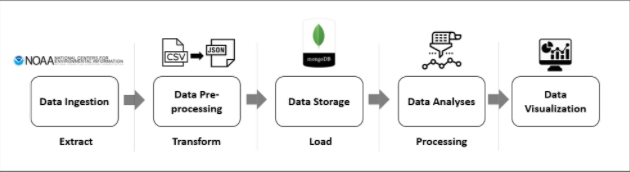
Sreeti Ravi

# Project A Report

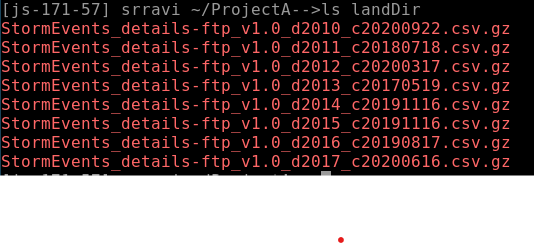
## **Intro**

Data pipelines facilitate the collection of large amounts of data from numerous sources, converting it to a usable format and delivering it to the designated directory where it can be processed to analyze and make decisions or show insights. The purpose of this project was to do that using data from the NOAA’s National Weather Service. The website contains a storms event database that contains data from January 1950 to July 2020. I used the different stages of a data pipeline (download, extract, transform, load, analyze and visualize) to produce a visualization that would show some insights into the data and maybe some trends. I used JetStream to create and configure a virtual machine that had MongoDB, Python, Pandas and Pymongo already installed. Using those libraries and the provided files, I was able to go through the different stages of the pipeline.

## **Pipeline Description and Execution**



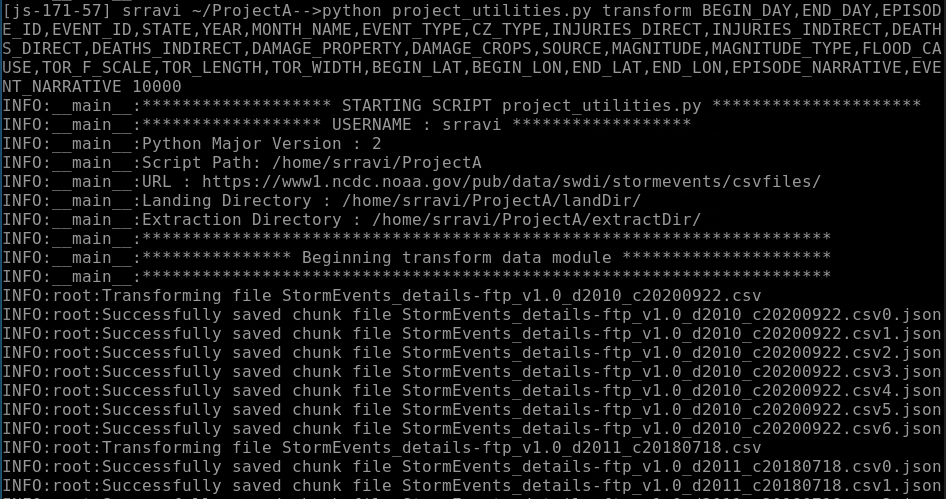
The pipeline had the stages of extract, transform, load and processing. Extracting is the process of downloading and reading data from a database. For this stage, I used the download function from project\_utilities.py to download the CSV files for storm events from the NCDC website for the years specified, which were 2010 to 2017. The data was downloaded as gz files into a landing directory.



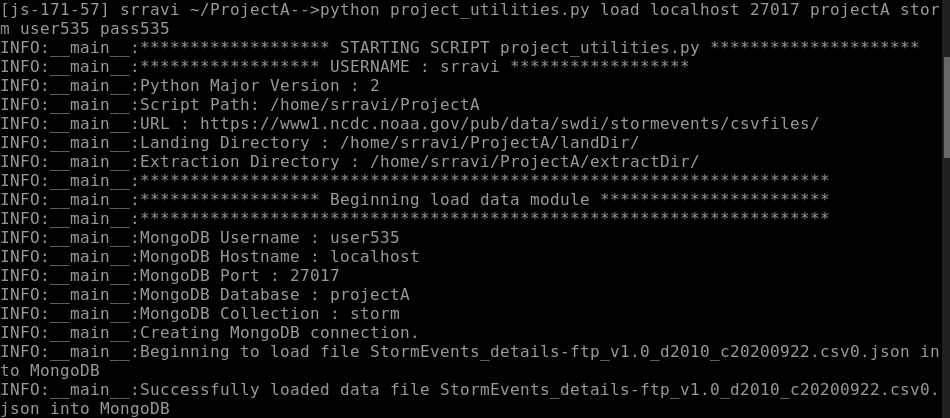
The CSV files were then extracted from each gz file into an extraction directory.

