5/1/2025

# Visual exploration of the dataset “global-renewable-energy”

Assignments Task 1 for the course: Explorative Data Analysis and Visualization

Alvine Gadeu Nee Djomatchie Touko
IU AKADEMIE

## Content

Introduction

1.Data Analysis

1.1. Data information
1.2. Location and variability
1.3. Remove duplicates
1.4. Handle missing Data
1.5. Outliers and bad data detections

2.Data visualization

2.1. Scatterplot of the total renewable energy per country over the years

2.2. Bar chart of energy production per country

2.3. Bar chart of energy production over the years

2.4. Heatmap to visualize the correlations between energy features

2.5. Relational plot for further investigation of the correlation

Conclusion

References

## Introduction

This part of the training focuses on data visualization. It is especially important today because data is everywhere and often abstract consisting of numbers, letters, and tables. To properly analyze and utilize data, visualization is essential and is becoming increasingly important. The key goal is to extract valuable and meaningful information from it. Once this is done, the data can be used to drive progress and make informed, data-driven decisions. Python has been established as the leading language for data-driven decision-making, and we will use it to visualize our dataset effectively.

As part of the course explorative Data Analysis and visualization, I will be practicing my ability to investigate a Dataset, summarize their main Characteristics and use visualizations methods to uncover their patterns, relationships and insights. For this purpose, I choose the dataset “Global Renewable Energy Production (2000-2023)” from Kaggle, because it best fit to my background and is easy for me to understand.

The dataset summarizes the annual data on renewable energy between 2000 and 2023 and reveals the growth and distribution of renewable energy worldwide. The contributions of solar, wind, hydro and other renewable energy sources are provided in the columns.

For this assignment, I will begin by examining the dataset to ensure it is clean and ready for analysis. Then, I will perform the analysis and create visualizations to identify trends over time and explore the relationships between the different renewable energy sources included in the dataset. Using Python as the primary tool allows me to leverage its powerful libraries for data visualization, such as Matplotlib, Seaborn, and Plotly. These tools enable the creation of clear, informative, and visually appealing charts that help uncover trends, relationships, and anomalies within the data.

Please note that the full project and code can be found on my GitHub repository using the following code: https://github.com/toukoalvine/Visual-exploration-project.

# 1. Data Analysis

## 1.1. Data information

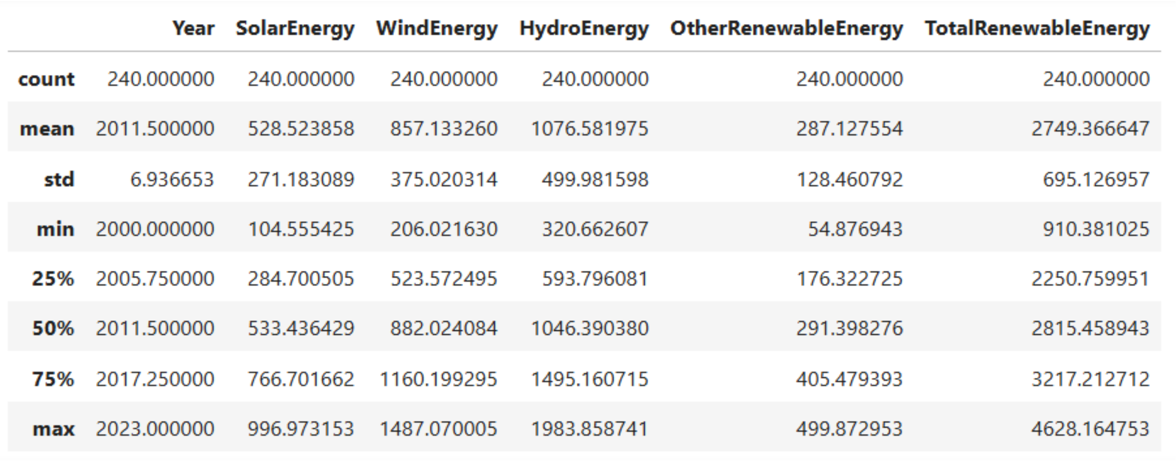
To understand the dataset, it was loaded and its structure inspected using *df.head()*, *df.dtypes*, and *df.info()*. The dataset consists of 7 columns and a total of 240 rows. The data types include 1 integer feature, 5 float features, and 2 object-type features.

