# UNSDG & Climate Change Process Book

UNSDGViz: Lorenzo Rovati, Sophia Ly, Julia Wälti

# 1 Introduction

Climate change is one of today's most pressing problems. Anthropogenous activities such as burning fossil fuels or deforestation lead to rising temperatures. This on the other hand has environmental impacts such as sea level rise, storms and wildfires. In order to reach the goals of the Paris agreement, countries have to prioritize environmental sustainability in their development strategies [1]. Economic growth is often linked with a higher consumption of goods and energy [2]. Finding strategies on how to decouple CO2 emissions from economic growth is therefore a challenge. Through the visualizations of this project, the relationship between development and environmental sustainability is explored. The website could be helpful to analyze trends to find environmentally friendly policies.

# 2 Methodology for attaining the final data visualization

We started the project by brainstorming about topics that we found interesting and meaningful. Since climate change is something that affects everyone and will be determined by the combined action of the whole world, we choose this broad topic. Afterwards, a more precise topic had to be defined for visualization. We did not want to visualize the consequences of climate change since this has been done a lot already. Something that is deeply interconnected with climate change is sustainability. Sustainability means balancing environmental, social and economic aspects. We believe that a sustainable development is the most effective way of combating climate change. It is also a topic that can be visualized in space and time. After some online research, we came to the conclusion that no website exists that visualizes the relationship between development and sustainability according to our vision. Therefore, we agreed to create a website with the goal to visualize the relationship between country development and environmental sustainability. A sketch of the methodology process can be seen in the figure 1 below.

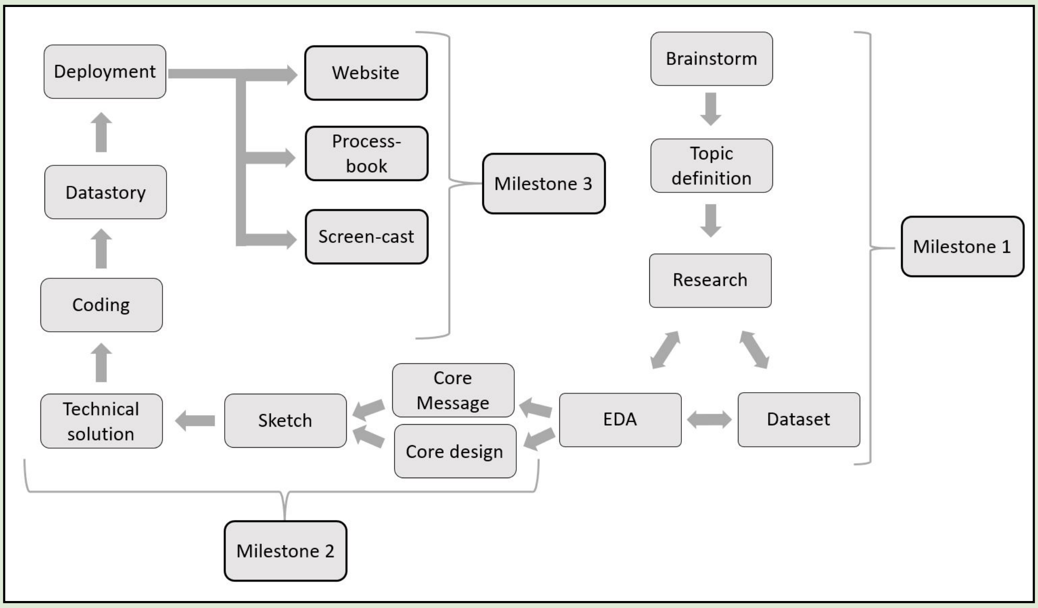


**Figure 1:** Process flow of the project, spanning all Milestones.

After the topic definition, we implemented an iterative process of research about the topic, related work and interesting visualization styles. For each suitable data-set we did an exploratory analysis. We had to condense the available information to the core message we wanted to communicate to the user of the website. In addition, a core design had to be defined. Since the main focus of the project is environmental sustainability, we decided to choose a color palette representing the topic. Namely blue and green, corresponding to the earth's color. We also use accent colors from the UNSDG goals color palette. Based on the knowledge gained from the research, we discussed about visualization design and used sketches to communicate our ideas. To work more efficiently, we broke the project down to independent pieces. With this, each member can work on an isolated visualization.

