



VRIJE  
UNIVERSITEIT  
BRUSSEL

# ENVIRONMENTAL PROGRAMMING

## ASSIGNMENT 5

**Tropical Cyclone impact data comparison between  
Wikimpacts2.0 and EM-DAT database**

NI LI  
PROF. DR. IR. ELGA SALVADORE

**2025-2026**





<b><u>1. INTRODUCTION</u></b>	<b>2</b>
<b><u>2. ASSIGNMENT</u></b>	<b>2</b>
<b>2.1. DATASET</b>	<b>2</b>
<b>2.2. TASKS</b>	<b>2</b>
<b>2.3. OPTIONAL TASK</b>	<b>3</b>
<b>2.4. LIBRARIES</b>	<b>3</b>
<b><u>3. SUBMISSION AND REPORT</u></b>	<b>3</b>
<b>3.1. DO YOU NEED CLARIFICATION ON THE ASSIGNMENT?</b>	<b>4</b>
<b>3.2. HOW IS THE ASSIGNMENT EVALUATED?</b>	<b>4</b>

# 1. INTRODUCTION

In recent years, the study of climate extreme events has garnered considerable attention due to the increasing frequency and intensity of such phenomena and their extensive impacts. Nevertheless, existing databases, such as EM-DAT, are limited by their focus on impact data at the national level, which constrains research on local impacts, climate attribution, and adaptation measures.

To address this gap, we introduce the new climate impact database, Wikimpacts 2.0, which includes impact information at the sub-national level. It is essential to align events from Wikimpacts with those in EM-DAT to establish a benchmark for data consistency regarding impacts.

In this project, you will be provided with the confidential Wikimpacts 2.0 database and tasked with matching the impact information of tropical cyclones to that of EM-DAT, as well as evaluating any discrepancies.

## 2. ASSIGNMENT

### 2.1. Dataset

1. Wikimpacts 2.0: contains data on the occurrence and impacts of climate extremes in country and sub-national scales. The database is inferred from Wikipedia and uses generative AI.
2. EM-DAT, please download from [Public EM-DAT platform](#), only "tropical cyclone" needed.

### 2.2. Tasks

1. Download the Wikimpacts 2.0 database in db format.
2. Load Data:
  - Read the database file and load all tables that start with "Total" into a DataFrame named `L1`.
  - Identify all tables that start with "Specific" and load them into separate DataFrames named `L3\_\*`, where `\*` represents impact categories, only load Deaths, Injuries and Damage.
3. Filter by "Tropical Storm/Cyclone":
  - Using the "Main\_Event", filter the Tropical Storm/Cyclone events from L1 into a new dataframe "L1\_TC"
  - Using "Event\_ID" from "L1\_TC", filter the "L3\_%" with only impact from Tropical Storm/Cyclone
    - "Start/End\_Date\_Year," "Start/End\_Date\_Month," and "Start/End\_Date\_Day" columns. If these date fields are missing in `L3\_\*`, fill them with the corresponding information from `L1\_TC`.
4. Aggregate by Administrative Area:
  - Using the "Administrative\_Area" column in each `L3\_\*` DataFrame obtained from Step 3, for the same "Event\_ID", aggregate the impact from the same "Administrative\_Area", name the new dataframe to `L3\_\*\_aggregated`

#### 5. Filter by Date:

- In each `L3\_\*\_aggregated` DataFrame, filter the records to include only those events that occurred after the year 1900. Name these filtered DataFrames as `L3\_\*\_aggregated\_1900`.

#### 6. Identify and Analyze same tropical cyclone (TC) Events:

- Using the ISO from EM-DAT, and Administrative\_Area\_GID in `L3\_\*\_aggregated`, and "Start/End\_Date\_Year," "Start/End\_Date\_Month," to identify the same TC events, and save a new dataframe as "EM\_DAT\_Wikimapcts\_Matched"
- Calculate the impact (e.g., Deaths) difference of these matched events. Using the relative difference, and category the difference to 5 categories, -50% less, -30% less, Perfect Match, +30% more, +50% more, and visualize the difference in a bar plot.
- Save the plot as "EM\_DAT\_Wikimpacts\_Impact\_comparison.png".

