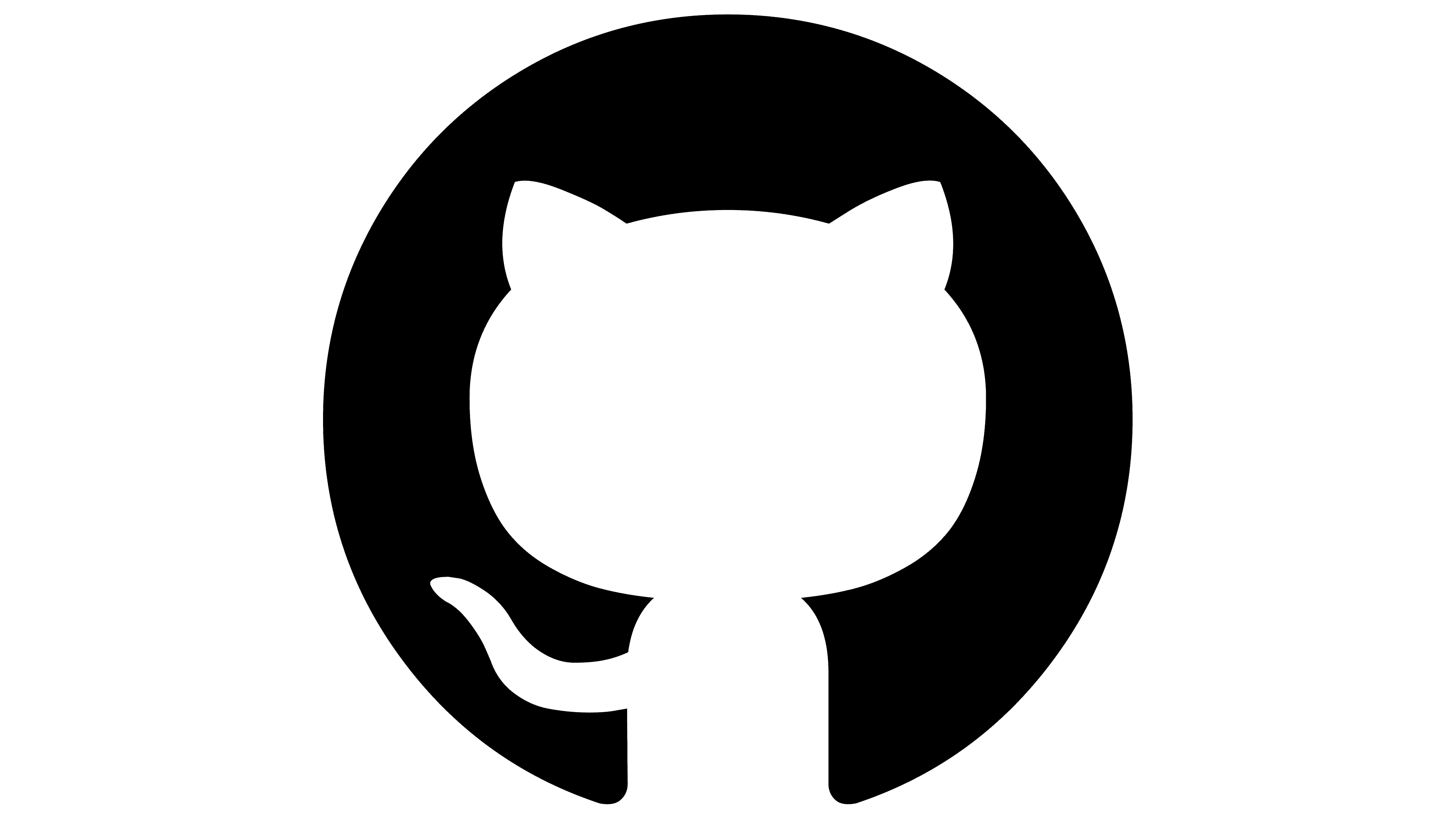
Ecosystem Modeling Laboratory – Project

[](https://github.com/Andy-Lewis-Sapner/EML_EcoCoders)[](https://colab.research.google.com/drive/1CoNpPhQpgkgO21lmb2uey-zbLuap1cWw?usp=sharing)Analysis of the Kishon River Ecosystem in terms of Measurements and Observations

Osher Ben Zazon (212240238), Andy Lewis Sapner (213298524).

Introduction

The Kishon River, which is the second largest coastal river in Israel, is highly polluted due to its long history of contamination [1]. This is one of the reasons the river is an excellent choice for analyzing its unique ecosystem. We decided to focus on the question of what the relations between specific measurements of the river and the total amount of animal and plant observations are there.

Overview of Existing Work

There are not many research papers that dealt with the Kishon River and analyzed produced data from the research.

Certain research [2] checked how nitrogen moves around near Israel's coastline and found that most of it comes from ocean currents, a bit from rain and air, and some from rivers (mainly sewage and runoff). Surprisingly, all that extra nitrogen did not really shake up the ecosystem much - it barely changed how much stuff grows in the water or which tiny critters dominate, except for some small local patches. Unlike other places where rivers are the main polluters, in this case it is the sea itself, bringing most of the nutrients.

They built a nitrogen budget model combining field measurements (nutrients, chlorophyll, productivity), seasonal sampling (2018 – 2019), and existing data. Used water column profiles, offshore vs. coastal comparisons, plus cross-correlation analysis (Pearson's r) to compare coastal vs. pelagic conditions. They estimated fluxes from rivers, atmosphere, ocean currents, and sediment, to balance sources vs. sinks.

Another research [3] kept checking for 20 years the Kishon estuary and even though pollution levels dropped a bit, the water's still not doing great - most of the time it's rated as bad or just so-so. The nutrients, algae levels, and oxygen often cross the danger lines. Because the water sits there for a long time without flushing out, they say we must look at both chemistry and biology together to really judge how the place is doing.

They ran forty field sampling campaigns over 20 years at 3 fixed stations. Measured nutrients (nitrate, phosphate, ammonium), chlorophyll-a, dissolved oxygen, pH, salinity, and temperature using in-situ sensors and lab analysis. Used standard threshold benchmarks (EU, NOAA) to classify ecological status. Nutrient measurements done with automated analyzers, biological indicators included toxic algae counts.

In another research [4], it was stated that the Kishon got trashed starting in the 1950s – when water was sucked away for farming, and the river ended up full of sewage, industrial junk, and farm runoff. For decades, nobody really fixed it because the government was more focused on building the country fast. But in the nineties, things shifted: better tech, stronger environmental laws, and a big scandal (when soldiers got cancer after training in the river) finally pushed people to clean it up.

They used government reports, committee documents, legal files, public records, and media coverage. Analyzed long-term policy decisions, legislation, public perception, and technological changes over decades. No fieldwork, this was political/environmental history based on archival analysis.

Around 2002, the Israeli government planned to just dump the industrial waste directly into Haifa Bay instead of the river. All six factories around Haifa were breaking pollution rules, spewing heavy metals, toxic chemicals, and nasty acidic stuff into the water. In research done by Greenpeace [5] they said that dumping it into the sea would only make things worse, mess up marine life and fishing, and that the real solution is much tougher regulations, real monitoring, and stopping the pollution at its source.

They reviewed the Israeli Ministry of Environment's ENVIRON report, analyzed old monitoring data (up to 20 years old), and compared it with international environmental standards (BAT, EQS, EQO). Evaluated industrial discharge reports, pollution levels in water, sediment, and marine life. Pointed out data gaps and inconsistencies using desk review, not field measurements.

In the last research [6], the researchers took a deep dive into the mud along the Kishon and found three clear zones: clean up top, super polluted down by the factories, and medium near the sea. Loads of heavy metals (like cadmium, mercury, zinc, and copper) and gunk from oil and fertilizers built up in the lower part. Sometimes floods wash some of this mess out into the ocean, but a lot of it just sits there, trapped in the riverbed.

They have done field sampling of riverbed sediments (1992-1993): collected surface sediments across the river, estuary, and drainage basin. Analyzed heavy metals (Cd, Hg, Zn, Cu, Pb), organic carbon, and minerals using acid digestion, ICP-AES, AAS, and carbon oxidation methods. Ran inter-lab quality controls and statistical tests (Wilcoxon, correlations) to ensure data accuracy.