The dataset was checked for data formats, data types, and column names, all of which were found to be correct. The column names are self-explanatory, and the following are the features included in the dataset:

- **Year**: The year of data collection (e.g., 2000, 2001, etc.)
- **Country**: The name of the country
- **SolarEnergy**: Annual solar energy production in gigawatt-hours (GWh)
- **WindEnergy**: Annual wind energy production in gigawatt-hours (GWh)
- **HydroEnergy**: Annual hydro energy production in gigawatt-hours (GWh)
- **OtherRenewableEnergy**: Annual energy production from other renewable sources (e.g., geothermal, biomass) in gigawatt-hours (GWh)
- **TotalRenewableEnergy**: Total annual renewable energy production in gigawatt-hours (GWh).

## 1.2. Location and variability

For the numeric features descriptive statistics can easily be obtained with the following code *df.describe()*. With this, we have the descriptive statistics that summarize the central tendency, dispersion of a dataset distribution, excluding NaN values. For our data, the result are summarized in the picture below.



| | Year | SolarEnergy | WindEnergy | HydroEnergy | OtherRenewableEnergy | TotalRenewableEnergy |
|---|---|---|---|---|---|---|
| **count** | 240.000000 | 240.000000 | 240.000000 | 240.000000 | 240.000000 | 240.000000 |
| **mean** | 2011.500000 | 528.523858 | 857.133260 | 1076.581975 | 287.127554 | 2749.366647 |
| **std** | 6.936653 | 271.183089 | 375.020314 | 499.981598 | 128.460792 | 695.126957 |
| **min** | 2000.000000 | 104.555425 | 206.021630 | 320.662607 | 54.876943 | 910.381025 |
| **25%** | 2005.750000 | 284.700505 | 523.572495 | 593.796081 | 176.322725 | 2250.759951 |
| **50%** | 2011.500000 | 533.436429 | 882.024084 | 1046.390380 | 291.398276 | 2815.458943 |
| **75%** | 2017.250000 | 766.701662 | 1160.199295 | 1495.160715 | 405.479393 | 3217.212712 |
| **max** | 2023.000000 | 996.973153 | 1487.070005 | 1983.858741 | 499.872953 | 4628.164753 |

***Table 1**: descriptive statistics of the dataset*

The table 1 provides information about the number of values (count), the mean, variance, minimum and maximum values, as well as the quartiles (25%, 50%, and 75%) of the dataset. It also highlights the minimum and maximum values for each numerical feature. The mean represents the center of the distribution and takes all observations of the feature into account. For example, the mean total production of renewable energy between 2000 and 2023 is 2749.36 gigawatt-hours (GWh). the larger standard deviation in total renewable energy production indicates that the data points are widely spread out around the mean. Conversely, a smaller standard deviation suggests that the data are tightly clustered around the mean. Quartiles divide the data into four equal parts. The median, also known as the second quartile, corresponds to the 50th percentile and is shown as the 50% value in Table 1.

### 1.3. Remove duplicates

The dataset has been checked in duplicates with the following line code:

***Code***

*print(df.drop_duplicates())*

And no row has been dropped from the data, so there were no duplicates in it.

### 1.4. Handle Missing Data

I used the codes *print(df.dropna()), print(df.dropna(how='all')) and print(df.dropna(thresh=2))* to investigate missing values in the data frame. There were no missing values in the data.

### 1.5. Outliers and bad data detections

Knowing how to identify and handle outliers is an important part of data cleaning phase. There are various methods available for this task. For the visual inspection I decided to use box plots and histograms, which provide a comprehensive understanding of data distribution and ensure robust outliers detection.

