Final Report

Chengze Li, Zhiqi Yu

December 2024

MERRA2 Data in 1980

Following Sai's instructions, we obtained the following data files:

- `TE_ready_MERRA2_198001.nc`
- `MERRA_filelist.txt`
- `TE_geos.sh`
- `MERRA2.topo.nc`

The specific codes are available in the 9.5 Report. We used the data from `TE_ready_MERRA2_198001.nc` to generate plots and examine its data characteristics. The resulting figures are shown below.

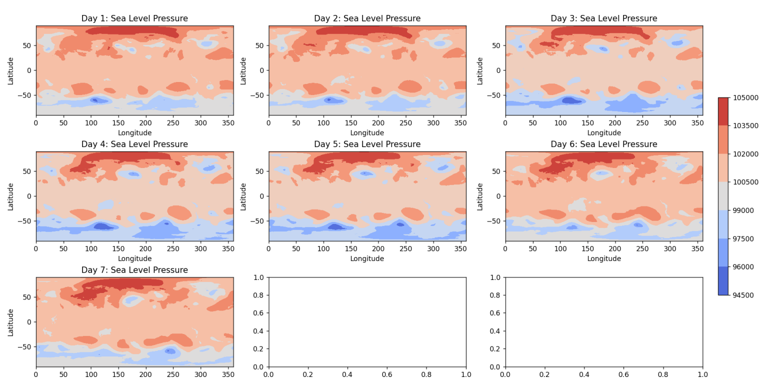


Figure 1: Different variables in dynamic time in SLP, Lon, Lat

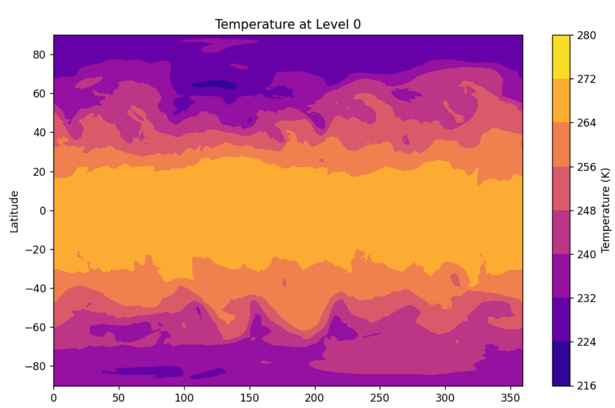


Figure 2: T, Lon, Lat

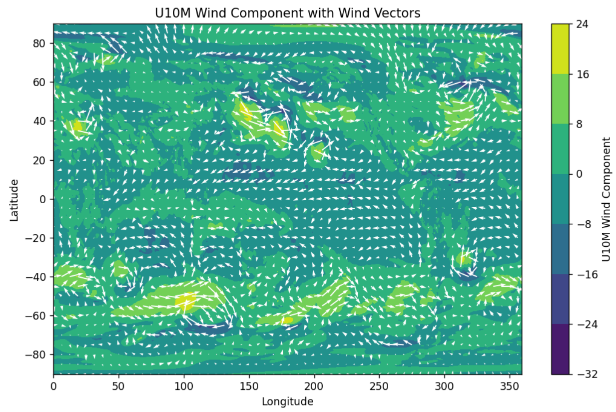


Figure 3: U10M, V10M, Lat, Lon

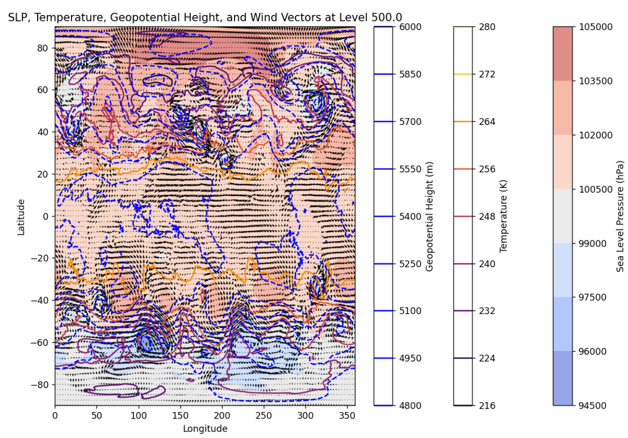


Figure 4: SLP, U10M, V10M, Temperature, Geopotential Height, lev

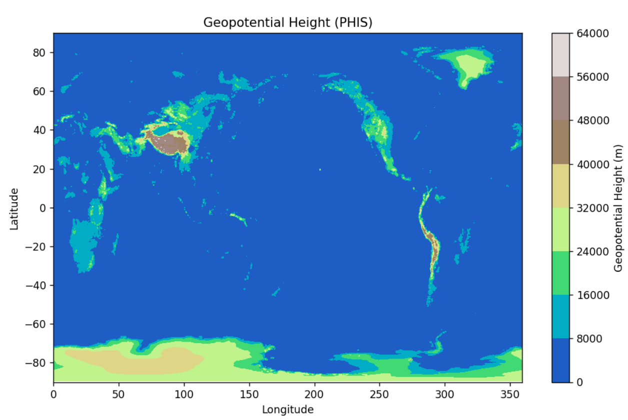


Figure 5: MERRA2.topo.nc Graph

After examining the characteristics of TE_ready_MERRA2_198001.nc, we performed a reanalysis using TempestExtremes. This was accomplished by executing the track_storms_in_1980.sh script available in the 9.12 Report directory. The contents of the input text file are as follows:

```
/home/skompella/MERRA2/TE_data/TE_ready_MERRA2_198001.nc;
/home/skompella/topo/MERRA2.topo.nc;
/home/skompella/topo/MERRA2.topo.nc
```

Upon successful execution, the script generated the file 1980_output_from_Tempestextremes.dat. This output file contains the reanalysis results of MERRA2 for January 1980.

Subsequently, within the 9.19 Report directory, we utilized Convert_data.py to label the data obtained through TempestExtremes. For the already labeled data, we employed the tracktry.py script in 9.26 Report to perform storm tracking analysis, resulting in the generation of storm_tracks_1980.png.

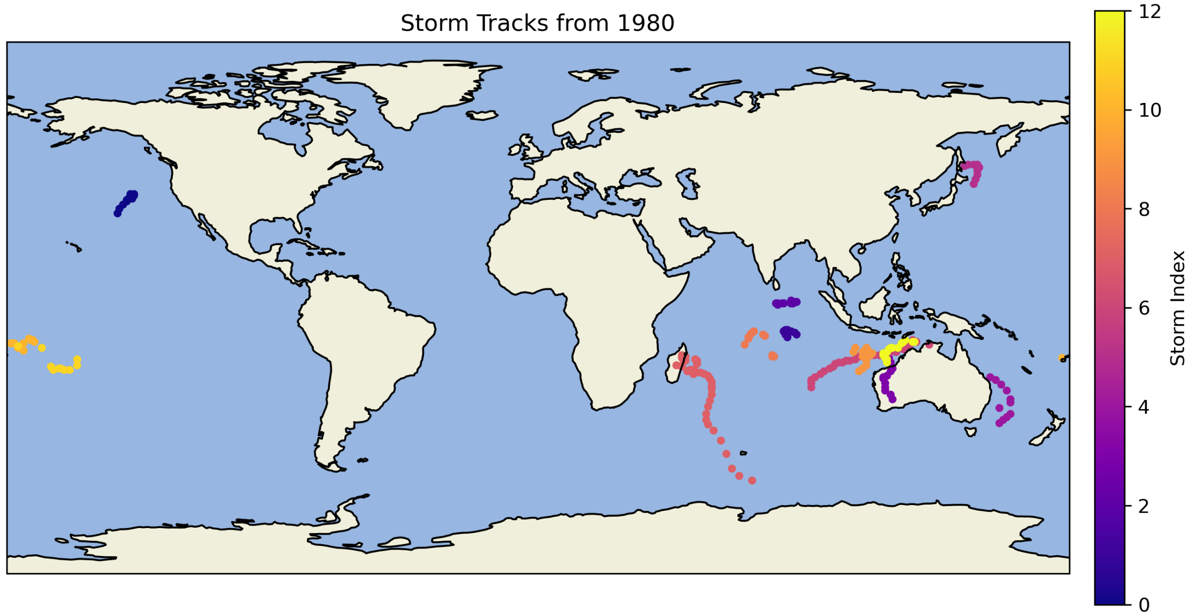


Figure 6: Storm Tracks for January 1980

IBTrACS Data in 2022

We analyzed the IBTrACS data for the year 2022 and classified the data into four key features for visualization:

- **Average Storm Speed**
- **Number of Unique Storms per Month per Basin in 2022**
- **Storm Tracks in 2022 and 1980**
- **Total Number of Unique Storms in 2022**

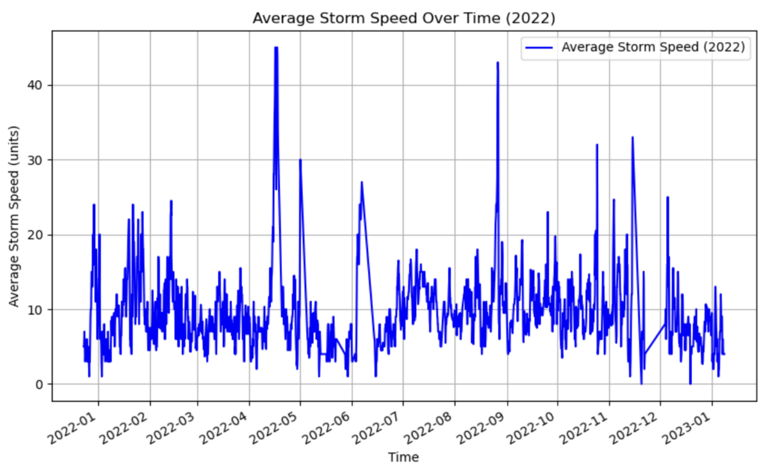


Figure 7: Average Storm Speed

For the number of unique storms per month in 2022, we selected only storms with a maximum wind speed > 34. Using this filtered data, we generated the plots. The detailed steps for generating these visualizations can be found in the Copernicus Climate Data Store documentation and 10.3.py.

