An Analysis of Historical Storm Data

By: Stephanie Brannen, John Singson, Sophia Muller, and Crosby Burdon

AKA: Rock Me Like a Hurricane

Project Goals/Summary

- Core Hypothesis: Hurricanes and Tropical Storms have gotten stronger and more numerous in the last few decades.
- Utilize Atlantic Hurricane Database Data from the National Hurricane Center to study storm characteristics
  - https://www.nhc.noaa.gov/data/#hurdat
- Analysis Questions:
  - Have Hurricanes and Tropical Storms gotten stronger in the last 35 years? -Results mixed.
  - Have they gotten more numerous? -Results trend towards yes.
  - Do storms of greater strength last longer? -Results indicate yes.
  - What cities have the highest frequency of storms making landfall? -Havenlock, NC and Tallahasee, FL
  - Of those cities, which ones are near the largest storms? -Wilmington, NC
  - What part of hurricane season has the most storms? -Late August - September

Background Data Information

- Primary Characteristics of a Hurricane (measured every 6 hours):
  - Start/End Date - For storm duration and whether it occurred during Hurricane Season.
    - June-November
  - Maximum Sustained Wind Speed (knots-mph) - measured inside the eyewall, represents average over one minute. Used to determine storm category with Saffir-Simpson Scale.
    - Tropical Storm - 39-73 mph
    - Category 1 - 74-95 mph
    - Category 5 - >156 mph
  - Barometric Pressure (mbars) - measures atmospheric pressure inside the storm. The lower the pressure, the stronger the storm. Standard atmospheric pressure is 1013.25 mbars.
    - Category 1 - >980 mbars
    - Category 5 - <920 mbars
  - Location (Latitude/Longitude in degrees) - location of the center (eye) of the storm.
  - Landfall - indicates when the storm hit land. Use landfall location with CitiPy to get nearest city.

Data Cleanup & Exploration Challenges

- Data was organized with the storm ID and name as a header row before each storm's data.
  - Solved by creating a loop that looked for header row as a variable and appended to a list with the rest of the storm's data.

```
AL092011,            IRENE,     39,
20110821, 0000,  , TS, 15.0N,  59.0W,  45, 1006,  105,    0,    0,   45,    0,    0,    0,    0,    0,    0,    0,    0,
20110821, 0600,  , TS, 16.0N,  60.6W,  45, 1006,  130,    0,    0,   80,    0,    0,    0,    0,    0,    0,    0,    0,
20110821, 1200,  , TS, 16.8N,  62.2W,  45, 1005,  130,    0,    0,   70,    0,    0,    0,    0,    0,    0,    0,    0,
20110821, 1800,  , TS, 17.5N,  63.7W,  50,  999,  130,   20,    0,   70,   30,    0,    0,    0,    0,    0,    0,    0,
20110822, 0000,  , TS, 17.9N,  65.0W,  60,  993,  130,   30,   30,   90,   30,    0,    0,   30,    0,    0,    0,    0,
20110822, 0600,  , HU, 18.2N,  65.9W,  65,  990,  130,   60,   60,   90,   40,   25,   20,   35,   25,    0,    0,    0,
20110822, 1200,  , HU, 18.9N,  67.0W,  70,  989,  160,   60,   60,   90,   40,
20110822, 1800,  , HU, 19.3N,  68.0W,  75,  988,  160,   60,   40,   90,   40,
20110823, 0000,  , HU, 19.7N,  68.8W,  80,  981,  160,   70,   50,  100,   70,
20110823, 0600,  , HU, 20.1N,  69.7W,  80,  978,  180,  120,   90,  130,   90,
20110823, 1200,  , HU, 20.4N,  70.6W,  80,  978,  180,  120,   90,  130,   90,
```

