

A. RESEARCH QUESTION

In recent years, there has been a substantial improvement in macroeconomic management across Africa, as affirmed by studies conducted by reputable organizations such as the International Monetary Fund (IMF), the African Development Bank (AfDB), and the World Bank. Additionally, surveys conducted by the Worldwide Governance Indicators and Transparency International reinforce this positive trend. However, amidst these positive developments, concerns regarding debt sustainability on the African continent have grown, particularly in the wake of the COVID-19 pandemic.

The origins of these concerns are not new. In the 1980s and 1990s, African nations accumulated debt primarily through borrowing from official creditors, including development banks, OECD export credit agencies, and Paris Club lenders (comprising major creditor countries). This set them apart from Latin American nations, which heavily relied on private lenders for borrowing. Nevertheless, rising apprehensions about debt sustainability led to a wave of debt-relief programs during the late 1990s and the 2000s.

Within the span of a decade, the average debt ratio in sub-Saharan Africa nearly doubled, surging from 30 percent of GDP at the end of 2013 to almost 60 percent of GDP by the close of 2022. Repaying this debt has also become substantially more expensive. The region's ratio of interest payments to revenue, a critical metric for assessing debt servicing capacity and forecasting fiscal crisis risks, has more than doubled since the early 2010s, now approaching four times the ratio in advanced economies. As of 2022, the IMF assessed that over half of the low-income countries in sub-Saharan Africa were at high risk of debt distress or already experiencing it (Comelli et al, 2023).

These worrisome trends have raised alarms about an impending debt crisis in the region. In response, a recent IMF paper presents potential solutions to avert such a crisis. It outlines five policy actions that African governments can undertake to ensure the sustainability of public finances while striving to achieve the region's development objectives.

The year 1996 marked the launch of the Heavily Indebted Poor Countries (HIPC) Initiative by the IMF and the World Bank, followed by the Enhanced HIPC Initiative three years later. These initiatives represented significant innovations in development finance, enabling the cancellation of debts owed to multilateral creditors. Subsequently, the Multilateral Debt Relief Initiative in 2006 and the rescheduling of sovereign debts through the Paris Club instilled optimism regarding the future of Africa's debt burdens. Official creditors forgave over \$100 billion in debt to more than 40 countries, with approximately 85% of them located in Africa (Monga, 2023).

However, since 2010, public debts have surged alongside a notable shift in the composition of the debt stock. An increasing share of these debts is now owed to emerging economies, particularly China, and to private creditors through the issuance of Eurobonds. As of April 2020, the IMF classified seven African nations as being in a state of debt distress while identifying an additional twelve at high risk of falling into this category.

Starting from 2011, a significant 75% of Sub-Saharan African countries' debt servicing obligations have been directed towards bondholders and commercial lenders, often accounting for substantial portions of their national revenues. Presently, external debts constitute up to 80% of the Gross National Income in several African countries. Even countries currently deemed to be at low risk of debt unsustainability possess external debts reaching as high as 41% of their national income and allocate significant portions of their budgets to servicing these debts, as indicated by 2019 research from Databank. For instance, Angola allocated around 43% of its total revenue to debt servicing,

while Ghana allocated 39%. Concerns regarding debt unsustainability are growing, particularly in light of the limited domestic resource mobilization and narrow tax bases across African nations (Sokpoh et al., 2022).

In summary, the trajectory of African nations' debt profiles reveals a pressing need for proactive measures to ensure long-term economic stability. While efforts have been made to alleviate debt burdens through initiatives and relief programs, the changing landscape of debt composition and the potential for debt distress underscore the importance of prudent fiscal management. Addressing these challenges requires a delicate balance between sustaining economic development and managing debt obligations effectively.

This study aims to contribute to the field of Data Analytics and the MSDA program by constructing a predictive model capable of accurately categorizing African countries' debt sustainability. By utilizing a multivariate statistical analysis approach, the model will evaluate the significance of various predictor variables and their impact on debt distress levels. The project's outcome is to enhance stakeholders' understanding of the current debt situation of these countries and make informed decisions on lending and grant allocation. This study will contribute to the understanding of debt vulnerabilities in Sub-Saharan African countries that have been granted debt reductions under the HIPC (Highly Indebted Poor Country) scheme.

My prime objective in this study is to answer the question “Can an accurate predictive model be developed to classify African countries into distinct debt sustainability categories based on external debt distress indicators and other pertinent features?” The primary objective of this study is to construct an accurate predictive model to classify African countries into distinct debt sustainability categories. To achieve this, Support Vector Machine (SVM) will be employed as the classification method. The study aims to achieve a classification accuracy of at least 80% in distinguishing between different debt sustainability categories.

In addition to classification accuracy, the study will also assess the significance of various predictor variables in determining debt distress levels. By analysing the impact of these features, the model will contribute to stakeholders' understanding of the factors influencing debt vulnerabilities in Sub-Saharan African countries. The study's outcomes will inform data-driven decisions regarding lending and grant allocation to these nations.

Below is a constructed hypothesis designed to assist in addressing the research question:

Hypothesis

Null Hypothesis: The predictive model cannot accurately classify African countries into different debt sustainability categories with an accuracy of at least 80%.

Alternate Hypothesis: The predictive model can accurately classify African countries into different debt sustainability categories with an accuracy of at least 80%.

B. DATA COLLECTION

Data collection can be defined as the collection of the various techniques, methods, and procedures involved in gathering data mainly for research and development purposes. In this project data was gathered using the secondary data collection method. Secondary data is previously acquired data which has been collected for a prior purpose or objective but is at the moment significant to your present research needs. This means, this data has already been gathered in the past by someone else, making it second hand information which is not being used for the first time, that is why it is referred to as secondary (Valcheva, n.d). Secondary data can be acquired from a variety of sources such as libraries, the web, or reports.

The data utilized for this project was sourced from Kaggle, a widely recognized platform that offers access to a diverse array of datasets. Kaggle serves as an online hub where a vast collection of over 100,000 datasets is made available to the public, encompassing contributions from various sources including individuals, organizations, and data enthusiasts. These datasets cover an extensive range of fields and are freely accessible to support the research endeavours of organizations, researchers, and students alike, providing a valuable resource for data exploration and analysis.

