DATA ANALYTICS AND MACHINE LEARNING ON WEATHER IN SINGAPORE AND IMPACT ON SENSORS PERFORMANCE

Ang Ian[1], Goh Kaisi, Ashley[2], Tan Choon Yong[3], Yu Guan-yu Victor[4], Koh Geng Hao Bernard[5]
[1] River Valley High School, 6 Boon Lay Ave, Singapore 649961
[2] Methodist Girls' School, 11 Blackmore Dr, Singapore 599986
[3] Catholic High School, 9 Bishan Street 22, Singapore 579767
[4] Hwa Chong Institution, 661 Bukit Timah Rd, Singapore 269734
[5] Defence Science and Technology Agency, 1 Depot Rd, Singapore 109679

Abstract

Radiofrequency (RF) is affected by weather conditions. The performance of sensors utilising RF can be significantly affected by the volatile weather patterns in Singapore. In this study, we analysed historical data of Singapore's weather to infer patterns and understand rainfall and temperature trends that affect Singapore. Using the XGBoost algorithm, we developed a Machine Learning (ML) Model that can predict the amount of daily rainfall using historical data. The predictions can be used to predict the extent of RF attenuation that sensors could face and be used to aid in developing mitigating measures to optimise sensor performance.

1. INTRODUCTION

1.1 Background & Objective

RF play a significant role in electronic warfare (EW), including electronic warfare support (ES), electronic attack (EA) and electronic protection (EP). From identifying and interfering with enemy forces to communicating with friendly forces, most—if not all—sensors used in EW utilise RF signals to obtain and transmit the necessary information. However, attenuation of RF signals in dense atmospheric conditions (e.g. rain, hazy, fog) severely impact the effectiveness of sensors that rely on RF signals to perform their tasks. As climate change progresses, Singapore's weather patterns have become even more volatile, making it even more important to accurately predict unfavourable weather and the subsequent impacts on sensor performance. Additionally, there is a need to analyse Singapore's weather patterns and assess the impacts of different weather conditions on RF. This project aims to use machine learning to make long-term predictions of Singapore's weather and to predict the signal attenuation such weather could pose to radiofrequencies. (Refer to Appendix 1 for the Fundamentals of Radiofrequency)

1.2 Impact of Weather Conditions on Radiofrequency

Air particles in the atmosphere cause a phenomenon known as atmospheric attenuation, where radio waves experience power loss due to absorption and scattering by particles in the transmission path. This is particularly prominent in Singapore as it is a tropical country, giving it high temperatures, high humidity and high annual precipitation. Singapore typically experiences sunny with high humidity, rainy and hazy weather.

In sunny weather, the main factors affecting atmospheric attenuation are temperature and humidity. While temperature has no direct impact on signal attenuation, humidity has a negative impact on signal strength. The higher the relative humidity, the more water vapour particles are present in the air, increasing signal attenuation via scattering and absorption. Scattering is the redirection of electromagnetic (EM) waves by particles with diameters comparable to their wavelengths via refraction against and in the particle. Absorption is the transfer of energy from the waves to the particles as they pass through, resulting in power loss. High levels of EM attenuation are observed particularly at higher frequencies, especially above 10GHz [1].

Singapore also faces near-annual hazy seasons as air pollutants produced in forest fires in Indonesia are carried by wind to Singapore, exacerbated by the Southwest Monsoon from June to September [3]. Haze heightens the concentration of smoke and dust particles in the atmosphere, among others, thus increasing the attenuation of EM waves during hazy seasons in accordance with the size and concentration of particles. Though power loss due to smoke is generally frequency-dependent, attenuation is rather high at all frequencies [4] [5] [6].

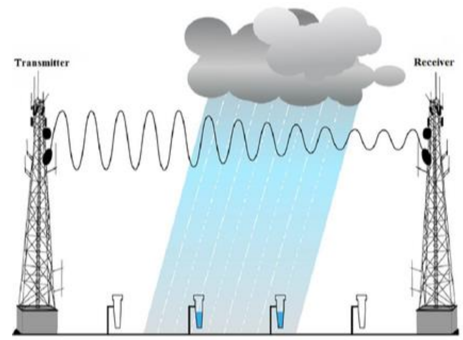


Fig. 1.2.1 Rain-induced attenuation [2]