In order to build the visualizations, each member researched about the technical solutions by looking at available tools and revisiting the lecture notes. For each visualization, the basic skeleton is first implemented, followed by some interactive aspect such as sliders and drop-down menus. Lastly, we brainstormed about extra ideas that enhance the visualizations or provide more insight about the data.

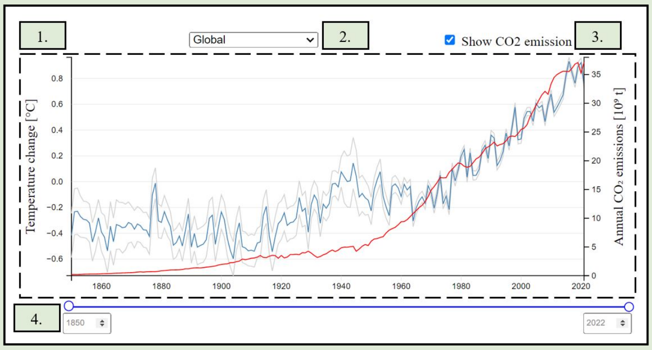


**Figure 2:** Steps of building a visualization shown on the example of the climate change time series. Content in the dotted box (number 1) is the minimal viable part of the visualization.

The final website is implemented by combining all independent visualizations and by writing the data story. To check the usability and impression of the website, we asked for feed-backs from peers and took them into consideration for final touches such as minor choice of color and placement. In addition to the website, a screen-cast is deployed to better explain the usage of the website.

## 2.1 Data preprocessing

At first, possible data-sets representing environmental, economic and social sustainability were found. Namely information about UNSDG indexes, $CO_2$ emissions, literacy rate, GDP, education rate and air pollution. Those datasets were merged according to the process described in Milestone 1. Outliers defined as values exceeding the $75^{th}$ percentile plus 1 standard deviation were converted to NaNs. Through exploratory data analysis an unexpected negative correlation between renewable energy share on the total energy consumption to the proportion of population relying on clean fuels was found. The UNSDG-index dataset does not further explain how the renewable energy share on the total energy consumption is calculated. After further investigation, we decided to replace the information about renewable energy in the UNSDG-index by the percentage of electricity covered by renewable since its information is in accordance with a validation dataset about Renewables per capita (kWh - equivalent).

For both heatmap visualizations, the data needed had to be extracted first. Since climate change and one of its main driver $CO_2$ is the focus of this website, only countries with information about this greenhouse gas is kept within the data-set. The features marked as interesting in Milestone 1 were then extracted. In the exploratory analysis we saw that the measurement interval of features are not the same. Therefore, we interpolated each feature between the measurement dates, grouped by country. Since no extrapolation was implemented, the start of the measurement varies per factor. Therefore, we split the data into four periods and calculated the percentage of missing values for each value. For each period we only kept features that had less than 30% missing values, since missing values can distort the true correlation. Lastly, the correlation of the remaining features was calculated.

For the bar plot of the SpearmanR result, we need at least three values to not be null from each distribution that will be compared to the $CO_2$ distribution. Thus we used the dataset created in the first preprocessing and selected each distribution that satisfies this condition. Then, we computed the Spearman R in Python. To make the visualization easier we created two very specific csv. The first one contains the indexes and their averaged Spearman R value. The second one contains the indexes and two string with the list of country and the values that correspond to the top and last 5 country. We chose to do this because it is easier to handle data transformations in Python than Javascript and that we knew exactly what we needed.

# 3 Discussion of visualizations

In order to be able to work independently, we ensured that each visualization is made in a container-div.

## 3.1 Hero image

We decided to implement an hero image as our website's landing page. Our goal was to captivate the user, ignite curiosity while also communicating the broad topic we plan to explore later on. Since the main motivation of our project is to talk about sustainability, we decided to show the beauty of our planet. We believe that this could invoke the intrinsic motivation to protect our planet in order to keep its beauty. Since we are based in Switzerland, we decided to go for a mountain scenery.