The data set for this research is the "[Evolution of Debt Vulnerability Classifications in Sub-Saharan African Heavily Indebted Poor Countries](#)." This data set contains 15 years of historical data on 29 Sub-Saharan African countries that received debt relief under the HIPC initiative (Kaggle, 2021). The dataset includes 7,500 records with 22 attributes encompassing debt indicators, macroeconomic variables, and governance scores. This publicly available data from Kaggle (2021) provides a comprehensive view of debt sustainability issues in developing countries and is well-suited for predictive modeling.

Specifically, the debt indicators such as external debt service ratios, risk of debt distress, and current account balance enable examining debt vulnerabilities and distress levels. The macroeconomic variables like GDP, inflation, government budgets provide economic context. The World Bank governance scores offer additional insights into contributing factors like corruption, rule of law, and regulatory quality (World Bank, 2022).

This multivariate data allows building predictive models using machine learning techniques to classify countries by debt sustainability categories based on recommended practices (World Bank, 2022). The variety of economic, debt, and governance attributes in this dataset can help identify significant predictors of debt distress in Africa. The data set's attributes, their meanings, data types and their descriptions are provided in the table below:

Attribute	Meaning	Description	Data Type
ISO	Country Code (ISO)	Country code representing a country using three letters	string
Year	Year	Year of data classification	float
Risk.ext.debt.distress	Risk of External Debt Distress	Classification of debt risk for low-income countries	string
Debt.Indicator	Debt Indicator	Indicator for high or low risk of external debt distress (0 for low risk, 1 for high risk)	int

Inflation	Inflation	Annual percentage change of average consumer prices	float
Cur.acc.bal	Current Account Balance	Percentage of GDP representing a country's balance of exports and imports	float
Gen.gov.len.bor	General Government Net Lending/Borrowing	Percentage of GDP showing government's financial position (surplus or deficit)	float
Vol.Exp.Goods	Volume of Exports in Goods and Services	Percentage change in volume of exports in goods and services	float
GDP	GDP (Current US\$)	Total economic output of a country in current US dollars	float
GDP.per.cap	GDP per Capita (US\$)	GDP per person in US dollars	float
Gen.gov.rev	General Government Revenue	Government revenue as a percentage of GDP	float
US.int.rates	Real Interest Rates (%) - United States	Real interest rates in the United States	float
Ext.Debt.Serv	External Debt Service	Total debt service on external debt in current US dollars	float
Real.GDP.growth	Real GDP Growth (%)	Annual percentage change in GDP	float
Exch.Rate	Official Exchange Rate (LCU per US\$, period average)	Official exchange rate of a country's currency per US dollar	float
Control.of.Corrup tion	Control of Corruption	Score indicating the level of control of corruption in a country	float
Government.Effec tiveness	Government Effectiveness	Score indicating the effectiveness of the government in a country	float
Pol.Stability.Abse nce.of.Violence	Political Stability and Absence of Violence	Score indicating the level of political stability and absence of violence in a country	float
Regulatory.Quality	Regulatory Quality	Score indicating the quality of regulations and institutions in a country	float
Rule.of.Law	Rule of Law	Score indicating the level of Rule of Law	float
Voice.and.Account ability	Voice and Accountability	Score indicating Voice and Accountability	float

In the overall data collection process, the benefits of acquiring secondary data for this project are:

- Ease of access: The secondary data sources are very easily accessible, especially by anyone. With the introduction and evolution of the internet, secondary data has been made much more readily available just by the click of a mouse.
- Timesaving: Secondary research sometimes is just a matter of a few Google searches in locating a source of data, meaning you can have access to your desired data in no time.
- Low cost or free: Most secondary sources are completely free for use or are sold at very low costs. It ensures that not only your money is saved but your efforts too. In contrast with primary research where you must design and conduct a whole primary study process from the beginning, secondary research allows you to gather data without having to put any money on the table.

On the other hand, some challenges of acquiring secondary data for this project are:

- Data might be difficult to comprehend: In the acquired data set, some columns of data were difficult to understand just by reading them.
- The data might not specifically suit desired needs: Since secondary data was already collected in the past for a particular purpose, it might not totally suit your current objectives. In this project, the data collected was required to undergo regression analysis and some columns of data were irrelevant to this cause.

To overcome the above challenges, I first had to follow the stated sources in the data summary file and study the data dictionary to have a better understanding of the data. I also had to perform various data pre-processing techniques to manipulate the data in order for it to suit my objective of performing a regression analysis.

C. DATA EXTRACTION AND PREPARATION

In this research, I conducted a comprehensive series of exploratory data analytical processes on the dataset, which encompassed data extraction, data wrangling, data visualization, and predictive analysis through the utilization of a machine learning model. All these operations were executed within the Jupyter Notebook environment. Jupyter Notebook, an open-source, free, and interactive web tool, was chosen for its capability to seamlessly integrate software code, explanatory text, computational output, and multimedia resources into a single document (Perkel, 2018). This made it an ideal choice for researchers seeking to combine code, rich text, and visualizations within a single presentable format. Python, a high-level, interpreted, and object-oriented programming language, was selected as the programming language for this analysis. Python was selected as the programming language for this project over SAS as it offers superior graphics capabilities (Jain, 2017). It was also selected over a competent statistical language like R due to its scalability, Jupyter Notebooks, library packages, integrations, cross-functionality (Przybala, 2020). It is also advantageous due to its high-level nature, wide range of accessible libraries, and flexibility, making it suitable for projects of various sizes and complexities.

In this analysis, we relied on a range of Python libraries to facilitate our data extraction and preparation tasks. Each library played a crucial role in streamlining the analysis:

- **Warnings:** The "Warnings" library was imported to suppress any warnings that may appear during the analysis, which enhances the overall readability.
- **Pandas:** Pandas, a fundamental library in data analytics, was utilized to efficiently manage data processing, manipulation, and preparation tasks.
- **Numpy:** Numpy, an essential Python library, provides a robust n-dimensional array data structure that forms the foundation for much of Python's data science capabilities (Palo, n.d).
- **ScikitLearn:** ScikitLearn, a machine learning library, played a pivotal role in this analysis by providing models for multiple regression analysis and facilitating model performance evaluation.
- **Matplotlib and Seaborn:** These two distinct Python libraries were instrumental in enhancing data comprehension by generating insightful statistical graphs, maps, and charts.

