

# CS-GY 6313 / CUSP-GX 6006: Data Visualization - Spring '24

---

## Homework #4: Interactive Visualizations

---

This homework is due during the final examination period: **Sunday, May 12th, 2024 at 11:59pm**

*This homework consists of three (3) required segments, totaling up to 15 points. Points are more likely to be awarded if code implementations come with comments explaining your process, even if the answer is incorrect. Submit your code as a **.zip** file together with your code and any other submittable items such as figures or writeups. You are permitted to use external Python libraries for your work.*

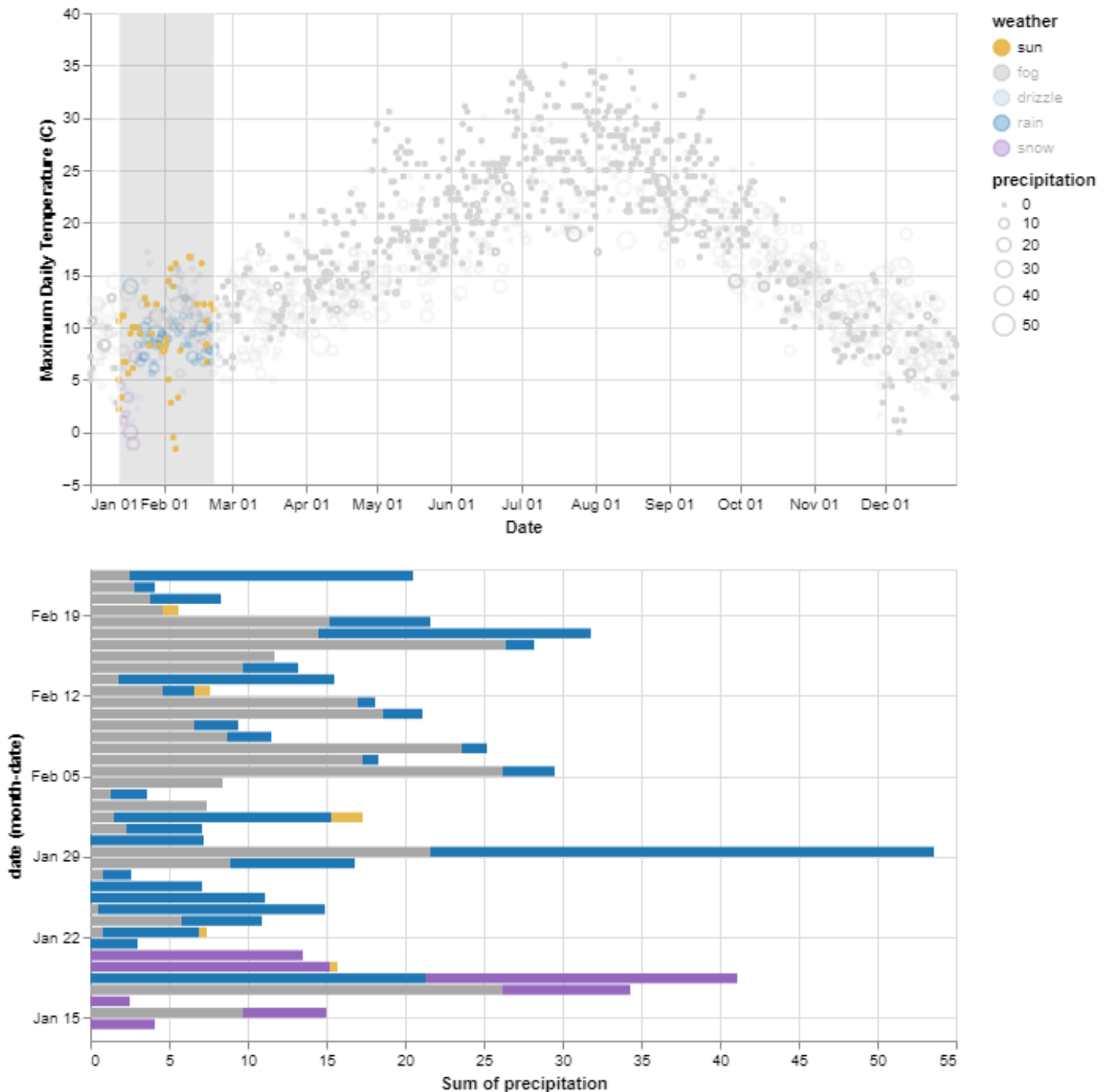
*Adhere to NYU's Academic Integrity rules. Your answers and implementation must be your own. If you cooperated with another student from the course, please list either their name or NetID either as a comment in the code implementation or in Markdown. If you relied on external resources, websites, publicly-available code, etc. to answer any part of this assignment, you are expected to add an attribution to the original source material (ex. a comment, or a URL). **Using ChatGPT to generate code is explicitly not allowed and will result in an automatic 0 points for this assignment.***