#### 7. Analyze the spatial differences between two databases

- Using the ISO from EM-DAT, and Administrative\_Area\_GID in `L3\_\*\_aggregated`, compute the number of impact data entries difference between two databases, and visualize the difference in a world map.
- Save the plot as "EM\_DAT\_Wikimpacts\_Spatial\_Impact\_comparison.png".

### 2.3. Optional Task

Develop a simple graphical user interface (GUI) to interactively execute your code.

**Tip:** You are free to use any open access Python package and free to design and create the layout of the interface. I suggest you to try Tkinter (<https://wiki.python.org/moin/TkInter>) but this is just a suggestion, many other options are available.

Internet is an incredible source of information (docs, videos, tutorials...), e.g. <https://www.geeksforgeeks.org/python/create-first-gui-application-using-python-tkinter/>

Just to list some websites and youtube videos...

<https://www.youtube.com/watch?v=eJRLftYo9A0&list=PLQVvaa0QuDcIKx-QpC9wntnURXVJqLyk&index=8>

<http://sebsauvage.net/python/gui/>

[https://www.tutorialspoint.com/python/python\\_gui\\_programming.htm](https://www.tutorialspoint.com/python/python_gui_programming.htm)

<https://docs.python.org/3/library/tkinter.html>

<http://www.tkdocs.com/tutorial/index.html>

### 2.4. Libraries

Python libraries you may use in the project: Pandas, Geopandas, Matplotlib, NumPy

## 3. SUBMISSION AND REPORT

Develop a software that solves the steps described in the previous section in a collaborative way using GitHub desktop (or GitHub CLI). Submit a link to your GitHub repository on the dedicated space on the CANVAS page of the course by 21<sup>st</sup> of December 2025.

Make sure the online GitHub repository contains: (1) all the Python codes you have developed to accomplish the tasks described in section 2.2, (2) a Jupyter Notebook to present and run your code (user manual/tutorial like), (3) a section of the Jupyter Notebook describing the use of Generative AI, (4) a section of the Jupyter Notebook describing the

contributions of each team member (if not clear from the GitHub contributions), (5) a flow chart of your codes (it can be included in the Jupyter Notebook), (6) a .yml file of the environment you have created for your project.

### **3.1. Do you need clarification on the assignment?**

Please contact Ni Li and Elga Salvadore through the dedicated CANVAS forum (name of your group). In the discussion forum you are able to post questions, share your code and request an appointment for an online session.

### **3.2. How is the assignment evaluated?**

EVALUATION MATRIX Assignment	Point
1) Does your software perform the required tasks providing the correct result(s)?	10 points
2) Is your code readable?	$\pm 0.5$ point
3) Are the name of the variables\functions\modules meaningful?	$\pm 0.5$ point
4) Are the data types appropriate for values the hold?	$\pm 0.5$ point
5) Are there too many code repetitions?	$\pm 0.5$ point
6) Does your code have a coherent structure?	$\pm 0.5$ point
7) Is the Jupyter Notebook clear, to the point and complete?	$\pm 1$ point
8) Is your code easy to use?	$\pm 1$ point
9) Is there a good balance of comments and code in the scripts?	$\pm 0.5$ point
10) Code reuse: are existing codes used appropriately? Code generation: appropriate use of Generative AI for programming	$\pm 1$ point
11) Code reuse: Did you forget to properly citing the source of codes you are reusing? Did you let ChatGPT (or similar) generate part of your code without mentioning it in your Jupyter Notebook?	- 3 point
12) How flexible is your code? For example, if I use a different data set with a different file name, would it still be working?	$\pm 0.5$ point
13) How easy would be for the next user to adapt/modify your code?	$\pm 0.5$ point
14) Is there any documentation (within the Jupyter Notebook and .yml file)? Did you create a flow chart?	$\pm 1$ point
15) Did you develop a GUI for your code?	Up to 2 points