Boulder Weather Forecast API

—

Ryan and Yousif

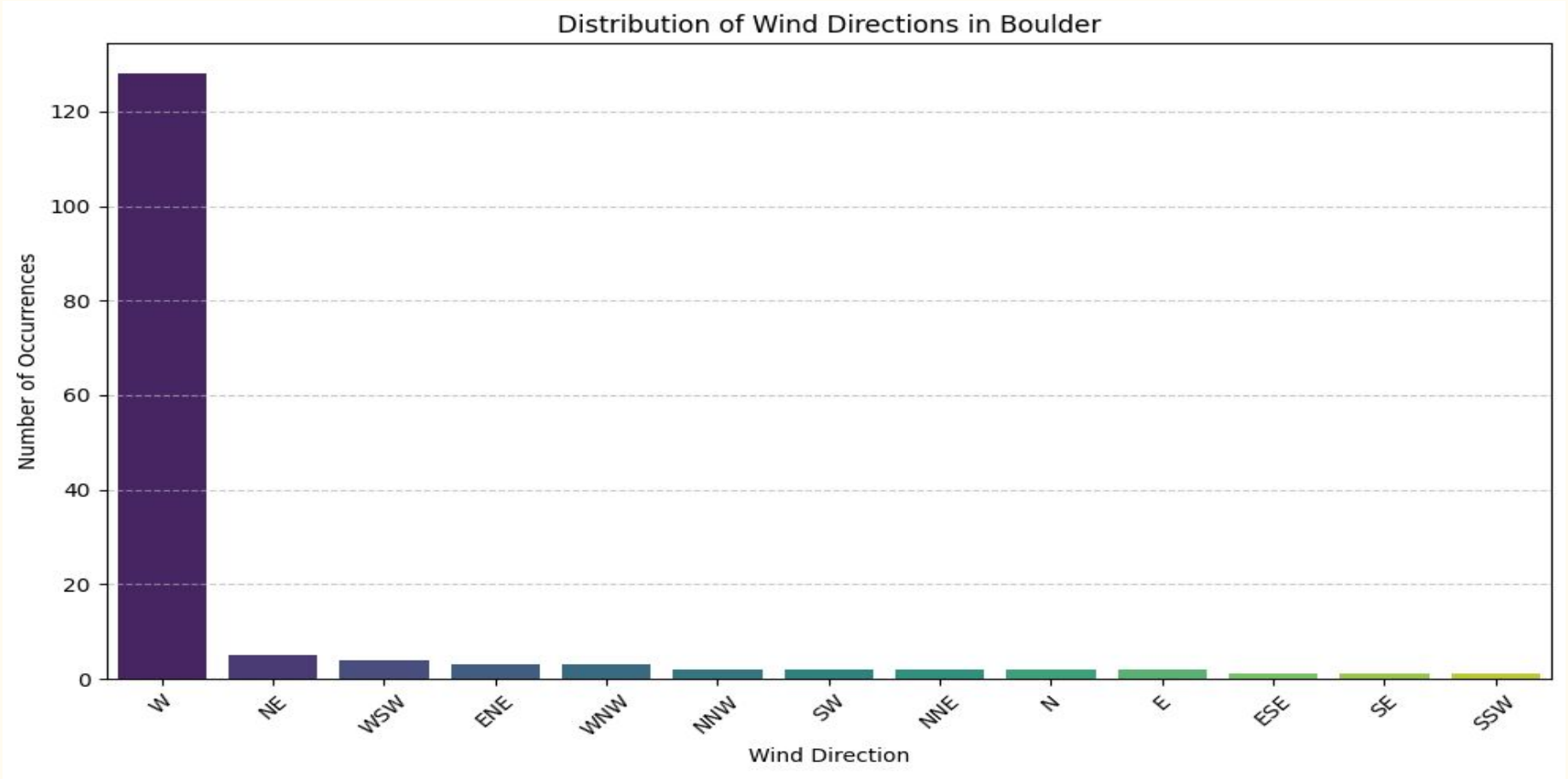
Why Boulder Forecast?

We wanted to choose something that was relatable, not only to us but to everyone sitting in this room. We all experience weather here in Boulder, and we figured that visualizing and interpreting different aspects of the predicted forecast will allow for a greater understanding of our shared habitat. The API endpoint that was chosen highlights an hourly update of the forecast from December 1st to December 7th: the current week that we are in. We chose to do an hourly forecast rather than a 12 hour forecast to truly dissect the small but meaningful shifts that Boulder's weather endures throughout the week.