The box plot with the following code:

```
import matplotlib.pyplot as plt
import seaborn as sns
plt.figure(figsize=(15, 10))
for i, col in enumerate(df.columns):
    plt.subplot(5, 5, i + 1)
    sns.boxplot(y=df[col], data = df)
    plt.title(col)
plt.tight_layout()
plt.show()
```

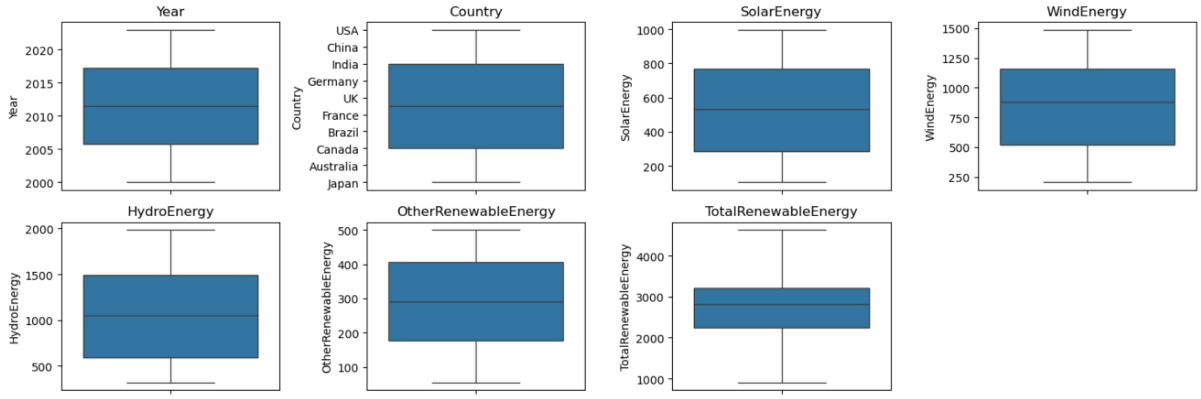


***Figure 1:*** *boxplots*

The Boxplots above clearly show that the data are all in the range. There is no outlier in this dataset.

This dataset is clean and requires no preprocessing for visualization and analysis.

# 2. Visualizations

## 2.1. Scatterplot of the total renewable energy per country over the years

The scatter plot in Figure 2 shows the evolution of total renewable energy production over the years for each country. It was created using the following piece of code.

***Code***

```
import plotly.express as px
fig = px.scatter(df, x="Year", y="TotalRenewableEnergy", color="Country", title="TotalRenewableEnergy over the years")
fig.show()
```

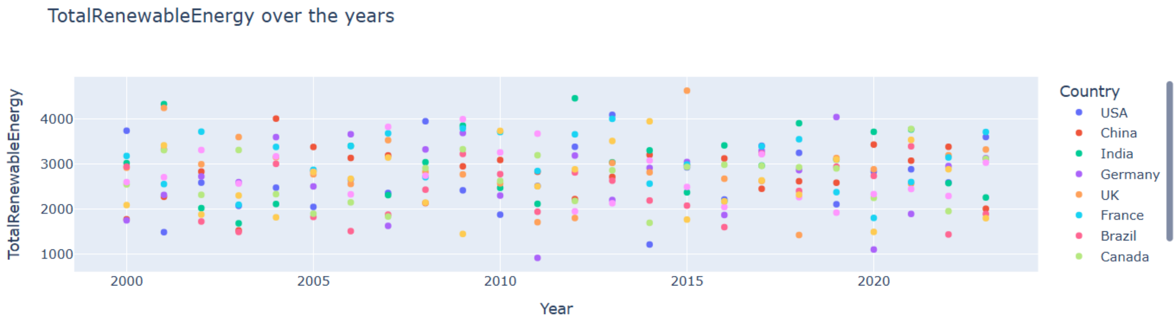


***Figure 2:*** *production of total renewable energy per country over the years*

The country with the highest renewable energy production varies from year to year, showing noticeable fluctuations. However, Brazil consistently ranks among the lowest in terms of total production. The peak in total renewable energy output occurred in the UK in 2015, followed by India in 2013 and again in 2001. The lowest recorded production was in Germany in 2011. In 2019, there was a significant gap between Germany and the other countries, highlighting a marked difference in energy output for that year.