	Storm_Id	Name	Date	Year	Status	Latitude	Longitude	Windspeed	Pressure	Landfall
0	0000000	UNNAMED	06-25-1851 000000	1851	HU	28.0N	94.8W	80	-999	
1	0000000	UNNAMED	06-25-1851 060000	1851	HU	28.0N	95.4W	80	-999	
2	0000000	UNNAMED	06-25-1851 120000	1851	HU	28.0N	96.0W	80	-999	
3	0000000	UNNAMED	06-25-1851 180000	1851	HU	28.1N	96.5W	80	-999	
4	0000000	UNNAMED	06-25-1851 210000	1851	HU	28.2N	96.8W	80	-999	L

Data Cleanup & Exploration Challenges

- Getting start and end date for entire storm while at least at Tropical Storm strength.
  - Solved by creating dataframe with min date, max date, and duration. Merge new columns to original file.
- Get nearest city for storms that made landfall.
  - Solved by creating dataframe to get only storms that reached landfall and used CitiPy Library to get nearest city with new columns: Landfall lat, long, max wind speed, and nearest city. Merged to original dataframe.

```
# Get min date (when storm became Tropical Storm) and Max Date (When storm is no longer a tropical storm)
# Calculate the duration

storm_gb = hurricane_df_clean.groupby('Storm_Id')
storm_sgb = storm_gb['Date']
start_date = storm_sgb.min()
end_date = storm_sgb.max()
duration = end_date - start_date
```

```
#Merge Start Date, End Date, and Duration to original dataframe.
start_end_df = pd.DataFrame({"Start Date": start_date
                             ,"End Date": end_date
                             ,"Duration" : duration
                            }).reset_index()

merge_df = pd.merge(hurricane_df_clean, start_end_df, how="outer", on="Storm_Id")

merge_df.head(100)
```

Data Cleanup & Exploration Challenges

- CitiPy doesn’t accept latitude and longitude coordinates that include cardinal direction.
  - Solved by removing cardinal directions from data and using negative numbers to indicate W and S.

```
# Use citipy to find the nearest city
landfall_df.loc[:, "Latitude"] = lats
landfall_df.loc[:, "Longitude"] = lons

# Change the column to numeric
landfall_df["Latitude"] = pd.to_numeric(landfall_df["Latitude"])
landfall_df["Longitude"] = pd.to_numeric(landfall_df["Longitude"])

# Convert Longitude column to negative
landfall_df["Longitude"] *= -1


# Use citipy to find the nearest city
latitude = landfall_df["Latitude"]
longitude = landfall_df["Longitude"]
coordinates = zip(latitude, longitude)
cities = []
for coordinate_pair in coordinates:
    lat, lon = coordinate_pair
    cities.append(citipy.nearest_city(lat,lon))
```

Data Cleanup & Exploration Challenges

- Get all values from row with max wind speed for each storm.
  - Solved by googling and finding idxmax() function to allow for this.

```
# Find row with max windspeed and return all columns in that row.
clean_storm_df =
storm_added_fields_df.iloc[storm_added_fields_df.reset_index().groupby(['Storm ID'])["Max Windspeed"].idxmax()]
clean_storm_df.head()
```

https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.idxmax.html

Data Visualization Challenges

- Original dataset was intended to only cover 15 years, but plots of storm variables showed no clear trends over that time period.
  - Solved by increasing it to 35 years to make the trends clearer.
- Couldn’t filter storms by category using a >= conditional, since they were initially named with a string format
  - Solved by adding a new column to the data frame indicating category with an integer. This was achieved by binning.

```
# Create Bins for each storm category according to https://en.wikipedia.org/wiki/Maximum_sustained_wind
min_wind = clean_storm_df["Max Windspeed"].min()
print(min_wind)
bins = [33, 63, 82, 95, 112, 136, 170]

# Create the names for the four bins
category_names = ['Tropical Storm', 'Category One', 'Category Two', 'Category Three', 'Category Four', 'Category Five']
category_values = [0,1, 2, 3, 4, 5]

# Create new category column
storm_category = pd.cut(clean_storm_df["Max Windspeed"], bins, labels=category_names)
category_value = pd.cut(clean_storm_df["Max Windspeed"], bins, labels=category_values)
```