To enhance the user's experience we decided to add some dynamic parts by adding a parallax scroll effect to give a 3D impression. As a first step, we drew clouds, mountain and forest silhouette and planned the parallel effect as seen in figure 3. A static container was created and for each layer a div was added. By adjusting the z-index, we created the first start image. A scroll-event listener was added in Javascript. The goal was to imitate the effect of different scrolling speed, where the background moves slower compared to the foreground. Therefore, the div containing the layer with content furthest away was fixed to the top of the page. The effect of slow scroll for the remaining background layers was implemented by adjusting the "top-position" of the corresponding absolute-positioned div compared to the actual scroll position. Once the foreground layers div-bottom is reached, all other layers are moved back to the top of the page and the foreground-layer's and title's z-index is set to 0 and made transparent to show the following websites content.

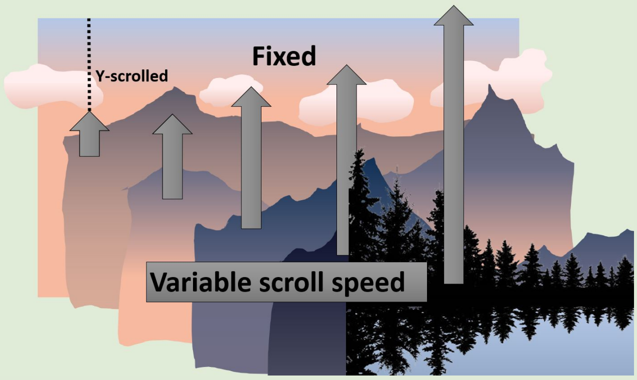


**Figure 3:** Planning of achieving the parallax effect.

After implementation however, the color scheme seem to not fit with the rest of the website content. Therefore, the layer-images were replaced with mountains silhouettes in the color of the UNSDG logo. The graphics were generated by taking mountain images from the web and adjusting the colors. Lastly, the transition had to be adapted and instead of changing the foreground-layer's z-index, an additional div was added right after the foreground layer with the same color as the background. In order to not let the title abruptly disappear, a delay of the z-Index change was implemented with setTimeout() function.

## 3.2 Climate change time series

The first graph should introduce the user to the topic. We therefore designed a visualization that shows the temperature change relative to the 1961-1990 average temperature on one axis and annual $CO_2$ emissions on the other axis. This way, the user can see the correlation between climate change and the greenhouse gas $CO_2$ emissions. In order to show more detail, we implemented an interactive range slider with which the user can choose the time period and a drop-down menu, with which the user can choose the interest-area. Lastly, a checkbox was implemented, in order to show the $CO_2$ emission. The biggest challenge of the figure was adding the interactivity. First, a static graph was generated, containing the temperature change with upper and lower confidence intervals and the $CO_2$ emissions. The year column was read as date-value. This ensures that potential missing values does not distort the graphic axis. A drop-down menu defining the area, a range slider with input number-box defining the years depicted and a checkbox defining the $CO_2$ depiction enable an interactive data exploration. In order to be able to update the graph on change of widget, the code for generating the static graph was rewritten as a function including removal of all child elements on function call. A main challenge was the dependency of the graph to all widget inputs.

As a first step, the temperature data was filtered according to the dropdown-menu input. The dropdown-menu was appended to an independent div through D3.js. Its values were found by extracting the unique values from the data. On change of the selected drop-down value, the data is newly filtered and the graph is updated.

The slider defining the years depicted was implemented in a different container-div. With html's input tag, four input fields were generated. An upper and lower range slider, respective number-box in which the range number can be directly typed or tuned with an up and down arrow. To ensure that it is properly displayed in multiple browsers, we defined the corresponding CSS code for WebKit rendering engine and Gecko rendering engine. First, a function had to built to convert the input values from string to integer. For both the upper and the lower slider, a function was created to ensure that the lower slider never overtakes the upper slider and to make sure that the number-box displays the value of the slider. Respectively, a function for both number-box ensure that the slider value displays the input through the input box. A function updating the range to be colored was created, and called on each update through the functions previously mentioned. Lastly, a function is created which is called on user-input and filters the area-filtered data according to the time range. It calls the slider, respective number-box, updating-function and the chart creation function.