## 2.2. Bar chart of energy production per country

A bar chart is a good choice when comparing data across different categories. The bars make it easy to visualize the differences at a quick glance. The data have been grouped per country and the sum of the numeric features has been calculated using the code line:

```
countrygroups=df.groupby(by="Country").sum(numeric_only=True)
countrygroups
```

And are visualized in a bar chart with the following code line:

```
fig=px.bar(data_frame=countrygroups,
    y=["HydroEnergy","WindEnergy",    "TotalRenewableEnergy",    "SolarEnergy",
"OtherRenewableEnergy", ],
    barmode="group",height=600,title="Mean Values by country")
fig.show()
```

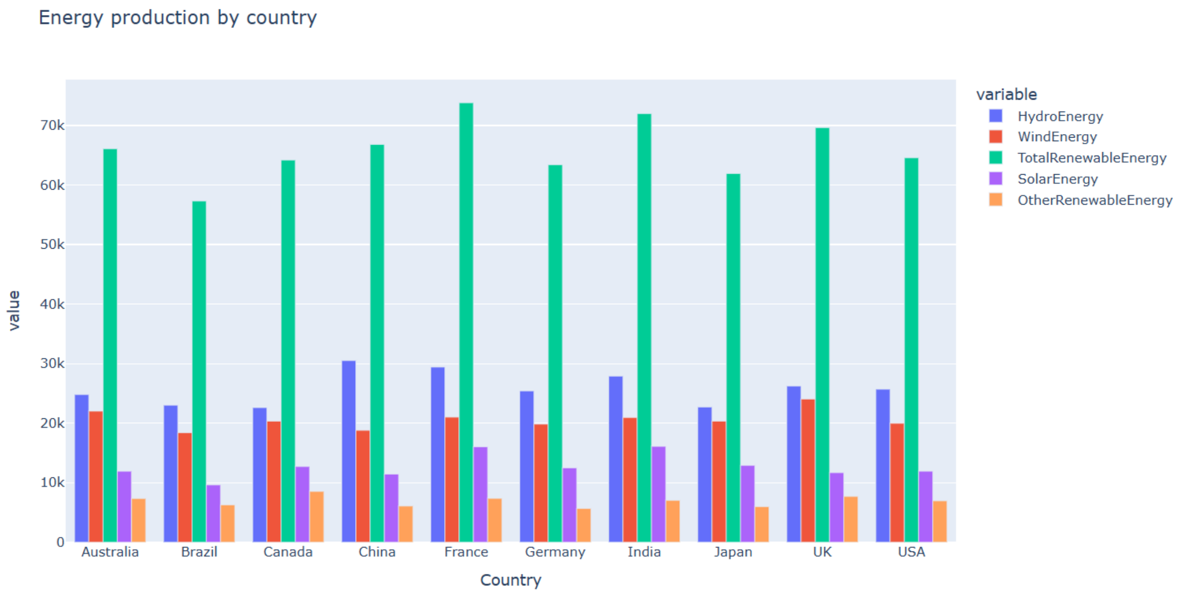


***Figure 3**: Histogram of energy production per country between 2000 and 2023*

The bar charts in the figure 3 shows that between 2000 and 2023, France produced the most renewable energy, followed by India. In contrast, Brazil had the lowest production of renewable energy. The United Kingdom produced the highest amount of wind energy, followed by Australia. China was the leading producer of hydro energy, followed by France.

### 2.3. Bar chart of energy production over the years

In this case for the visualization of the changes over the years, data have been grouped per year and the sum of the numeric features has been calculated (see table 2) using the line of code:

```
yeargroups=df.groupby(by="Year").sum(numeric_only=True)
```

*yeargroups*

The code below was used for visualization.

```
fig=px.bar(data_frame=yeargroups,
    y=["HydroEnergy","WindEnergy",      "TotalRenewableEnergy",      "SolarEnergy",
"OtherRenewableEnergy", ],
    barmode="group",height=600,title="Yearly Renewable Energy Production")
fig.show()
```