Methods Used and Findings

There are many ways to analyze the data surrounding the Kishon River, as discussed before. Our approach was to use a couple of different known methods, each of which will allow us to look at the data from a different unique perspective.

The data we have collected by searching it on the internet includes measurements done by the Kishon River Authority [7], and observations done by volunteers who submitted them to the Global Biodiversity Information Facility (GBIF) [8], when the dataset is one that is written under the Israel Nature and Parks Authority [9]. The collected data is around the time of the years 2022 to 2024.

The data, as seen in figure 1, includes species from various families. The common families were Poaceae (13.1%), Fabaceae (11%), Asteraceae (8.5%) and others. When checking the distribution of species based on whether they are aquatic species or not, we got that only 67 out of 4461 (1.5%) occurrences are aquatic. Therefore, in this analysis, we have decided to show findings regarding all species, rather than dividing them into two categories (aquatic and others).

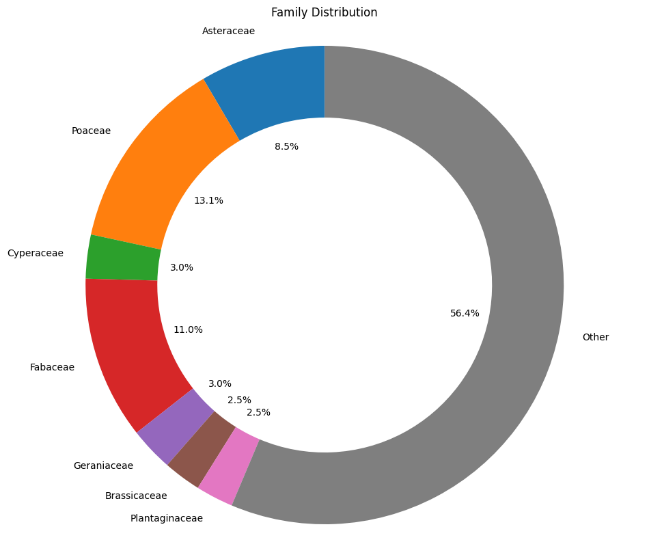


Figure 1: Species Common Families Distribution

The first method we have used involves using PCA. Principal component analysis (PCA) is a multivariate technique that analyses a data table in which observations are described by several inter-correlated quantitative dependent variables. Its goal is to extract the valuable information from the table, to represent it as a set of new orthogonal variables called principal components, and to display the pattern of similarity of the observations and of the variables as points in maps [10].

In figure 2, each point represents a point in time where there were measures of various measurements done in the Kishon River and the total amount of observations done at the same time in the whole river. This figure contains less variables than the ones measured in the river, since there were months when not all measures were measured, so we picked only the ones that are available in most of the months. As seen in the figure, most points are closer to four variables – individuals (the total amount of observations), coliform bacteria, opacity, and specific electrical conductivity. Therefore, we have decided to stick with these four measures when analysing the data.

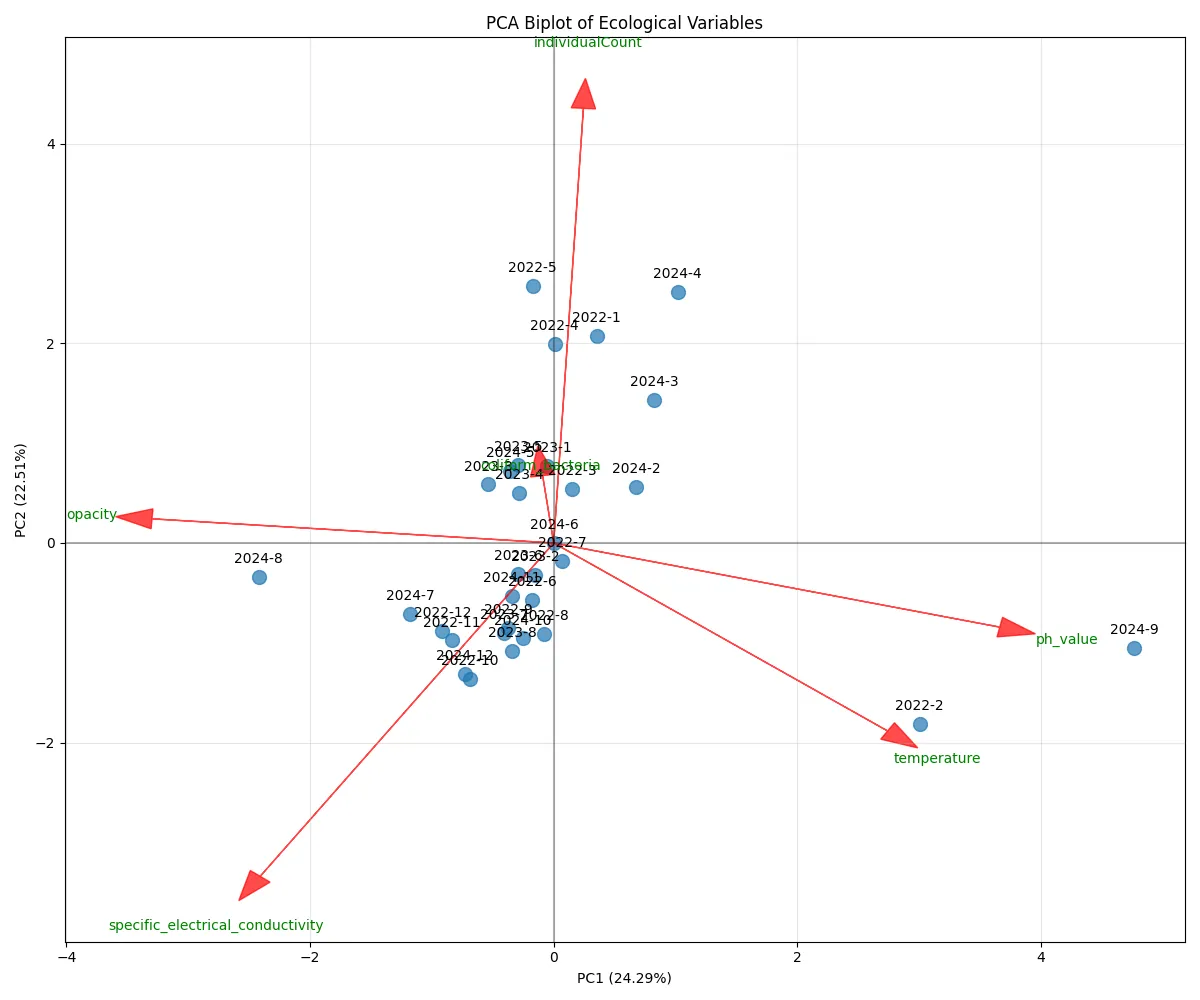


Figure 2: PCA Biplot of Ecological Variables

When we have used PCA, it created a correlation matrix. This matrix can be seen in figure 3. Since we focused only on four variables, we will only look at the relationships between them. PCA allows us to analyze the linear relationship between each two variables.

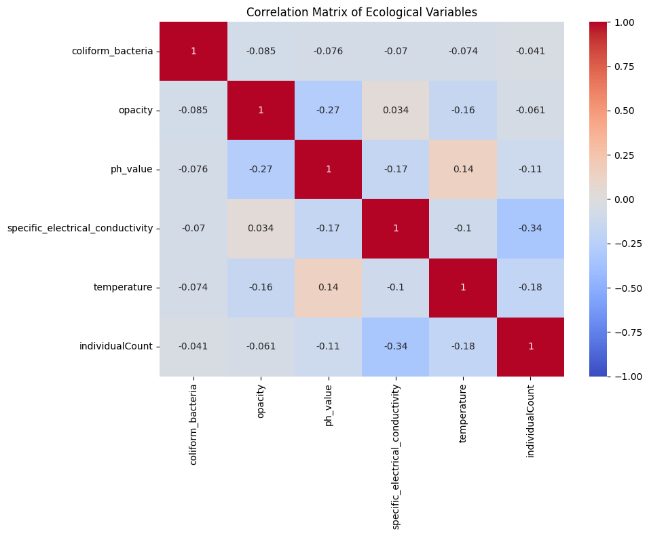


Figure 3: Correlation Matrix of Ecological Variables

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Individuals Count | Coliform Bacteria | Opacity | Specific Electrical Conductivity |
| Individuals Count | - | Negative | Negative | Negative |
| Coliform Bacteria | Negative | - | Negative | Negative |
| Opacity | Negative | Negative | - | Positive |
| Specific Electrical Conductivity | Negative | Negative | Positive | - |

Figure 4: Correlation Table of Ecological Variables

Figure 4 can be derived from figure 3, where each negative value in the correlation matrix is labeled as “Negative” and any positive value as “Positive”. A negative relationship means that the more there is from one measure, the less there is from the other. A positive value means that the more there is from one measure, the more there is from the other as well. The same conclusions can be derived from figure 5, which shows how the measures are changed each month. This figure focuses on the Morad Reservoir Kfar Baruch measuring station in the Kishon River and shows only data from the 2022 year.

