COMP 4304 / 6934 – Assignment 4 (7%)

MEMORIAL
UNIVERSITY

Due: 11:59pm, Feb. 16, 2024

Learning Objectives

The goal of this assignment is to become familiar with Seaborn and alluvial diagrams.

Data

Download the `sleep.csv`, `temps.csv` and `migration.csv` data sets from Brightspace.

The `sleep.csv` data set is the same data set about recommended hours of sleep from Assignment 2.

The `migration.csv` data set contains information about immigration and emigration between various regions of the world. We have used this data set before in Lecture 11.

The `temps.csv` data set comes from NASA and contains the average global temperature changes over the past 140 years (the Land-Ocean Temperature Index). This data set is available here: https://data.giss.nasa.gov/gistemp/. It contains the average temperature deviation from the 1951-1980 baseline on a month-by-month basis.

Instructions

Using the provided data sets, create a Jupyter notebook to answer the following questions.

You may only import the pandas, Matplotlib, Seaborn, math or NumPy packages.

Additionally, the Python script `alluvial.py` found on the assignment instructions will be needed for Question 3. This code comes from https://github.com/vinsburg/alluvial_diagram. Calling `plot()` will create an alluvial diagram (see Question 3).

Question 1 – Seaborn: (30 pts)

Seaborn is a high-level interface to Matplotlib. It allows for the creation of a number of different plot types with one-line commands. It can use pandas DataFrames directly. It includes a number of statistical features. And the default plots generally look pretty good (especially compared to default Matplotlib).

Re-create the grouped bar plot from Question 1 of Assignment 2 using Seaborn's `barplot()` function (see below).

Make sure that the category labels, axes labels, and colours are all correct. There should be a gap between bars within each age category.

Your visualization should be created using only the `barplot()` function. That is, the creation of your visualization should be no more than 1 line of code. You are allowed to perform data manipulation beforehand, but there should be no other Matplotlib or Seaborn functions called except for the one line of code for `barplot()`.

(Note that the title or legend present in the original image do not need to be present for this question, as they cannot be added through `barplot()`.)
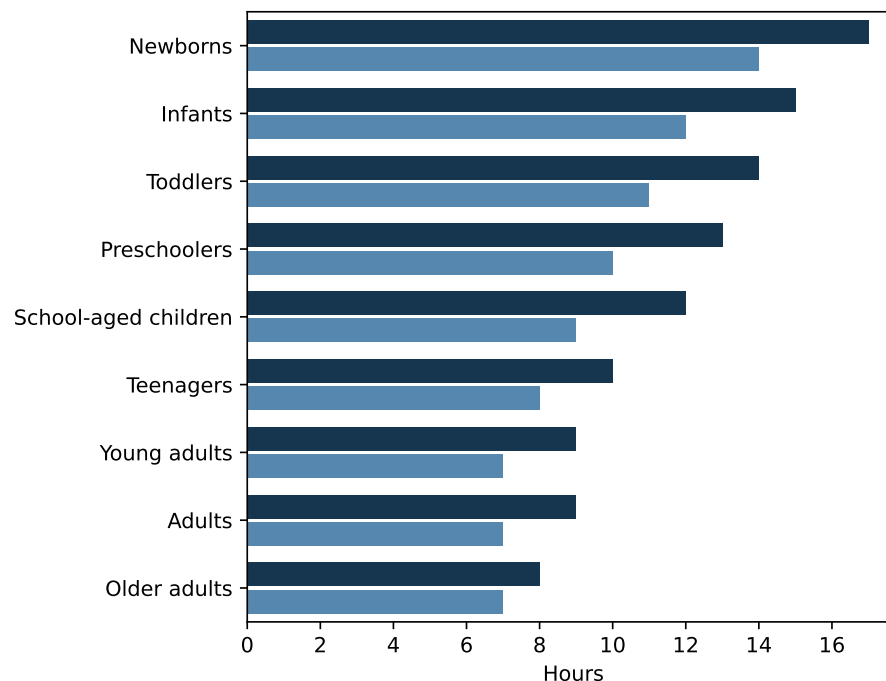
Question 2 – Histograms: (35 pts)

Re-create the visualization below using Matplotlib and Seaborn. This image is a series of histograms of the average monthly temperature deviations per decade. The vertical dashed white line represents the average global temperature from 1951–1980.