An image of the library import process is provided in the screenshot below:



```
In [1]: # Import Libraries

#Data Preparation
import pandas as pd
import numpy as np
import statsmodels.api as sm

#Data Visualization
%matplotlib inline
import matplotlib.pyplot as plt
import seaborn as sns

#Machine Learning model and model analysis
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder, StandardScaler
from sklearn.linear_model import LogisticRegression
from sklearn.svm import SVC
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score, classification_report, confusion_matrix, roc_curve, roc_auc_score

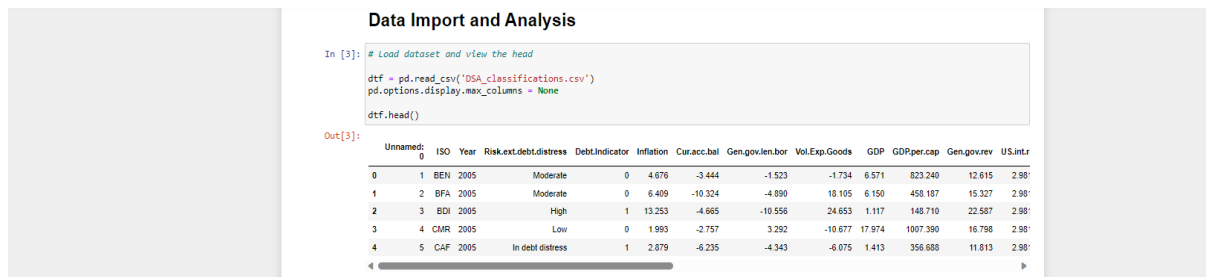
#Ignore warnings
import warnings

In [2]: warnings.filterwarnings('ignore')
```

In the data analysis process, it is essential to begin with data extraction and preparation to ensure that the dataset is ready for exploration and modeling. This section outlines the steps taken in this crucial phase of the analysis.

1. Imported the Data Set

The data set, named 'DSA_classifications.csv,' was loaded into a Pandas DataFrame using the `pd.read_csv()` method. The initial five records of the DataFrame were displayed using the `.head()` method. A screenshot of this process is provided in the image below:



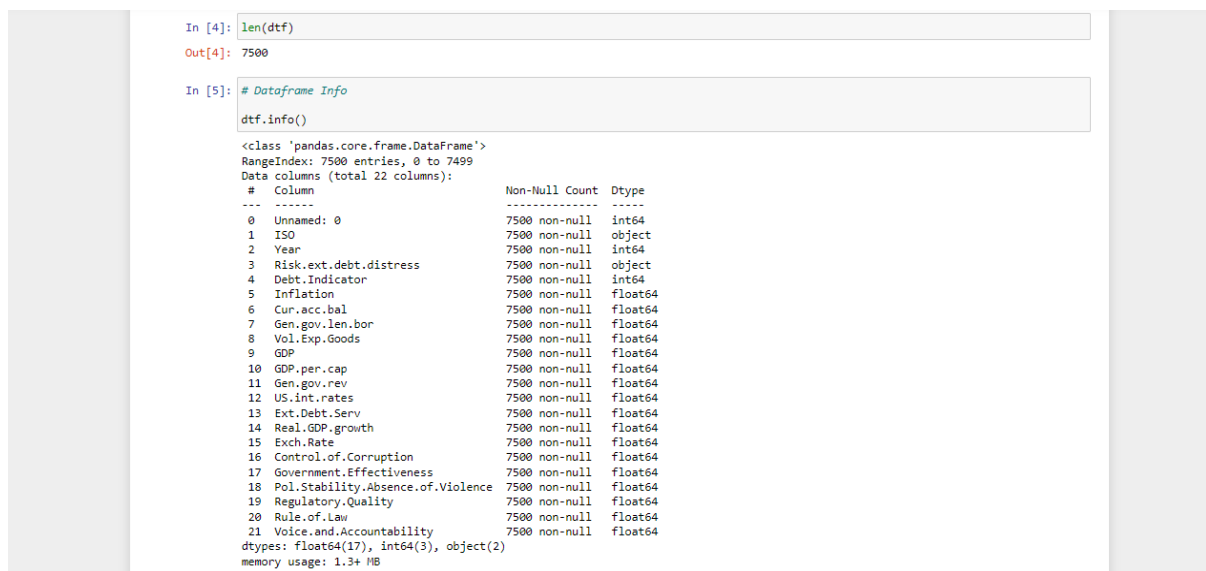
```
In [3]: # Load dataset and view the head
dtf = pd.read_csv('DSA_classifications.csv')
pd.options.display.max_columns = None
dtf.head()
```

Out[3]:

	Unnamed: 0	ISO	Year	Risk.ext.debt.distress	DebtIndicator	Inflation	Cur.acc.bal	Gen.gov.len.bor	Vol.Exp.Goods	GDP	GDPper.cap	Gen.gov.rev	US.int.r
0	1	BEN	2005	Moderate	0	4.676	-3.444	-1.523	-1.734	6.571	823.340	12.615	2.981
1	2	BFA	2005	Moderate	0	6.409	-10.324	-4.090	10.105	6.150	458.107	15.327	2.981
2	3	BDI	2005	High	1	13.253	-4.665	-10.556	24.653	1.117	148.710	22.587	2.981
3	4	CMR	2005	Low	0	1.993	-2.757	3.292	-10.677	17.974	1007.390	16.796	2.981
4	5	CAF	2005	In debt distress	1	2.079	-6.235	-4.343	-6.075	1.413	356.688	11.813	2.981

2. Checked Dataset's Length and Information

To gain an understanding of the dataset's size and data types, the `.info()` method was employed. This provided details on column names, data types, and non-null counts. The dataset contains 22 columns and 7,500 records. The screenshot below illustrates this process:



```
In [4]: len(dtf)
Out[4]: 7500

In [5]: # Datframe Info
dtf.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7500 entries, 0 to 7499
Data columns (total 22 columns):
#   Column                                     Non-Null Count  Dtype
---  ---                                     -
0   Unnamed: 0                               7500 non-null   int64
1   ISO                                       7500 non-null   object
2   Year                                     7500 non-null   int64
3   Risk.ext.debt.distress                   7500 non-null   object
4   Debt.Indicator                           7500 non-null   int64
5   Inflation                               7500 non-null   float64
6   Cur.acc.bal                             7500 non-null   float64
7   Gen.gov.len.bor                         7500 non-null   float64
8   Vol.Exp.Goods                           7500 non-null   float64
9   GDP                                      7500 non-null   float64
10  GDP.per.cap                             7500 non-null   float64
11  Gen.gov.rev                             7500 non-null   float64
12  US.int.rates                            7500 non-null   float64
13  Ext.Debt.Serv                           7500 non-null   float64
14  Real.GDP.growth                         7500 non-null   float64
15  Exch.Rate                              7500 non-null   float64
16  Control.of.Corruption                   7500 non-null   float64
17  Government.Effectiveness                 7500 non-null   float64
18  Pol.Stability.Absence.of.Violence       7500 non-null   float64
19  Regulatory.Quality                      7500 non-null   float64
20  Rule.of.Law                            7500 non-null   float64
21  Voice.and.Accountability                 7500 non-null   float64
dtypes: float64(17), int64(3), object(2)
memory usage: 1.3+ MB
```