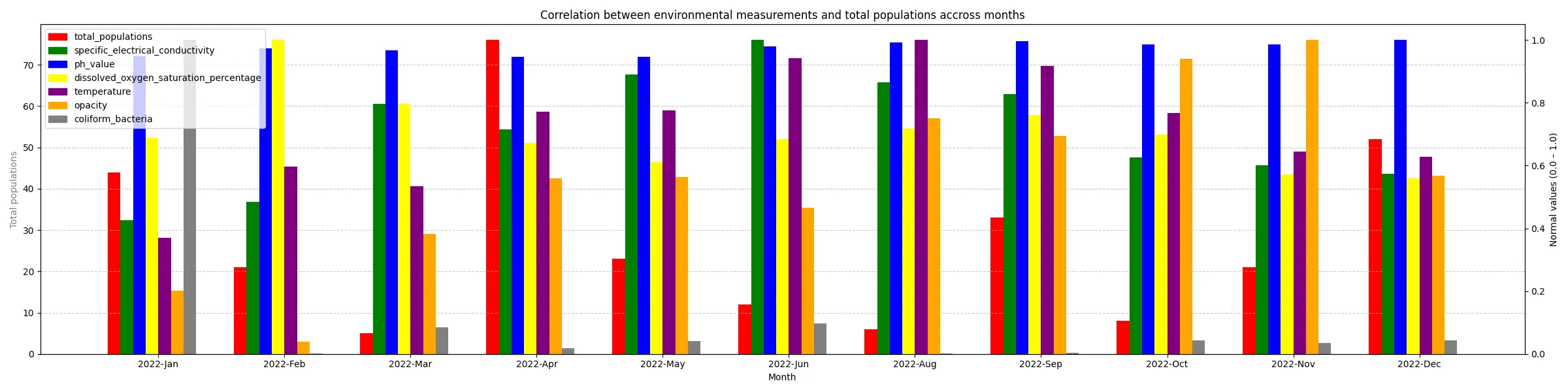


Figure 5: Correlation between Ecological Variables

The second method we have used when analyzing the data is the Kriging method, or specifically the Ordinary Kriging method. Ordinary Kriging is the most widely used kriging method. It serves to estimate a value at a point of a region for which a variogram is known, using data in the neighbourhood of the estimation location [11]. Since there are only 7 measuring stations in the Kishon River, including Morad Reservoir Kfar Baruch, Quarry Station Jalama, Bridge Iri Yagur, Gypsum Mountain, Histadrut Bridge, Julius Simon Bridge and Stone Pier, Kriging can be used to demonstrate what can be the value of each measure in the environment of the river. It can predict the values in each of the points that were not recorded in the environment.

As seen in figure 6, which shows the Kriging visualization of February 2022, with having only seven geographical points, which are the stations, we can predict how will opacity behave in the environment of these points. From this, we can analyse what were the observations in areas with a prominent level of opacity, and the same where there is less of it.

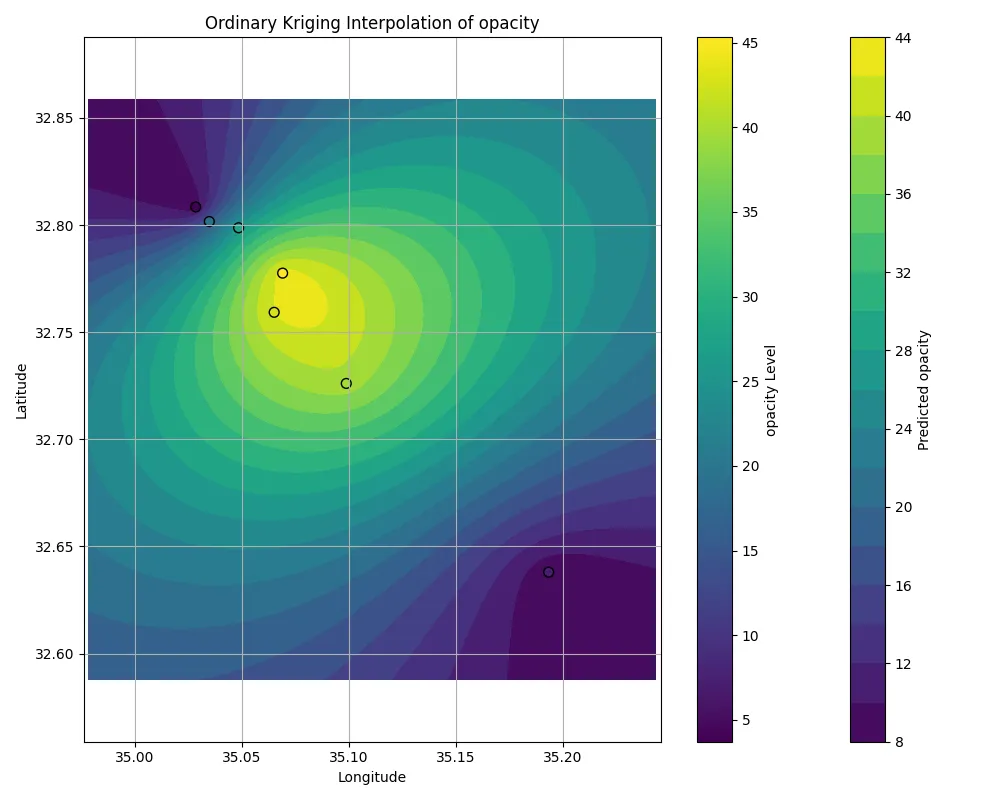


Figure 6: Ordinary Kriging Interpolation of Opacity

To do so, we can use all our available data and create a heatmap visualization plot, where the axes represent both the opacity and the electrical conductivity, while the map of colors is the individual count for each range of these values. The result can be seen in figure 7. From the heatmap plot, it can be interpreted that in the opacity range of 63.11 to 83.01 NTU and in the conductivity range of 11.88 to 14.24 ms/cm, there were the most individuals present – an amount of 638 individuals.