The background colour is '#6ec5fa', and the colour palette for the histograms is 'coolwarm'. The specific colour for each histogram chosen based on the mean temperature deviation for that decade (the mean temperature deviation for the 2010s is roughly +0.8 degrees C).

Make sure the background colour is present, as is the title, text annotations, and tick mark labels. Each histogram should be outlined with a black line. The number of bins of each histogram is 11.

(Hint: `sns.color_palette()` can be used to create a colour palette for sampling colours. Also, `zorder=` can be helpful for specifying how elements overlap one another.)

The Average Global Temperature Keeps Climbing
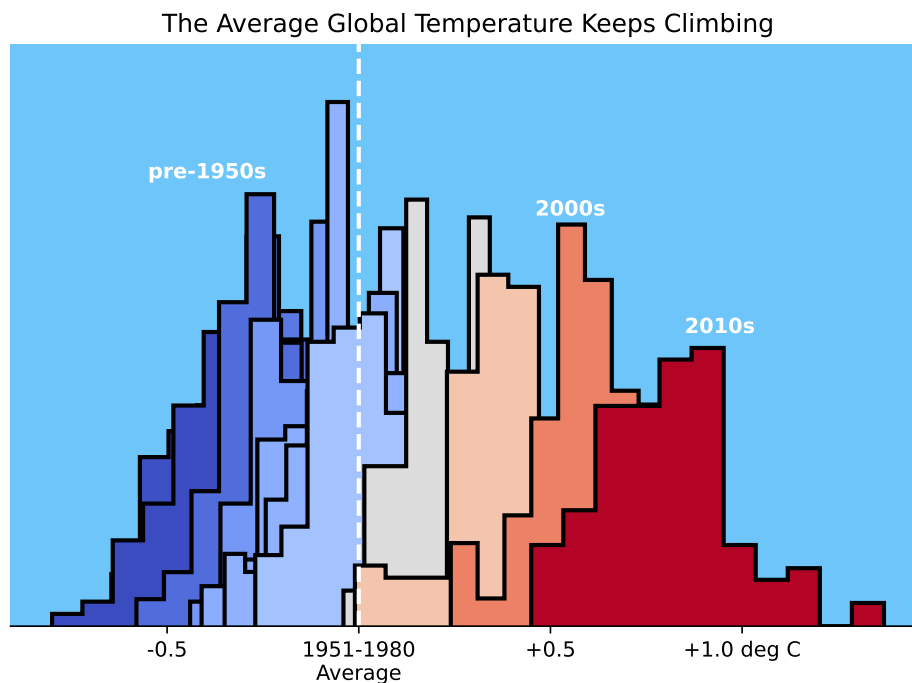
6934 Students Only – Question 3 - Alluvial Diagram: (35 pts)

Alluvial diagrams are a form of Sankey diagram. They generally form a two-sided display representing the flow of items from one set of states to another set of states. The width of the arrows represent the quantitative value of the flow. It is possible to extend to three or more sets.

