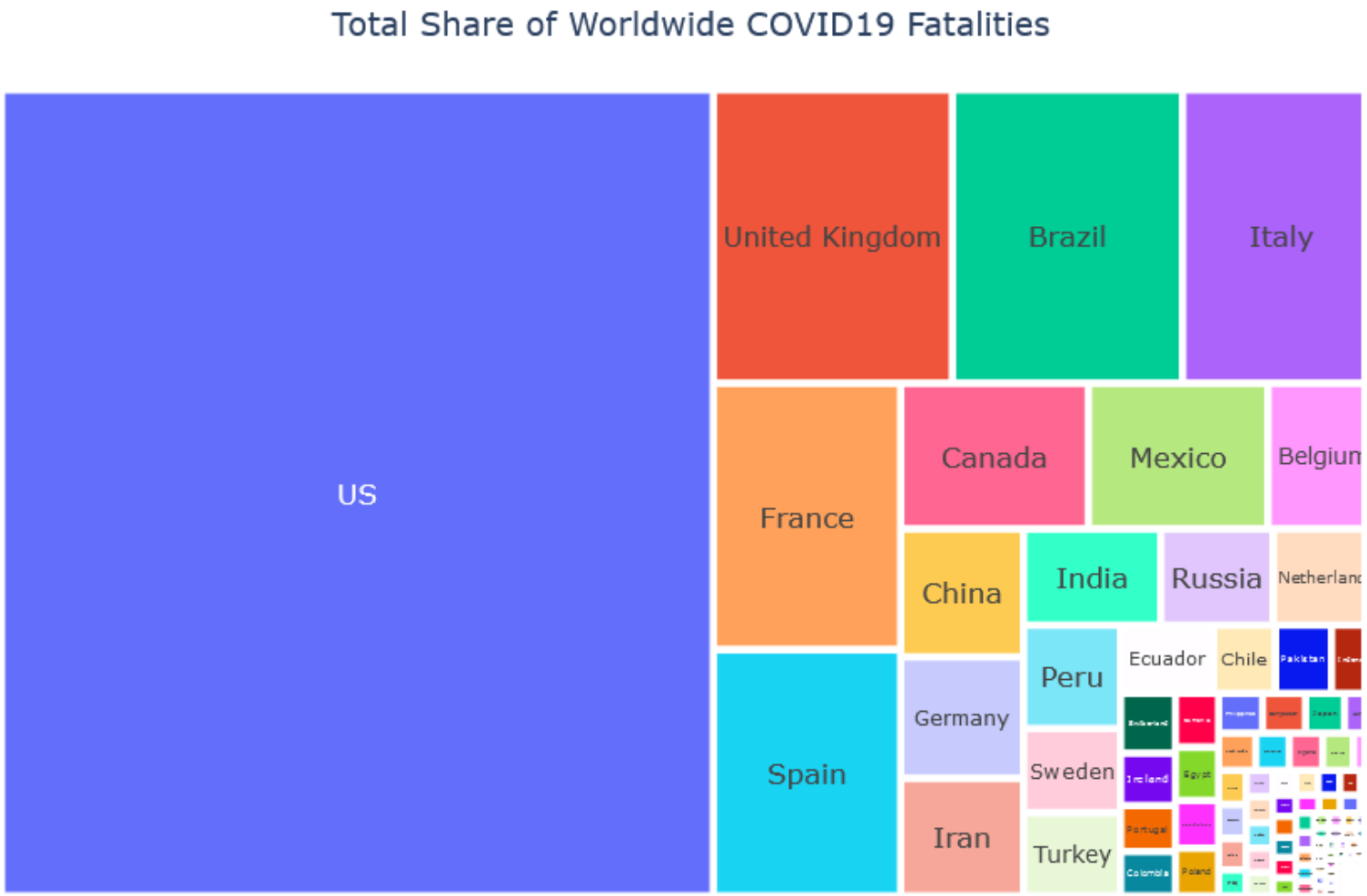
+ Code + Markdown

```
#machine learning Library
from sklearn.ensemble import RandomForestRegressor
from sklearn.model_selection import GridSearchCV, KFold
from sklearn import ensemble
from sklearn.preprocessing import OrdinalEncoder
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.pipeline import Pipeline
from sklearn import metrics
```

Pie chart


6.64%
4.09%
2.73%
2.33%
2.31%
2.21%
1.89%
51.8%
China
Chile
Mexico
Pakistan
Saudi Arabia
Bangladesh
Qatar
Belgium
South Africa
Netherlands
Sweden
Belarus
Ecuador
Colombia
United Arab Emirates
Egypt
Singapore
Portugal
Indonesia
Kuwait