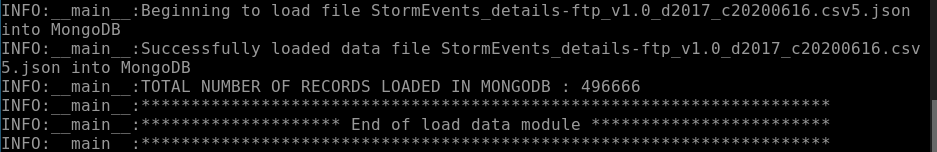


The next stage is transform and this is the process of converting the extracted data from its previous form into the form it needs to be in so it can be placed into another directory or database. The purpose of this stage was to transform the data from CSV into JSON, remove null key-value pairs and only keep the columns that were necessary. After specifying the columns and chunksize, the transform data was saved into the extraction folder.

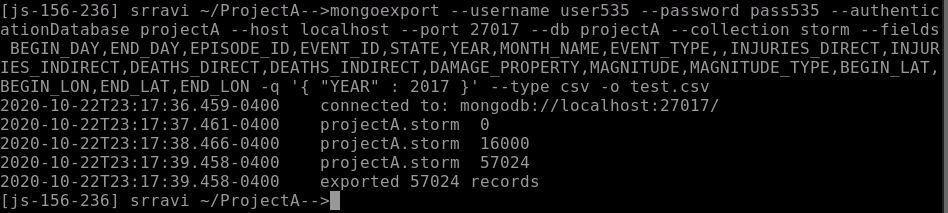


The next stage is load and this is the process of moving the transformed data into the target database, which was MongoDB. This was accomplished by using the load function and the pymongo module and specifying the hostname, port, database, collection, username and password.





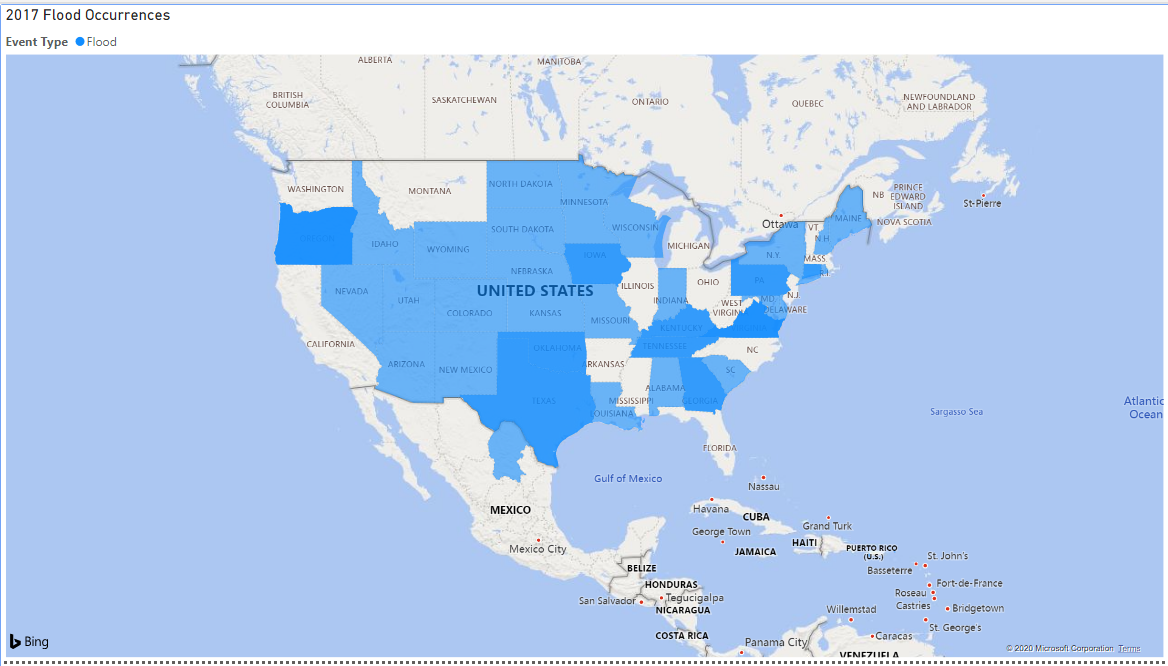
The next stage is processing, and this is where the data file was generated by running a query. I practiced with inserting and updating records in the collection and deleting and querying from the collection. After practicing querying in MongoDB, I chose to run a query that included the columns I wanted for 2017. I chose this because I wanted to look at all the major storm events that occurred that year to see if there were any trends in location.