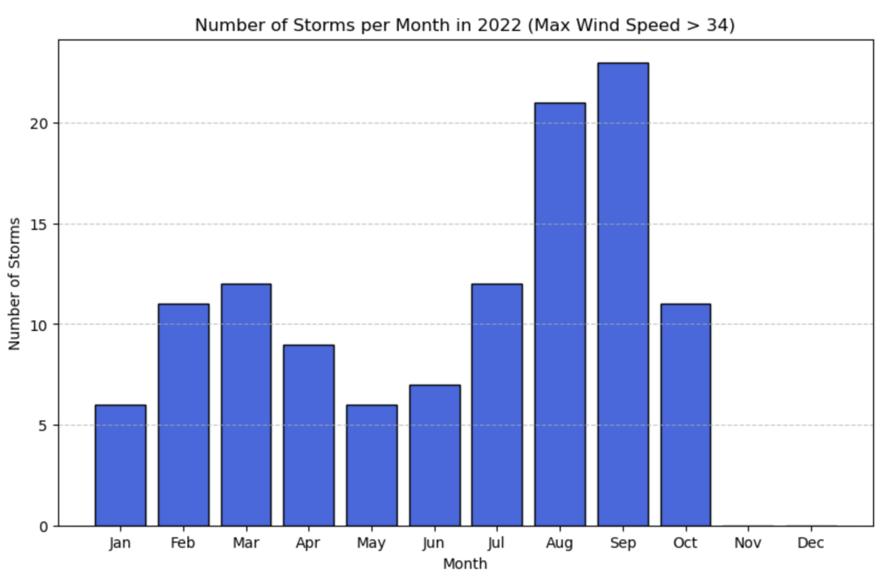


Figure 8: Number of Unique Storms per Month in 2022 ($MaxWindSpeed > 34$)

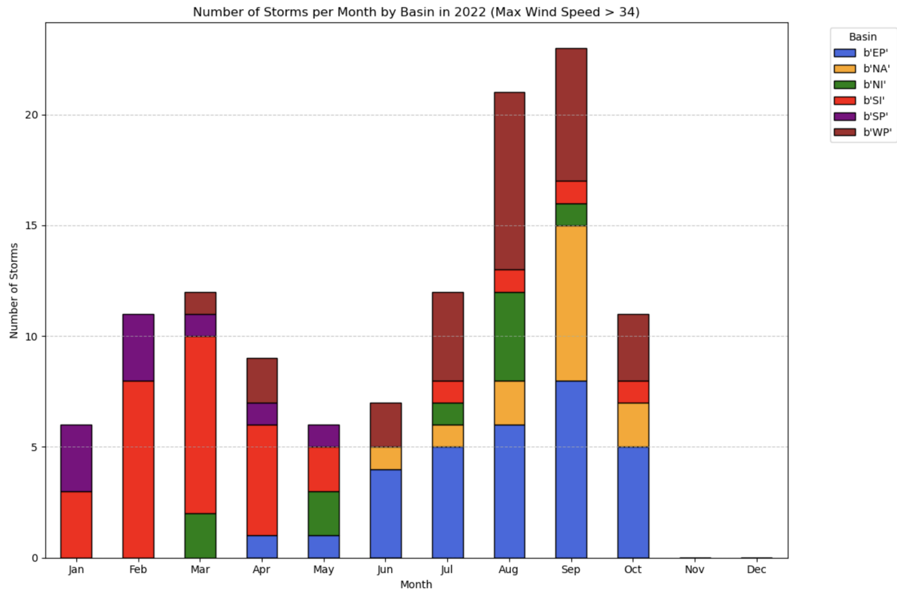


Figure 9: Number of Unique Storms per Month per Basin in 2022 ($MaxWindSpeed > 34$)

We visualized the IBTrACS data for the years 1980 and 2022 to observe the changes and trajectories of storms over time. Additionally, we conducted a longitudinal comparison between the IBTrACS data for 1980 and the MERRA-2 data for the same year. The detailed steps for generating these visualizations can be found in the trackstorm.py script.