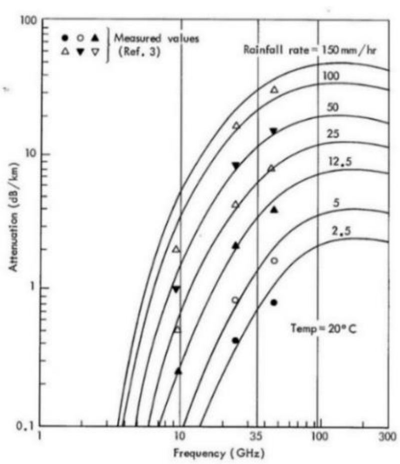


Fig. 1.2.2 Logarithmic plot of calculated values of attenuation by rain in dB/km [8]

Finally, heavy rain is the greatest cause of RF attenuation in Singapore, since the country lacks snow and tropical cyclones. The size and abundance of raindrops during heavy rains and thunderstorms significantly increases RF attenuation, as they absorb and scatter waves much more in the transmission link; rain can also cause depolarisation of signals. At 20°C, rain-induced attenuation of radio waves tends to be more severe in frequencies around 35-95GHz but is negligible in frequencies less than or equal to 10GHz [7]. Attenuation increases with rainfall rates; in Singapore, rainfall rates span from 0.5mm/h to 32mm/h on particularly rainy days [8], resulting in attenuation of a range of 2-11dB/km (Fig. 1.2.2).

1.3 Impact of Weather on Sensors' Performance

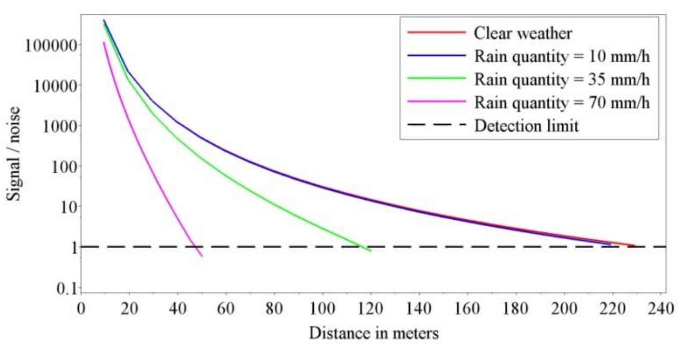
The impact of weather on sensors' performance varies greatly depending on the types of sensors, such as radar and electro-optical sensors, as different sensors use different RF ranges for transmission.

<u>Precipitation (Rain, snow, hail)</u>

Most RADARs used in EW use frequencies of 2-18GHz (e.g. 9GHz waves are used by maritime navigation radars). RADAR sensors can operate beyond adverse weather conditions but still require the assistance of a perception system to improve decision robustness. Such systems are not significantly attenuated at short distances, but rain backscattering or rain clutter can decrease the maximum range of detectability [9] [10], as well as result in false alarms [11]. Supplementing perception systems can include GNSS systems for accurate positioning and orientation information [11]. Adaptive sensor fusion algorithms can be applied to dynamically adjust the weighting and integration of sensor data based on the current environmental conditions and sensor performance. This can help mitigate the effects of adverse weather on the overall perception system.

Millimetre wave (mmWave) communication occurs within the range of 30GHz to 300GHz. Since signal attenuation becomes more severe at extremely high frequencies (i.e. more than 10GHz), the high frequencies of mmWaves limit them to travelling very short distances and prevent them from penetrating buildings and objects [12]. This, coupled with Singapore's frequent intense rainstorms, increases mmWave attenuation and decreases their range [1]. Conversely, frequencies below 10GHz experience lower signal attenuation due to rainfall.

According to Mie's solution to Maxwell's equation, any transmission wavelength (λ) that is similar or smaller to the droplet diameter of 6 mm will be subject to Mie scattering [13]. LiDARs transmitting in the 905 nm and 1550 nm wavebands will be heavily affected by Mie scattering from rain at longer distances [14]. However, within the range usually required for rangefinders on AVs, LiDAR susceptibility to rain is not as noticeable until more severe rain rates occur [15].



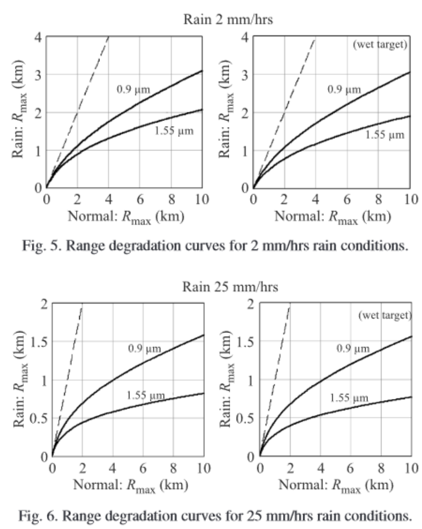


Fig. 1.3.1 Variation of the signal/noise ratio as a function of the rain rate and distance between LiDAR and target for a rain droplet radius equal to 3 mm [15]

Fig. 1.3.2 Range degradation curves for 2 mm/hrs and 25 mm/hrs rain conditions [14]

Temperature and Humidity

Relative humidity has negative correlation with signal strength. As relative humidity increases, RF signals have a weaker signal strength and vice versa. Higher humidity results in higher concentration of water vapour in the air. When RF signals propagate through the air, it is likely that more water vapour in the air absorbs energy from the waves, weakening them further [16].