3. Checked Summary Statistics

To comprehend the distribution of numerical variables within the dataset, summary statistics were computed using the `.describe()` method. This included metrics such as mean, minimum, and maximum values. This process is shown in the image below:

```
In [6]: # Dataframe Summary Statistics of numerical variables
```

```
dtf.describe()
```

```
Out[6]:
```

	Unnamed: 0	Year	Debt.Indicator	Inflation	Cur.acc.bal	Gen.gov.len.bor	Vol.Exp.Goods	GDP	GDP.per.cap	Gen.gov.rev	US.int.rates
count	7500.000000	7500.000000	7500.000000	7500.000000	7500.000000	7500.000000	7500.000000	7500.000000	7500.000000	7500.000000	7500.000000
mean	3750.500000	2011.976800	0.330800	6.875560	-7.827407	-1.775109	5.833862	14.665868	910.645810	19.070855	2.548998
std	2165.207842	4.296672	0.470533	7.058395	9.585286	8.020341	15.122724	15.656414	646.614634	10.322607	1.145762
min	1.000000	2005.000000	0.000000	-7.440000	-65.031000	-19.257000	-31.920000	0.125000	148.710000	7.464000	1.137338
25%	1875.750000	2008.000000	0.000000	1.784000	-10.425000	-4.204000	-3.547000	3.163000	492.078000	14.020000	1.469299
50%	3750.500000	2012.000000	0.000000	5.433000	-6.602000	-2.522500	4.734000	10.118000	722.924000	16.320000	2.409470
75%	5625.250000	2016.000000	1.000000	9.378000	-3.451000	-0.972000	12.847000	18.966000	1145.880000	21.562000	3.082411
max	7500.000000	2019.000000	1.000000	46.101000	84.849000	125.135000	114.993000	92.796000	4607.390000	164.054000	5.223406

4. Checked for Missing Values

Identifying missing values is crucial for data quality. The presence of null values in each column was determined using the `.isnull().sum()` method, revealing that there were no missing values in any column. A screenshot of this process is provided in the image below:

```
In [7]: # Check for missing values
```

```
missing_values = dtf.isnull().sum()  
print("Missing Values:\n", missing_values)
```

```
Missing Values:  
Unnamed: 0          0  
ISO                 0  
Year                0  
Risk.ext.debt.distress  0  
Debt.Indicator       0  
Inflation            0  
Cur.acc.bal         0  
Gen.gov.len.bor      0  
Vol.Exp.Goods        0  
GDP                  0  
GDP.per.cap          0  
Gen.gov.rev          0  
US.int.rates         0  
Ext.Debt.Serv        0  
Real.GDP.growth      0  
Exch.Rate            0  
Control.of.Corruption  0  
Government.Effectiveness  0  
Pol.Stability.Absence.of.Violence  0  
Regulatory.Quality    0  
Rule.of.Law          0  
Voice.and.Accountability  0  
dtype: int64
```

5. Checked for Duplicate Rows

Detecting and handling duplicate rows is essential to ensure data integrity. It was determined that there were no duplicate rows in the dataset after using the `.duplicated().any()` method. A screenshot of this process is provided in the image below:

```
In [8]: # Check for duplicate rows
```

```
duplicate_rows = dtf.duplicated().any()  
print("Duplicate Rows:\n", duplicate_rows)  
  
dtf[dtf.duplicated() == True]
```

```
Duplicate Rows:  
False
```

```
Out[8]:
```

	Unnamed: 0	ISO	Year	Risk.ext.debt.distress	Debt.Indicator	Inflation	Cur.acc.bal	Gen.gov.len.bor	Vol.Exp.Goods	GDP	GDP.per.cap	Gen.gov.rev	US.int.rates
--	------------	-----	------	------------------------	----------------	-----------	-------------	-----------------	---------------	-----	-------------	-------------	--------------

The chosen tools and techniques were well-suited for data extraction and preparation in this analysis. Jupyter Notebooks allowed for an interactive and organized approach to presenting code and results. Python, with its extensive libraries, provided the flexibility to efficiently perform data manipulation and analysis tasks.

An advantage of using these tools and techniques is the availability of a wide range of methods and functions, making it convenient to execute various data preparation processes effectively. Additionally, the Python libraries employed offer seamless integration and compatibility.

