**Data Analysis Project**

(Graduation project NTI Internship)

**Team1:**

Mohamed Saeed

Fatma Najah

Shahd Mohamed

Mahmoud Saeed

**Supervisor:**

Eng.Menna Tarek

**23/7/2025**

**Data Cleaning by python:**

**1. Data Loading and Initial Exploration**

The first steps involve loading the necessary libraries and the dataset, and then performing an initial assessment of the data's quality.

* **Import Libraries:**  
  import pandas as pd  
  import numpy as np  
  import seaborn as sns  
  import matplotlib.pyplot as plt  
    
  This imports the essential libraries for data manipulation (pandas, numpy), and visualization (seaborn, matplotlib).
* **Load Data:**  
  df = pd.read\_csv(r"E:\Data analysis NTI\Final project\governance\_dataset\_dirty\_2000\_rows.csv")
* **Then df to print the dataset and understand it**



* **Initial Checks:**
* df.duplicated().any(): This checks for any duplicate rows in the dataset.
* df.isnull().any(): This checks each column for the presence of null (missing) values.
* df.info(): This provides a summary of the DataFrame, including the data types of each column and the number of non-null values.

**2. Data Cleaning and Preprocessing**

This section details the cleaning process for each column to ensure data quality and consistency.

**Company\_ID**

* **Splitting the Column:**  
  df[['Company\_Prefix', 'Company\_Number']] = df['Company\_ID'].str.extract(r'([A-Za-z]+)(\d+)')  
    
  The Company\_ID is split into a prefix (text part) and a number.
* **Dropping and Renaming:**  
  df=df.drop(columns='Company\_Prefix')  
  df=df.drop(columns='Company\_ID')  
  df = df.rename(columns={'Company\_Number': 'Company\_ID'})  
    
  The original Company\_ID and the extracted Company\_Prefix are dropped, and the Company\_Number is renamed back to Company\_ID.
* **Type Conversion:**  
  df["Company\_ID"]=df["Company\_ID"].astype('int')  
    
  The Company\_ID is converted to an integer data type.

**Text-Based Columns (Company\_Name, Sector, CEO)**

* **Standardization:**  
  df['Company\_Name'] = df['Company\_Name'].str.lower().str.strip()  
  df.drop\_duplicates(subset=['Company\_Name'], keep='first')  
  df['Sector'] = df['Sector'].str.lower().str.strip()  
  df['CEO'] = df['CEO'].str.lower().str.strip()  
    
  These columns are converted to lowercase and leading/trailing whitespace is removed to ensure consistency. Duplicate company names are dropped.
* **Handling Missing CEO:**  
  df['CEO'] = df['CEO'].fillna('Unknown')  
    
  Missing CEO names are filled with the string 'Unknown'.

**Numerical Columns (Board\_Size, Independent\_Directors, Corruption\_Risk\_Score, Year\_Established, ESG\_Score)**

* **Board\_Size:**
* pd.to\_numeric(df['Board\_Size'], errors='coerce'): Converts the column to a numeric type; any non-numeric values become NaN.
* The median of the Board\_Size is calculated and used to fill the NaN values.
* The column is then converted to an integer type.
* **Independent\_Directors:**
* pd.to\_numeric(df['Independent\_Directors'], errors='coerce'): Converts to numeric, with non-numeric values becoming NaN.
* The median is used to fill NaN values.
* The column is converted to an integer type.
* **Corruption\_Risk\_Score:**
* pd.to\_numeric(df['Corruption\_Risk\_Score'], errors='coerce'): Converts to numeric.
* The median is used to fill NaN values.
* **Year\_Established:**
* pd.to\_numeric(df['Year\_Established'], errors='coerce'): Converts to numeric.
* The median year is used to fill NaN values.
* The column is converted to an integer type.
* **ESG\_Score:**
* The column is converted to a numeric type.
* The median is used to fill NaN values.
* A histogram is plotted to visualize the distribution.
* df["ESG\_Score"] = df["ESG\_Score"].clip(lower=0, upper=100): The values are capped between 0 and 100 to handle outliers.

**3. Outlier Detection and Handling (Num\_Employees)**

The Modified Z-score method is used to detect and handle outliers in the Num\_Employees column.