Hence, when Singapore's weather is more humid, signal strength of RF-based systems is likely to be weakened. Temperature, on the other hand, has little to no effect on RF signals.

Fog and other Atmospheric Gases

In contrast to vision-based and laser scanners, RADAR sensors are robust against environmental conditions, such as fog or other changes in luminosity [17]. RADAR-based object detection is less affected by the weather, especially in foggy scenarios, where recognition from data from optical sensors such as cameras and LiDAR fails at a very short-range depending on the fog density; hence, RADAR is shown to be a good solution for dense fog perception [18].

LiDAR systems and cameras, whose operating wavelengths are less than fog particles, are subject to Mie scattering [11]. In addition, near-infrared signals are also subject to significant attenuation by fog [19]. With air-light interference (the scattering of light from the interference of particles), objects within the immediate vicinity of the light source are impossible to perceive by light-based sensors such as cameras [11]. In addition, the higher the reflectivity of an object, the darker it would appear in fog [20]. Evidently, there are many significant factors that affect the performance of electro-optical sensors in fog, and these devices are best supplemented by other types of sensors, such as RADAR, to mitigate these shortcomings.

At higher frequencies, there is a greater loss of energy of the millimetre waves as they propagate through the atmosphere, due to interactions between the waves and gas molecules like oxygen, nitrogen dioxide and water vapour in the troposphere. The loss of energy in millimetre waves causes attenuation of the signal [21]. The attenuation varies with the amount of water vapour present in the atmosphere [12].

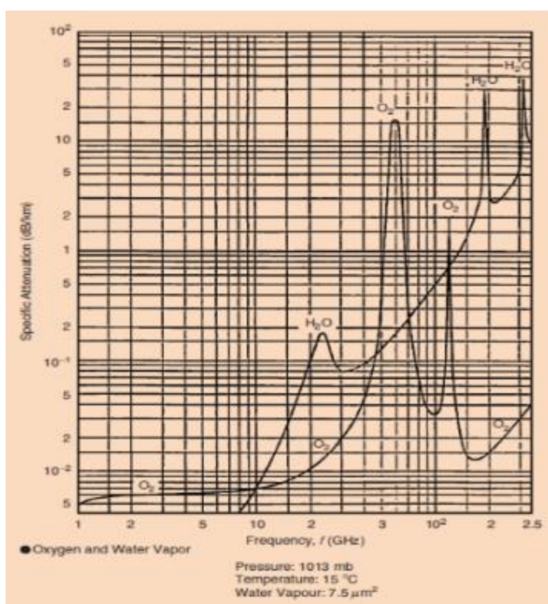


Fig. 1.3.3 Specific attenuation due to different atmospheric gases [21]

2. METHODOLOGY

2.1 Data Analytics of Past Weather across different regions of Singapore

As weather conditions significantly impact sensor performance, it is crucial to gain a better understanding of weather patterns in Singapore to complement EW technology and enhance Singapore's defence capabilities. To achieve this, we sought historical hourly data to analyse climate trends, and to make a predictive model for future weather. This understanding will enable defence technology to prepare for and adapt to weather changes that may affect sensor performance.

To achieve this, we used the Open-Meteo free weather API [22], which provides a python script (Refer to Appendix 2A for the script), among other methods, to download weather data. To represent the North, South, East, West and Central regions of Singapore, we used data from Sembawang, Sentosa, Changi, Jurong West and Ang Mo Kio Park respectively. These five regions would set the basis for a more comprehensive understanding on how varying weather conditions in one region in Singapore might affect the other. Furthermore, we chose a range of January 1991 to November 2024 to have a well-rounded understanding of historical trends. The variables in the data used are the temperature, relative humidity, dew point, rain, surface pressure, cloud cover, wind speed, and wind direction, as these factors appear to be the most relevant to weather conditions in Singapore’s tropical climate.