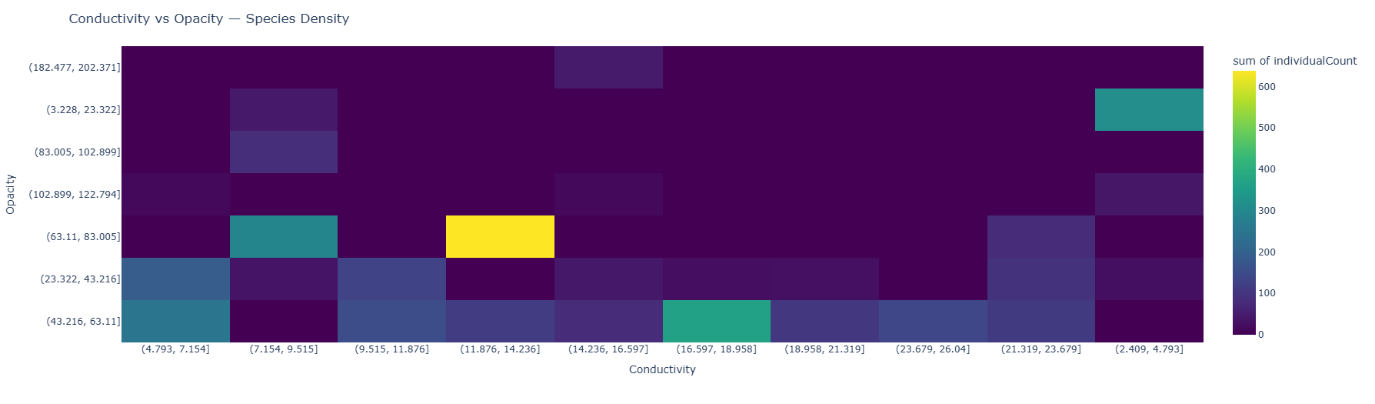


Figure 7: Electrical Conductivity and Opacity with Individuals Count

However, when we decide to look at the data from only one perspective, we get other results. In figure 8a, the electrical conductivity was divided into ten ranges and for each of them, the amount of individuals count was appended. The same was done in figure 8b for opacity ranges. When looking at the opacity table (figure 8b), it can be seen immediately that most of the individuals were recorded in the lower ranges of opacity (96.65% vs 3.35% of all individuals). In contrast, with conductivity, we would need to sum the values in the first half of the ranges and the other half. We get that in the lower half, there were 67.33% of all individuals, while in the higher half, there were 32.67% of all individuals.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Conductivity Ranges | Individuals Count | Individuals Count Percentage |  | Opacity Ranges | Individuals Count | Individuals Count Percentage |
| (2.41, 4.79] | 377 | 10.97% |  | (3.23, 23.32] | 357 | 10.39% |
| (4.79, 7.15] | 452 | 13.15% |  | (23.32, 43.22] | 563 | 16.38% |
| (7.15, 9.52] | 453 | 13.18% |  | (43.22, 63.11] | 1310 | 38.11% |
| (9.52, 11.88] | 280 | 8.15% |  | (63.11, 83.01] | 1009 | 29.36% |
| (11.88, 14.24] | 752 | 21.88% |  | (83.01, 102.9] | 83 | 2.41% |
| (14.24, 16.6] | 185 | 5.38% |  | (102.9, 122.79] | 67 | 1.95% |
| (16.6, 18.96] | 387 | 11.26% |  | (122.79, 142.69] | 0 | 0% |
| (18.96, 21.32] | 132 | 3.84% |  | (142.69, 162.58] | 0 | 0% |
| (21.32, 23.68] | 282 | 8.2% |  | (162.58, 182.48] | 0 | 0% |
| (23.68, 26.04] | 137 | 3.99% |  | (182.48, 202.37] | 48 | 1.4% |
| Figure 8a: Conductivity ranges vs  Individuals count | | |  | Figure 8b: Opacity ranges vs  Individuals count | | |

There is also another measure that we did not yet discuss – the coliform bacteria measure. When also including this measure, we will get the same ranges of conductivity and opacity. This can be seen in figure 9, where rows with 0 individuals were omitted for clarity, in the ranges of 11.88 to 14.24 ms/cm of conductivity, 63.11 to 83.01 NTU for opacity and 17.2 to 93,535.48 cfu/100ml of coliform bacteria, we get the most total amount of individuals – 638 of them. If we divide only the coliform bacteria into two halves, we get, as seen in figure 10, that in the first half, in the lowest levels, there are 98.73% of all individuals, while the highest levels include only 1.27% of them.

|  |  |  |  |
| --- | --- | --- | --- |
| Conductivity Ranges | Opacity Ranges | Coliform Bacteria Ranges | Individuals Count |
| (11.88, 14.24] | (63.11, 83.01] | (17.2, 93835.48] | 638 |
| (16.6, 18.96] | (43.22, 63.11] | (17.2, 93835.48] | 363 |
| (2.41, 4.79] | (3.23, 23.32] | (17.2, 93835.48] | 314 |
| (7.15, 9.52] | (63.11, 83.01] | (17.2, 93835.48] | 292 |
| (4.79, 7.15] | (43.22, 63.11] | (281472.037, 375290.32] | 250 |
| (4.79, 7.15] | (23.32, 43.22] | (17.2, 93835.48] | 188 |
| (9.52, 11.88] | (43.22, 63.11] | (17.2, 93835.48] | 151 |
| (23.68, 26.04] | (43.22, 63.11] | (17.2, 93835.48] | 137 |
| (9.52, 11.88] | (23.32, 43.22] | (17.2, 93835.48] | 129 |
| (11.88, 14.24] | (43.22, 63.11] | (17.2, 93835.48] | 114 |
| (18.96, 21.32] | (43.22, 63.11] | (17.2, 93835.48] | 105 |
| (7.15, 9.52] | (83.01, 102.9] | (93835.477, 187653.76] | 83 |
| (14.24, 16.6] | (43.22, 63.11] | (17.2, 93835.48] | 80 |
| (21.32, 23.68] | (63.11, 83.01] | (17.2, 93835.48] | 79 |
| (21.32, 23.68] | (23.32, 43.22] | (17.2, 93835.48] | 51 |
| (14.24, 16.6] | (182.48, 202.37] | (17.2, 93835.48] | 48 |
| (7.15, 9.52] | (3.23, 23.322] | (17.2, 93835.48] | 43 |
| (14.24, 16.6] | (23.32, 43.22] | (844381.72, 938200] | 42 |
| (2.41, 4.79] | (102.9, 122.79] | (93835.477, 187653.76] | 38 |
| (7.15, 9.52] | (23.32, 43.22] | (17.2, 93835.48] | 35 |
| (18.96, 21.32] | (23.32, 43.22] | (93835.477, 187653.76] | 27 |
| (2.41, 4.79] | (23.32, 43.22] | (17.2, 93835.48] | 25 |
| (21.32, 23.68] | (43.22, 63.11] | (17.2, 93835.48] | 25 |
| (16.6, 18.96] | (23.32, 43.22] | (17.2, 93835.48] | 24 |
| (14.24, 16.6] | (102.9, 122.79] | (17.2, 93835.48] | 15 |

Figure 9: Conductivity, Opacity, Coliform Bacteria, and Individuals Count

|  |  |  |
| --- | --- | --- |
| Coliform Bacteria Ranges | Individuals Count | Individuals Count Percentage |
| (17.2, 93835.48] | 2856 | 86.65% |
| (93835.48, 187653.76] | 148 | 4.49% |
| (187653.76, 281472.04] | 0 | 0% |
| (281472.04, 375290.32] | 250 | 7.58% |
| (375290.32, 469108.6] | 0 | 0% |
| (469108.6, 562926.88] | 0 | 0% |
| (562926.88, 656745.16] | 0 | 0% |
| (656745.16, 750563.44] | 0 | 0% |
| (750563.44, 844381.72] | 0 | 0% |
| (844381.72, 938200] | 42 | 1.27% |

Figure 10: Coliform Bacteria ranges vs Individuals count.

As we have seen, using methods such as PCA and Kriging, and then plotting the data in a certain way can show us some interesting details about ecological data in the river. We will next discuss what are the implications of all our findings.

Discussion

Analyzing the data through some means brought us to some conclusions. What we wanted to check earlier are the relationships between measures of the river and the total amount of observations found in the river. Although the data had many measurements, to understand the relationships deeper, we have used PCA. From this method, we understood that we need to select only a few of the measures. The ones chosen were specific electrical conductivity, opacity, coliform bacteria, and individuals.

Using Kriging, we understood that there are certain locations in the river where measures can be higher, so it led us to research more on what happens when the levels of measurements are either low or high. From the findings, we can conclude that at lower levels of all measures, the number of individuals was higher, although with specific electrical conductivity the situation was not as clear as in other measures, as we needed to calculate the percentages of each half to understand it. It is also implied by the correlation matrix produced by PCA, where the correlation between individuals and specific electrical conductivity was much stronger than with other measurements.

Therefore, with the data we got in our hands, we have been able to define the relationships between the river’s measures and the total amount of individuals recorded in each month. This brought us to the conclusion that we want to maintain the measures in the lower values, sometimes in the middle, so the number of individuals will increase.