### Interactive Weather Plots (15 points)

There are many Python-based packages to help you generate interactive 2D graphs. In this assignment, we'll be covering **Altair**, which gives you plenty of options on how to generate some interactivity with 2D plots.

We'll be looking at weather data in Seattle from 2012 to 2015 and visualizing them as a combination of scatter and horizontal bar plots. The means to import and access the Seattle data is provided in code, and we also provide some starter code to get you started understanding how to use Altair. It's now your turn to update the visualization so that it has some cool mouse interactions.

## Seattle Weather: 2012-2015



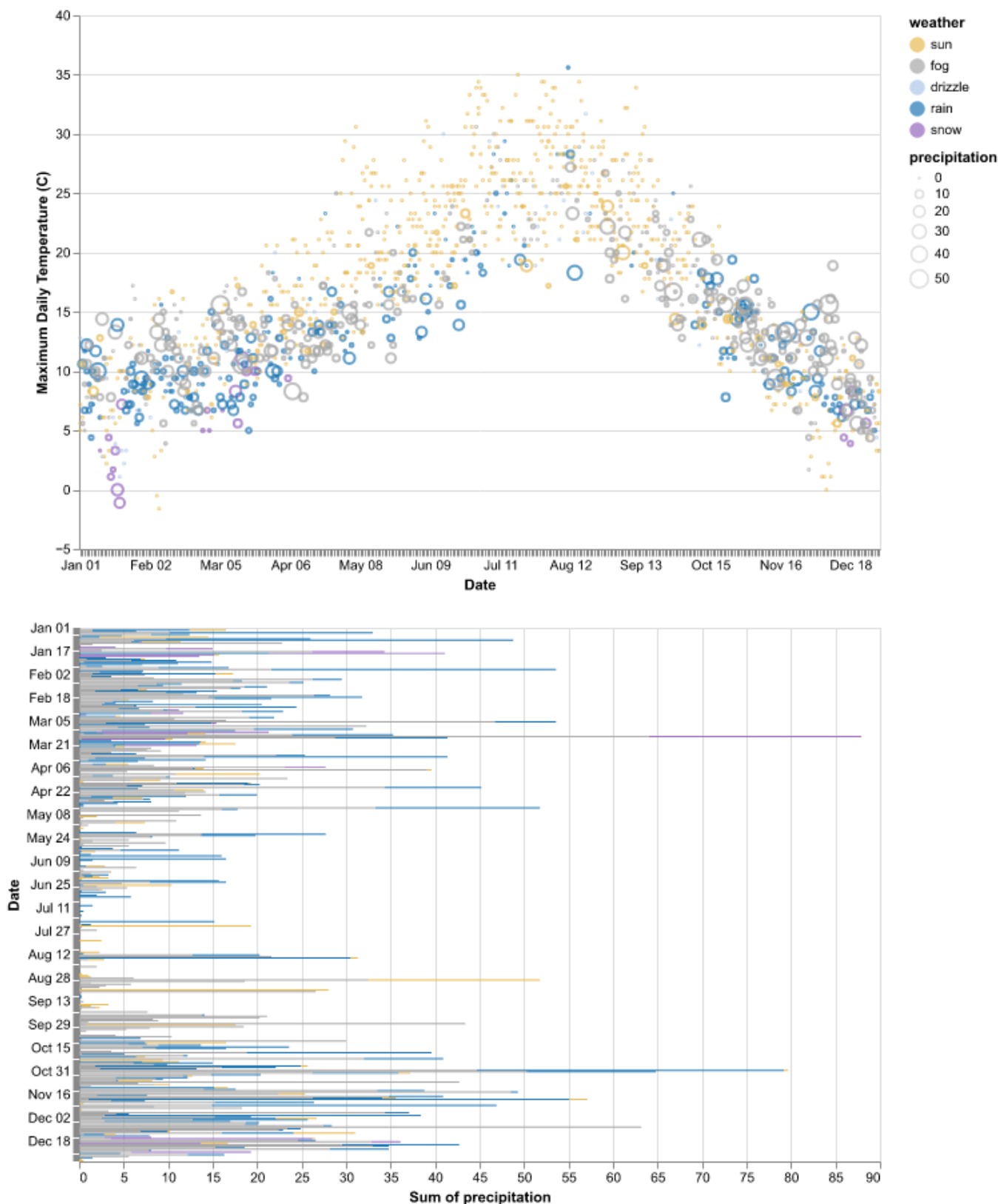
### Part 1: Changing Plots (5 points)