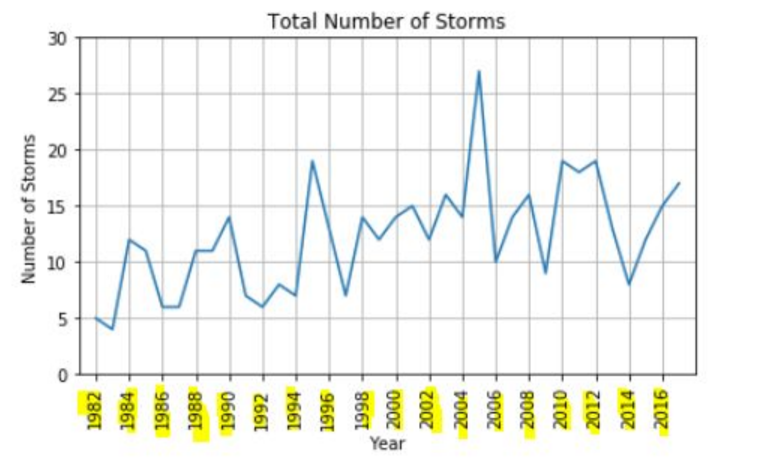
Data Visualization Challenges

- When filtering the storms for category 3 or stronger, some years didn't have any of that strength. This caused them to disappear from the dataset.
    - Solved by first creating a data frame with counts grouped in years. Then adding a new column to said dataframe with the count of storms category 3 or stronger, using a .fillna(0) command on a dataframe containing all the years.

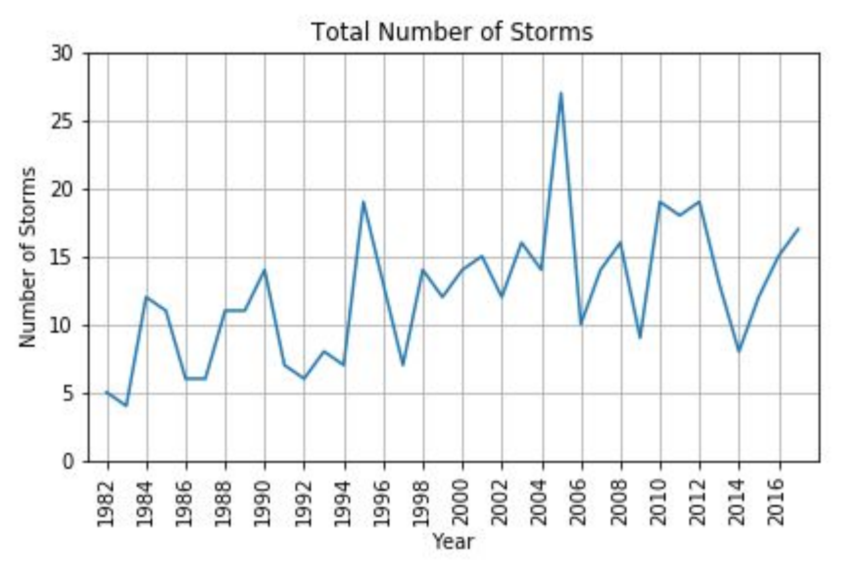
```
# Create df with counts to ensure no year is left behind
count_df = grouped.count()
# Count storms category 3 or stronger
over3_count = clean_storm_df[clean_storm_df['Category Value'] >= 3].groupby(['Year']).size()
# Add count over 3 to count data frame
count_df['Over 3'] = over3_count
# Replace NaN values with zeros
count_df = count_df.fillna(0)
```