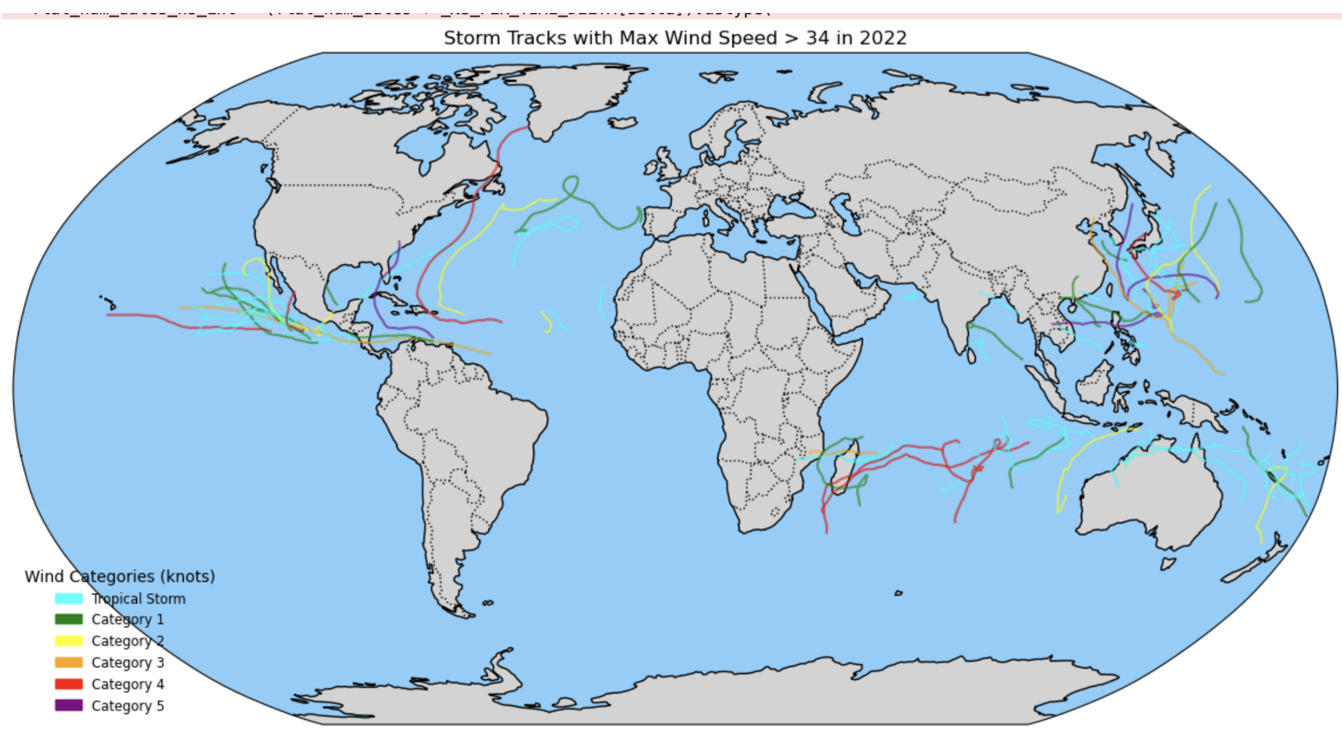


Figure 10: Track of Storms in 2022

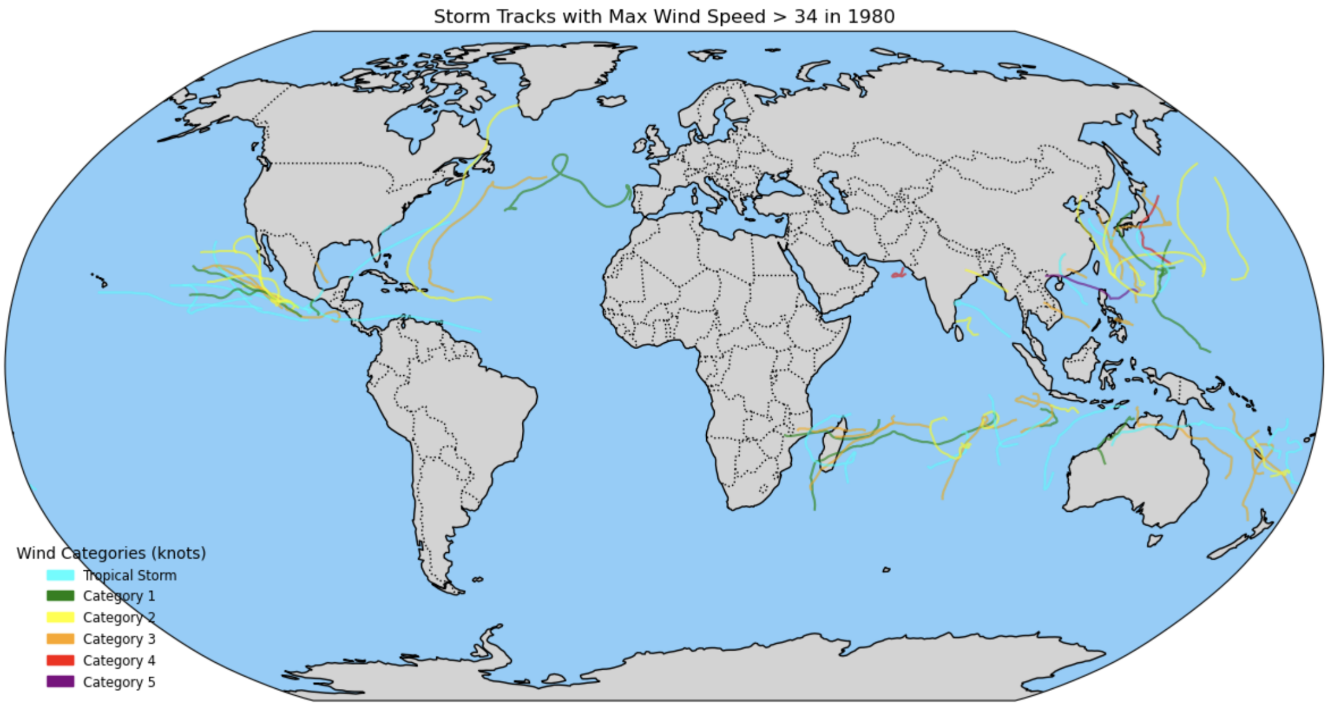


Figure 11: Track of Storms for IBTrACS in 1980

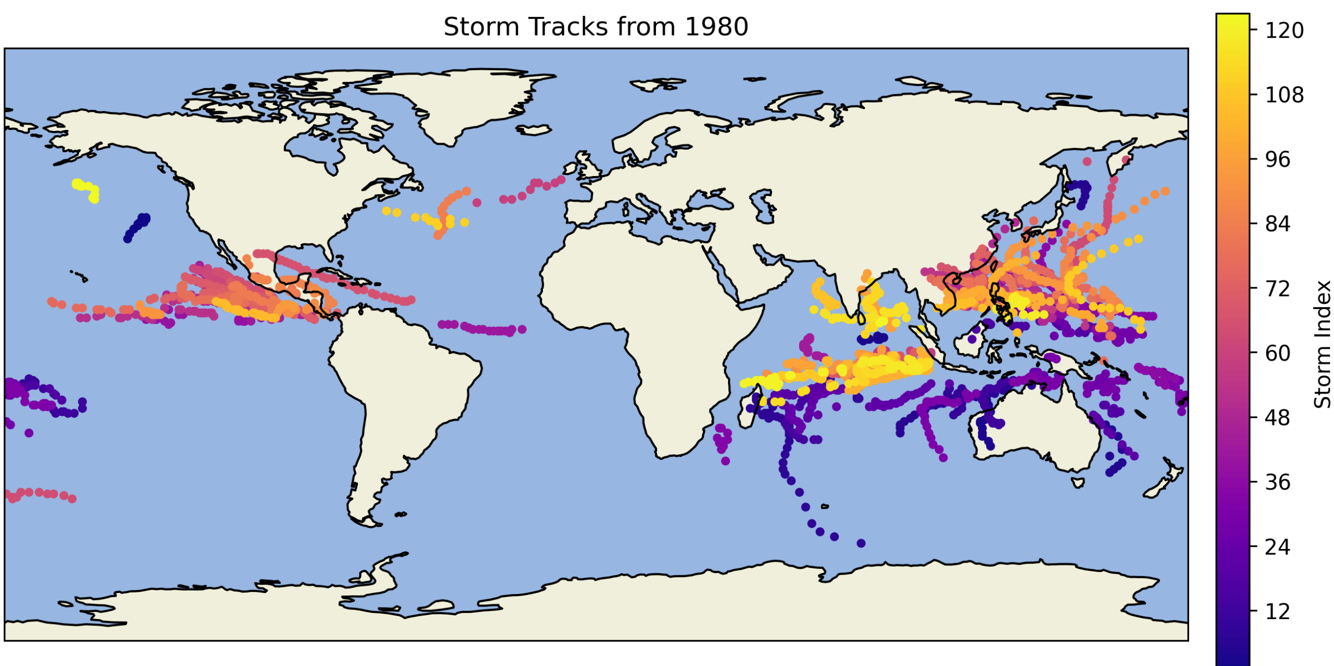


Figure 12: Track of Storms for MERRA-2 in 1980

ERA Data in 2022

We obtained the ERA5 data from the Climate Dataset Store. First, we processed the extracted variables to ensure the dataset contains only the required variables, such as `U10M`, `V10M`, and `SLP`, which are necessary for TempestExtremes reanalysis.

Next, we processed the attributes of each variable by adding essential attributes to ensure that the modified dataset could successfully undergo reanalysis in TempestExtremes. After multiple iterations of adding and removing attributes, we determined that the attribute `unit` is critical for TempestExtremes to function correctly. More details process can be seen in attribute.py. We then extracted the ERA5 data for March 2022 and generated the following visualizations. The detailed procedures and steps can be found in the ERA5 in 2022 directory.

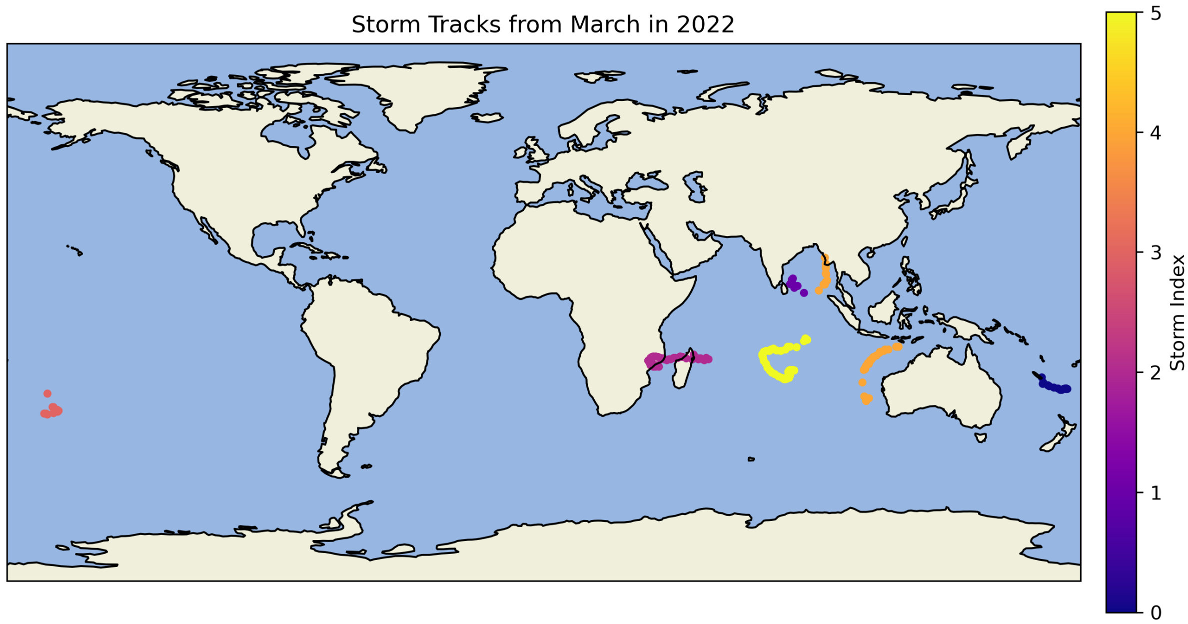


Figure 13: Storms Track for ERA5 in March 2022

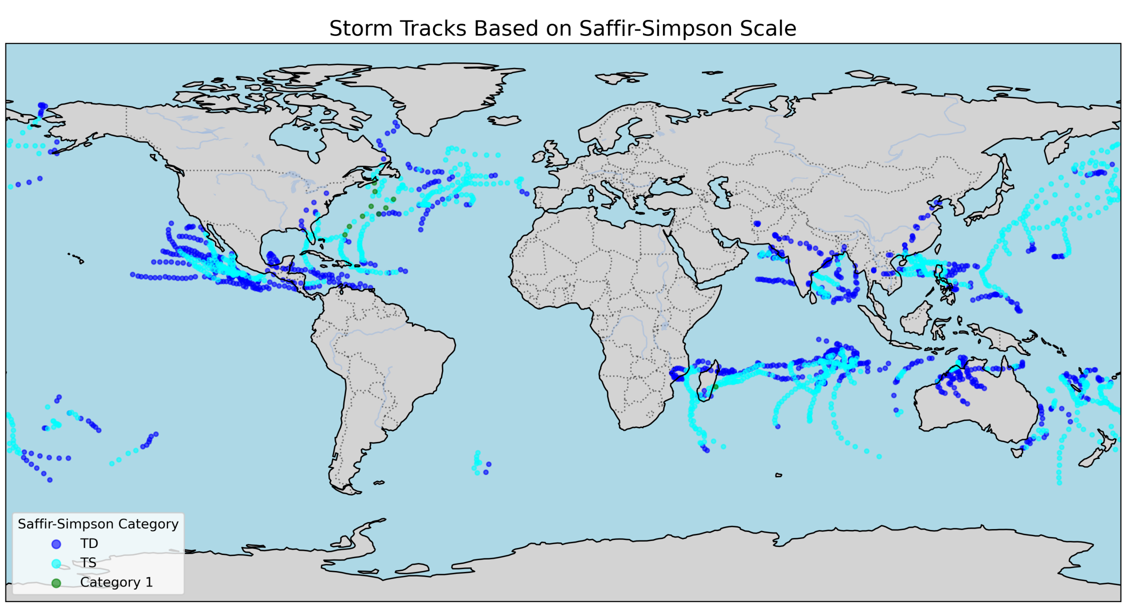


Figure 14: Storms Track for ERA5 in 2022

For Figure 18, after completing the initial adjustments, we obtained the full-year ERA5 data from the Climate Dataset Store. First, we cleaned the data to retain only the required variables and removed unnecessary attributes, ensuring that the `unit` attribute is present for each variable. Then, we input the processed dataset into TempestExtremes for reanalysis. Using the reanalyzed data, we generated visualizations to observe the storm trajectories and their changes over the year. The detailed steps and procedures can be found in the ERA5 Data Processing directory and the drawing strategy is based on Saffir-Simpson criteria.