The future work that can be done is to check more relations of all measures and even relationships between them more thoroughly. Also, since our data was scarce, both in time and number of observations, with more data, we will be able to find more precise relationships between these.

Studio Feedback – SUS Grade

The SUS grade as we calculated is 92.7. It implies high user satisfaction and positive user experience. Therefore, the system’s usability is considered to be great.

Maintenance Guide

The project was made using the [Google Colab](https://colab.research.google.com/) environment. Therefore, it is a single Jupyter Notebook file containing various cells that are categorized with headers. The description of the file will contain the name of the header, the sub-header and a description of all the code featured there.

Installations, Import, Enums and Constants

This part includes all what is needed to be installed when running the notebook (such as gradio package for the interface and pykrige package for Kriging). Then, it has all the used imports, data files that are used, and enums with constants that are used in the project.

Data Analysis

This part contains all functionality related to what was discussed in this document. All data tables, plots, PCA and Kriging are included in it.

The kishon\_occurences and kishon\_measurements variables hold all data we have collected about the Kishon River.

A merged\_df variable is created to change the rows and columns of kishon\_measurements, to be used in other functions.

Functions:

|  |  |
| --- | --- |
| **Function Name** | **Description** |
| biplot\_pc1\_pc2() | Creates a biplot plot of the first two PCA components. It uses the pca\_result variable, which is created by standardizing the merged\_df data and applying PCA to it. |
| correlation\_matrix\_heatmap() | Creates a correlation matrix heatmap. Use the correlation\_matrix variable which is created by calling the Pandas corr() function. |
| find\_best\_model(x, y, values, xpoints, ypoints, models=('linear', 'power', 'gaussian', 'spherical', 'exponential')) | Finds the best fitting variogram model for Kriging. Since data is scarce, not all data can fit in with one model, therefore the function is used to choose the one with the best score. |
| plot\_ordinary\_kriging(kriging\_longitudes, kriging\_latitudes, kriging\_values, measurement\_type\_to\_krig) | Creates two Ordinary Kriging plots – interpolation and variance plots. |
| plot\_kriging\_at(year, month, data\_source, measurement\_type) | Processes the data to be given to the plot\_ordinary\_kriging function. The function can get as input one of two data tables - monitoring\_df, which includes measurements from the Kishon River or occurences\_df, which includes the observations of individuals in the river. |
| plot\_kriging\_for\_month\_and\_year(measurement\_type, month, year) | Chooses the right data table according to the given measurement and calls plot\_kriging\_at() function. |
| generate\_density\_heatmap(x\_measure, y\_measure) | Creates the density heatmap for two measures, when the axes are the measures and the heatmap levels are the individuals count. |

There are also some other variables that are used for this document. These are data tables that show the ranges of certain measurements and how do they compare them to the individuals count measurement. They are called sum\_by\_conductivity\_bin, sum\_by\_opacity\_bin and sum\_by\_coliform\_bacteria\_bin.

Map Screen

This part includes the content displayed in the map section of the application. The content includes a map of historical data of animals and plants observations around the Kishon River and a map of forecasted data.

The variable kishon\_occurences\_grouped groups the kishon\_occurences by species, per month, year and location. Then some operations are used to create the integrated\_df, which has the number of individuals per species and river’s measurements, in each month, year and location. Then, the data is used to train a machine learning model to forecast observations called XGBoost. A data frame called historical\_proportions is created, which is used when in the end of the forecasting.

Functions:

|  |  |
| --- | --- |
| **Function Name** | **Description** |
| find\_nearest\_station(point\_geom, stations\_gdf\_proj) | Maps each location in the observations data to the nearest measuring station in the Kishon River. It makes the forecasting more precise, since the measurements are only per station. |
| normalize\_proportions(df\_group) | Normalizes the proportions of observations in each location to all observations in the same location. It will be used when forecasting the individuals with XGBoost and then returning to original locations. |
| get\_forecasted\_map\_data(target\_month, target\_year, integrated\_df\_full, model, species\_encoder, features, historical\_proportions, progress=gr.Progress()) | Creates a map with forecasted observations. It uses the XGBoost model and historical proportions to first create the forecast and then return it to the original locations. A gr.Progress() component is used to display the progress of the forecasting, since it takes some time. |
| create\_map\_per\_month\_and\_year(month, year) | Creates a map containing the historical data of all observations in a certain month and year. It also returns a plot with sums of individuals per month until the chosen month. |
| create\_future\_map\_per\_month\_and\_year(month, year, progress=gr.Progress()) | Creates a map containing the forecasted data of observations in a certain month and year. As the previous function, it also returns a plot with sums of individuals, including the new forecasted month. |
| create\_sum\_count\_plot(df\_to\_plot, title\_suffix="") | Makes the individuals count sum plot. Its input is the data frame, which can be either historical data or forecasted data and is used in the previous two functions. |

Graph and Info Screen

This part includes all the content displayed on the Info page. It has a diagram with all measurements and individuals count and some information from Wikipedia about species mentioned in the observations.

A station\_list\_str variable is created, which is used to display all available stations to look at when generating the diagram. Also, species\_list is also created for display – it includes all observed species in the river’s area.

Functions (Graph Part):

|  |  |
| --- | --- |
| **Function Name** | **Description** |
| get\_months\_axis(start\_year, start\_month, end\_year, end\_month) | Creates a list of string representations of all months between one month and another. |
| normalize\_list(lst) | Normalizes the list by using 0 as the minimum value and the maximum value from the data. |
| get\_values\_list\_by\_time\_periods\_and\_measurement(start\_year, start\_month, end\_year, end\_month, measurement, station, file) | Creates a dictionary with months as keys and measurements as values. |
| are\_dates\_valid(start\_year, end\_year, start\_month, end\_month) | Checks if the dates are valid (the start date is before or equal to the end date). |
| generate\_plot(start\_year, end\_year, start\_month, end\_month, station) | Generates the plot of all measurements and individuals counts from one date to the other, in a chosen measuring station. |

Functions (Info Part):

|  |  |
| --- | --- |
| **Function Name** | **Description** |
| fetch\_species\_info(species\_name) | Uses the Wikipedia API to get data about a certain species. |
| create\_species\_occurrence\_map(species\_name) | Creates a map of all occurrences of a certain species around the river. |
| update\_graph\_info\_content(species\_name) | Used as part of the interface to update the components to include the data and the map of the new selected species. |

App

This part includes the interface of the application. It uses the gradio package. There are various used components such as gr.Blocks() for the whole interface, gr.Row() and gr.Column() to organize each page in the app, gr.Column() is also used to show content for a certain page, gr.Markdown() to display stylized text, gr.Button() for functional buttons, gr.Image() to display an image, gr.Plot() for displaying plots, gr.DataFrame() for data frames (for instance, pandas DataFrame objects), gr.HTML() to display HTML content, in our case – folium maps and gr.Dropdown() for displaying several options.

Listeners can be assigned to buttons (click listeners) and to dropdowns (change listeners), with inputs and outputs being gradio components.

Functions:

|  |  |
| --- | --- |
| **Function Name** | **Description** |
| navigate(screen\_name) | Changes the visible content on the page to create a multi-page gradio interface (since gradio doesn’t have native multi-page interface, it can be achieved by changing the visibility of gr.Column() components). |
| show\_historical\_map() | Show the historical map only on the map page. |
| show\_future\_map() | Show the forecast map only on the map page. |