Lastly, a checkbox element is created with html's <label> tag. A global boolean variable was initialized to keep track if the $CO_2$ emissions should be displayed. It is updated on change of the checkbox element. Only if the checkbox is checked, the $CO_2$ emissions and the corresponding second y axis is visualized. Parallel to the temperature line-chart, the area-filtered data is filtered according to the slider values and the graph updated. In the beginning, the $CO_2$ data was diveded by $10^9$ to make the visualization more clearer.

## 3.3 Introduction to carbon cycle

In order to give the reader more context about the subject, an introduction about the carbon cycle is given. Through this the greenhouse gas effect, the driver of climate change, can be understood. To make it visually more appealing and interactive, an image of the carbon flows is added. For each process causing a material flow, two divs were added. One with the process name and one describing the process. In order to be able to switch between the process name and the process description, a click event was added. A variable storing the number of clicks was added to keep track if the click is an even or odd number. For the first case, the process name was omitted with opacity setting and the z-index set lower compared to the description. For the later case, the description was omitted and the z-index set lower compared to the process name. This ensures that the div which is visible, is clickable. The position of all processes were set manually to match the image position. During the first iterations of this visualization implementation the divs containing the carbon processes were placed using relative units with respect to screen or viewport size. We realized only towards the end of the process that this choice caused the viz to break for certain screen types and browser window dimensions. After some deeper research we were able to define the position of the boxes as absolute values (percentages) with respect to the underlying image, so that the viz could be more adaptive to various devices.

## 3.4 Presentation of the indexes

Towards the end of the project, we realized that we never introduced the UNSDG. Hence, before going into detail about the analysis, we wanted to present them. To do so, we used a visualization of all of the images of the goals from the UNSDG program. Then, we divided all of the features (indexes) from our dataset into the different goals. To have some interactivity, we decided to use click events on the images of the goals to show the associated indexes. For this, we used D3.js and basic html and javascript. We use the same technique as the carbon cycles clicks to be able to switch between text and image. To have some stability in the position and have the text always be at the good height, we used two images: one with the goal and one of a unified background. Then, we change the content of the html div depending om the clicks. The bland image is of the same color as the original goal to keep a cleaner and cohesive look. This part serves as a transition between the introduction and the rest of the website.

## 3.5 Correlation heat-map

While "Plotly" is a strong data visualization tool that can generate a heat-map correlation matrix directly from Python, we decided to use D3.js instead. Partly because we already use this library for other visualizations and do not want to import "Plotly" in addition, which comes with a big bundle size where only a small part is used for this visualization. Another reason is that the JavaScript code generated is very lengthy and without white-space, which makes customization much more complex. Therefore, we faced the challenge of transforming the correlation-matrix to a structure that facilitates visualization in D3.js. We solved this by melting the matrix dataframe, followed by re-indexing. With this we generate a CSV-file containing the x-axis feature, the y-axis feature and the corresponding correlation value for each period.

First, the basic static visualization of a correlation heat-map was generated by using D3.js scaleBand function for building the axis. For each correlation, a rectangle with color representing the value was added to the SVG. We only wanted to visualize a correlation once, and not show the inverse pair to make the design easier to understand (e.g. Only "$CO_2$ emission -Growth rate" and not additionally "Growth rate- $CO_2$ emisison". Therefore, we had to go back to our Python code and apply a mask to the upper triangle of the correlation matrix. Afterwards, we filtered the empty values out in the Javascript code. The colorbar was added by appending a SVG with a rectangle. This rectangle should be filled with the same color gradient as the one created for the heatmap. We faced the problem that the gradient is filled according to the y-axis. Color-bars however are filled from bottom to top. Therefore, we had to generate an additional color-gradient, which is the inverse of the displayed color gradient. Lastly, a scale label is added to the colorbar with d3.axisRight function.