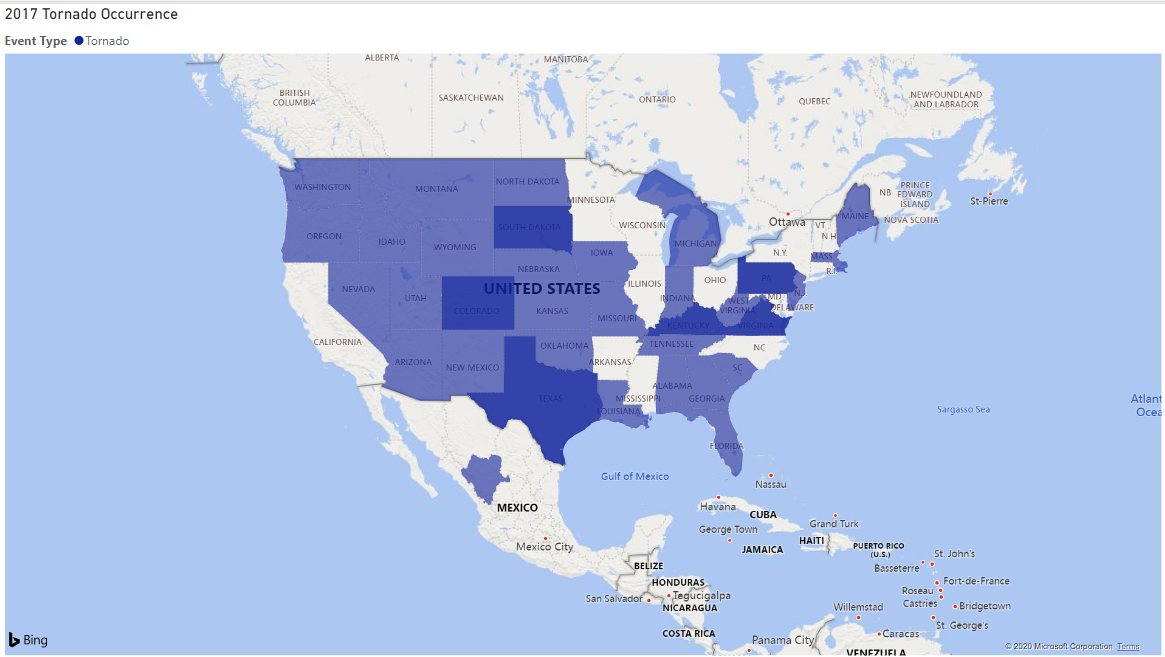


The final stage is to create a data visualization. Data visualizations can be powerful when done correctly. They help present a large amount of data in an easier form to understand by showing trends and outliers. It reduces noise and is a great way to represent data to anyone. I chose to use PowerBI for the visuals I created because I had the most experience with it. I wanted to look at any trends in location for the tornados and hurricanes. I chose these two storm events because they both had latitude and longitudes, while some of the other events were not. Out of all the choices between storm events that did have latitude and longitude, I decided on these two because these were the two that we normally hear the most about.