The visualization below is an alluvial diagram showing the migration of people between different regions of the world. The left and right sides represent the origin and destination (people migrating from one region to another region).
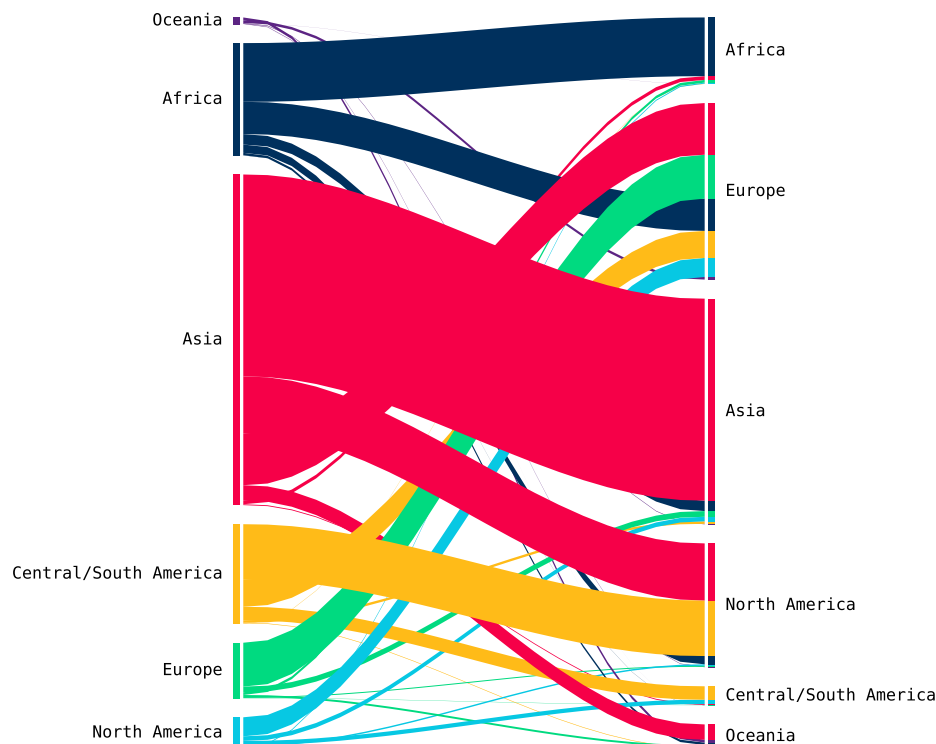
Re-create this alluvial digram using the `migration.csv` data set and the `alluvial.py` source code. Make sure that the colours are correct, the source and destination values are correct, as well as the labelling and ordering of items within each set.

The set of colours used are '#07c8e3' (North America), '#00da80' (Europe), '#ffbb18' (Central/South America), '#f60048' (Asia), '#00305d' (Africa), and '#5c2483' (Oceania).

Importantly, the alluvial fans (flows) must be layered so that wider bars are on top of thinner bars. For example, migration from Asia to Asia is the largest bar. It overlaps all others. Asia to Oceania is under the flow from Central/South America to North America, but above the flow from Central/South America to itself. You will need to modify the provided source code to enable this.

Submission

Submit your Jupyter notebook (`.ipynb`) through Brightspace. The modified alluvial diagram source code should be embedded inside your notebook (preferably not as a standalone piece of code).

Late submissions will be subject to a 10% penalty for each hour past the deadline.

Attribution

Submissions should include an attribution section indicating any sources of material, ideas or contribution of others to the submission.

Submissions must represent your independent work.

You are encouraged to use any resources to help with your solution, but your solution must represent independent work. If your submitted work includes unacknowledged collaboration, code materials, ideas or other elements that are not your original work, it may be considered plagiarism or some other form of cheating under MUN general regulations 6.12.4.2 (4.12.4.2 for graduate students) and academic penalties will be applied accordingly.

Avoid academic penalties by properly attributing any contribution to your submission by others, including internet sources and classmates. This will also help distinguish what elements of the submission are original. You may not receive full credit if your original elements are insufficient, but you can avoid penalties for plagiarism or copying if you acknowledge your sources.

Github

I encourage you to store and version your work on GitHub. It is good practice to do so as everyone uses git in the real world.

However, **it is a requirement that git repositories containing assignment material be private.** University regulations (undergraduate 6.12.4.2 and graduate 4.12.4.2) consider it cheating if you allow your work to be copied. There will be zero tolerance for this.