If you print the starter code, you should notice several things about the plots:

- The top chart is a scatter plot that shows the correlation between maximum daily temperature (x-axis) and the daily temperature range (y-axis).
- The bottom chart is a horizontal bar chart that counts the number of days that correspond to each weather type.

We want to modify both of these charts so that they show different aspects of the data. The top chart should be modified so that the x-axis now represents the days in a year, while the y-axis represents the maximum daily temperature of each day. The bottom bar chart should be converted into a stacked horizontal bar chart that shows the sum total amount of precipitation for each day in our weather dataset. Its x-axis must represent the sum amount of precipitation that day, its y-axis must represent the date (in

month-date format), and the stacking must indicate the percentage each weather type contributed to the total sum. The stacking must also be color-coded to reflect the colors in the `scale` provided. Be sure to look at the example below to understand what each plot should look like:



**Grading Criteria:** This part will be graded on the following:

1. Is the top chart changed so that the x-axis represents the date range of a year and the y-axis represents the maximum daily temperature? (2 points)

- Is the chart a horizontal bar chart that shows the total precipitation for each day in the dataset? (2 points)
- Is the bar chart stacked, showing how each weather type contributed to the total precipitation for that day? (1 point)

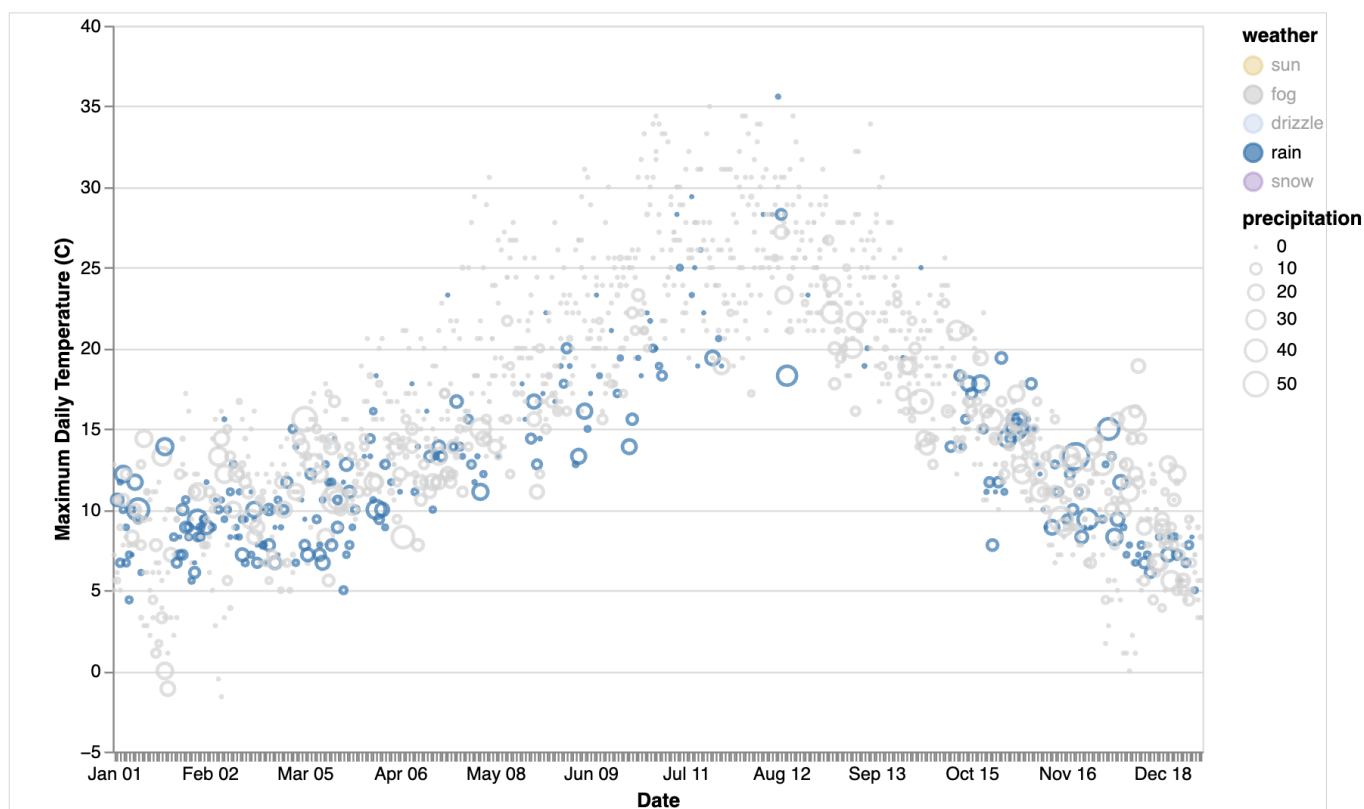
### Hints:

- Look at how Altair uses strings to represent different operations. For example, to provide a count of the number of days, it uses `'count()'` rather than a function. You might want to look up the Altair documentation for what operations are available.
- Inside the `encode(...)` function used in the code provided, you should see that you can use strings to tell Altair what data they should use for each axis.

### Part 2: Weather Type Selection (5 points)

With the changes we made to both charts, let's begin to add interactivity to the top chart. This time, we will want to add that interactivity to the legend of the top chart. Indeed, Altair allows you to bind actions to the legend.

Your task for this part is to modify the top graph so that if you click the weather name in the legend, all data points with different weather types are grayed out in the top chart. If I click anywhere else, the legend is reset and all colors are returned to the top chart.



**Grading Criteria:** This part will be purely graded on whether the interactivity described above is implemented. The method through which this is achieved is up to you as the implementer. (5 points)

### Hints:

- To understand how to add interactive elements to an Altair plot, make sure to check out [Altair's user documentation](#), which is available online and gives you insights into how to add particular

interactions.

- You will likely have to make use of some Altair functions such as `alt.selection_point(...)` and `alt.condition(...)` to add the required functionality.

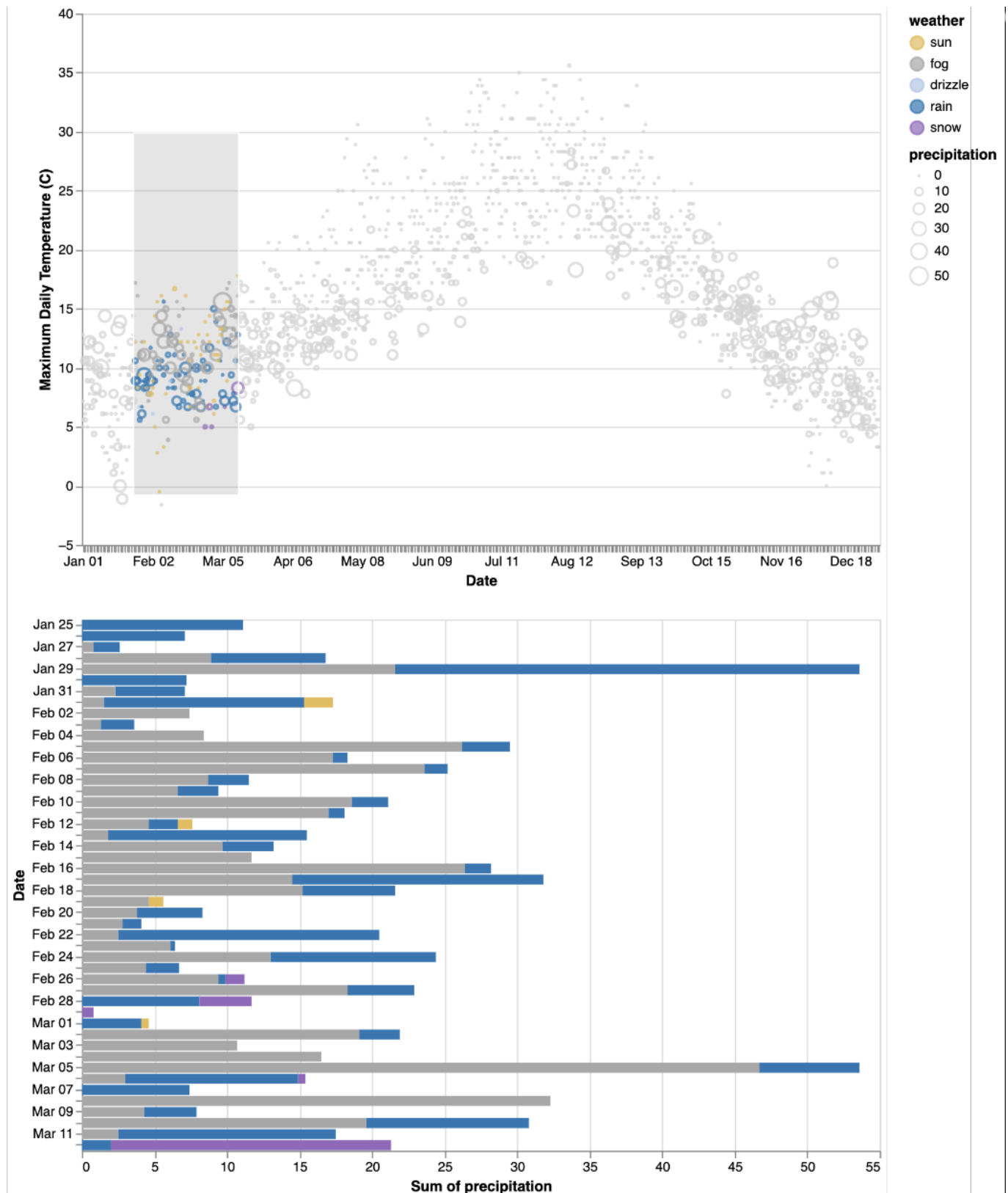
### Part 3: Interval Selection (5 points)

Let's see if we can add another layer of interactivity to the top chart. This time, I want to be able to select an interval of data points on the top chart by clicking-and-dragging to make a selection. Similar to how one might select multiple files on a computer screen.

Your task for this part is to modify the top chart such that if you create a selection of data points by clicking and dragging your mouse, you will see:

1. The bottom horizontal bar chart reflect the data points in the selection only. For example, if I collect a sample of points such that the data from February 10 to February 20 is selected, I should only see Feb 10 to Feb 20 on the y-axis in the bottom chart.
2. All points outside the selection will be grayed out, similar to Part 2.

We should also make it so that if I click on a weather type in the legend, only the points in my selection will have their colors adjusted; all other points outside the selection are still grayed out.



**Grading Criteria:** This part will be graded on the following:

1. Mouse interactivity allows me to select a subset of points on the top chart. The y-axis on the bottom chart reflects only the points in the selection. (2 points)
2. All points outside of the selection are grayed out. Points inside the selection are colored. (1 point)
3. With an interval selected, clicking an item in the legend should affect only the points in the selection. (2 points)

**Hints:**

- To understand how to add interactive elements to an Altair plot, make sure to check out [Altair's user documentation](#), which is available online and gives you insights into how to add particular interactions. Additional searches on websites like Google should give you an idea of how to add mouse interactions to Altair charts.
- Altair allows you to add more than just one condition to a particular characteristic such as color. You can combine conditions together using conjunctions like `|` or `&`.