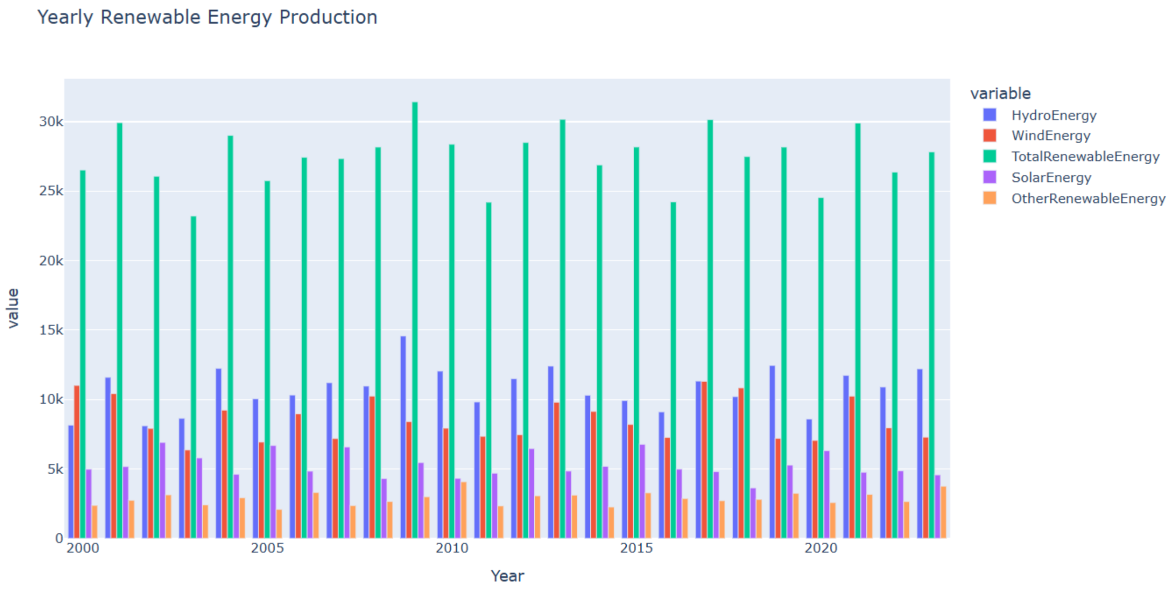


***Figure 4**: bar chart of the energy production over the years*

We observe in the bar chart above fluctuations in total energy production over the years, with the peak occurring in 2009 at around 32000 gigawatt-hours (GWh), while the lowest production was in 2003. According to the chart, hydro energy contributed the most to renewable energy production, followed by wind energy. There is no increasing or decreasing trend overall.

## 2.4. Heatmap to visualize the correlations between energy features

A heat map is a two-dimensional representation of data used to investigate the correlation between variables. The correlation coefficients, ranging from -1 to +1, are displayed in the squares and measure the linear relationship between two variables. The map has been created with the code below:

```
selected_cols=df[["SolarEnergy","WindEnergy","HydroEnergy","OtherRenewableEnergy","TotalRenewableEnergy"]]

# Calculate the correlation matrix

corr_matrix = selected_cols.corr()

plt.figure(figsize=(7, 5))

sns.heatmap(corr_matrix, annot=True, cmap="coolwarm", fmt=".2f", square=True)

plt.title("Correlation Heatmap of renewable energy production")

plt.show()
```