* **Modified Z-score Calculation:**  
  abs\_median = abs(df["Num\_Employees"]-df["Num\_Employees"].median())  
  MAD = abs\_median.median()  
  mod\_z\_scores = 0.6745\*((df["Num\_Employees"]-df["Num\_Employees"].median())/MAD)  
    
  This calculates the Modified Z-score for each value in the Num\_Employees column.
* **Outlier Replacement:**  
  placeholder = 9999999  
  df['Num\_Employees'] = df['Num\_Employees'].replace(placeholder, np.nan)  
  median\_employees = df['Num\_Employees'].median()  
  df['Num\_Employees'].fillna(median\_employees, inplace=True)  
  df['Num\_Employees'] = df['Num\_Employees'].astype(int)  
    
  The placeholder value 9999999 is replaced with NaN. These NaN values are then filled with the median number of employees, and the column is converted to an integer type.

**4. Handling Dates (Last\_Audit\_Date)**

* **Date Conversion and Imputation:**  
  df['Last\_Audit\_Date'] = pd.to\_datetime(df['Last\_Audit\_Date'], errors='coerce')  
  most\_recent\_date = df['Last\_Audit\_Date'].max()  
  df['Last\_Audit\_Date'].fillna(most\_recent\_date, inplace=True)  
    
  The Last\_Audit\_Date column is converted to a datetime format. Any missing or invalid dates are filled with the most recent valid date in the column.

**5. Modeling for Missing Revenue\_USD**

A RandomForestRegressor model is used to predict the missing values in the Revenue\_USD column based on other features.

* **Model Preparation:**
* A copy of the DataFrame is created for modeling.
* Categorical features (Sector, Country) are one-hot encoded.
* The data is split into a training set (where Revenue\_USD is known) and a prediction set (where it is missing).
* **Training and Prediction:**  
  model = RandomForestRegressor(n\_estimators=100, random\_state=42)  
  model.fit(X\_train, y\_train)  
  predicted\_scores = model.predict(X\_predict)  
  df.loc[df['Revenue\_USD'].isnull(), 'Revenue\_USD'] = predicted\_scores  
    
  The model is trained on the available data and then used to predict the missing revenue values.

**6. Saving the Final Dataset**

The cleaned and processed DataFrame is saved to both CSV and Excel formats.

* **Saving to CSV:**  
  df.to\_csv('governance\_data\_final.csv', index=False)
* **Saving to Excel:**  
  ESG = pd.DataFrame(df)  
  file\_path = "E:\Data analysis NTI\Final project/Cleaned\_dataGovernance22.xlsx"  
  ESG.to\_excel(file\_path, index=False)

***Tableau Dashboard***

The instructions cover creating essential calculated fields, building a variety of charts for different analytical purposes, and applying advanced statistical plots.

**1. Creating Calculated Fields**

Calculated fields allow you to create new data from the data that already exists in your data source. Here’s how to create several useful calculated fields for ESG (Environmental, Social, and Governance) analysis.

To create a calculated field, right-click anywhere in the **Data pane** (the left-hand sidebar) and select **Create Calculated Field**.

**A. ESG Risk Category**

This calculation categorizes companies into risk levels based on their ESG score.

* **Field Name:** ESG Risk Category
* **Formula:**
* IF [ESG\_Score] >= 75 THEN "Low Risk"
* ELSEIF [ESG\_Score] >= 50 THEN "Medium Risk"
* ELSE "High Risk"
* END

**B. Board Independence Ratio**

This calculation determines the proportion of independent directors on a company's board.

* **Field Name:** Board Independence Ratio
* **Formula:**
* [Independent\_Directors] / [Board\_Size]

**C. Company Age**

This calculation determines the age of a company in years based on its establishment date.

* **Field Name:** Company Age
* **Formula:**
* YEAR(TODAY()) - [Year\_Established]

**2. Building Worksheets: Core Visualizations**

Follow these instructions to create individual worksheets that visualize your data from different perspectives.

**Sheet 1: Geographic ESG Score Overview (Map)**

This map visualizes the average ESG score by country, providing a clear geographical perspective on performance.

1. **Set Mark Type:** In the **Marks** card, change the dropdown from Automatic to **Map**.
2. **Add Geographic Detail:** Drag Country from the Data pane to the **Detail** property on the Marks card.
3. **Generate Coordinates:** Drag Latitude (generated) to the **Rows** shelf and Longitude (generated) to the **Columns** shelf.
4. **Color by ESG Score:** Drag ESG\_Score to the **Color** property on the Marks card. Right-click the ESG\_Score pill, go to **Measure**, and select **Average**. Click the **Color** property again and choose a **Green-Yellow-Red Diverging** palette to visually distinguish performance levels.
5. **Enhance Tooltip:** Drag Company\_Name and Sector to the **Tooltip** property on the Marks card. Click the **Tooltip** property to edit the text and add COUNTD(Company\_ID) to show the number of companies in each country.