To add interactivity a div showing the details value of the correlation is implemented. With the functions "mouseover", "mousemove" and "mouseleave", it was possible to make the div appear as a hover element which moves with the mouse position. At the end, a drop-down menu was added to display the correlation for different periods. We chose to define the correlation heat-map figure size as a constant value with varying rectangle size because this results in a calmer transition between the figures.

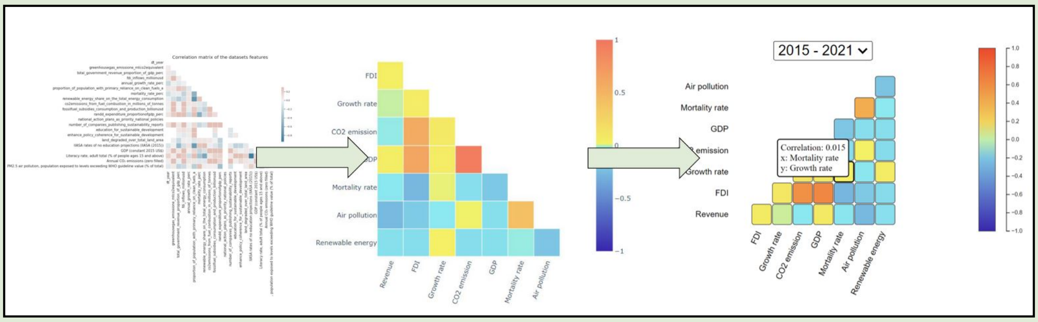


**Figure 4:** Process of building the visualization. From the Python result of all features, to the visualization of selected features built with Ploty to the final visualization built with D3.js including interactivity.

## 3.6 Detailed information of individual countries

While the goal of other visualizations is to give insight on the data on a global scale, we also wanted to take advantage of the detailed datasets we were working with. Hence, we decided that it was necessary to include access to data in a more focused way. Abandoning the time frame, the countries become the focus of this visualization.

When first sketching this visualization, the main idea was to give the opportunity to easily visualize correlation between CO2 emissions and other features inside a country. This would be reached by using a scatterplot plotting CO2 emissions against another feature, and then showing the correlation between the two through a linear regression. This design was promising but it was lacking an interesting interaction aspect (the ones we could come up with were already used elsewhere, e.g. a dropdown box used to choose a feature to visualize).

After looking for inspiration through common D3.js visualizations, we decided to take a step back and rethink the use of the scatterplot. The final version of the visualization embodies this second iteration, in which the scatterplot shows all (normalized) features compared against CO2 emissions for the selected country. This brings a lot more data to the plot, which is something planned in order to invite the user to interact. By hovering on the scatter plot, the visualization reacts highlighting all data points linked to the feature of the point currently being hovered. This is to clear up the plot and allow the user to draw insight from viewing a single feature at a time.

The plot being settled, we just needed an easy way for the user to select a country. Since we already implemented an interaction system through world maps (described in the next subsection), we thought about reducing the cognitive charge of the user by using the same kind of map. By having access to a world map with zoom-in and zoom-out functionalities, the user can easily click on a country to select it and visualize on the scatter plot the corresponding data.

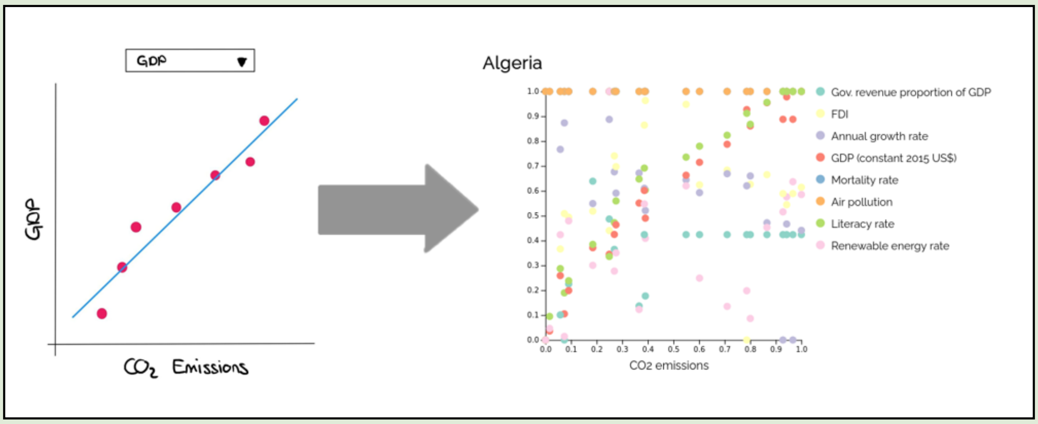


**Figure 5:** Country detailed information. On the left, the original idea sketched showing a single feature and the linear regression. On the right, the final version showing various features at the same time.