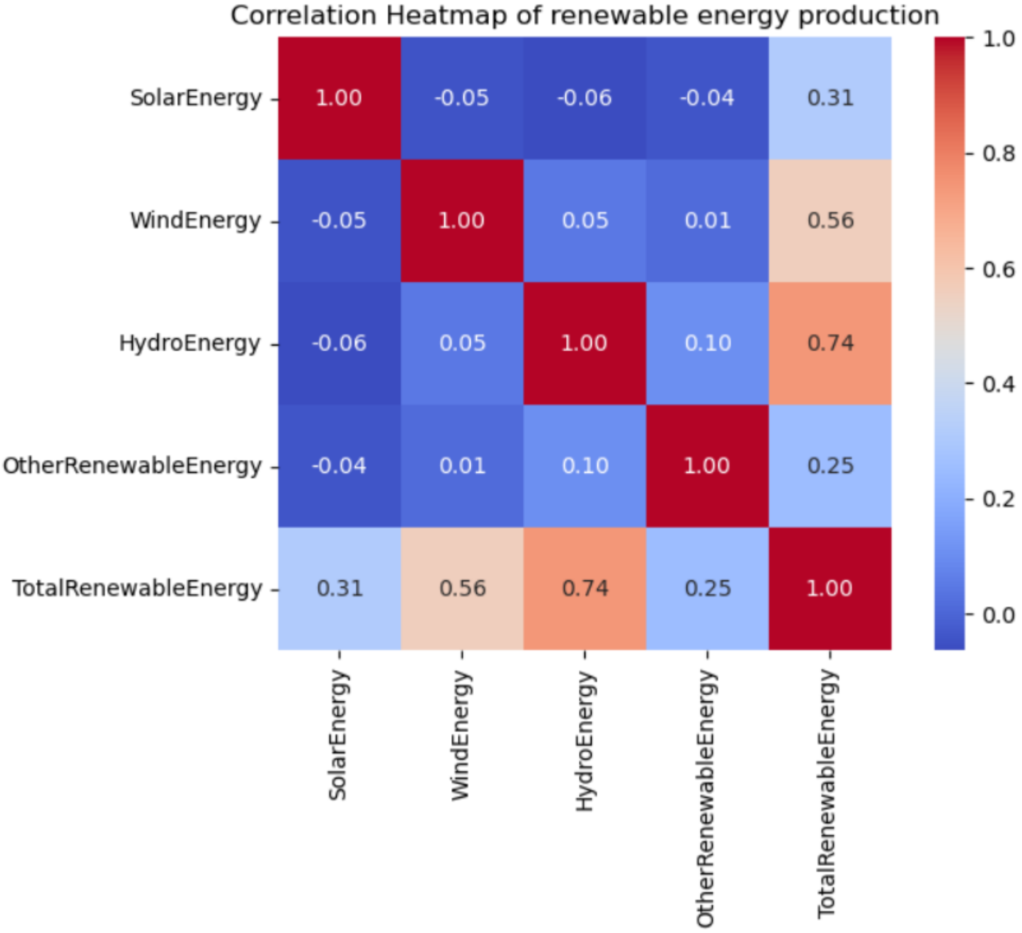


***Figure 5:** Heatmap of the different energy sources*

The map in figure 5 shows no significant correlation between solar, wind, hydro, and other renewable energy sources. The correlation coefficient between solar and total renewable energy is weak (0.31). Wind and hydro energy show moderate correlations with total renewable energy, at 0.56 and 0.74 respectively. These relationships will be further investigated using relational plots.

### 2.5. Relational plot for further investigation of the correlation

The relational plot in figure 6 illustrates the relationship between the hydro energy and total renewable energy. I then included the contribution of wind energy as an additional variable. The graph was generated using the following code:

```
sns.relplot(data=df, x="HydroEnergy", y="TotalRenewableEnergy", hue="WindEnergy")
plt.show()
```

The resulting visualization is shown in the following figure 6.