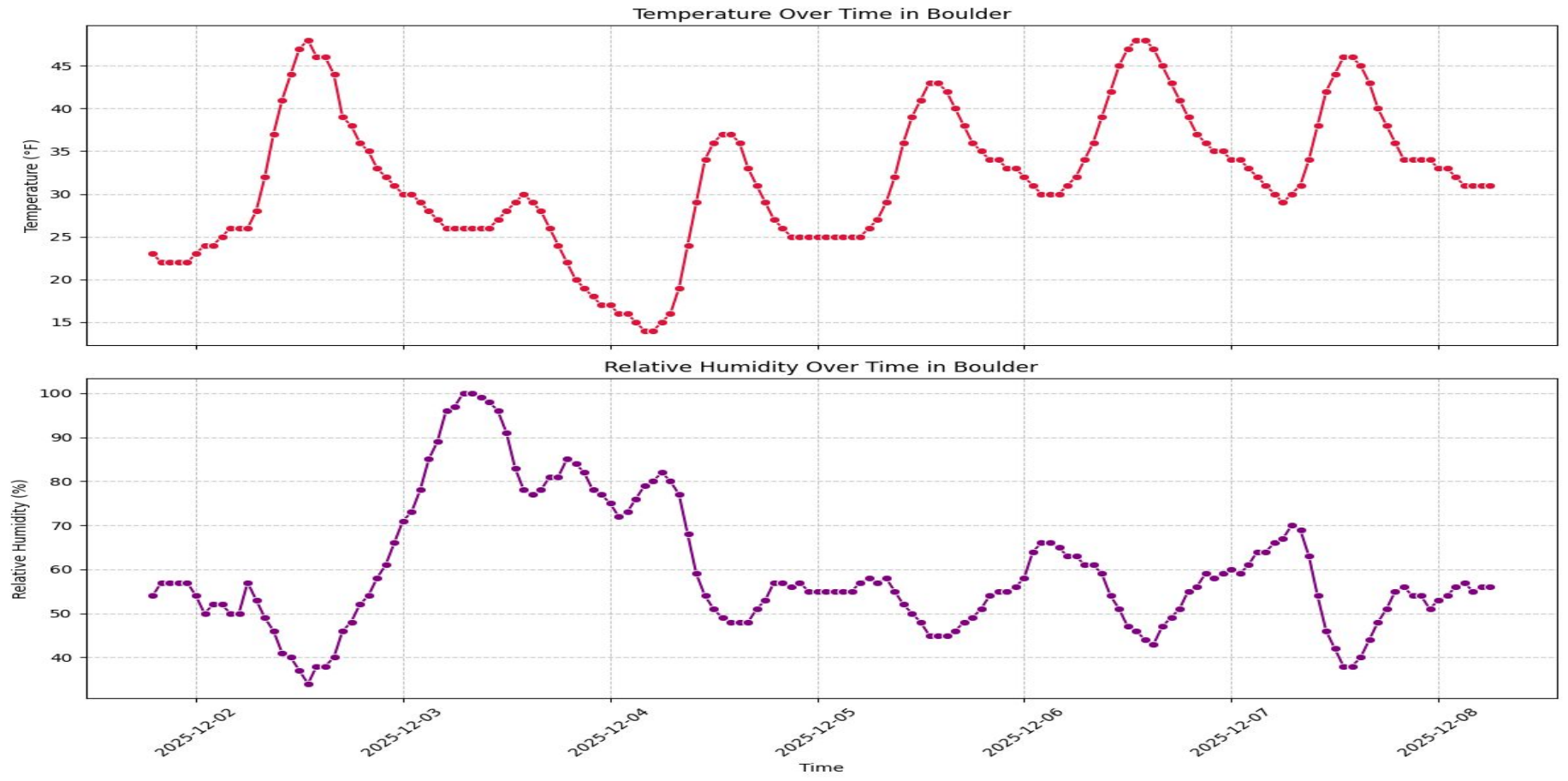
Overcoming Challenges

Although our project is relatively straightforward, we still had to make decisions regarding the direction that we wanted our project to go in. We first investigated the 12 hour forecast, which can be accessed from the same API endpoint, but we ultimately decided that we wanted more data to work with in order to ensure the statistic significance of our findings. Once this was decided, we had to choose variables that were both non-constant over the time period, but also familiar and understandable to a large audience. Among the variables chosen, some of the most notable are the temperature, the probability of precipitation, and the humidity. Once chosen, the final step of our data visualization process emerged: we had to decide which variables to compare against each other for a well rounded and readable visualization.