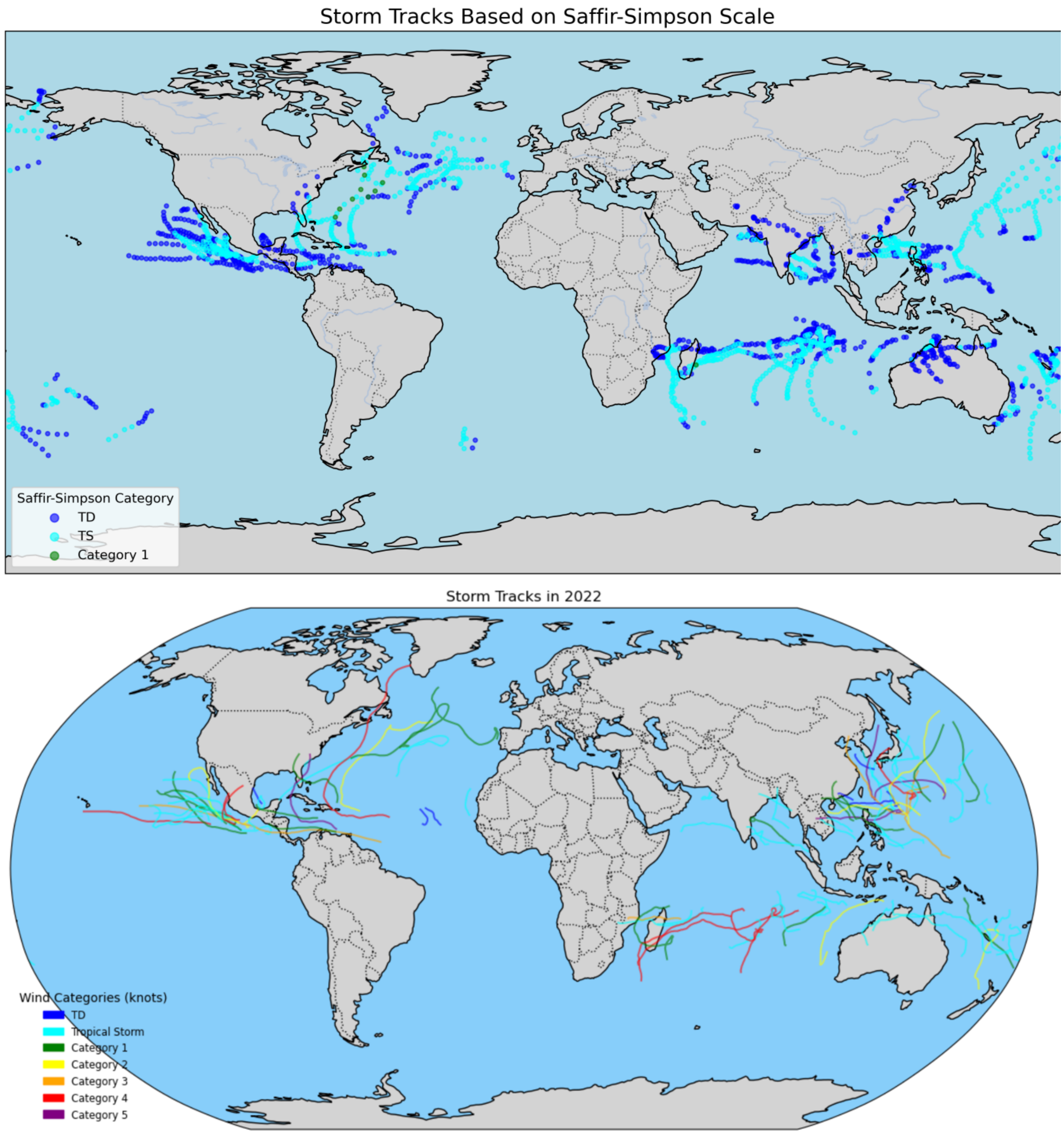


Figure 15: Storms Track for ERA5 in 2022

For Figure 19, we also conducted a longitudinal comparison between the ERA5 data for 2022 and the IBTrACS data for 2022 to observe the similarities and differences between the two datasets based on reanalysis performed using TempestExtremes.

After finishing the processed work, we focused on reproducing EAR5 images from the article Neural General Circulation Models for Weather and Climate. We adjusted the parameters in the shell script based on page 75, section 1.2 of the paper. For specific operations, please refer to the 10.24 report. After that, run the 0.25°_csv_convert.py and 1.4°_csv_convert.py scripts to get CSV data respectively. Finally, we plotted the graph with 1.4-degree and 0.25-degree resolutions via combine.py.

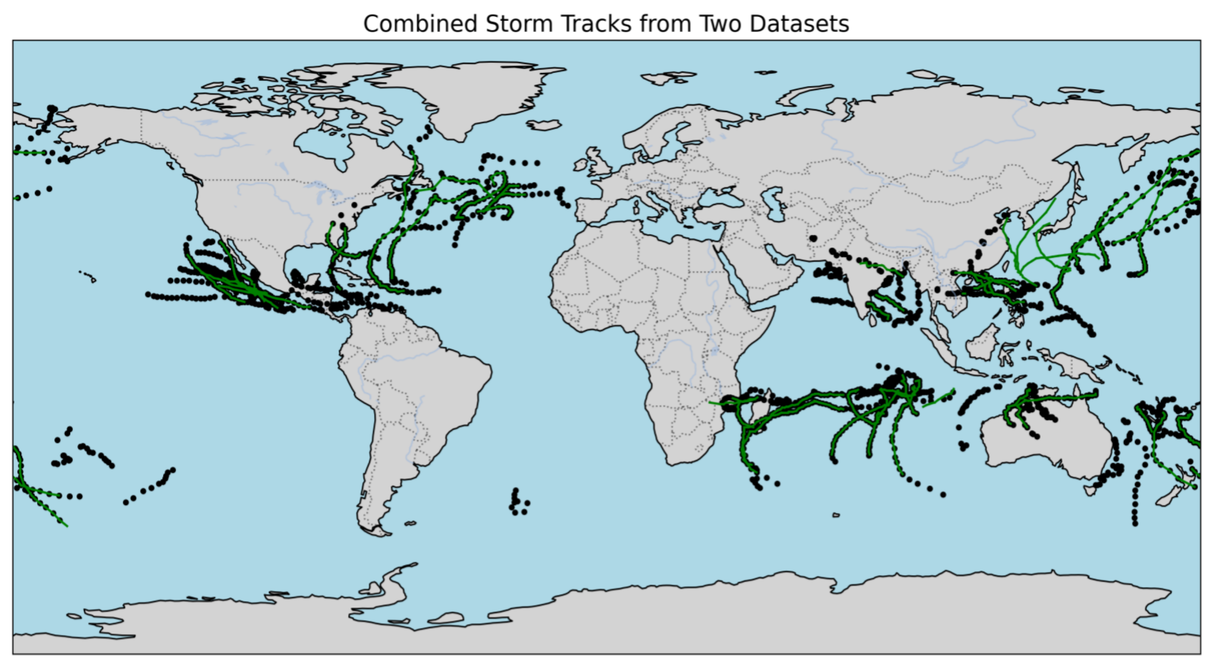


Figure 16: Combined Strom Tracks From Two Datasets

NeuralGCM Data in 2020

Subsequently, we proceeded to visualize the prediction data generated by NeuralGCM for the year 2020. Initially, we extracted the original NeuralGCM data following the guidelines outlined in the Forecasting quick start documentation. We then modified the original data generation script provided in the NeuralGCM repository by adapting it to addingunits_onemonth.py. This modification enabled us to generate all necessary variables for use in TempestExtremes while eliminating redundant variables.

The resulting dataset encompasses the period from July 1 to July 30. For each initial time point within this period, the model generates forecasts 15 days ahead, producing data at four specific times each day (00:00, 06:00, 12:00, and 18:00). It is important to note, as illustrated in Figure 21, that the NeuralGCM output contains several oblique lines. Each of these lines operates independently when running TempestExtremes. Consequently, performing reanalysis with TempestExtremes necessitates executing the process 30 separate times to accommodate each initial time point.