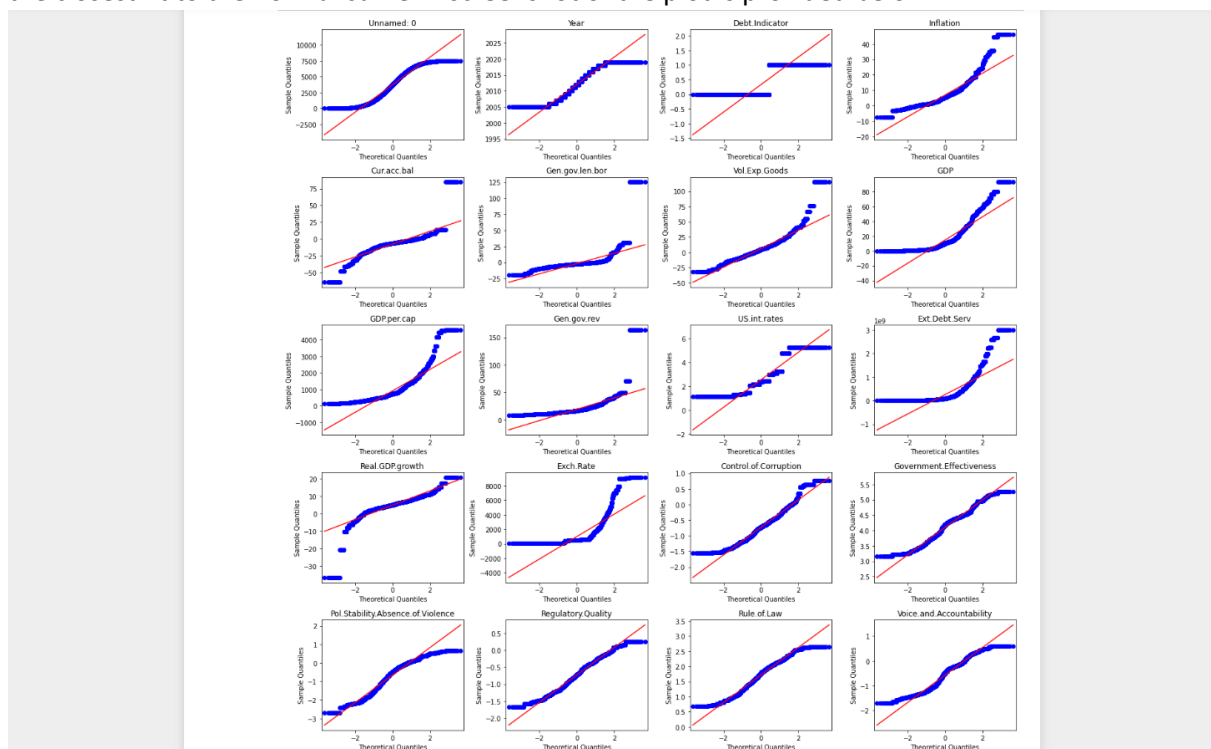
Moreover, no noticeable disadvantages were encountered when using these tools for data extraction and preparation in this section. Python and Jupyter Notebooks proved to be a robust combination for this phase of the analysis, allowing for a smooth transition to exploratory data analysis and modeling in subsequent stages.

D. ANALYSIS

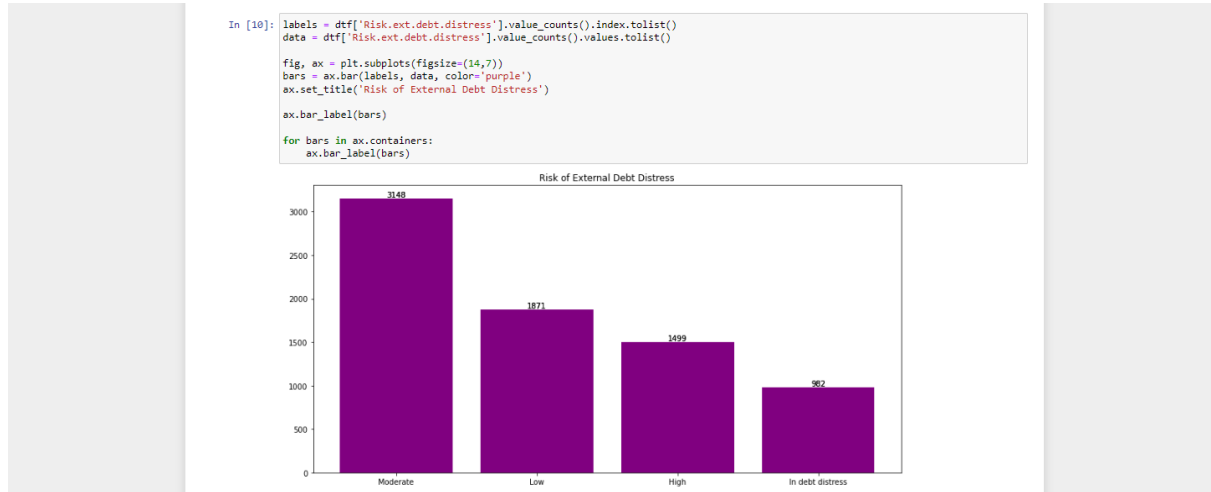
Data analysis can be defined as the process of cleaning, altering, and modelling data to discover meaningful information for business decision-making. The aim of Data Analysis is to obtain useful information from data and further making decisions based upon the data analysis (Johnson, 2022). Data analysis is imperative to research as it makes studying and understanding data a lot simpler and more accurate. It also helps the researchers interpret the data in a straightforward manner so that they do not exclude anything that could help them derive insights from it (Amadebai, n.d.).

In this phase, the initial step was to visualize data. Data visualization is simply the visual or graphical representation of data. It is beneficial in highlighting the most valuable insights from a data set, making it easier to recognize patterns, outliers, trends, and correlations (Stevens, 2021). Various attributes of the data were plotted as either bar charts, histograms, or pie charts. There was a total of four visualizations created using the matplotlib and seaborn libraries. These visualizations were:

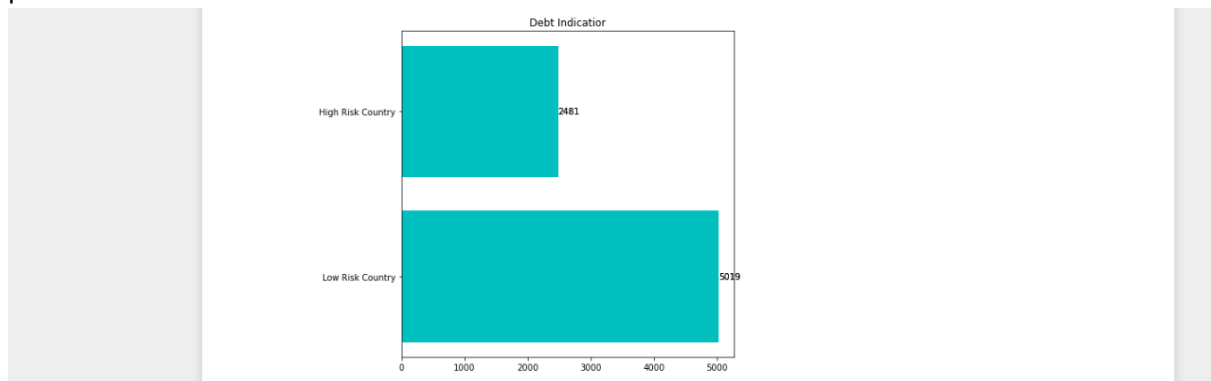
- **Q-Q plots:** Here, I used the `sm.qqplot` function from the `statsmodels` library to create a Q-Q plot for each numerical variable in the dataset. A Q-Q plot compares the quantiles of a variable with the quantiles of a normal distribution. This helps to check if the variable is normally distributed or not. The closer the points are to the diagonal line, the more normal the distribution is. The Q-Q plots reveal that majority of the continuous variables follow an approximate normal distribution, with some minor deviations in the heads and tails. Control.of.Corruption, Government.Effectiveness, Regulatory.Quality, Rule.of.Law, and Voice.and.Accountability exhibit the closest fit to the normal curve. A screenshot of the plot is provided below:



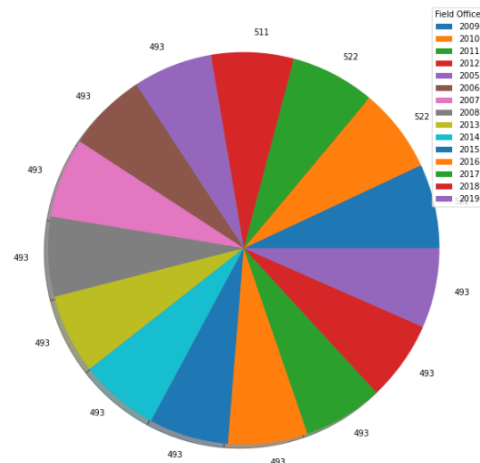
- **Bar chart:** I applied the `ax.bar` function from the `matplotlib.pyplot` library to create a bar chart for the `Risk.ext.debt.distress` variable. A bar chart shows the frequency of each category in a categorical variable. This helps to compare the relative size of each category. From the chart, it can be seen that most of the countries have a moderate risk of external debt distress, followed by low, high, and in debt distress. Below is a screenshot of the chart:



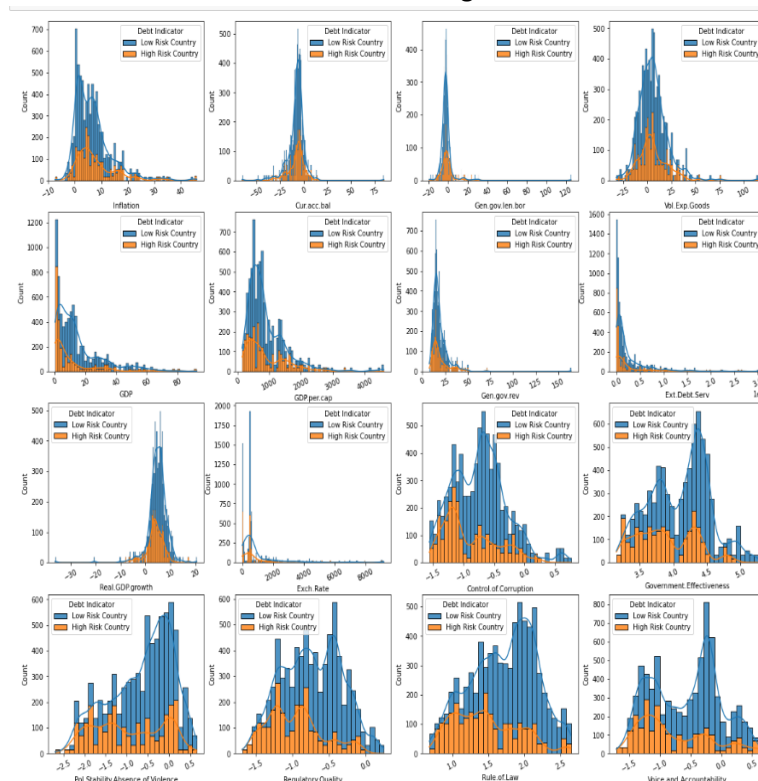
- **Horizontal bar chart:** I utilized the `ax.barh` function from the `matplotlib.pyplot` library to create a horizontal bar chart for the `Debt.Indicator` variable. Horizontal bar charts also show the frequency of each category in a categorical variable. This helps to compare the relative size of each category. From the chart, the horizontal bar plots of risk levels show low risk countries outnumber high risk countries by a ratio of about 2 to 1. The image below illustrates this entire process:



- **Pie chart:** Here, I used the `plt.pie` function from the `matplotlib.pyplot` library to create a pie chart for the `Year` variable. A pie chart shows the proportion of each category in a categorical variable. This helps to visualize the share of each category in the entire dataset. The pie chart illustrates a relatively even distribution of records across years, with a slight peak from 2010 to 2013 which together make up 25% of the data. No single year dominates. A screenshot of the chart is provided below:



- Histograms:** The `sns.histplot` function from the `seaborn` library was used to create a histogram for each numerical variable in the dataset. A histogram shows the distribution of a numerical variable by grouping the values into bins and counting the frequency of each bin. This helps to explore the shape, center, spread, and outliers of the variable. From the histograms, some variables have skewed distributions, such as `Gen.gov.len.bor`, `Gen.gov.rev`, and `Ext.Debt.Serv`. Some variables have bimodal distributions, such as `Control.of.Corruption`, and `Government.Effectiveness`. Below is a screenshot of the histogram:



After visualizing the data the next step was data preprocessing. Data preparation or preprocessing is the sorting, cleaning, and formatting of raw data so that it can be better used in business intelligence, analytics, and machine learning applications (Wolff, 2021). In this analysis data preparation comprised of an eight-step series of processes. These steps were:

1. Dropping Insignificant Columns

In this step, columns that were considered insignificant for the analysis were removed from the dataset. These columns were 'Unnamed: 0', 'ISO', 'Debt Indicator', and 'Risk.ext.debt.distress'. This process helps to reduce unnecessary data and focus on relevant attributes.

2. Encoding Categorical Values

To prepare the data for machine learning analysis, categorical values were encoded into numerical values using one-hot encoding. This was applied to the 'Year' column, which was converted into separate binary columns for each year. Each of these columns represents a specific year and has a value of 1 if the data corresponds to that year and 0 otherwise. This encoding allows the model to work with categorical data effectively.

3. Separating Features and Target Variable

The dataset was divided into two components: features (X) and the target variable (y). The 'Debt.Indicator' column was designated as the target variable (y), while all other columns were considered as features (X). This separation is crucial for supervised machine learning, where the model learns patterns in the features to predict the target variable.

4. Standardizing Features

Standardization was applied to the feature columns (X) to ensure that all features have zero mean and unit variance. This step helps in making the features more comparable and can improve the performance of machine learning algorithms, especially those sensitive to feature scales.