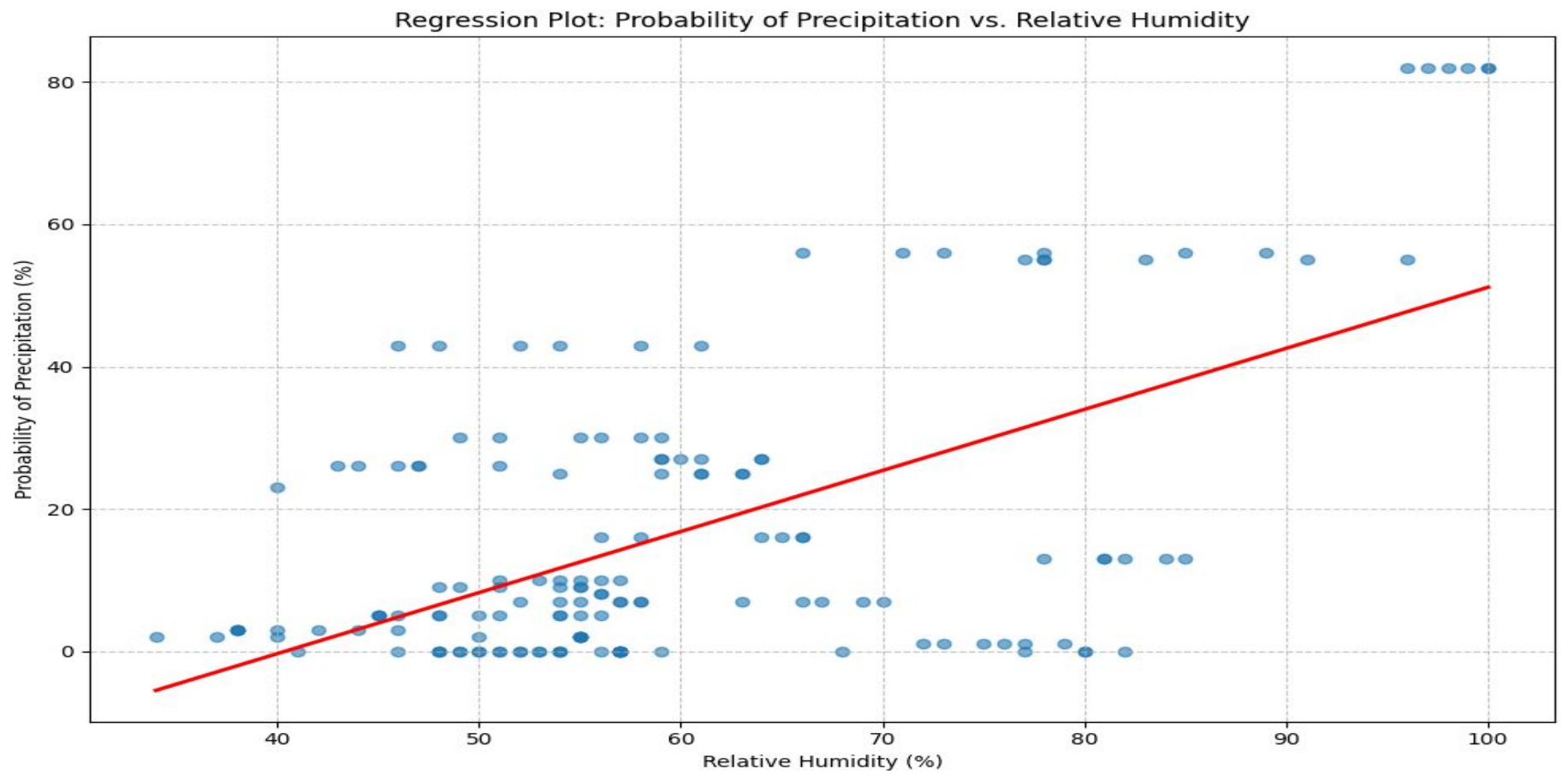
Frequencies of Wind Direction



Time vs Temperature & Time vs Humidity



Correlation between Humidity and Precipitation



Limitations

Although the weather API provides many interesting variables that are useful in data visualization, we as future information scientists always prefer more than less. Our variables were limited, and it would have been helpful to have additional variables such as percentage of cloud cover, air pressure, solar radiation, etc. In addition to this, it would also be nice to have a yearlong forecast. Although impractical, this would allow us to have more data observations, and also account for seasonal variables that change slowly over time. In a perfect world where data was infinitely plentiful, we could visualize many different combinations of these variables. This combined with a much larger amount of observations could greatly improve the statistical significance of our findings, which is practically the only area our project could improve in.

How Does Programming Simplify Data?

Atmospheric scientists deal with complex systems found in nature, but programming helps break that complexity into patterns we can actually interpret. Hourly weather data can look overwhelming, yet code lets us clean, organize, and visualize it in ways that reveal meaningful atmospheric behavior. In this project, programming turned raw humidity, temperature, and precipitation values into clear relationships that help us understand Boulder's weather over the week. And that's where the fun part comes in: Ryan and I ended up being a really complementary duo—my ATOC background explains why these variables behave as they do, while his data science perspective focuses on how to process and visualize them. Programming bridges both worlds and makes the data understandable and meaningful (something we only realized while making these slides).