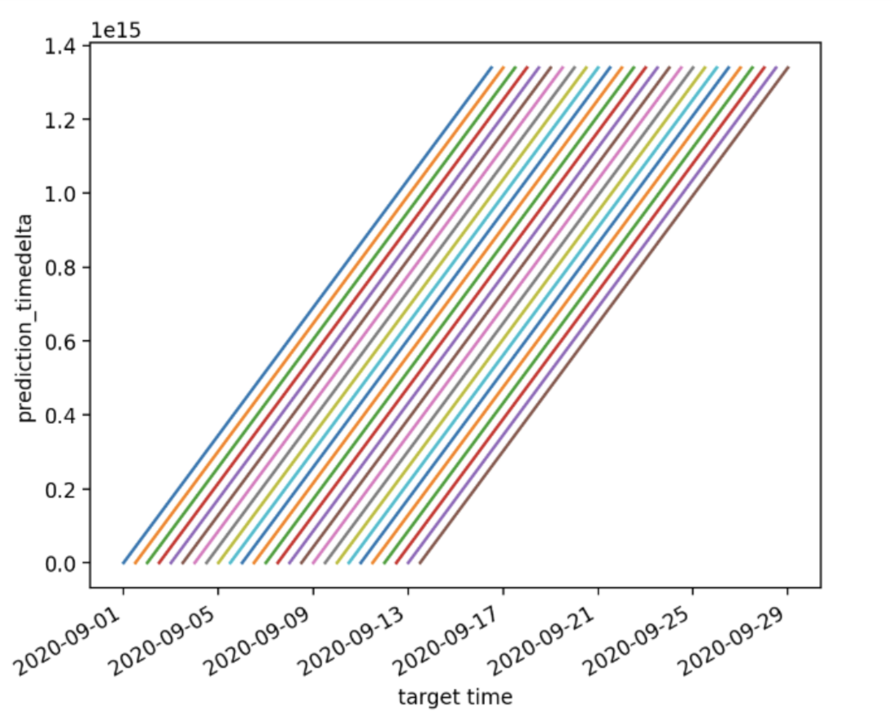


Figure 17: The Data Form of NeuralGCM

After we got the output of NeuralGCM within one month called **predictions_output_onemonth_updated.nc**. The path is as follow:

`/data0/cl4460/predictions_output_onemonth_updated.nc`

Then run the TEready_onemonth.py script to adjust the variable name and units. Finally, we obtained **predictions_output_onemonth_TE_ready.nc** with the path:

`/home/cl4460/NeuralGCM/predictions_output_onemonth_TE_ready.nc`

Notify that the output **predictions_output_onemonth_TE_ready.nc** only contains one initial day with corresponding 15 prediction days. The name of the file above may cause ambiguity. Therefore, We change the initial time and name of the files, run **addingunits_onemonth.py** got the three dataset called:

```
predictions\_output\_0701.nc
predictions\_output\_0702.nc
predictions\_output\_0703.nc
```

with path:

`/data0/zy2608/predictions_output_0701.nc`

The dataset **predictions_output_0701.nc** contains multiple levels and attributes that are not required for executing the TE model. Consequently, only the levels at 300, 500, and 850 are considered. The script Cut_obtained_dataset.py was subsequently executed, producing the output file **predictions_output_TE_ready_0701.nc** with the path:

`/data0/zy2608/predictions_output_TE_ready_0701.nc.nc`

The updated dataset was then utilized to run the TE model, and the resulting outputs were visualized accordingly.

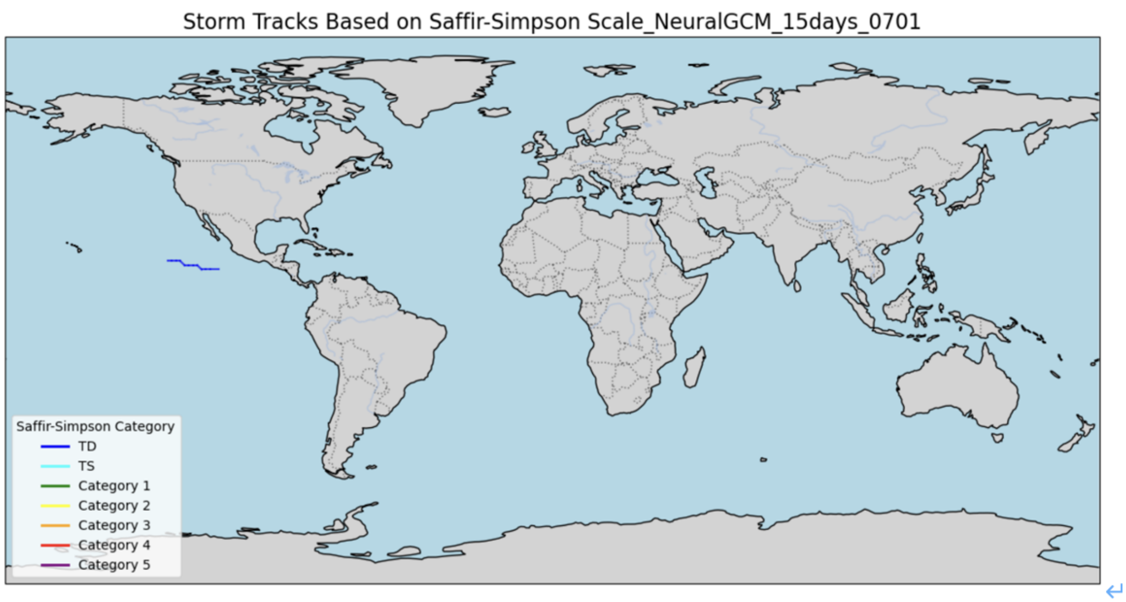


Figure 18: Storm Track With Initial Time 0701

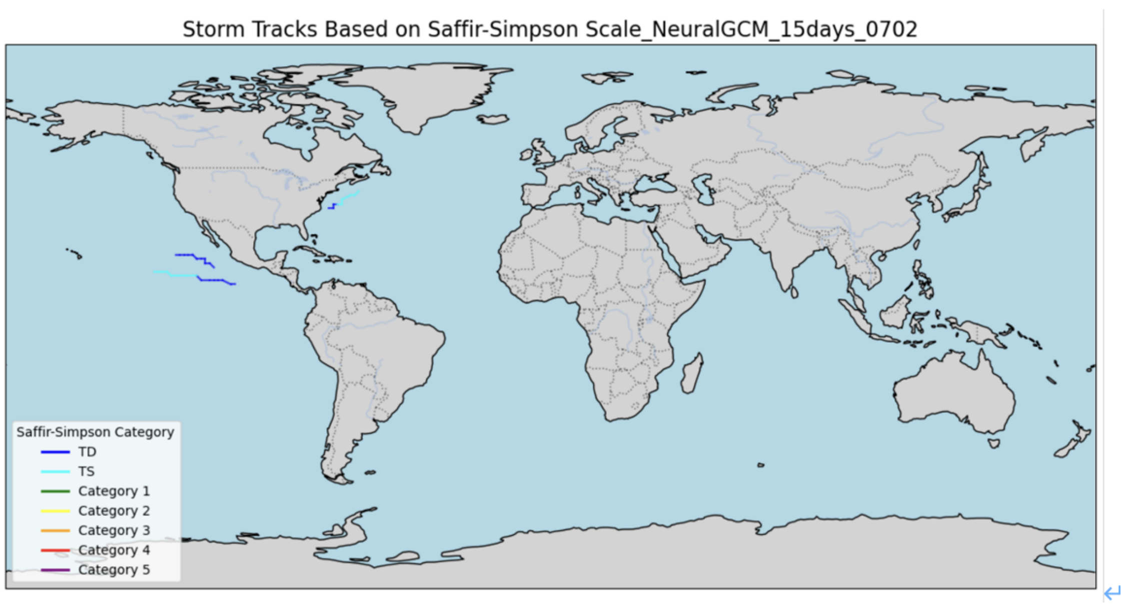


Figure 19: Storm Track With Initial Time 0702

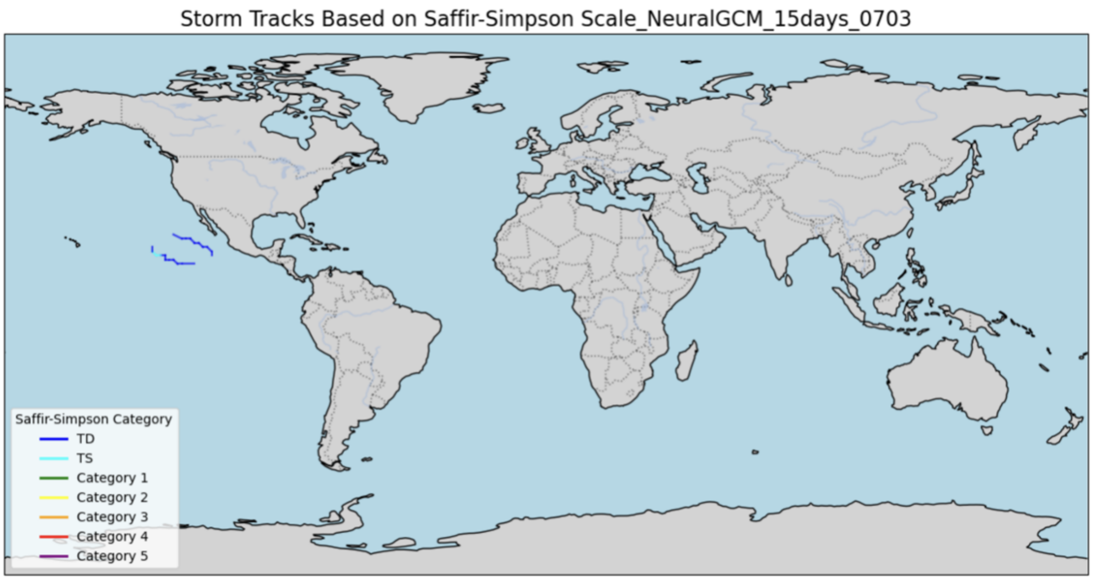
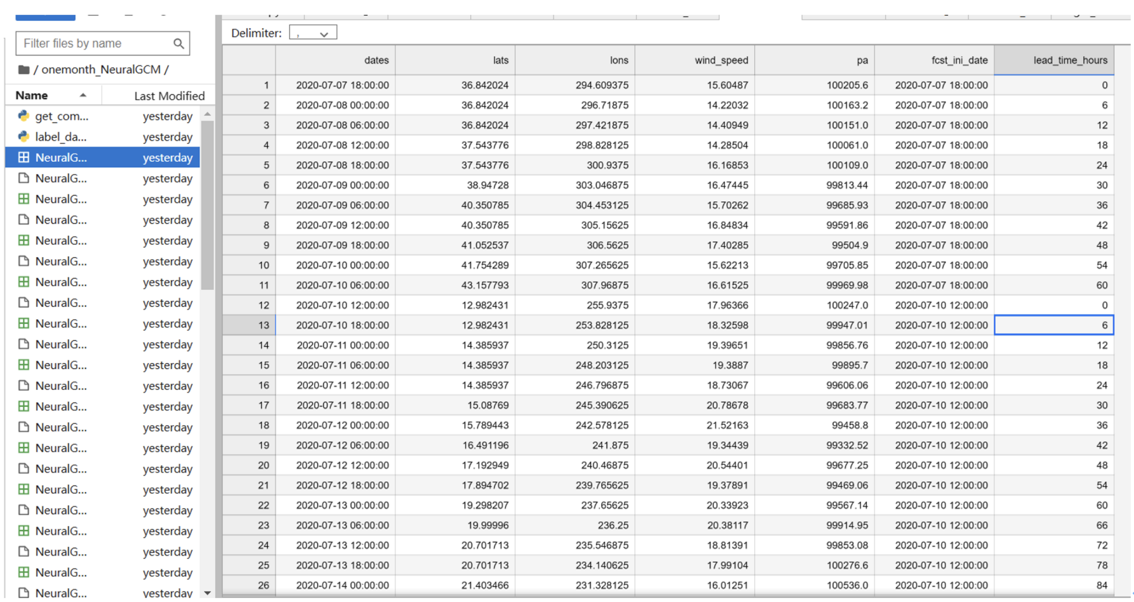