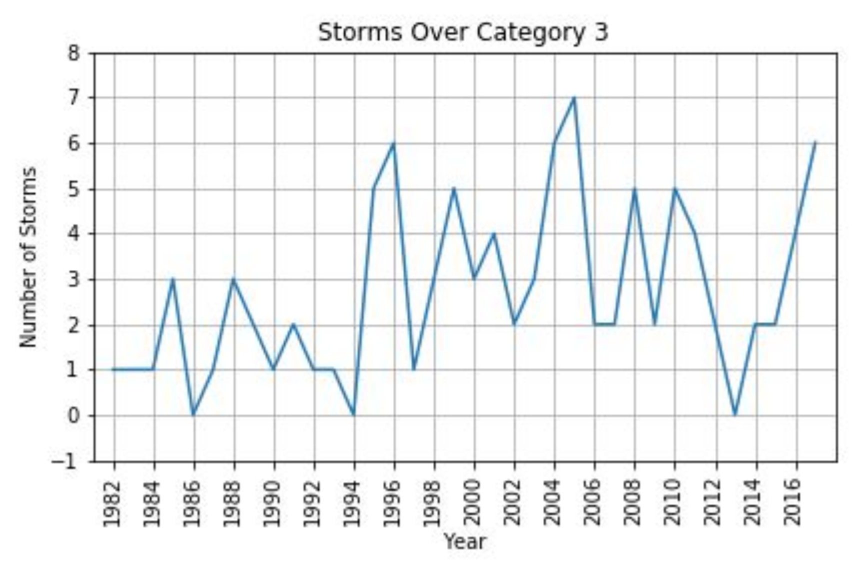
Data Visualization Challenges

- The graph tick labels for years were too crowded and hard to read.
  - Solved by: Setting up the tick labels to only show every two years and making them vertical using the .xticks() command and lengthening the graph by using the .tight_layout() command.