**Sheet 2: ESG Score by Sector (Bar Chart)**

This bar chart compares the average ESG score across different business sectors.

1. **Create Bars:** Drag Sector to the **Rows** shelf and ESG\_Score to the **Columns** shelf. Tableau will default to a bar chart.
2. **Set Aggregation:** Right-click the ESG\_Score pill on the Columns shelf, go to **Measure**, and ensure it's set to **Average**.
3. **Color by Score:** Drag AVG(ESG\_Score) from the Columns shelf to the **Color** property on the Marks card.
4. **Sort Data:** Click the **Sort** icon on the axis or in the toolbar to sort the sectors in descending order by their average ESG score.

**Sheet 5: Top 10 Companies by ESG Score (Table)**

This table highlights the top-performing companies based on their average ESG score.

1. **Build the Table:**
   * Drag Company\_Name, CEO, and Sector to the **Rows** shelf.
   * Drag ESG\_Score to the **Columns** shelf. Right-click the pill and set the measure to **Average**.
2. **Apply Top N Filter:**
   * Drag Company\_Name from the Data pane to the **Filters** shelf.
   * In the filter dialog, go to the **Top** tab.
   * Select **By field**.
   * Configure it to show the **Top 10** by ESG\_Score with the aggregation set to **Average**.

**Sheet 6: Employee Distribution by Sector (Treemap)**

A treemap is excellent for visualizing hierarchical data as a proportion of a whole. Here, it shows the distribution of employees across sectors.

1. **Set Mark Type:** In the **Marks** card, change the dropdown from Automatic to **Treemap**.
2. **Define Size:** Drag Num\_Employees to the **Size** property on the Marks card. Ensure the measure is set to **SUM**.
3. **Define Color & Label:**
   * Drag Sector to the **Color** property.
   * Drag Sector and SUM(Num\_Employees) to the **Label** property to display them inside the rectangles.

**3. Advanced Analysis Techniques**

These plots offer deeper statistical insights into your data's distribution and variability.

**Box and Whisker Plot: ESG Score Distribution by Sector**

This plot shows the range, median, and interquartile range of ESG scores within each sector, revealing variability and identifying potential outliers.

1. **Initial Setup:** Drag Sector to **Columns** and ESG\_Score to **Rows**.
2. **Disaggregate Data:** Go to the **Analysis** menu at the top and uncheck **Aggregate Measures**. This plots a mark for every company's individual score.
3. **Select Plot Type:** In the **Show Me** panel (top-right), select the **Box-and-Whisker** plot. Tableau will automatically rearrange the pills and structure the plot.
4. **Add Color (Optional):** Drag Sector to the **Color** property on the Marks card to assign a unique color to each sector's box plot for easier comparison.

**Histogram: Distribution of Board Sizes**

A histogram reveals the frequency distribution of a measure, helping you understand common patterns. This chart will show the most common board sizes.

1. **Create Bins:**
   * In the **Data Pane**, right-click the Board\_Size field.
   * Select **Create** > **Bins**.
   * In the dialog box, you can accept the default suggested bin size or define your own (e.g., a size of 2 to group board sizes like 8-9, 10-11, etc.). Click **OK**.
2. **Build the Histogram:**
   * Drag the newly created Board\_Size (bin) field to the **Columns** shelf.
   * Drag Company\_ID to the **Rows** shelf. Right-click its pill, go to **Measure**, and select **Count (Distinct)** to get the number of unique companies in each bin.
3. **View:** The resulting chart is a histogram showing the frequency of companies for each board size bucket.

**Sheet 1: Geographic ESG Score Overview (Map)**

**Question:** Which geographic regions exhibit the highest and lowest average ESG scores, and where are the highest concentrations of companies in our dataset located?

**2. Sheet 2: ESG Score by Sector (Bar Chart)**

**Question:** Which sectors are leading and which are lagging in terms of overall average ESG performance?

**3. Sheet 5: Top 10 Companies by ESG Score (Table)**

**Question:** Who are the top 10 performing companies based on average ESG score, and in which sectors do they operate?

**4. Sheet 6: Employee Distribution by Sector (Treemap)**

**Question:** Which sectors are the largest in terms of total number of employees, and how do they compare proportionally?