Next, we performed a detailed analysis of the data. Using Power BI, we plotted graphs of the total monthly rainfall and the mean monthly temperature for each of the five different locations based on the datasets obtained. This allowed us to visualise trends and variations in rainfall and temperature across the regions over the 43-year period.

Moreover, the historical total hourly rainfall data of the five locations was analysed, to identify rainfall trends across years in the respective locations. The datasets are also used to calculate the mean total rainfall of each month across the years (e.g. mean rainfall in January from 1991-2024), enabling us to identify seasonal rainfall patterns.

2.2 Machine Learning Model

To predict future weather conditions and their possible impacts on sensors’ performance, we tried out a few different ML models to identify one that is best suited to perform this job.

Initially, we tested the Random Forest Regressor, a model known for its ensemble-based approach of combining multiple decision trees to reduce overfitting and improve predictive performance. However, Random Forests treat data as independent and do not explicitly consider sequential dependencies, which are crucial in time-series data like weather patterns. Additionally, the model struggled with capturing long-term temporal trends, making it less suitable for our objective of accurately predicting weather impacts on sensors.

We also explored the Transformer model for time series forecasting. Although not originally intended to predict sequential data, the Transformer's positional encoding effectively retains temporal orders, allowing it to understand the sequential nature of time series. Furthermore, its self-attention mechanisms dynamically weigh the importance of different time steps, enabling the model to identify the most relevant weather patterns for its predictions. Also, the root mean square error (RMSE) of our first trial was notably low at 1.17mm compared to the actual rainfall data. However, the training process was extremely slow. Thus, we concluded that the Transformer model was not a feasible option within the timeframe of our investigation.

Finally, we settled on the XGBoost model, which constructs decision trees sequentially, with each tree attempting to correct the errors of its predecessors. This iterative approach enables the model to capture complex, non-linear relationships effectively. XGBoost is also computationally efficient for large datasets, leveraging optimisations like parallel processing and sparse data handling—particularly valuable for irregular rain data with many zero values. Its ability to model intricate dependencies and incrementally reduce errors made it a stronger candidate for our prediction model. Furthermore, XGBoost's regularisation techniques help prevent overfitting, ensuring robustness even when training on datasets with complex weather patterns.

Preparing the dataset is a crucial step for building an effective model that can uncover meaningful relationships. Weather data is inherently temporally dependent, as each observation is influenced by past weather conditions. Recognising this, we implemented three key preprocessing techniques to maximise the model's ability to learn from the sequential nature of the data (Refer to Appendix 2B for the data preparation code).

Firstly, we lagged each variable in the dataset over a 24-hour period, adding one lagged feature per hour to the training data. This allows the model to capture temporal dependencies by considering the influence of past weather variables on current conditions.

Secondly, because time is a cyclical variable (e.g., December leads directly into January, and midnight transitions into a new morning), we employed sine-cosine normalisation. By transforming time-related variables into two components—sine and cosine—we ensured the model recognized these cyclical patterns, which enhanced its ability to identify seasonal trends and periodic behaviour. (Refer to Appendix 2C for implementation details.)

Thirdly, we used the data to train a model to predict each variable in the future for each region. During the training process, we initially used K-Fold cross-validation to divide the data into eight separate folds, training a model on each fold to evaluate its generalisation across subsets of the data. However, we discovered that this approach was inappropriate for time-series data. K-Fold cross-validation assumes independence between data points and can mix future data into training sets, neglecting the temporal order critical for time-series forecasting. To address this, we adopted Time Series Cross-Validation (TSCV), which ensures that training sets always precede validation sets in time. This approach preserves the temporal structure and prevents data leakage. Additionally, we introduced a 24-hour gap in our TSCV implementation between training and validation datasets to reduce the risk of overfitting on immediate temporal correlations.

Our dataset used for the modelling includes hourly meteorological data collected from Ang Mo Kio Park, Changi, Jurong West, Sembawang and Sentosa. The data spans from January 1, 1991 (12am) to November 30, 2024 (11pm) and includes the following variables [22]:

1. Temperature (°C) [2m] - Air temperature at 2 metres above ground
2. Relative humidity (%) [2m] - Relative humidity at 2 metres above ground