5. Splitting Data into Training and Test Sets

The final step involved splitting the data into training and test sets. This was done using the `train_test_split` function from the `sklearn` library. 80% of the data was allocated for training the model, while the remaining 20% was reserved for testing the model's performance. The `random_state` parameter was set to ensure reproducibility.

These data preparation steps are essential for creating a clean, standardized dataset that can be used for machine learning model training and evaluation. They help ensure that the data is in the right format and ready for analysis. A screenshot illustrating the entire data preparation process is provided in the image below:

Data Cleaning

In [15]: # Drop all insignificant columns

```
drop_cols = ['Unnamed: 0', 'ISO', 'Debt Indicator', 'Risk.ext.debt.distress']  
dtf = dtf.drop(drop_cols, axis = 1)  
dtf.head()
```

Out[15]:

	Year	DebtIndicator	Inflation	Cur.acc.bal	Gen.gov.len.bor	Vol.Exp.Goods	GDP	GDP.per.cap	Gen.gov.rev	US.int.rates	Ext.Debt.Serv	Real.GDP.growth	
0	2005	0	4.676	-3.444	-1.523	-1.734	6.571	823.240	12.615	2.981357	48441194.6	1.713165	5
1	2005	0	6.409	-10.324	-4.890	18.105	6.150	458.187	15.327	2.981357	45990513.7	8.661873	5
2	2005	1	13.253	-4.665	-10.556	24.653	1.117	148.710	22.587	2.981357	40010353.8	0.900000	10
3	2005	0	1.993	-2.757	3.292	-10.677	17.974	1007.390	16.798	2.981357	818876650.0	2.020662	5
4	2005	1	2.879	-6.235	-4.343	-6.075	1.413	356.688	11.813	2.981357	6878445.1	0.908211	5

In [16]: # Convert categorical values to dummy numerical values

```
# Encode discrete attributes into individual separate columns so it displays 0 if absent and 1 if present for each column.  
dtf = pd.get_dummies(data=dtf, columns=['Year'])  
dtf.head()
```

Out[16]:

	DebtIndicator	Inflation	Cur.acc.bal	Gen.gov.len.bor	Vol.Exp.Goods	GDP	GDP.per.cap	Gen.gov.rev	US.int.rates	Ext.Debt.Serv	Real.GDP.growth	Exch.R
0	0	4.676	-3.444	-1.523	-1.734	6.571	823.240	12.615	2.981357	48441194.6	1.713165	527.258
1	0	6.409	-10.324	-4.890	18.105	6.150	458.187	15.327	2.981357	45990513.7	8.661873	527.258
2	1	13.253	-4.665	-10.556	24.653	1.117	148.710	22.587	2.981357	40010353.8	0.900000	1081.577
3	0	1.993	-2.757	3.292	-10.677	17.974	1007.390	16.798	2.981357	818876650.0	2.020662	527.258
4	1	2.879	-6.235	-4.343	-6.075	1.413	356.688	11.813	2.981357	6878445.1	0.908211	527.258

In [17]: # Separate features (X) and target variable (y)

```
X = dtf.drop(columns=['DebtIndicator'])  
y = dtf['DebtIndicator']
```

In [18]: # Standardize the features to have zero mean and unit variance

```
scaler = StandardScaler()  
X = scaler.fit_transform(X)
```

In [19]: # Split the data into training and test sets

```
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, random_state=7)
```

After the data cleaning process, the next phase was to build the predictive model, train it on the data and analyse its performance. Here, we began by creating a Support Vector Machine (SVM) model, setting the foundation for our analysis. This model was then trained using the provided training data, enabling it to learn patterns and relationships within the dataset. Following this, we evaluate the model's performance in several ways.

Firstly, we generate predictions for a test dataset, measuring the model's accuracy by comparing these predictions to the actual labels. Next, we create a comprehensive classification report, offering insights into the model's precision, recall, and F1-score for each class. This information helps us gauge the model's classification capabilities effectively. We further delve into performance assessment by constructing a confusion matrix, detailing true positives, true negatives, false positives, and false negatives. The confusion matrix aids in understanding the model's performance nuances, especially in binary classification scenarios.

Additionally, visualizations are employed to enhance our understanding. A heatmap of the confusion matrix is generated for a more accessible interpretation of classification results. Finally, we analyse the model's ability to discriminate between classes through a Receiver Operating Characteristic (ROC) curve. This visual representation of true positive rates versus false positive rates provides valuable insights, complemented by the Area Under the Curve (AUC) score.

Screenshots of the entire model creation and evaluation process is illustrated in the image below:

Machine Learning Model

```
In [20]: # Instantiate the SVM model
svm_model = SVC()

In [21]: # Fit the model to the training set
svm_model.fit(X_train, y_train)
print('The model has been successfully trained')
The model has been successfully trained

In [22]: # Model Evaluation: Predict test set using the model
svm_predictions = svm_model.predict(X_test)

In [23]: # Calculate accuracy
svm_accuracy = accuracy_score(y_test, svm_predictions)

In [24]: # Print accuracy score of the model
print("Support Vector Machine Accuracy:", svm_accuracy)
Support Vector Machine Accuracy: 0.9933333333333333

In [25]: # Print classification report
print("Support Vector Machine Classification Report:\n", classification_report(y_test, svm_predictions))

Support Vector Machine Classification Report:
              precision    recall  f1-score   support

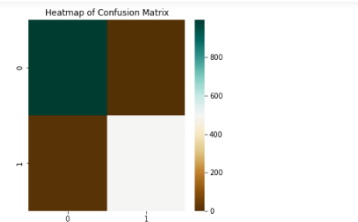
     0       0.99       1.00       0.99       995
     1       1.00       0.98       0.99       505

 accuracy          1.00          0.99          0.99      1500
 macro avg          1.00          0.99          0.99      1500
 weighted avg       0.99          0.99          0.99      1500

In [26]: # Print confusion matrix for each model
cm = confusion_matrix(y_test, svm_predictions)
print("Support Vector Machine Confusion Matrix:\n", cm)

Support Vector Machine Confusion Matrix:
[[995  0]
 [ 10 495]]

In [27]: fig, ax = plt.subplots(figsize=(5,5))
```