Tree Map

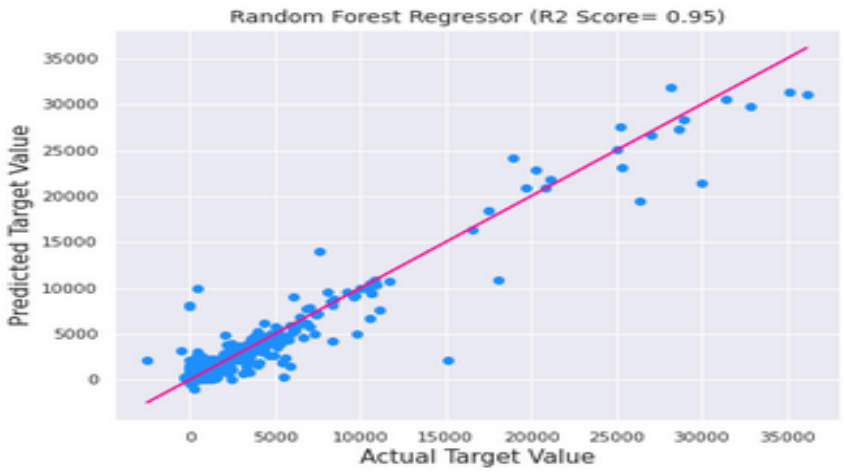

Total Share of Worldwide COVID19 Fatalities
US
United Kingdom
Brazil
Italy
France
Canada
Mexico
Belgium
China
India
Russia
Netherlands
Spain
Germany
Peru
Ecuador
Chile
Pakistan
Iran
Sweden
Turkey

Random Forest

```
model2 = RandomForestRegressor(n_jobs=-1)
estimators = 100
model2.set_params(n_estimators=estimators)

pipeline2 = Pipeline([('scaler2' , StandardScaler()),
                      ('RandomForestRegressor: ', model2)])
pipeline2.fit(X_train , y_train)
prediction = pipeline2.predict(X_test)
```

+ Code + Markdown

```
plt.figure(figsize=(8,6))
plt.plot(y_test,y_test,color='deeppink')
plt.scatter(y_test,prediction,color='dodgerblue')
plt.xlabel('Actual Target Value',fontsize=15)
plt.ylabel('Predicted Target Value',fontsize=15)
plt.title('Random Forest Regressor (R2 Score= 0.95)',fontsize=14)
plt.show()
```




```
print('RMSE of model2 =', np.sqrt(metrics.mean_squared_error(y_test,prediction)))
print('R2 Score of model2 = ',metrics.r2_score(y_test,prediction))
```

```
RMSE of model2 = 74.06029329294243
R2 Score of model2 =  0.9411116054253269
```

```
import xgboost as xgb
```

```
xgbr= xgb.XGBRegressor(n_estimators=300, learning_rate=0.01, gamma=0, subsample=.7,
                       colsample_bytree=.7, max_depth=10,
                       min_child_weight=0,
                       objective='reg:squarederror', nthread=-1, scale_pos_weight=1,
```

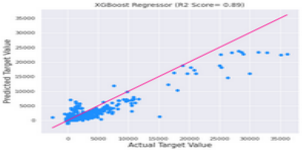
```
xgbr.fit(X_train, y_train)
```

```
XGBRegressor(base_score=0.5, booster='gbtree', colsample_bylevel=1,
             colsample_bynode=1, colsample_bytree=0.7, gamma=0, gpu_id=-1,
             importance_type='gain', interaction_constraints='',
             learning_rate=0.01, max_delta_step=0, max_depth=10,
             min_child_weight=0, missing=nan, monotone_constraints='()',
             n_estimators=300, n_jobs=-1, nthread=-1, num_parallel_tree=1,
             random_state=27, reg_alpha=6e-05, reg_lambda=1, scale_pos_weight=1,
             seed=27, subsample=0.7, tree_method='exact', validate_parameters=1,
             verbosity=None)
```

+ Code + Markdown

```
prediction_xgbr=xgbr.predict(X_test)
```

```
plt.figure(figsize=(8,6))
plt.scatter(x=y_test, y=prediction_xgbr, color='dodgerblue')
plt.plot(y_test,y_test, color='deeppink')
plt.xlabel('Actual Target Value',fontsize=15)
plt.ylabel('Predicted Target Value',fontsize=15)
plt.title('XGBoost Regressor (R2 Score= 0.89)',fontsize=14)
```




```
print('RMSE_XGBoost Regression=', np.sqrt(metrics.mean_squared_error(y_test,prediction_xgbr)))
```

```
RMSE_XGBoost Regression= 115.77534027448156
R2 Score_XGBoost Regression= 0.8568859835345886
```

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

Future Scope

- Making better decision and plan to handle pandemic.
- Making better decision and plan to utilize Resources.
- Gives different result based on different regions.
- Output based on real time input data.
- Can be upgraded for future use of handling pandemic.

Table of Contents

1. Introduction
2. Purpose Of the Project
3. Existing Problem
4. Proposed Solution
   - Technology Stack
   - Dataset
   - Data Analysis
   - Data Cleaning
   - Data Visualization
   - Machine Learning Model
5. Results
6. Project Screenshots
7. Future Scope
8. References

References

- https://github.com/CSSEGISandData/COVID-19/tree/master/csse_covid_19_data/csse_covid_19_time_series
- https://github.com/CSSEGISandData/COVID-19
- https://devdocs.io/scikit_learn/
- https://matplotlib.org/stable/contents.html
- https://en.wikipedia.org/wiki/COVID-19_pandemic_lockdown_in_India

Thank You