**5. Box and Whisker Plot: ESG Score Distribution by Sector**

**Question:** Which sectors exhibit the most consistency in ESG scores (i.e., a narrow distribution), and which show the widest range of performance among their constituent companies?

**6. Histogram: Distribution of Board Sizes**

**Question:** What is the most common board size across all companies, and what is the overall frequency distribution of board sizes?

***Power BI******Dashboard***

After cleaning the data using Python, we loaded it into Power BI for visualization. We ensured the data types were correct, formatted the Revenue column to display two decimal places, and added new calculated columns: *Corruption\_Level*, *Company\_Age*, and *Company\_Age\_Group*.

**1-Corruption\_Level**

Corruption\_Level = SWITCH(TRUE(),ESG[Corruption\_Risk\_Score]<=2.5,"Very Low", ESG[Corruption\_Risk\_Score]<=5,"Low",ESG[Corruption\_Risk\_Score]<=7.5,"Moderate",ESG[Corruption\_Risk\_Score]<=10, "High")

**2- Company\_Age**

Company\_Age = YEAR(TODAY())-(ESG[Year\_Established])

**3- Company\_Age\_Group**

Company\_Age\_Group=SWITCH(TRUE(),ESG[Company\_Age]<=5,"Startup", ESG[Company\_Age]<=10,"Young",ESG[Company\_Age]<=30,"Established", ESG[Company\_Age]>30,"Old")

**Visualization**

The dashboard consists of four main pages designed for effective visualization:

* 1. **Home page**  
     Includes the dashboard title, logo, and navigation icons to move between pages.
  2. **Questions page**

Presents the key questions addressed through the dashboard using various charts and visuals

* 1. **Details page**

Provides deeper insights into the data using matrix tables and a decomposition tree.

**What questions we needed answer it?**

* What is the percentage of companies with a large number of independent board members?
* How are ESG scores distributed across different countries?
* How do sectors vary in terms of corruption levels and ESG scores?
* What is the revenue size in different countries in relation to corruption?
* Is there a relationship between companies’ governance (ESG) scores and corruption?

**KPIs :**

-Average of Revenue\_USD: 2.5 bn

- Average of Corruption\_Risk: 1.84

- Average of ESG\_Score: 48.78

- Total Board\_Size: 17k

- Total Independent\_Directors: 7.47k

**Excel Dashboard**

**1. Chart: Average ESG Score & Corruption by Sector**

**Description:**

A clustered bar chart that shows the average ESG scores and average corruption risk scores across different sectors such as Technology, Retail, Manufacturing, etc.

**Purpose:**

To compare sustainability performance and corruption risk across sectors.

**Key Question:**

**❓ Which sector has the best ESG performance and lowest corruption risk?**

**2. Chart: Total Revenue by Board Size (Pie Chart)**

**Description:**

A pie chart illustrating the distribution of total company revenue based on board size categories (e.g., 5, 7, 9, etc.).

**Purpose:**

To explore the relationship between board size and total revenue generated.

**Key Question:**

**❓ Which board size is associated with the highest total revenue?**

**3. Chart: Total Companies by Country**

**Description:**

A column chart showing the number of companies per country, including Brazil, Egypt, France, Germany, and more.

**Purpose:**

To identify the countries with the highest representation of companies in the dataset.

**Key Question:**

**❓ Which country has the highest number of companies?**

**4. Chart: Total Companies by Headquarters City**

**Description:**

A bar chart showing the number of companies based on their headquarters locations (e.g., Berlin, Cairo, Mumbai, New York, etc.).

**Purpose:**

To analyze the geographical distribution of company HQs.

**Key Question:**

**❓ Which city hosts the largest number of company headquarters?**

**5. Chart: Average Revenue by Last Audit Date**

**Description:**

A line chart displaying the average revenue over different years, based on the last audit date (from 2015 to 2023).

**Purpose:**

To examine trends and fluctuations in average revenue over time.

**Key Question:**

**❓ How has average company revenue changed over the years from 2015 to 2023?**

**6. Interactive Filters (Slicers): Audit Year, Country, Sector, Year Established**

**Description:**

Slicers (dropdown filters) that allow users to dynamically filter data by audit year, country, sector, and year of establishment.

**Purpose:**

To enable customized, interactive analysis for different use cases.