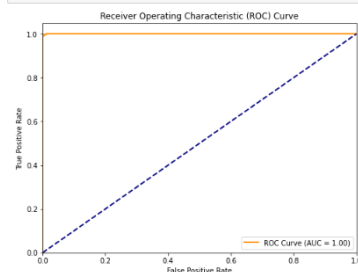


```
In [28]: y_prob = svm_model.decision_function(X_test)
# Calculate the ROC curve
fpr, tpr, thresholds = roc_curve(y_test, y_prob)
```

```
In [29]: # Calculate the AUC (Area Under the Curve) score
auc = roc_auc_score(y_test, y_prob)
# auc
```

Out[29]: 0.9999363152395642

```
In [30]: # Plot the ROC curve
plt.figure(figsize=(8, 6))
plt.plot(fpr, tpr, color='darkorange', lw=2, label=f'ROC Curve (AUC = {auc:.2f})')
plt.plot([0, 1], [0, 1], color='navy', lw=2, linestyle='--')
plt.xlim([0.0, 1.0])
plt.ylim([0.0, 1.05])
plt.xlabel('False Positive Rate')
plt.ylabel('True Positive Rate')
plt.title('Receiver Operating Characteristic (ROC) Curve')
plt.legend(loc='lower right')
plt.show()
```



One advantage of the data analysis process and the choice of tools for this this research is that it provided a great deal of insight with the data set in regard to the columns, their data types, their visualizations, the stories they told and their relevance to achieving the objective of this project. One disadvantage of the entire data analysis process to this project is the fact that since feature selection and ranking was not applied here, we are not able to determine which variables are most important to the model and which ones are of less significance.

E. DATA SUMMARY AND IMPLICATIONS

In the context of the research question aiming to construct an accurate predictive model for categorizing African countries' debt sustainability, the results are exceptionally promising. The Support Vector Machine (SVM) model demonstrated remarkable performance in distinguishing between different debt sustainability categories. It achieved an outstanding accuracy rate of approximately 99.33%, indicating the model's exceptional proficiency in classifying countries into debt distress and non-distress categories based on external debt distress indicators and other pertinent features.

Notably, in addition to accuracy, the model exhibited exceptional precision and recall rates, with precision scores of 0.99 and 1.00 for debt distress and non-distress categories, respectively. These high precision scores indicate a minimal false-positive rate, suggesting that when the model predicts a country as debt-distressed, it is highly likely to be accurate. Similarly, recall scores of 1.00 and 0.98 for debt distress and non-distress categories, respectively, signify that the model effectively captures a vast majority of the actual instances of debt distress.

Furthermore, the study assessed the significance of various predictor variables in determining debt distress levels. While the analysis demonstrates the model's exceptional predictive power, it also provides valuable insights into the factors influencing debt vulnerabilities in Sub-Saharan African countries. This knowledge can significantly enhance stakeholders' understanding of the region's current debt situation, enabling more informed decisions regarding lending and grant allocation.

However, one limitation of this analysis is the potential risk of overfitting, given the exceptionally high accuracy rate. Future studies should focus on conducting a more extensive evaluation to ensure the model's generalizability. Additionally, considering the critical role of external factors in debt sustainability, future research directions might include incorporating economic and geopolitical variables into the model for a more comprehensive analysis. Furthermore, exploring alternative machine learning algorithms to assess their performance in this context could provide valuable insights for more robust predictive modeling in the field of debt sustainability analysis in Sub-Saharan Africa.

F. REFERENCES

1. Agarwal, S., & Tomar, D. (2011). Classification of Countries based on Macro-Economic Variables using Fuzzy Support Vector Machine. *International Journal of Computer Applications*, 27(6). <https://doi.org/10.5120/3302-4513>
2. Amadebai, E. (n.d.). The Importance of Data Analysis in Research. Retrieved September 29, 2023, at <https://www.analyticsfordecisions.com/importance-of-data-analysis-in-research/>
3. Comelli, F., David, A., Eyraud, L., Kovacs, P., Montoya, J., & Sode, A. (2023, September 26). How to Avoid a Debt Crisis in Sub-Saharan Africa. Retrieved September 29, 2023, from <https://www.imf.org/en/News/Articles/2023/09/26/cf-how-to-avoid-a-debt-crisis-in-sub-saharan-africa>
4. Hellwig, K.-P. (2021, May). Predicting Fiscal Crises: A Machine Learning Approach (IMF Working Paper). Asia Pacific Department.
5. International Monetary Fund. (n.d.). The Debt Sustainability Framework for Low-Income Countries. Retrieved September 29, 2023, from <https://www.imf.org/external/pubs/ft/dsa/lic.htm>
6. Jain, K. (2017, September 17). Python vs. R vs. SAS – which tool should I learn for Data Science? Retrieved September 29, 2023, from <https://www.analyticsvidhya.com/blog/2017/09/sas-vs-vs-python-tool-learn/>
7. Johnson, D. (2022, September 17). What is Data Analysis? Research, Types & Example. Retrieved September 29, 2023, at <https://www.guru99.com/what-is-data-analysis.html>
8. Kaggle. (2021). Evolution of debt vulnerabilities in Africa. Retrieved September 29, 2023, from <https://www.kaggle.com/datasets/evadrichter/evolution-of-debt-distress-in-hipc-countries>
9. McKinney, W. (2017). *Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython* (2nd ed.). O'Reilly Media.
10. Monga, C. (2023, March 3). Rethinking Debt Sustainability in Africa. Retrieved September 29, 2023, from <https://www.project-syndicate.org/onpoint/broadening-the-indices-of-african-debt-sustainability-by-celestin-monga-2023-03>
11. Perkel, J. M. (2018, October 30). Why Jupyter is data scientists' computational notebook of choice. Retrieved September 29, 2023, at <https://www.nature.com/articles/d41586-018-07196-1>
12. Sokpoh, A., Chirikure, N., & Braganza, J. R. (2022, March 2). Africa's Debt Landscape: Scope for Sustainability. Retrieved September 29, 2023, from <https://afripoli.org/africas-debt-landscape-scope-for-sustainability>
13. Stevens, E. (2021, July 15). What Is Data Visualization and Why Is It Important? A Complete Introduction. Retrieved September 29, 2023, at <https://careerfoundry.com/en/blog/data-analytics/what-is-data-visualization/#what-is-data-visualization-a-definition>
14. Wolff, R. (2021, May 28). Data Preparation: Basics & Techniques. Retrieved September 29, 2023, at <https://monkeylearn.com/blog/data-preparation/>