Machine Learning for Sustainable Development Goal 1: No Poverty

1. Introduction

Project Objective:

To use machine learning to address challenges in No Poverty, aiming to support SDG 1 by predicting the which country need the financial aid around the world using some socioeconomic and health factors.

Problem :

KMU NEED International have been able to raise around $ 20 million. Now the CEO of the NGO needs to decide how to use this money strategically and effectively. So, CEO has to make decision to choose the countries that are in the direst need of aid. Hence, your job as a Data Scientist is to find the countries using some socioeconomic and health factors that determine the overall development of the country. Then you need to predict the countries which the CEO needs to focus on the most.

2. Data Collection

Data Source : Kaggle Dataset(eg., "Country_Data Dataset")

Dataset Description :

Features :

Dataset Attributes	Description
country :	Name of the country
child_mort :	Death of children under 5 years of age per 100 live birth
exports :	Exports of goods and services per capita. Given as % age of the GDP per capita
health :	Total health spending per capita. Given as % age of GDP per capita
imports :	Imports of goods and services per capita. Given as % age of the GDP per capita
income :	Net income per person
inflation :	The measurement of the annual growth rate of the Total GDP
life_expec :	The average number of years a new born child would live if the current mortality patterns are to remain the same
total_fer :	The number of children that would be born to each woman if the current age-fertility rates remain the same
gdpp :	The GDP per capita. Calculated as the Total GDP divided by the total population

Size : 167 rows by 10 columns
Target Variable : Initially it is unsupervised learning problem but we apply feature engineering to get target variable (needed_aid) {binary }

3. Exploratory Data Analysis (EDA)

Summary Statistics : Mean, std, min, max and quartile of the data by using the describe() for float and int type and for categorical we see the description of the data by using the describe()

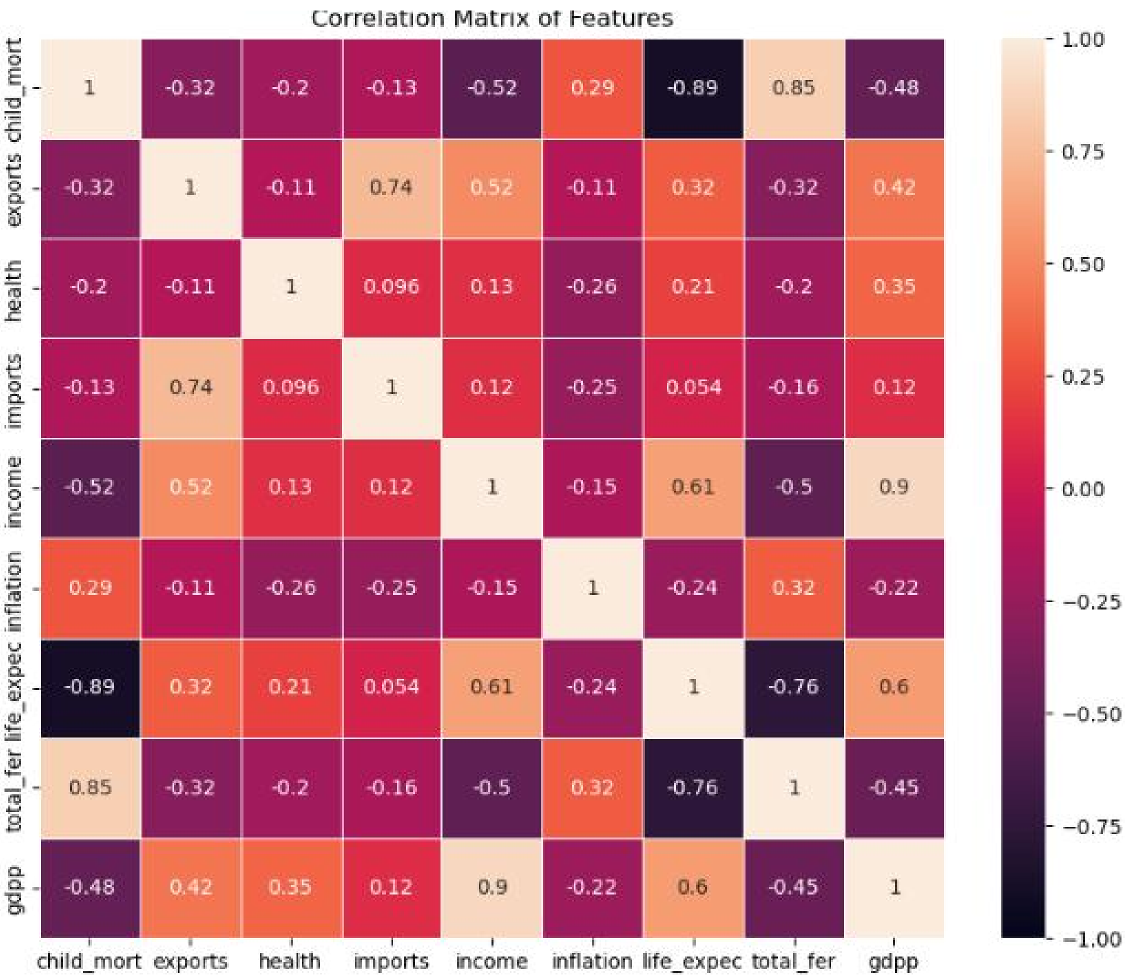
Visualization :

Histogram : for all numerical variable in the data set we see the distribution of each variable.