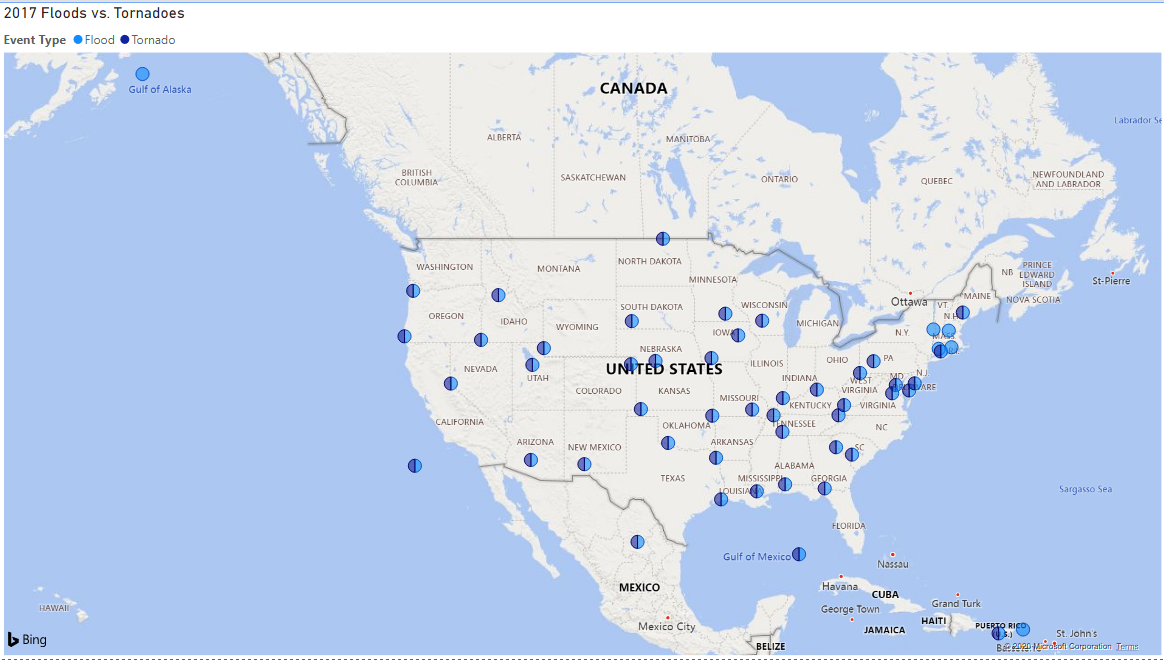
## **Visualizations and Results**

For my visualization, I used Power BI to show where the occurrences of tornados and floods were located. I filtered the storm events to just floods and tornadoes because those two had latitudes and longitudes. I put the two on separate maps to see where they were occurring on a map.





I realized the two had a lot of similarities on where they occurred, so I placed both of them on a different map visual that would show locations of both together. The visualization I used above let me put two different events on the map, but you would only be able to see the top color, so I separated the two and created a different visual for both of them combined.



I thought this was fascinating because as you can see a lot of the places had both floods and tornadoes occurring. I was interested in the correlation between the two and did some research on whether floods and hurricanes tend to happen in the same area. I found out that this is a common occurrence and in the last few years, researchers have found that “every year about 400 tornado and flash-flood warnings are issued within 30 minutes of each other for the same area” (Nielsen). This is an example of how visualizations can be powerful. I was able to notice a trend and when plotted together, the correlation was more obvious and I was able to learn some new information about the two storm events occurring simultaneously.

## **Challenges**

I had some challenges with JetStream when I suspended my VM. The first day that I was working on the project, I got to the load stage and the files were taking some time to load. I kept waiting and eventually it got stuck and the VM froze. I could not click in the browser and it would not let me even force quit using ‘ctrl + C’ or ‘ctrl + z’. I had to close the web shell and when I went to open a new web shell, I got a message saying it could not connect, so I looked online for troubleshooting steps. I found the IU JetStream Confluence page and read through the Troubleshooting and FAQ page. From the information on that page, I refreshed my browser and waited 5-10 minutes, redeployed, rebooted and did a hard reboot and the Web Shell would still not open, so I had to delete the instance and start over.

The second time, I got a little bit farther and all my files loaded, but the VM froze after that, so I deleted and restarted again. I thought maybe the VM was timing out if I wasn’t actively using it for a few minutes, but I realized it was a connection issue because the Web Shell would open and give me an error message to try again. The third time, I was finally able to finish all the stages of the pipeline and I suspended the VM that night. When I wanted to work on the project the next day, the VM was stuck on networking, so I waited for over thirty minutes. After that I followed the troubleshooting steps and refreshed the browser, redeployed, rebooted and hard rebooted. After the hard reboot, it was working again. I also had issues connecting to MongoDB after I had suspended the VM once. I would be able to login to MongoDB, but I would not be able to run any commands, such as checking the count of data records or inserting a record, etc. Luckily, I was able to practice all of that the first time I was working through the project. I wanted to run some queries to make sure they were working before I exported files with those queries, but it would not let me and I did not want to redo everything, so I just ran the ‘mongoexport’ query and would read the error messages if the query did not work.

## **Conclusion and Suggestion**

A suggestion for improving the pipeline is to include validation in any of the steps, but probably during processing. While working on my visualization, I ran into issues with not all of the events having a latitude and longitude. I did not realize this until I was working on my visualization during the last step of the pipeline. It would have been better to figure out what I was going to do for my visualization earlier in the process, so I knew what data I would need for it. This would have reduce the amount of data I was loading and exporting earlier, which is usually best practice for optimal speed. After realizing some of the storm events did not have latitude and longitude, I had to change my visual, but the end result showed me a correlation between tornadoes and floods and after some research I learned that they happen together frequently.

## **Resources**

When Tornadoes and Flash Floods Occur Simultaneously by Erik Nielsen. <https://physicstoday.scitation.org/do/10.1063/PT.5.4020/full/#:~:text=Until%20recently%2C%20most%20research%20has,flash%20floods%20as%20isolated%20events.&text=Our%20research%20reveals%20that%20every,was%20more%20frequent%20than%20expected>.

Mongoexport help.

<https://docs.mongodb.com/manual/reference/program/mongoexport/>

<https://coderwall.com/p/ejxgjg/mongoexport-query-with-using-date>

NOAA website. <https://www.ncdc.noaa.gov/stormevents/details.jsp>