3. Dew point (°C) [2m] - Dew point temperature at 2 metres above ground
4. Total rainfall (mm) - Liquid precipitation of the preceding hour including local showers and rain from large scale systems
5. Surface pressure (hPa) - Pressure on mean sea level
6. Cloud cover (%) - Total cloud cover as an area fraction
7. Cloud cover low (%) - Low level clouds and fog up to 3 km altitude
8. Cloud cover mid (%) - Mid level clouds from 3 to 8 km altitude
9. Cloud cover high (%) - High level clouds from 8 km altitude
10. Wind speed (km/h) [10m] - Wind speed at 10 metres above ground
11. Wind speed (km/h) [100m] - Wind speed at 100 metres above ground
12. Wind direction (°) [10m] - Wind direction at 10 metres above ground
13. Wind direction (°) [100m] - Wind direction at 100 metres above ground

Data from January 1, 1991 to December 31, 2019 (about 85% of the total), is used to train the model, while data from January 1, 2020 to November 31, 2024 (about 15% of the total), is used for testing the model. We used the different meteorological parameters in the dataset to predict future rainfall.

3. RESULTS & DISCUSSION

3.1 Analysis of Weather across different regions of Singapore

After plotting various graphs to gain a deeper understanding on the weather patterns in Singapore over the last 3 decades, we sought to explain the trends we observed (Refer to Appendix 3A for rainfall and temperature graphs across the 5 locations of Singapore from 1981 January to 2024 November).

Monsoon Seasons

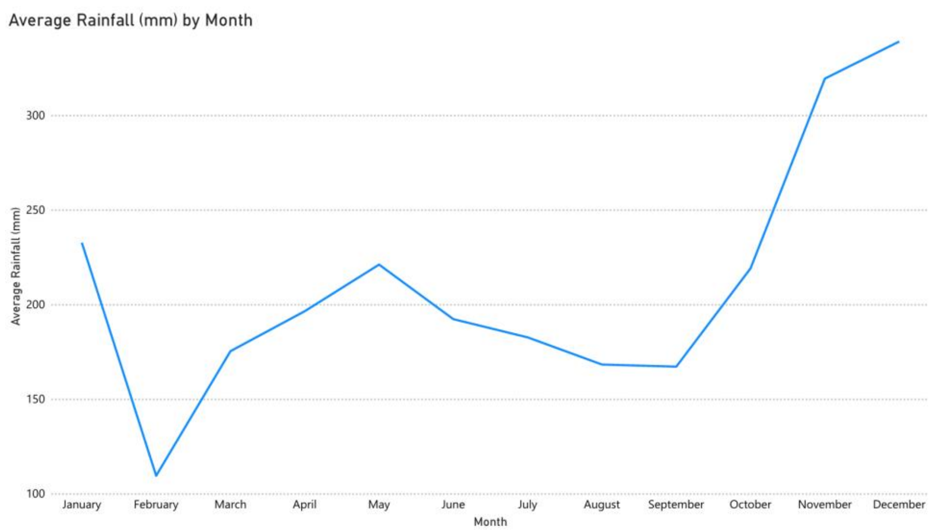


Fig. 3.1.1 Graph of Average Monthly Rainfall (mm) Trend across Ang Mo Kio Park, Changi, Jurong West, Sembawang and Sentosa (1981 January - 2024 November)

Within a year, the average monthly rainfall typically hits a peak in December; the lowest average monthly rainfall normally occurs in February. There is consistent rainfall throughout the year due to Singapore's tropical climate. Singapore's climate also explains the peak and low average monthly rainfall experienced during December and February respectively.

Singapore's climate is characterised by two monsoon seasons — the Northeast and Southwest Monsoons [23]. In Singapore, the effects of the Northeast Monsoon and Southwest Monsoon typically occur between December and early March, and between June and September respectively. During the Northeast Monsoon, from December to early January, Monsoon surges cause continuous, widespread moderate to heavy rain, which leads to the increased development of afternoon and early evening showers. On the other end of the spectrum, from late January to March (dry phase of Northeast Monsoon), Singapore typically experiences windy and relatively dry weather, with February being the driest month of the year on average. During the Southwest Monsoon, from June to September, short duration showers or thunderstorms in the afternoon are common, resulting in erratic weather patterns (Refer to Appendix 3B for the graph of the average mean temperature trend across months and the dual axis graph of the average monthly rainfall and mean temperature trend across months).