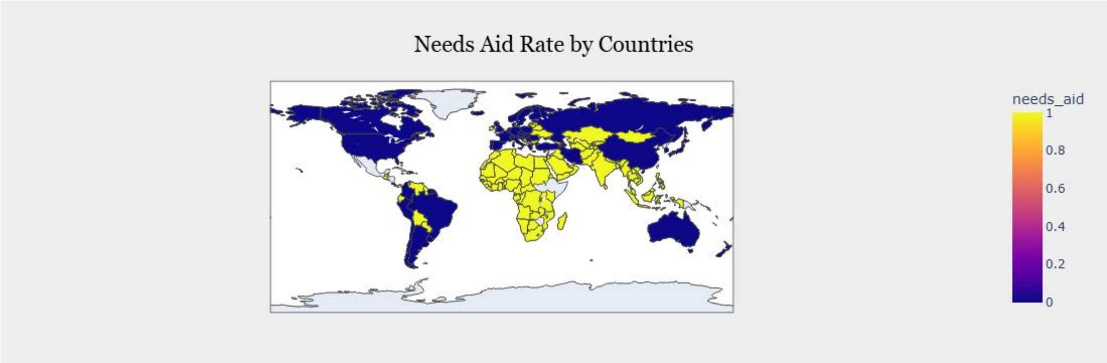
Box plot : Along with box plot we plot the kde for better visualization kde provides a smoothed representation of the underlying distribution of the data set and box plot for outliner detection.

Scatter Plot : plot all the numerical variable along the gdpp to visualize the relationship between the all variable v/s gdpp and us the box plot to see relation between all variable v/s gdpp.

Correlation matrix : To understand the relationship between all variable.



Choropleth plot : seeing the all column in geographical way we plot two choropleth plot
-> which show the choropleth and
-> without the choropleth
This is conclusion plot to show which country need aid this plot after the model implementation



Insights : by analyzing the distribution we know the details of variable after ploting the choropleth we see that some country gdpp is good but not other factor are good

4. Data Prepossessing

Handling Missing Value : There is no missing value in the data we verify by heatmap

Feature Engineering :

Trade_balance : The difference between the value of a country's exports and imports for a given period.

Trade balance = exports - imports

Where (A positive balance indicates an economy with more exports than imports, which may indicate strong trade stability.)

Health_risk :

Health risk is the chance or likelihood that something will harm or otherwise affect the health. It is not a guarantee that something will bad happen, but rather a possibility.

Health risk = child mort / Life expec

Where higher child mortality and lower life expectancy increase health risk.

Economic_stability :

Economic stability is the condition where an economy is free from various factors disrupting its smooth functioning and growth.

Economic stability = gdpp / inflation

Where higher gdp per capita and lower inflation increase stability.

Social_stability :

Social stability is the degree to which a society and its institutions remain predictable and reliable.

Trade balance = exports - imports

Where higher life expectancy and moderate fertility rates indicate stringer social stability.

5. Machine learning Model Selection

Model Choices:

Decision Tree Classifier : (a flow-chart like structure used to make decisions or predictions)
Random Forest Classifier : (for binary classification)
Support Vector Machine : (for best hyper plane)

Why Scikit-Learn: Easy implementation, variety of algorithms, and effective performance metrics.

Evaluation Metric: Accuracy, Precision, Recall, and F1-Score due to the critical nature of accurately identifying contamination.

6. Model Implementation

Data Splitting :
Split data set into 80% training and 20% testing sets using train_test_split from from Scikit-learn.

Hyper parameter Tuning:
Used GridSearchCV for Random Forest to identify optimal number of estimators and max depth.
Cross-validation with 5 folds to improve model generalization.

Code

```
# Declare the dependent and independent variable

x = df.drop(columns=['country', 'needs_aid','gdpp_quartile'])
y = df['needs_aid']

# Split the data for training and testing

from sklearn.model_selection import train_test_split, GridSearchCV

x_train, x_test, y_train, y_test = train_test_split(x, y, test_size=0.2, random_state=42)

from sklearn.ensemble import RandomForestClassifier

# Hyper parameter tuning for Random Forest

param_grid = {
   'n_estimators': [50, 100, 200],
   'max_depth': [10, 20, 30]
}

# Initialize the Random Forest Classifier and fit the model

from sklearn.ensemble import RandomForestClassifier

rf = RandomForestClassifier(random_state=42)
grid_search = GridSearchCV(estimator=rf, param_grid=param_grid, cv=5, scoring='f1')
grid_search.fit(x_train, y_train)

# Best model and evaluation

best_model = grid_search.best_estimator_
y_pred = best_model.predict(x_test)

print("Accuracy:", accuracy_score(y_test, y_pred))

print("Confusion Matrix:\n", confusion_matrix(y_test, y_pred))

print("Classification Report:\n", classification_report(y_test, y_pred))
```

7. Results and Evaluation

Model Performance :
Random Forest Achieved an accuracy of 0.9411764705882352 which is 94%, F1 score of 94%
And precision / recall value are same indicating the model strength in predicting

Feature Importance :
needed aid is the feature which is created by using all the feature and setting 3 as threshold limit to create needed aid variable

Confusion Matrix : Visualized true vs. predicted values to identify common misclassifications.

Classification report : show the precision, recall, f1-score, accuracy, macro avg and weighted avg

8. Conclusion and Future Work

Key Takeaways: Machine learning models effectively predict which country need aid by using socioeconomic and health factors. The project demonstrates potential for real-time fund to needed country allocation.

Future Improvements:

Adding additional data such as environment effect means (some sudden cause to change the economy)

Expanding to a broader data set covering multiple regions data.

9. References

Kaggle Dataset
Scikit-Learn Documentation