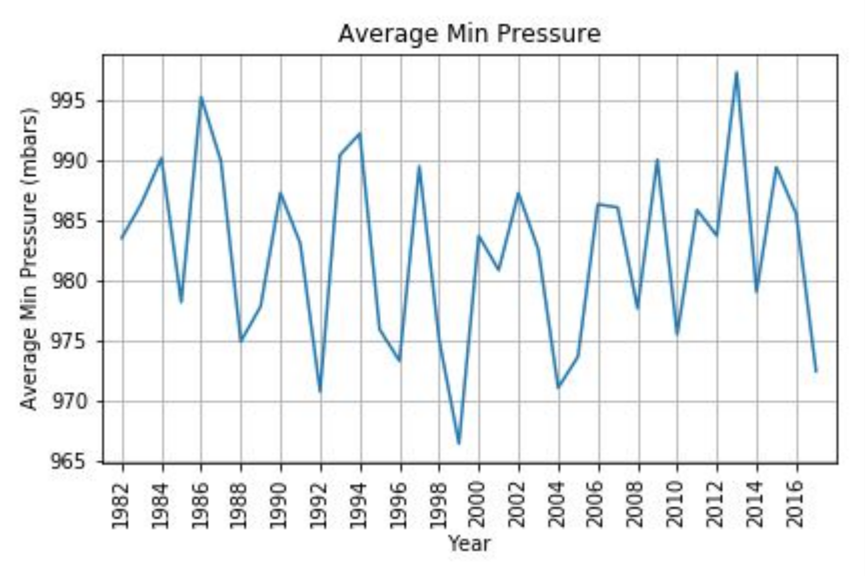


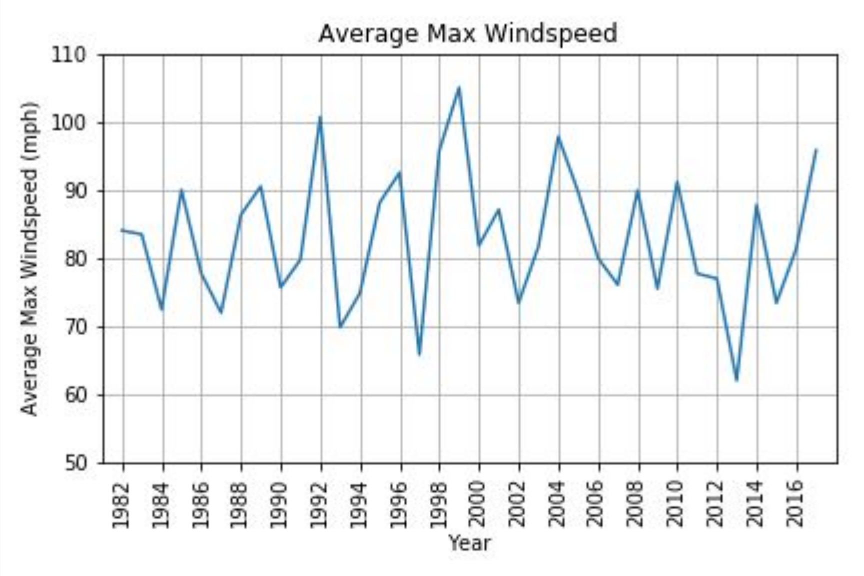
Storm Strength Analysis



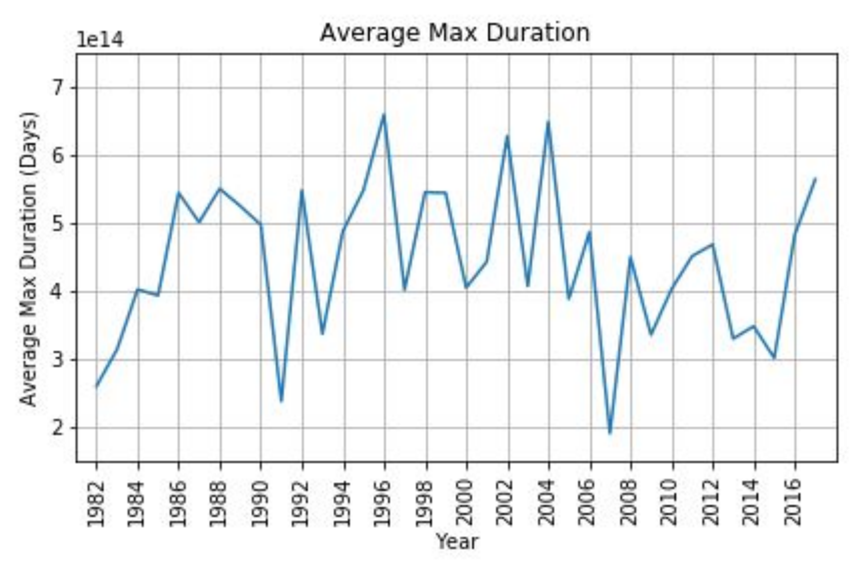


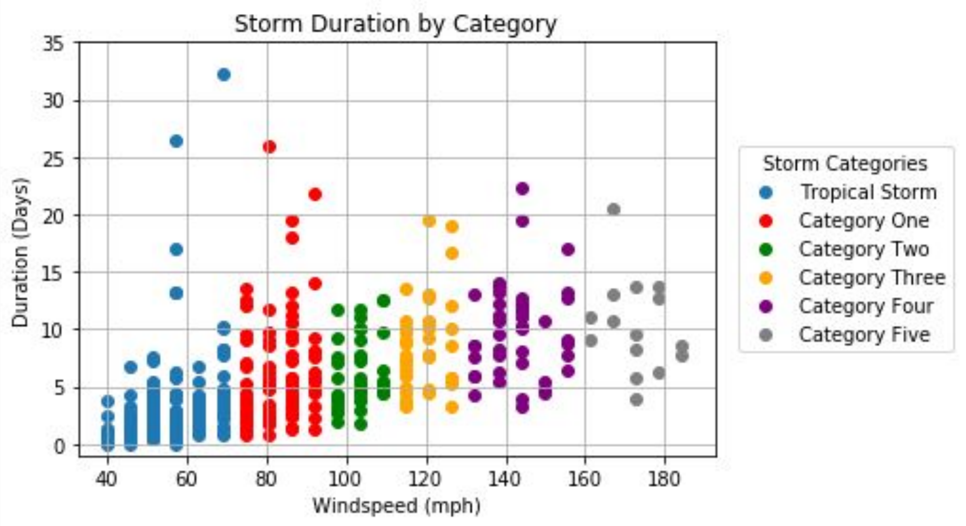
Storm Strength cont.