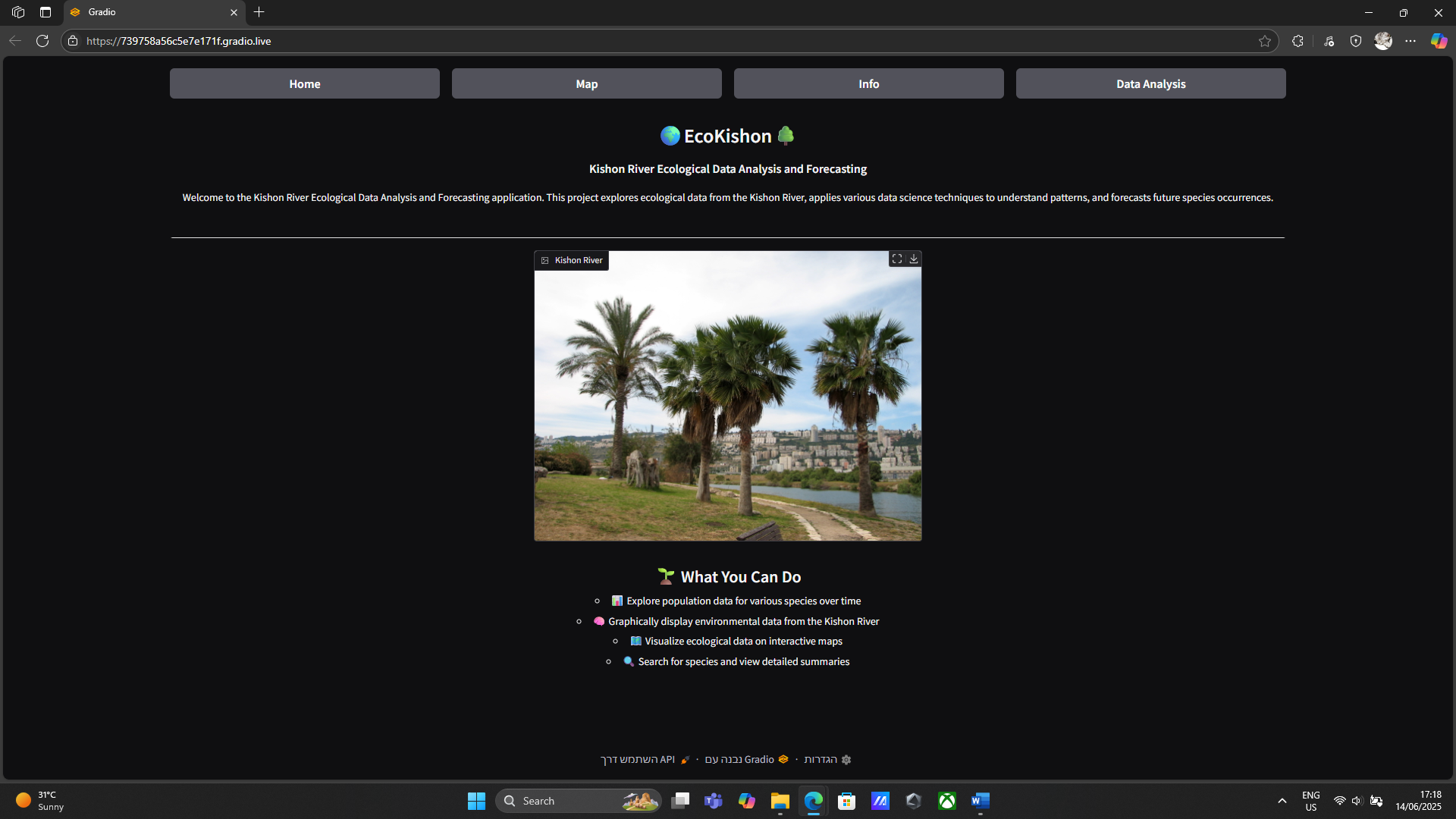
User Guide

[](https://drive.google.com/file/d/1Be2BzCfgcGYaAClkNUxzMZcM7Rs-xo5h/view?usp=sharing)Video of the application:

The app can be accessed via the [Google Colab link](https://colab.research.google.com/drive/1CoNpPhQpgkgO21lmb2uey-zbLuap1cWw?usp=sharing). To use it, the “Run all” button in the notebook’s interface can be clicked. The button is also located in the “Runtime” menu, or instead “Ctrl+F9” can be used.

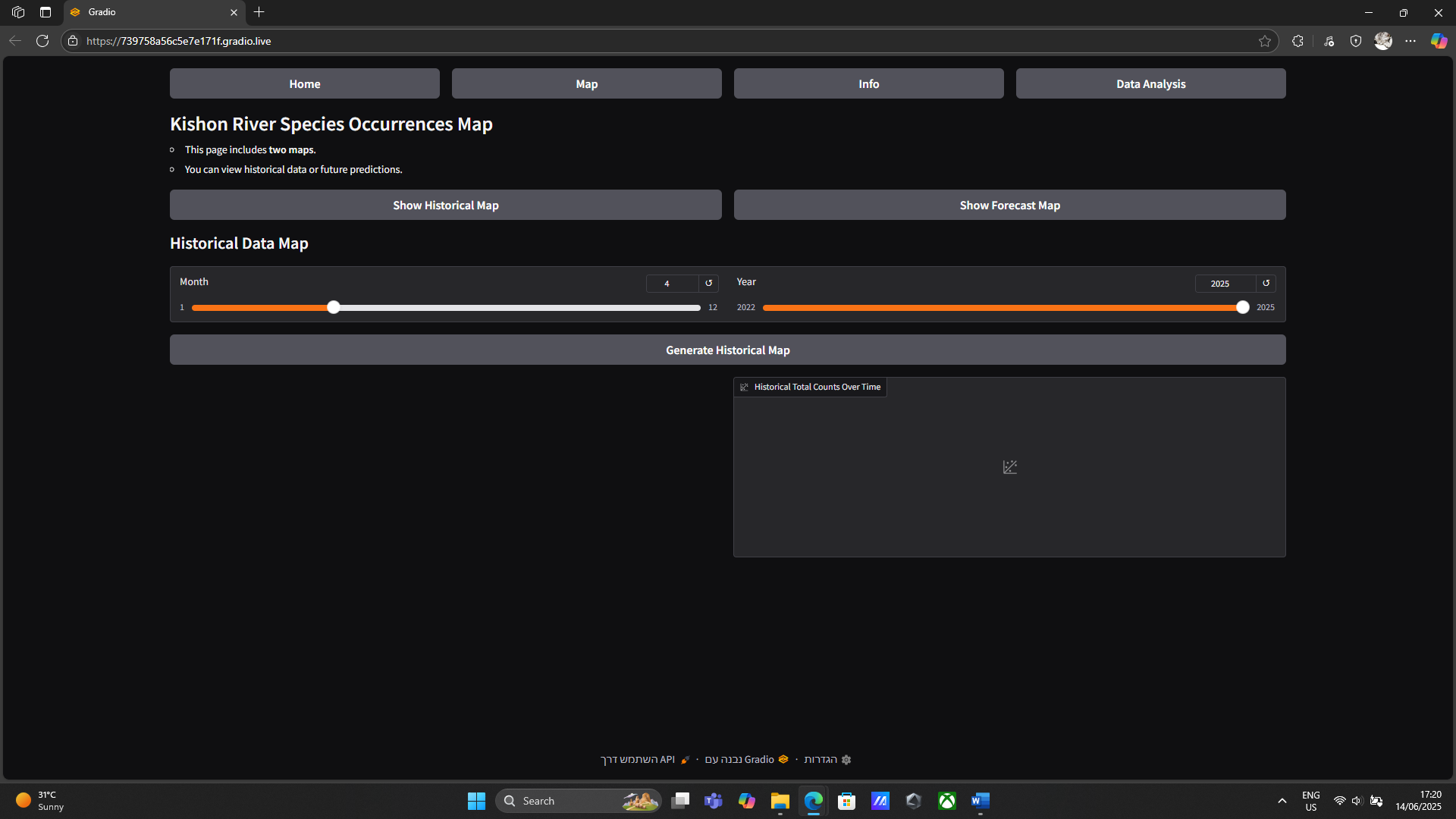
The application will be displayed in the App section of the notebook. In the output, there will be a public URL. It can be pressed to open the application in a new window.