<u>The Walker Circulation and El Niño Southern Oscillation</u>

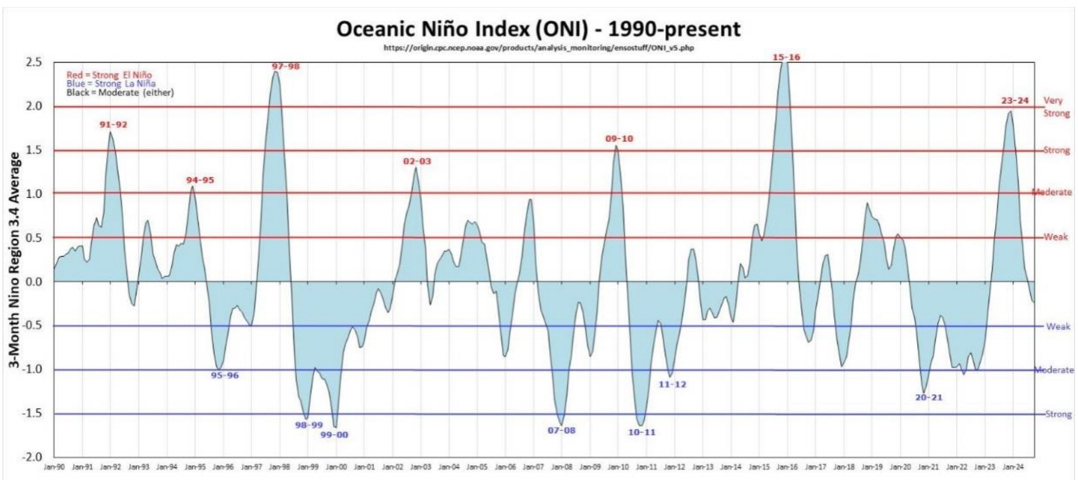


Fig. 3.1.2 Oceanic Niño Index (1990 - present) [24]

Over the years, Singapore's weather patterns have fluctuated due to the influence of the Walker Circulation, where trade wind directions change in line with the El Niño Southern Oscillation which occurs in the Pacific Ocean. When the Walker Circulation is in full force (La Niña conditions), pressure differences between landmasses on the West of the Pacific Ocean (Southeast Asia (SEA) and Australia) and on its East (the Americas) bring trade winds over the Pacific Ocean to SEA, resulting in heavier rainfall and lower temperatures during La Niña [25]. This is seen during the La Niña seasons from 1999 - 2000, 2007 - 2008 and late 2020 - 2021 (Fig. 3.1.2), where Singapore experienced hikes in rainfall amounts and dips in monthly temperature. When oceanic currents reverse under El Niño conditions, trade winds weaken and sometimes change direction, bringing heavy rainfall to the Americas and drier, hotter weather to SEA and Australia. Singapore's climate reflected these changes during strong El Niño conditions, as evidenced by the higher temperatures observed in 1997 and 2015 - 2016, and decrease in rainfall amounts in 1997 and early 1998.

Global Warming

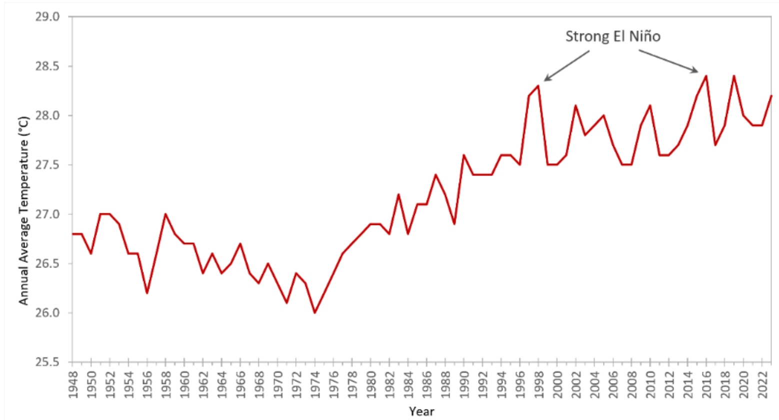


Fig. 3.1.3 Graph of Annual Average Temperature (°C) in Singapore from 1948 to 2023 [26]

Generally, the annual average temperature in Singapore has shown an increasing trend from 1948 to 2023. In Singapore, the annual average temperature has risen by an average of 0.25°C per decade between 1948 and 2023, which is about double the global trend of 0.12°C per decade between 1951 and 2012. The difference is likely due to rapid urbanisation in Singapore [27]. In 1998 and 2016, a strong El Niño had played into causing a huge increase in annual average temperature compared to the years before the respective year.

The general trend, while alarming, is not something that was unforeseeable, with many climate scientists and experts having pointed it out decades ago. The Enhanced Greenhouse Effect, driven by a massive increase in greenhouse gas emissions, has also exacerbated global warming, resulting in a rapid increase in not just Singapore's but the global annual average temperatures (Refer to Appendix 3C for more information regarding the Enhanced Greenhouse Effect and its impacts on Singapore's weather and sensors).