### 3.7 Spatial and time visualization of countries indexes

After getting insight on individual countries, the next step is to compare indices of individual countries against each others. We hence decided to reduce the level of detail of the data, to bring back the time frame, and to add the spatial frame. The design of this visualization has been straightforward, we had a precise idea of what we wanted the end result to be. The best and easiest way to compare data across countries is through a world map, where data is channeled through the color of a country. It was important for us to keep this visualization closely related to the previous one, so we decided to show the same features present in the detailed view of individual countries. We take advantage of the available time frame to set up an animation of the data through the years, which can be completely controlled by the user through a slider and a play button.

The main challenges for this visualization were technical ones. First of all, we had to get familiar with GeoJSON[1] data used to create SVG paths of the countries, and then with D3.js functionalities for creating and showing maps. A second challenge was to decide how to import the data in order to be shown on the map. A more straightforward solution would have been to use data in a csv format since it was readily available from the processing of milestone 1. The bindings between map and csv data appeared in the end to be non-trivial and overly complicated from a code perspective, so we decided to opt for a different solution. We invested some extra time embedding data for the various countries directly in the GeoJSON file used to draw the map. This greatly reduced the difficulty of showing the data on the map and more in general felt like a more natural approach to the issue. This came at the cost of having a relatively large GeoJSON file (around 2MB) which might impact loading time of scripts for devices with slow connectivity.

This visualization should have initially included the one exposed in the previous subsection. The user would have been able to click on a country and a detailed view of the corresponding data would have popped up below the maps. After some initial prototyping we were not at all convinced by this choice. It appeared to bring too many features on a single visualization, over complicating it. Moreover, the space taken by the maps was forcing the detailed view to partially appear off-screen, something we thought not to be pleasant from a browsing experience point of view. It was hence decided to remove the feature from this visualization and delegate it completely to a new one.

It is important to mention that the choice of the map has been a challenging task. The ideal solution would have been to find a map including the exact same countries of the datasets we were working with, but we could not find a free access to such a map, and designing one from scratch would have been too time consuming. We sadly had to settle on a free GeoJSON file provided online, which does not reflect borders and sovereignties globally recognized. This also created the need to process the

[1] `https://en.wikipedia.org/wiki/GeoJSON`

data once again in order to be able to bind the data from the processed datasets to the countries available in the map.

### 3.8 Comparison of the distribution of the CO2 emissions and other features

The last graph that we created is a presentation of SpearmanR results. This quantity allows to compared the distribution of two variables. We felt that it was an interesting result as the previous statistical metric showed only looks at linear relationships.

The SpearmanR is computed through the whole timseries for each features and each countries. We then average the results by features. This shows us the features that have evolved the same as CO2 emissions in each country and that could for instance be related. For instance, when a country's GDP increases, so does its CO2 emissions. This further confirms the results that the two are connected. We used Python to created the results and then exported them to csv to be used in javascript.

The most meaningful result is the aggregation by feature. Hence it is what we wanted our figure to show. However, this felt a little too superficial, hence to add more detail we added a tooltip that shows more details about the different countries results for a given feature. At first we wanted to show the results for all the different locations. However, we realized that with more than 100 countries in the dataset, it would be too much. We compromised for a list of the top 5 and last 5 SpearmanR values and the associated country.
We used d3 to do this plot as it offers a nice tooltip tool. The rest is a simple barplot and parsing of the csv files. The color palette was chosen to resemble some of the UNSDG goals.