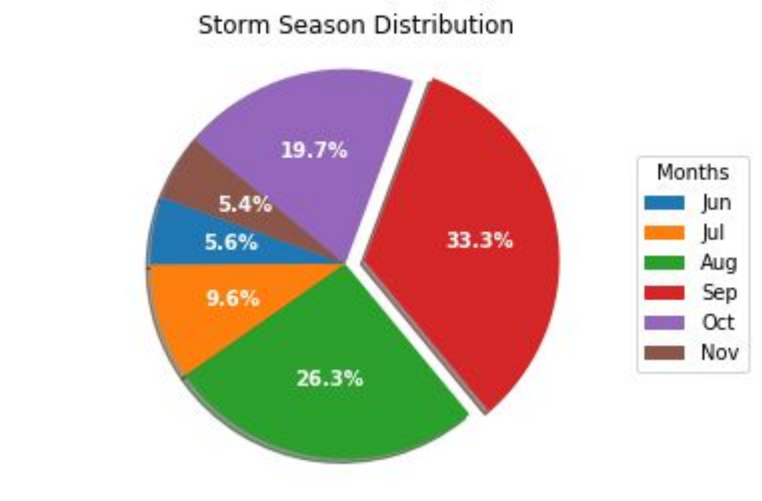


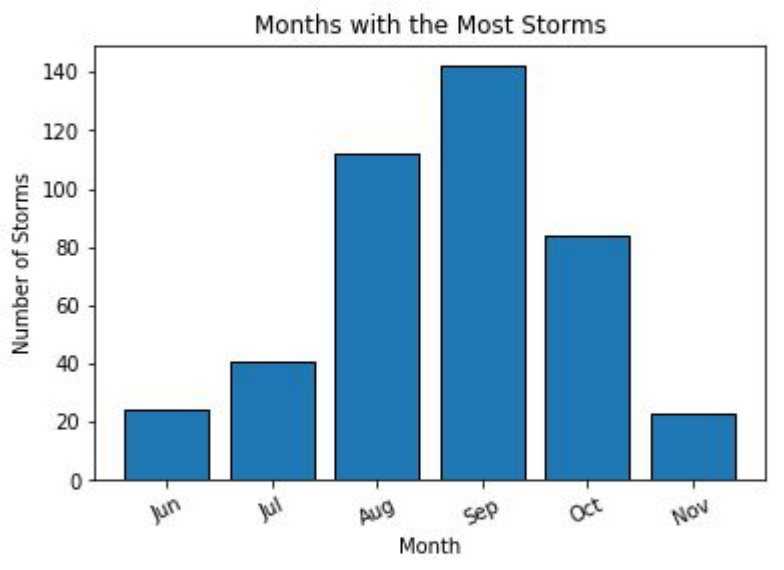
Storm Duration



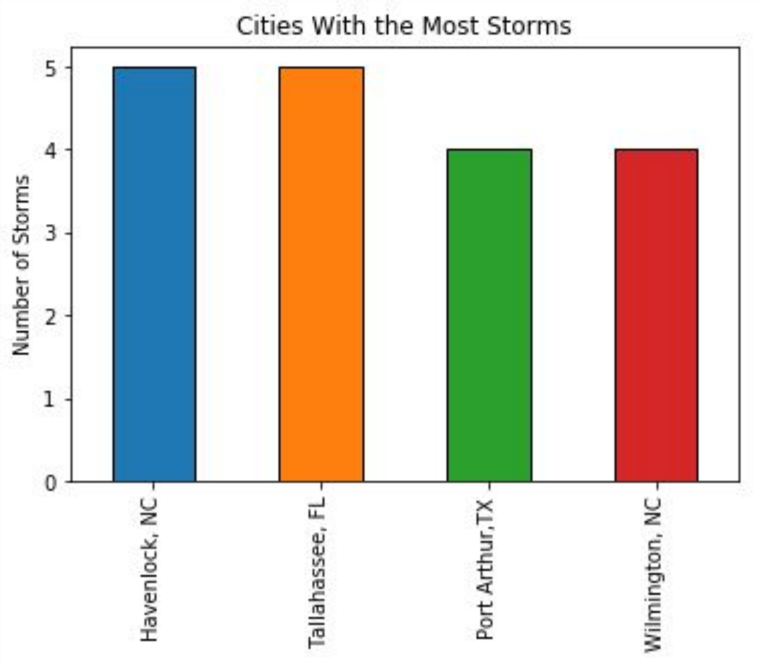


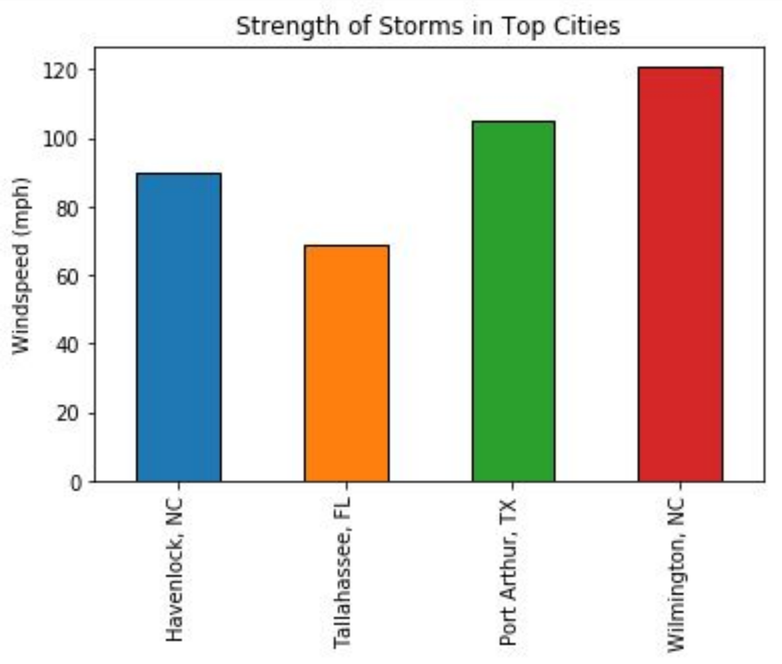
Hurricane Season





Most Storm-Prone Cities





Discussion

1. Storm Strength - number of storms clearly on the rise for all categories, major storms show a positive trend. Trends are not as clear with barometric pressure, though the tendency is for each periodic valley to have a lower minimum value. Max wind speed also has periodic peaks every few years that are increasing in maximum value. Each of the strength-related graphs ends with a trend towards stronger storms in 2017. Expected to see a clearer trend here.
2. Duration - Peak values of year vs. duration graph show increasing trend. Clear increase in minimum storm duration with category.
3. Hurricane Season - September is clearly the most dangerous month with 33% of total storms, but August is a close second with 26.3%. This confirms what we already knew.
4. Hurricane Cities - North Carolina coastal cities (Wilmington, Havenlock) and Tallahassee Florida are the clear winners for number of nearby storms, but Port Arthur Texas is a close second in terms of highest average wind speed.

Post Mortem

- We would be interested to see this dataset over a much longer time period (i.e. 100 years) to get a better idea of the long term trends of storm strength. This would help answer our primary question more clearly.
- Creating a heat map of the most dangerous cities would more efficiently show our results.
- Our current search doesn’t include small coastal towns, doing so would likely change our dangerous city results radically.
- Adding population and FEMA data on affected cities would also be useful to get a better idea on storm damage.
- Specific location data for the northeast quadrant of the storms (most dangerous) when they made landfall would also help with figuring out high risk areas.

Q & A