The first page that will open is the home page:



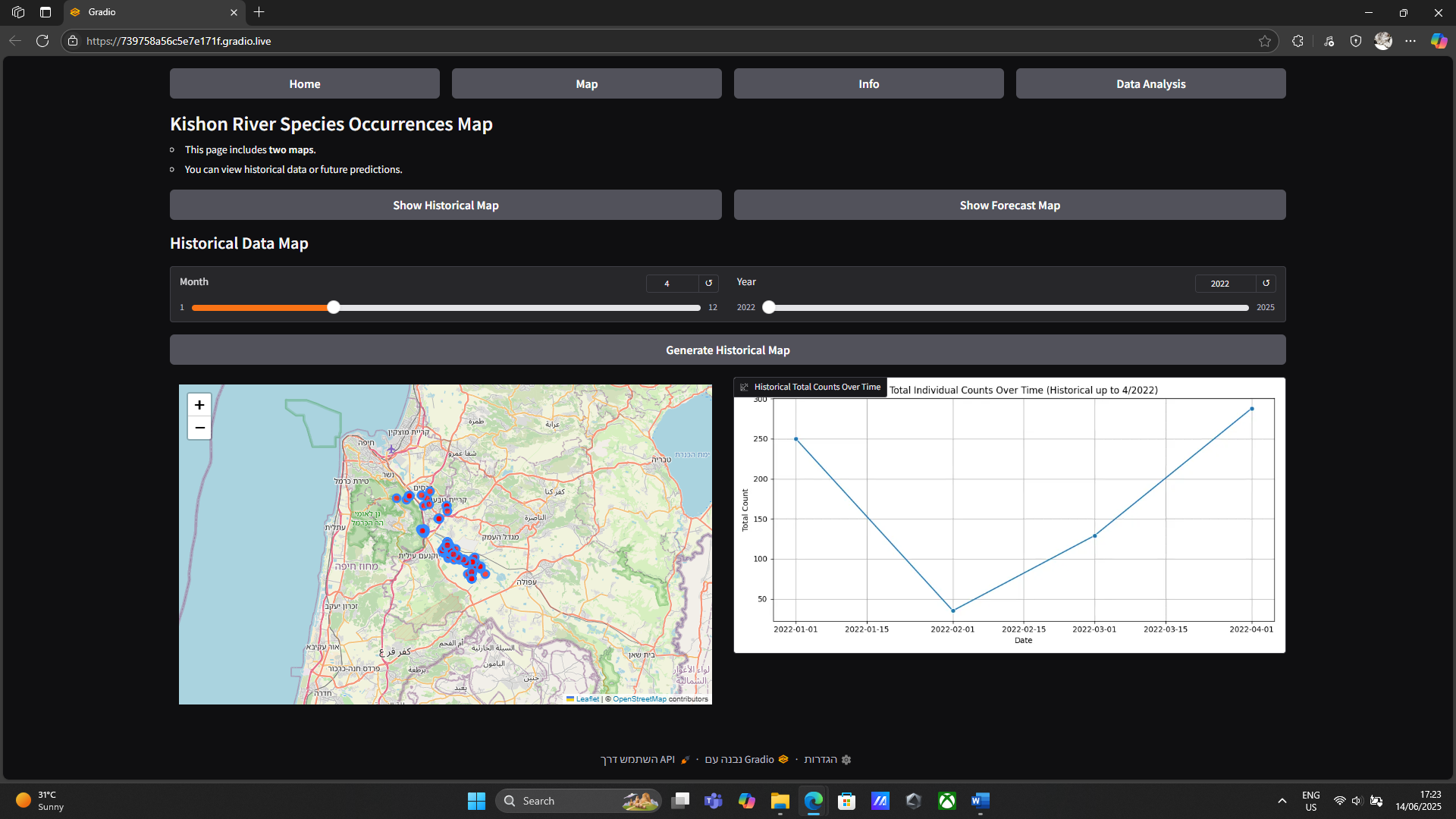
It includes some information about the app. To navigate through the app, the navigation bar at the top can be used. It includes four buttons – “Home” to return to the current page, “Map” to navigate to the map page, “Info” to the graph and information page and “Data Analysis” to some other plots and data frames.

The Map page is:



This page has two types of maps that can be displayed. One is the historical map, which includes all observations made from January 2022 to May 2025. It is accessed with the “Show Historical Map” button, and to generate it, a date should be chosen and then the “Generate Historical Map” button should be pressed.

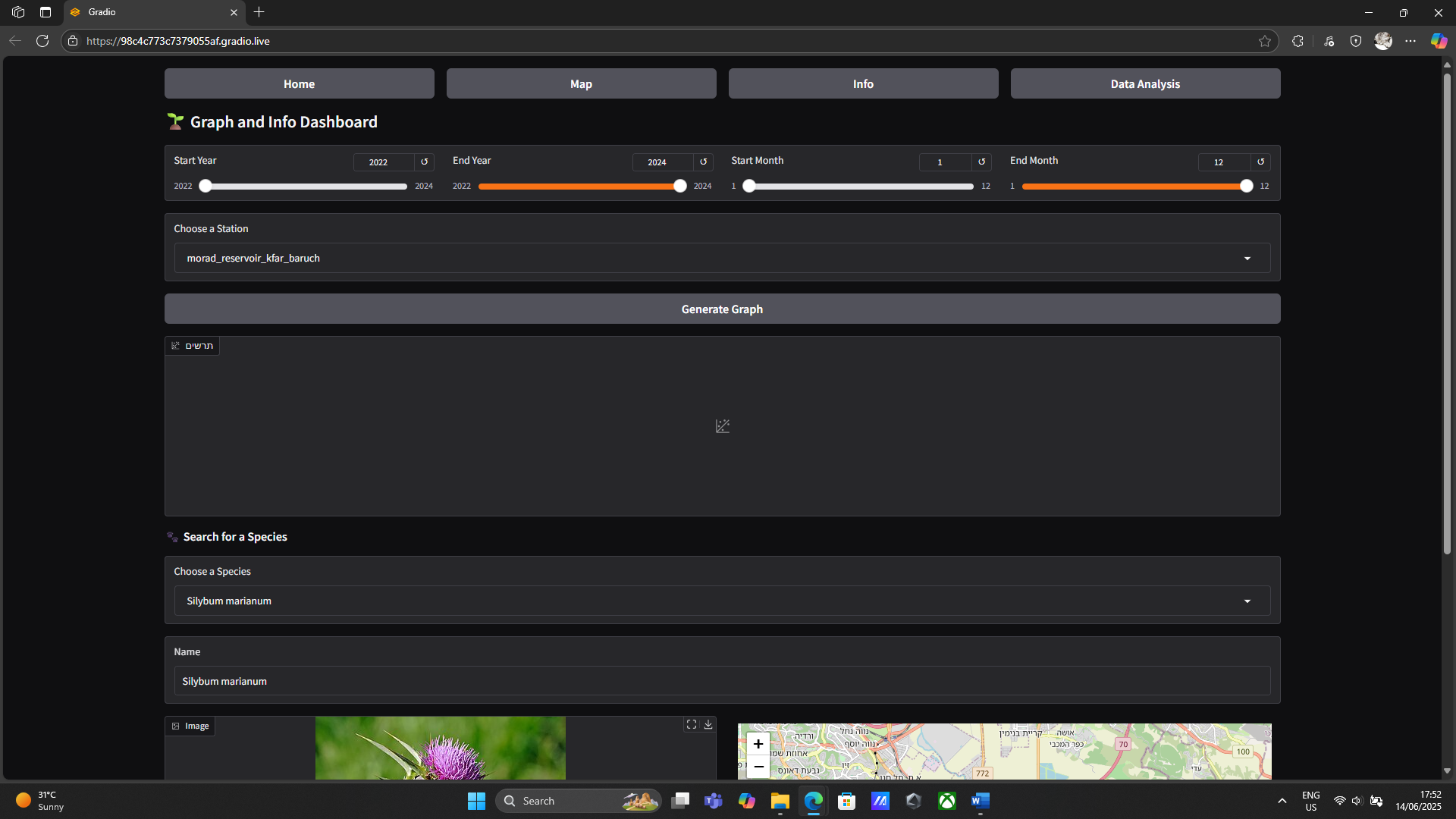
Doing so will create the map:



In the left side, the map will be displayed, while in the right side, a plot that counts the sum of all occurrences in each month is shown. Choosing a date after May 2025 may result in a failure message.

When choosing the “Show Forecast Map” and generating it, it will show the same – a map and a plot. An error is displayed when choosing a historical date.

The Info screen is:



It includes two parts; one is a graph that shows measurements of the Kishon River in a certain station with individuals around the same station. It is created by choosing a starting and ending dates and a station, then pressing “Generate Graph”:

A screenshot of a computer

AI-generated content may be incorrect.