Figure 20: Storm Track With Initial Time 0703

Due to CPU usage limitations in Google Colab, we were unable to download the necessary data independently. The script historical_sim.py contains the primary code used to generate the forecasts, while full_sim.py demonstrates how to generate forecasts for a list of start dates. Consequently, we sought the invaluable assistance of Abhinab in the data provisioning process. Abhinab executed the aforementioned scripts on DOE equipment, thereby enabling us to obtain the data required for our experiments.

After acquiring the 30 files corresponding to different initial days, we initiated the execution of the TE model. However, running the TE model 30 times individually was time-consuming. To streamline this process, we modified the shell script modified_TE.sh, enabling the simultaneous execution of these files. Subsequently, we converted all outputs into CSV format using label_data.py



	dates	lats	lons	wind_speed	pa	fcst_ini_date	lead_time_hours
1	2020-07-07 18:00:00	36.842024	294.609375	15.60487	100205.6	2020-07-07 18:00:00	0
2	2020-07-08 00:00:00	36.842024	296.71875	14.22032	100163.2	2020-07-07 18:00:00	6
3	2020-07-08 06:00:00	36.842024	297.421875	14.40949	100151.0	2020-07-07 18:00:00	12
4	2020-07-08 12:00:00	37.543776	298.828125	14.28504	100061.0	2020-07-07 18:00:00	18
5	2020-07-08 18:00:00	37.543776	300.9375	16.16853	100109.0	2020-07-07 18:00:00	24
6	2020-07-09 00:00:00	38.94728	303.046875	16.47445	99813.44	2020-07-07 18:00:00	30
7	2020-07-09 06:00:00	40.350785	304.453125	15.70262	99685.93	2020-07-07 18:00:00	36
8	2020-07-09 12:00:00	40.350785	305.15625	16.84834	99591.86	2020-07-07 18:00:00	42
9	2020-07-09 18:00:00	41.052537	306.5625	17.40285	99504.9	2020-07-07 18:00:00	48
10	2020-07-10 00:00:00	41.754289	307.265625	15.62213	99705.85	2020-07-07 18:00:00	54
11	2020-07-10 06:00:00	43.157793	307.96875	16.61525	99969.98	2020-07-07 18:00:00	60
12	2020-07-10 12:00:00	12.982431	255.9375	17.96366	100247.0	2020-07-10 12:00:00	0
13	2020-07-10 18:00:00	12.982431	253.828125	18.32598	99947.01	2020-07-10 12:00:00	6
14	2020-07-11 00:00:00	14.385937	250.3125	19.39651	99856.76	2020-07-10 12:00:00	12
15	2020-07-11 06:00:00	14.385937	248.203125	19.3887	99895.7	2020-07-10 12:00:00	18
16	2020-07-11 12:00:00	14.385937	246.796875	18.73067	99606.06	2020-07-10 12:00:00	24
17	2020-07-11 18:00:00	15.08769	245.390625	20.78678	99683.77	2020-07-10 12:00:00	30
18	2020-07-12 00:00:00	15.789443	242.578125	21.52163	99458.8	2020-07-10 12:00:00	36
19	2020-07-12 06:00:00	16.491196	241.875	19.34439	99332.52	2020-07-10 12:00:00	42
20	2020-07-12 12:00:00	17.192949	240.46875	20.54401	99677.25	2020-07-10 12:00:00	48
21	2020-07-12 18:00:00	17.894702	239.765625	19.37891	99469.06	2020-07-10 12:00:00	54
22	2020-07-13 00:00:00	19.298207	237.65625	20.33923	99567.14	2020-07-10 12:00:00	60
23	2020-07-13 06:00:00	19.99996	236.25	20.38117	99914.95	2020-07-10 12:00:00	66
24	2020-07-13 12:00:00	20.701713	235.546875	18.81391	99853.08	2020-07-10 12:00:00	72
25	2020-07-13 18:00:00	20.701713	234.140625	17.99104	100276.6	2020-07-10 12:00:00	78
26	2020-07-14 00:00:00	21.403466	231.328125	16.01251	100536.0	2020-07-10 12:00:00	84

Next, we utilized get_combined_graph.py to visualize the entire July storm events, resulting in the following figure:

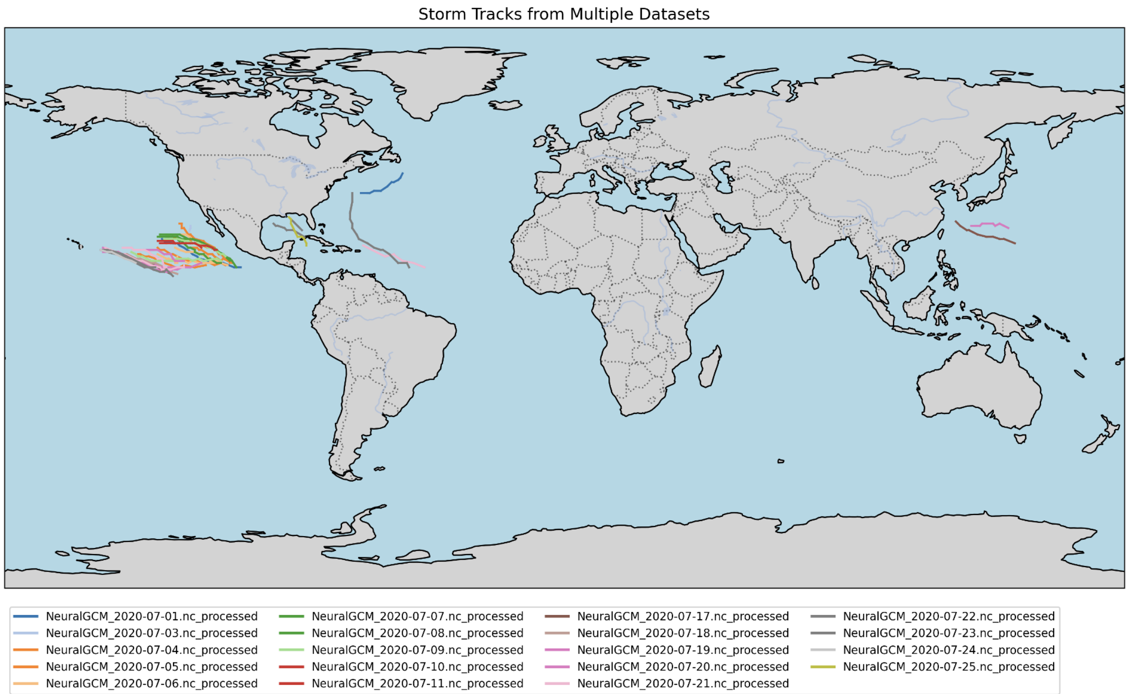


Figure 21: Predicted Storm with Different Initial Time

Subsequently, we generated the NeuralGCM output for the entire year of 2020 with 1.4°× 1.4° and 1.4°× 0.25°. The script 1.4_label_data.py was employed to convert the dataset into a CSV file and assign storm IDs. Following this, 1.4_resolution_getgraph.py was implemented to plot the corresponding graphs, as illustrated below:

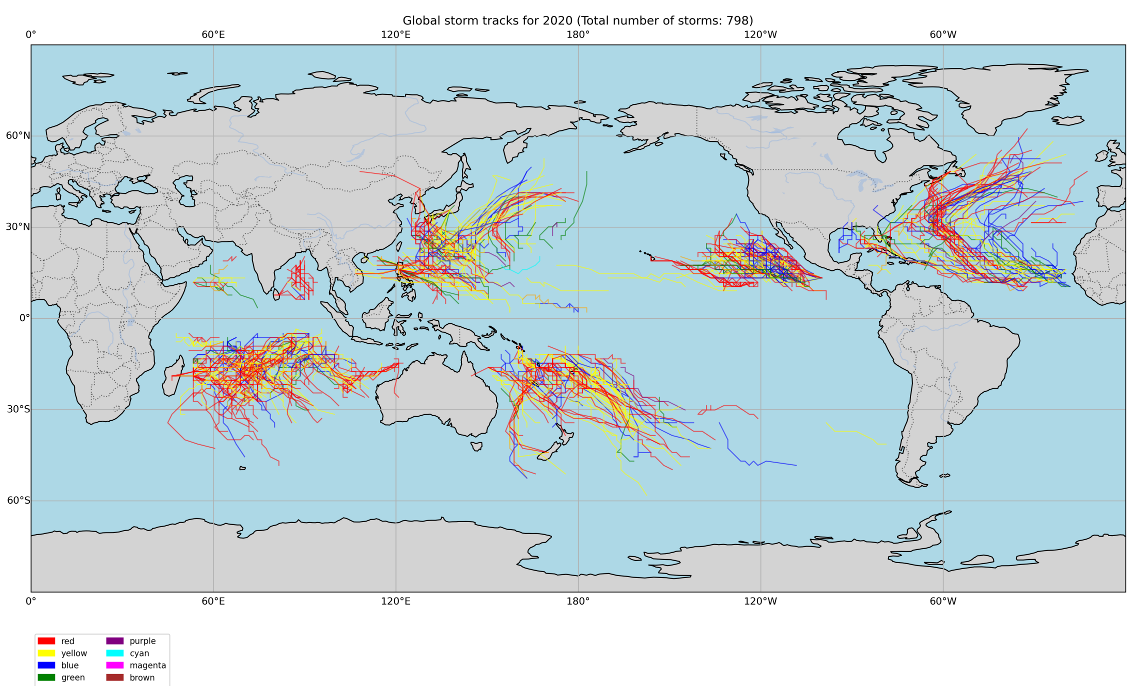


Figure 22: 1.4 Degree Dataset in 1.4 Resolution Global Storm Tracks

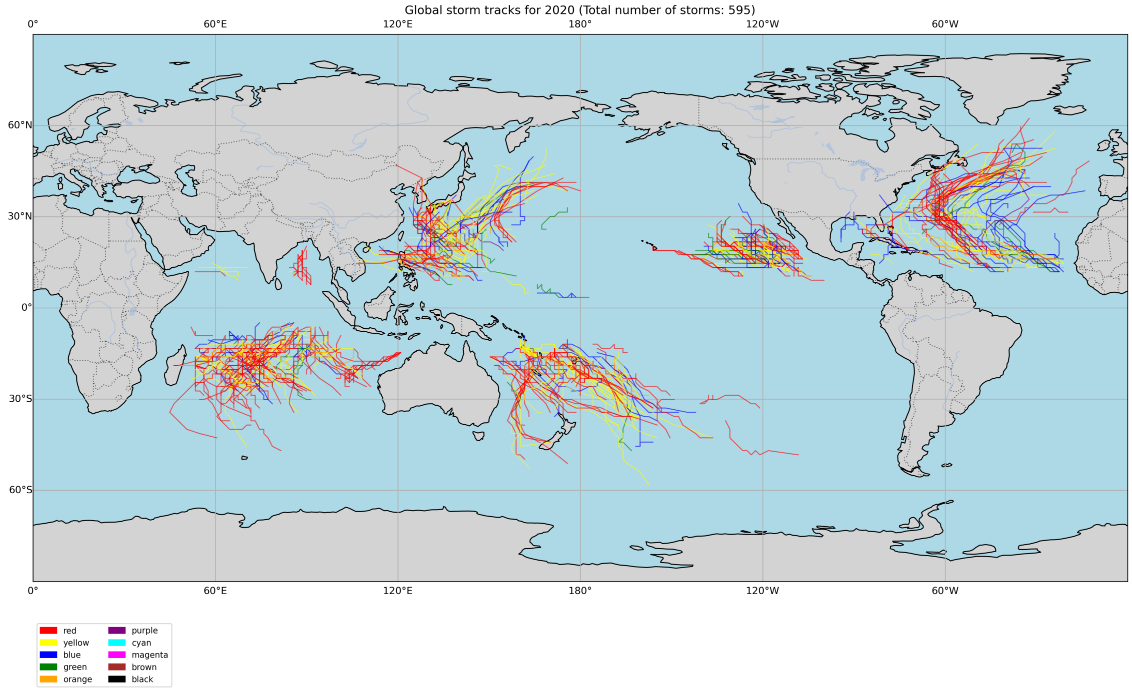
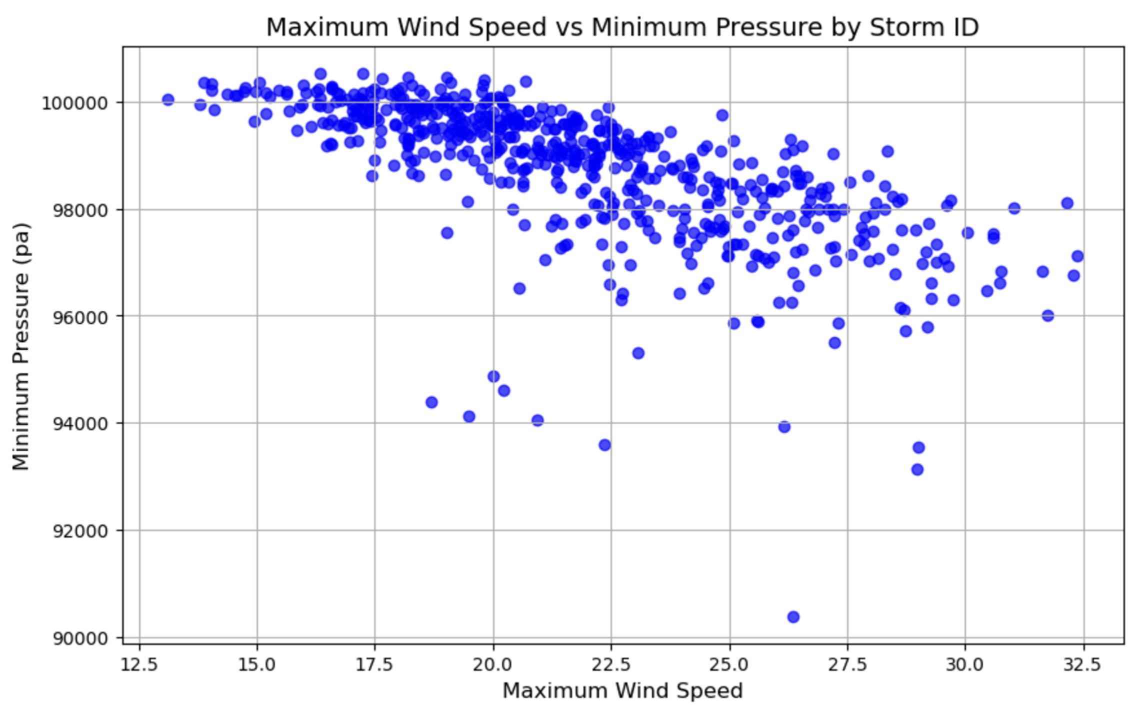
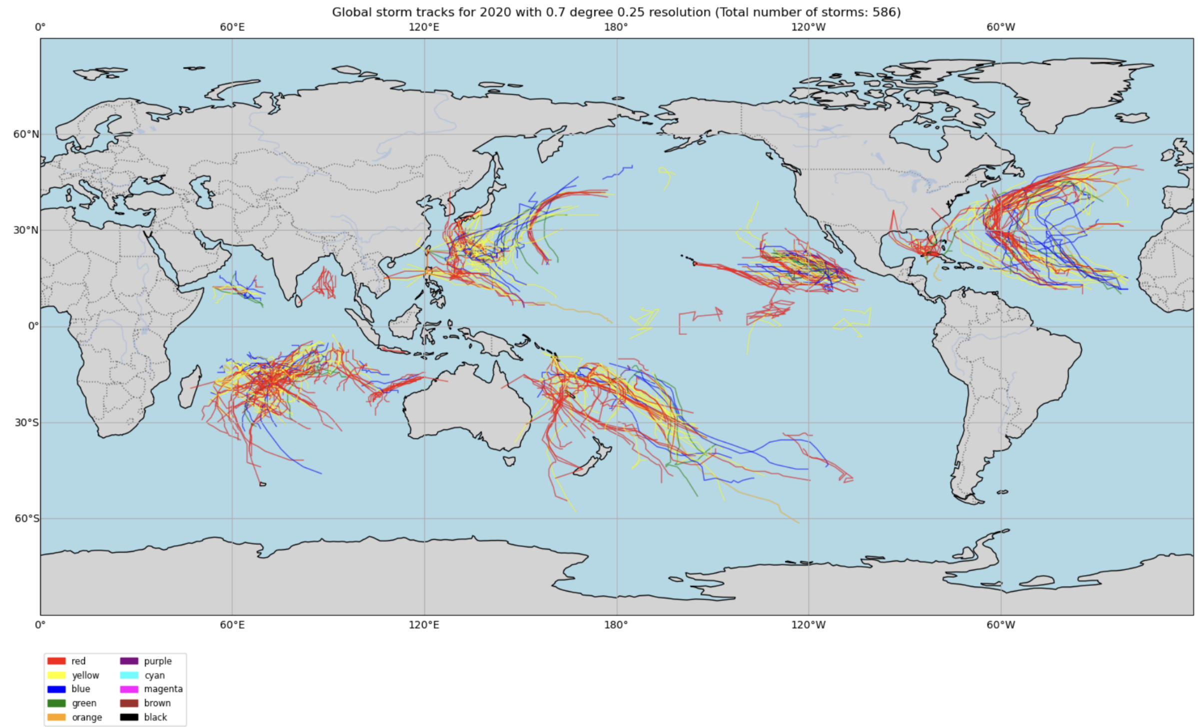