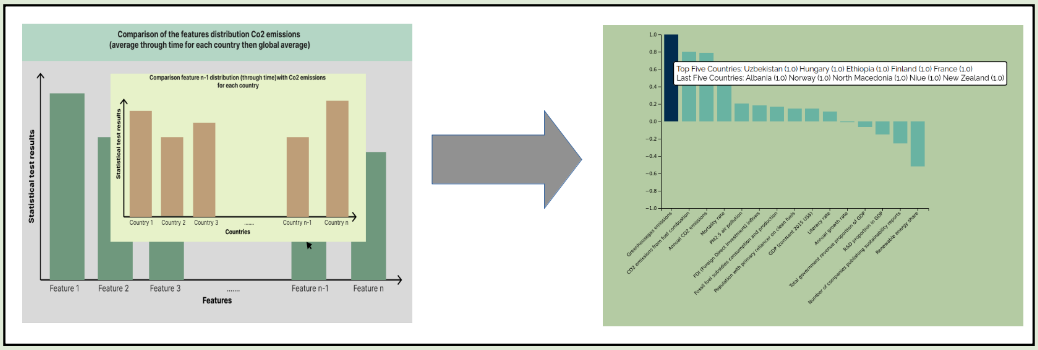


**Figure 6:** Initial idea for the Spearman R plot and actual visualization

## 4 Peer Assessment

At the start of each milestone we held a meeting to discuss the overall strategy and broke down the task to smaller chunks in order to assign them to each member. While we worked independently, regular updates were given to each other and if necessary meetings were held to solve technical or conceptual questions. Below is a breakdown of the project completed by each member:

Lorenzo Rovati:

- Research on interesting topic and dataset
- Build initial website with basic skeleton
- Website content: Spatial representation through world heat-maps, detailed information, final structure

Sophia Ly:

- Research on interesting topic and dataset
- Research about related work
- Data preprocessing
- Website content: Hero-image, climate change time series, introduction to carbon cycle, correlation heat-map

Julia Wälti:

- Research on interesting topic and dataset
- Exploratory Data Analysis
- Data preprocessing
- Website content: UNSDG Indexes presentation, statistical comparison of each features with CO2 Emissions

# 5 Conclusion

At the beginning of this project, we were really excited to have found a dataset and problematic that interested us all and that was meaningful. After doing the first exploration of the data, we realized that it was more complex than we initially though given the number of features, the spatial and temporal components and the missing values. We added new datasets to make up for the missing values in the original one. This however was not as easy as initially thought since the datasets do not contain a consistent key to allow perfect merging.

Then it was time to start visualizing. We worked separately on visualizations that have the different constraints so we used different preprocessing and therefore final dataset for each plot. If we had to do it again, we would spend more time discussing this together and creating one dataset that works for all the visualizations.

Since the topic is very scientific, it was a challenge to do something creative and visually exciting. In the beginning we focused mainly on creating plots that hold interesting information that we want to convey. This however resulted in a website heavy on scientific plots. In the end we added two more visual data-story type of plots to communicate our ideas and provide a gentle introduction to the scientific plots. Given more time, we would have liked to add more visually interesting plots in order to better balance the scientific ones.

Lastly, some visualizations could have been improved if more time was available. The parallax effect in the hero plot does not always run smoothly, since the positional settings are computed on scroll. Some interactions with D3.js elements based on mouse hovering do not always behave as expected: for unknown reasons in some cases the mouse pointer switches to a text highlighting tool when hovering on plots.

All in all, there are still some improvement ideas that could be implemented with more time. But overall we are content with having created our first cohesive and interesting website. We are happy to have spent time on something we are passionate about and to have been able to work alongside peers with the same motivations and goals.

# References

[1] H.-O. Pörtner, D. C. Roberts, H. Adams, C. Adler, P. Aldunce, E. Ali, R. A. Begum, R. Betts, R. B. Kerr, R. Biesbroek *et al.*, *Climate change 2022: Impacts, adaptation and vulnerability*. IPCC Geneva, Switzerland:, 2022.

[2] P. Benoit, "Energy and development in a changing world: A framework for the 21st century," *Center on Global Energy Policy*, 2019.