A new graph won’t be generated if the dates are invalid, such as when selecting a start date that is after the end date.

In the second part, information about a specific species can be displayed. Choosing a species will generate information and a map of where it was seen in some observations around the river:

A screenshot of a computer

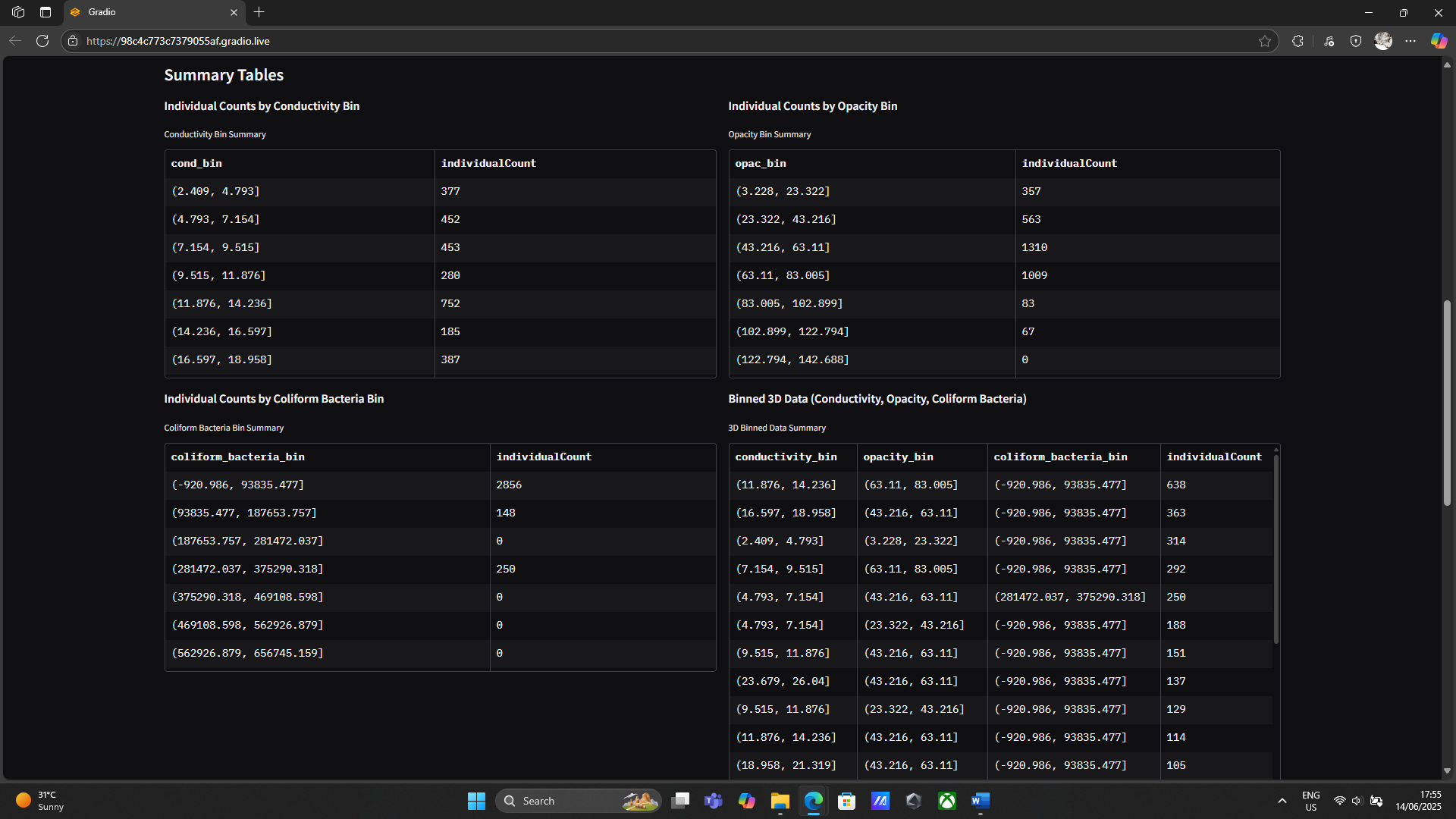
AI-generated content may be incorrect.

The last page is the data analysis page:

A screenshot of a computer

AI-generated content may be incorrect.

It includes various plots, including the PCA biplot and the correlation matrix (as shown), tables that were used for the research:



A density plot that can be generated for two measurements:

A screenshot of a computer

AI-generated content may be incorrect.

And a Kriging plot that can be generated for one of three measurements, for a certain month and year:

A screenshot of a computer

AI-generated content may be incorrect.

Challenges

There were many challenges we experienced when working on the project. First, choosing the ecosystem was no easy task. We tried to choose a system that can interest us, but not one with many things that affect it. Second, after choosing the Kishon River, scarcity of available data on the internet was an issue. Since we didn’t find a lot of it, it wasn’t easy to research it, therefore our results can be only specifically true according to the data we have collected. Third, using gradio, which is a new package for us, took us time, and thanks to finding examples on the internet of how to use gradio and integrate plots and maps into it, it wasn’t as complicated as we thought.

Studio Feedback – Notes about the System

|  |  |  |
| --- | --- | --- |
| **Feedback Comment** | **Do you think there is a need for a change in the system following the comment?** | **Reason** |
| Dense user interface. | Yes. | The user interface was indeed quite dense. We have decided to change the positions of some components, thus creating a more spacious UI. |
| Adding light mode. | No. | Light and dark mode are managed by the gradio framework – which is the UI package in our project. Therefore, light mode is already implemented and is applied according to the system’s preferences. |
| Use real data. | No. | Real data is already used as part of the project. We have collected data that was published by the Kishon River Authority and GBIF. |

References

1. Adi Oren, Zeev Aizenshtat, Benny Chefetz. (2005). Persistent organic pollutants and sedimentary organic matter properties: A case study in the Kishon River, Israel. <https://doi.org/10.1016/j.envpol.2005.08.039>.
2. Ben Ezra, T., Tsemel, A., Suari, Y., Berman-Frank, I., Tchernov, D., & Krom, M. D. (2024). Determining the Fluxes and Relative Importance of Different External Sources and Sinks of Nitrogen to the Israeli Coastal Shelf, a Potentially Vulnerable Ecosystem. *Water*, *16*(18), 2585. <https://doi.org/10.3390/w16182585>.
3. Herut, B., Gertner, Y., Segal, Y., Sisma-Ventura, G., Gordon, N., Belkin, N., & Rahav, E. (2023). Long-Term (2002–2021) Trend in Nutrient-Related Pollution at Small Stratified Inland Estuaries, the Kishon SE Mediterranean Case. Water, 15(3), 484. <https://doi.org/10.3390/w15030484>.
4. Golan, T. (2016). The Fall and Rise of the Kishon River. Water, 8(7), 283. <https://doi.org/10.3390/w8070283>.
5. Brigden, K. & Stringer, R. (2001). A critical assessment of the Kishon River Masterplan Report published by the State of Israel Ministry of Environment, July 2001. <https://www.greenpeace.to/publications/Kishon%20Haifa%202002.pdf>.
6. Herut B., Shirav M., Kress N., Homung H., Ilani S., Yoffe O. (1993). Sediment composition and pollution levels along the Kishon River and its drainage basin. <https://www.gov.il/BlobFolder/reports/reports-1993/he/report_1993_Herut-Shirav-Sediments-Pollution-Levels-Kishon-River-Drainage-Basin-GSI-23-93-IOLR-25-1993.pdf>.
7. Kishon River Authority [https://kishon.org.il/kishon-river-authority](https://kishon.org.il/kishon-river-authority/).
8. Global Biodiversity Information Facility (GBIF) <https://www.gbif.org/>.
9. Israel Nature and Parks Authority <https://exhibition.parks.org.il/home/en/>.
10. Herve Abdi, Lynne J. Williams. (2010). Principal Component Analysis. <https://doi.org/10.1002/wics.101>.
11. Wackernagel, H. (2003). Ordinary Kriging. In: Multivariate Geostatistics. Springer, Berlin, Heidelberg. <https://doi.org/10.1007/978-3-662-05294-5_11>.