Figure 23: 1.4 Degree Dataset in 0.25 Resolution Global Storm Tracks

Furthermore, to evaluate the accuracy of the NeuralGCM outputs, we plotted graphs of maximum wind speed versus minimum pressure. The detailed procedure is outlined in the script SLP_windspeed.py. The resulting graph for maximum wind speed and minimum pressure using 1.4° resolution data is presented below:

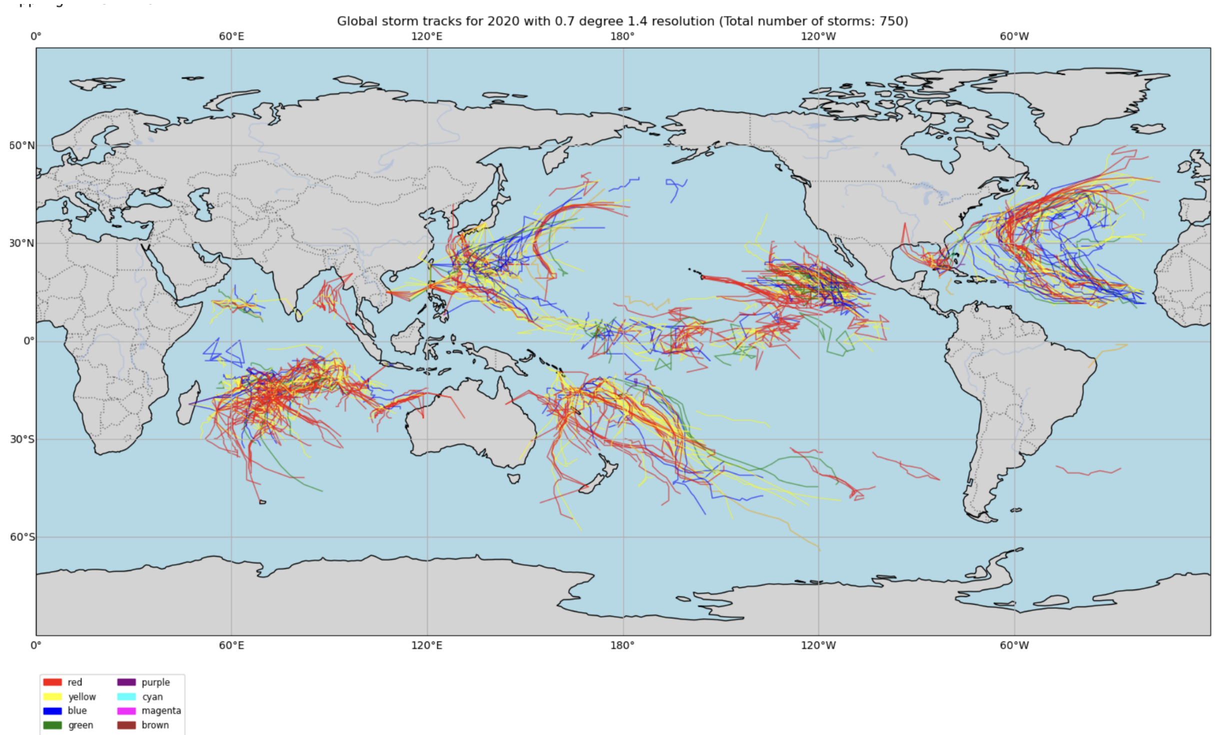


Additionally, we generated graphs for all storms using datasets with 0.7°× 1.4° and 0.7°× 0.25° resolutions employing the same methodology described above. This allowed us to verify the correctness and consistency of the data across different resolutions.



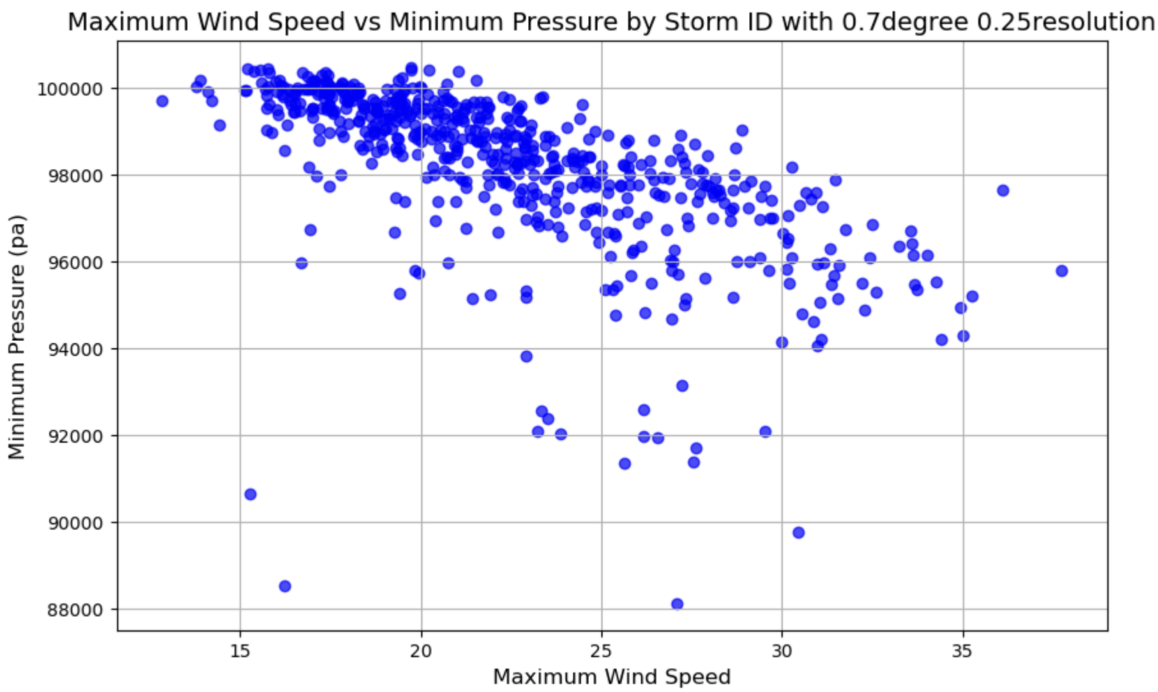

```
Image saved as 0.7_0.25resolution_global_storm_tracks_2020.png
Total number of storms: 586
```

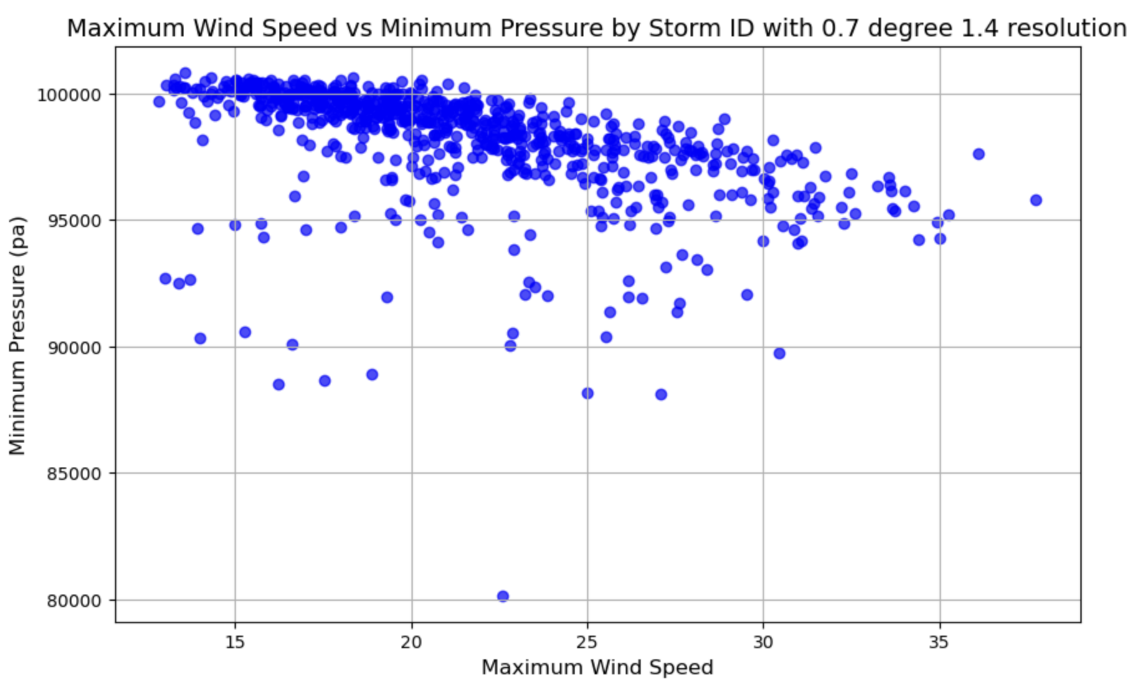
Figure 24: 0.7 Degree Dataset in 0.25 Resolution Global Storm Tracks




```
Image saved as 0.7_1.4resolution_global_storm_tracks_2020.png
Total number of storms: 750
```

Figure 25: 0.7 Degree Dataset in 1.4 Resolution Global Storm Tracks


Maximum Wind Speed vs Minimum Pressure by Storm ID with 0.7degree 0.25resolution
Minimum Pressure (pa)
Maximum Wind Speed
100000
98000
96000
94000
92000
90000
88000
15
20
25
30
35


Maximum Wind Speed vs Minimum Pressure by Storm ID with 0.7 degree 1.4 resolution
Minimum Pressure (pa)
Maximum Wind Speed
100000
95000
90000
85000
80000
15
20
25
30
35