Outliers in Rainfall Amounts

The peak in rainfall seen across regions between December 2006 and January 2007, is attributed to Typhoon Utor which first struck the Philippines before moving Westward toward the rest of SEA [28] [29]. Particularly heavy rainfall and flooding was experienced throughout SEA, including most of Singapore (Refer to Appendix 3D for elaboration on other outliers).

3.2 XGBoost Machine Learning Model

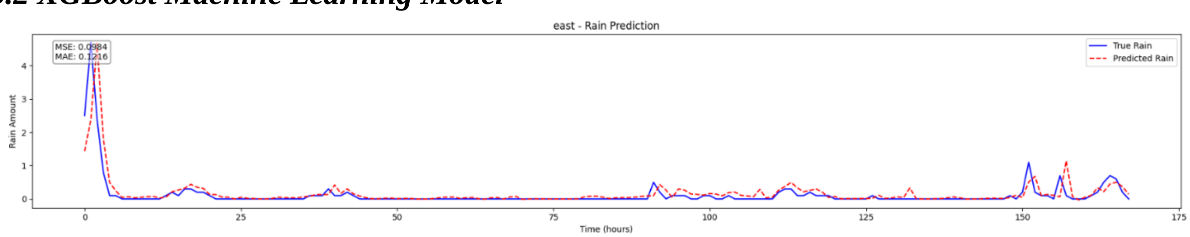


Fig 3.2.1 Graphs of hourly rainfall prediction against actual hourly rainfall for 175 hours (east region) (Refer to Appendix 3E for more graphs regarding all regions)

The ML Model achieved a mean absolute error of 0.325mm and a root mean squared error of 0.960mm with an accuracy of 79.8% across the 5 regions.

From the graphs, it is evident that the predicted rainfall is very close to the true rainfall, and it can usually reliably predict the significant presence of rainfall accurately with the exception of some outliers due to the unpredictability of rain patterns. The RMSE, 0.960mm, is significantly lower than the previous investigation's RMSE which was 2.13mm [30] (Refer to Appendix 3F for code that checks accuracy of predictions).

4. CONCLUSION & FUTURE WORK

Of all the phenomena affecting rainfall, the Northeast monsoon has the greatest impact on Singapore’s weather, making December and January the wettest months of the year and February, the driest. The Walker Circulation and global warming, too, affect the amount of rainfall Singapore faces, with the Walker Circulation bringing general rises and falls to rainfall amounts, and global warming making weather impacts more severe in both wet and dry seasons.

Our ML model, which can predict hourly rainfall amounts in Singapore’s various regions up to 79.8% accuracy and a MAE of 0.325mm, can be used by the relevant agencies to estimate signal attenuation of sensors at any one point in time. To evaluate the signal attenuation of sensors in varying rain rates, we drew lines on a graph (Fig 1.3.1) [15] illustrating a LiDAR sensor’s maximum detection range under varying rain rates, specifically for those in Singapore. Although open-source graphs of RF sensor detection ranges under differing rain conditions were unavailable, a similar approach would be utilised to estimate signal attenuation of RF sensors. Since the model is rather accurate in predicting rainfall spikes, mitigating measures can be put in place pre-emptively to mitigate the impact of such signal attenuation, like installing complementary sensors and sensor settings to improve Signal to Noise ratios (SNR) for EW.

Further studies on this topic could explore, in more detail, the impact of SNR degradation on sensor performance according to the various rainfall rates that Singapore experiences. Our ML model can also be trained by all parameters concurrently. Future work can also experiment with applying the same hyperparameters and normalisation to different models, to see if their root mean squared error is reduced by doing so. We also plan to utilise the ML model to predict the following day’s weather and countercheck with OpenMeteo to check its accuracy. Additionally, the ML model can be used to predict the amount of signal attenuation faced in Singapore. The difference in power loss between sensors over sea and over land may also be investigated, as maritime air masses have higher humidity than continental air masses [31].

5. ACKNOWLEDGEMENTS

We would like to express our gratitude to our mentor, Bernard, for his help and guidance throughout this project, and for his dedication in caring for us during our time in DSTA. We would also like to appreciate Regine and Eric for the wisdom they imparted on us, as well as Gabriel, Yi Teik and Daniel for their hospitality toward us over the course of our internship.

REFERENCES

[1] E. Alozie, A. Musa, N. Faruk, A. L. Imoize, A. Abdulkarim, A. D. Usman, Y. O. Imam-Fulani, K. S. Adewole, A. A. Oloyede, O. A. Sowande, S. Garba, B. A. Baba, Y. A. Adediran and L. S. Taura, "A review of dust-induced electromagnetic waves scattering theories and models for 5G and beyond wireless communication systems," *Scientific African*, vol. 21, p. e01816, September 2023.