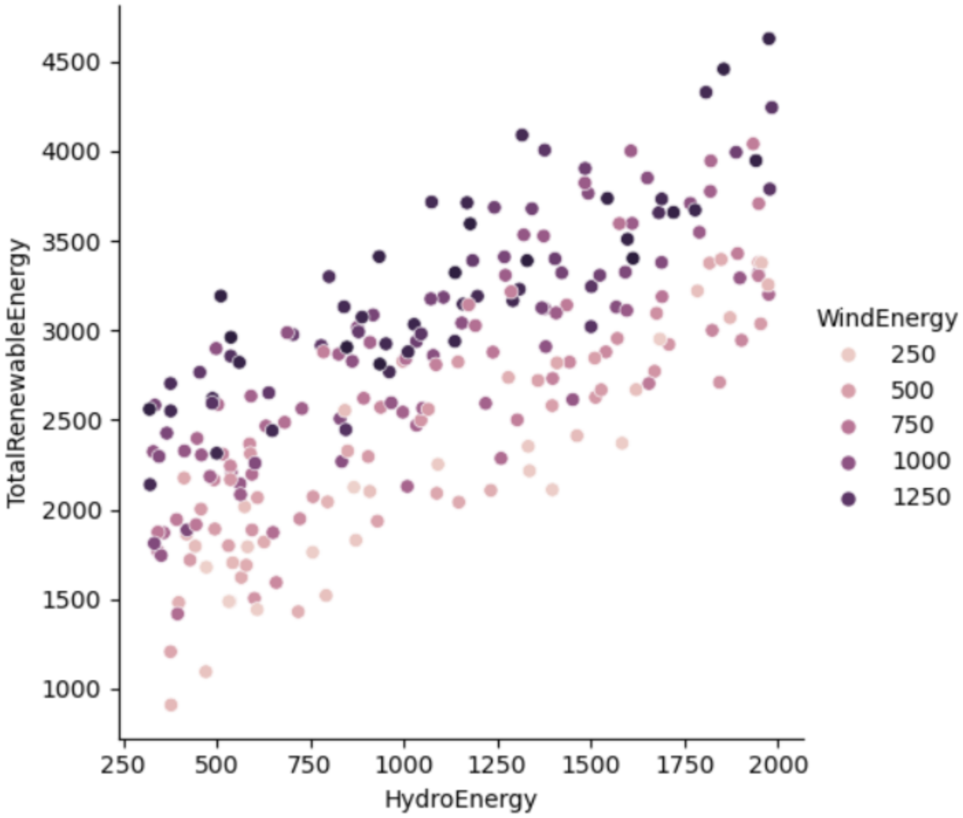


***Figure 6:*** *relational plot between Hydro and total renewable energy*

It can clearly be seen that higher values of hydro energy are associated with higher values of total renewable energy, which explains the positive correlation of 0,74 between these variables. We also observe that wind energy has a significant impact on total renewable energy, the higher its production, the greater the amount of total energy produced.

# Conclusion

in summary, this assignment offers an excellent opportunity to apply and reinforce the concepts learned in data visualization. By working with a real-world dataset on global renewable energy production from 2000 to 2023, I practiced essential steps of exploratory data analysis, including data cleaning and the creation of meaningful visual representations. These steps are crucial not only for understanding the data itself but also for being able to communicate insights effectively to a wider audience.

This analysis provided a comprehensive overview of the “Global Renewable Energy Production (2000–2023)” dataset and demonstrated key steps in data exploration, cleaning, and visualization using Python. It demonstrates the value of data visualization in uncovering patterns, relationships, and trends in complex datasets. Scatter plot showed significant variation among the counties, the bar chart grouped by country revealed that France led in overall renewable energy production, followed by India, the bar chart grouped by year showed fluctuations in energy production over time, peaking in 2009. The heatmap revealed that hydro energy had the strongest correlation with total renewable energy (0.74), while wind energy had a moderate correlation (0.56). The relational plot confirmed that higher hydro and wind energy production levels were associated with higher total renewable energy output, reinforcing the correlation results and providing a more intuitive visual understanding of the relationships between these variables.

Ultimately, the goal of this exercise is not just to practice data visualization techniques, but also to develop a deeper understanding of how data can inform solutions to real-world challenges, this has strengthened my practical skills in exploratory data analysis and visualization using Python and increased my confidence in interpreting and communicating data-driven insights.

,

# References

https://www.kaggle.com/datasets/ahmedgaitani/global-renewable-energy/code

https://github.com/toukoalvine/Visual-exploration-project (GitHub repository)

https://campus.datacamp.com/courses/introduction-to-data-visualization-with-plotly-in-python/introduction-to-plotly-1?ex=1

https://seaborn.pydata.org/generated/seaborn.heatmap.html

Rainer Schnell (1994) Graphisch gestützte Datenanalyse Verlag Oldenbourg, München 1994 ISBN: 978-3-486-23118-2

https://maucher.pages.mi.hdm-stuttgart.de/python4datascience/09MachineLearningInaNutshell.html

McKinney,W.(2012). Python for Data Analysis: Data Wrangling with Pandas, NumPy and IPython. O'ReillyMedia