[2] Kamtchoum, E. V., Armand, N. B., & Djamegn, C. T. (2022, November). Using Machine Learning Techniques for Rainfall Estimation Based on Microwave Links of Mobile Telecommunication Networks. ResearchGate. https://doi.org/10.1007/s42979-022-01458-6

[3] National Environment Agency, "Managing Haze," 7 August 2024. [Online]. Available: https://www.nea.gov.sg/our-services/pollution-control/air-pollution/managing-haze

[4] A. C. M. Austin, "Wireless Channel Characterization in Burning Buildings Over 100-1000 MHz," *IEEE Transactions on Antennas and Propagation*, vol. 64, no. 7, pp. 3265-3269, 2016. https://doi.org/10.1109/TAP.2016.2562671

[5] S. G. Hong, H. Lee, H. Yang, J. Jin, H. Lee and K. Lee, "Experimental Study of Microwave Attenuation in a Compartment Fire," *Journal of Electromagnetic Engineering and Science*, vol. 21, no. 3, pp. 249-251, 2021.

[6] Y.-w. Li, Y. Hong-yong, L. Yang, X. Ru-feng, F. Ming, Y. Mengqi and H. Ling, "Experimental studies of electromagnetic wave attenuation by flame and smoke in structure fire," *Fire Technology*, vol. 53, no. 1, pp. 5-27, 20 July 2016.

[7] C. C. C., "Attenuation of Electromagnetic Radiation by Haze, Fog, Clouds, and Rain," *NASA STI/Recon Technical Report N, 76, 10364*, 1975. https://api.semanticscholar.org/CorpusID:129738399

[8] G. S. Robert Huva, "WRF Model Sensitivity to Spatial Resolution in Singapore: Analysis for a Heavy Rain Event and General Suitability," *Atmosphere*, vol. 13, no. 4, p. 606, 2022. https://doi.org/10.3390/atmos13040606

[9] A. Hassen, "Indicators for the Signal degradation and Optimization of Automotive Radar Sensors under Adverse Weather Conditions," Tu-darmstadt.de, 5 January 2007.

[10] J. Huang, S. Jiang and X. Lu, *Int. J. Infrared Millimeter Waves*, vol. 22, no. 6, pp. 917-922, 2001.

[11] J. Vargas, S. Alsweiss, O. Toker, R. Razdan and J. Santos, “An Overview of Autonomous Vehicles Sensors and Their Vulnerability to Weather Conditions,” *Sensors*, vol. 21, no. 16, p. 5397, 2021.

[12] Y. Banday, G. Mohammad Rather and G. R. Begh, "Effect of atmospheric absorption on millimetre wave frequencies for 5G cellular networks," *IET Commun.*, vol. 13, no. 3, pp. 265-270, February 2019.

[13] S. Bertoldo, C. Lucianaz and M. Allegretti, "77 GHz automotive anti-collision radar used for meteorological purposes," 2017.

[14] J. Wojtanowski, M. Zygmunt, M. Kaszczuk, Z. Mierczyk and M. Muzal, "Comparison of 905 nm and 1550 nm semiconductor laser rangefinders’ performance deterioration due to adverse environmental conditions," *Opto-Electron. Rev.*, vol. 22, no. 3, January 2014.

[15] M. Hadj-Bachir and P. d. Souza, "LIDAR sensor simulation in adverse weather condition for driving assistance development," *HAL open science,* 29 January 2019.

[16] N. C. Yi Lim, L. Yong, H. T. Su, A. Yu Hao Chai, C. K. Vithanawasam, Y. L. Then and F. Siang Tay, Review of Temperature and Humidity Impacts on RF Signals, 2020, pp. 1-8

[17] F. De Ponte Müller, "Survey on ranging sensors and cooperative techniques for relative positioning of vehicles," *Sensors (Basel),* vol. 17, no. 2, p. 271, January 2017.

[18] M. Sheeny, E. De Pellegrin, S. Mukherjee, A. Ahrabian, S. Wang, and A. Wallace, "RADIATE: A radar dataset for automotive perception in bad weather," 2020.

[19] E. D. Kaplan and C. Hegarty, Understanding GPS/GNSS : principles and applications, Boston; London: Artech House, 2017.

[20] S. Zang, M. Ding, D. Smith, P. Tyler, T. Rakotoarivelo and M. A. Kaafar, "The impact of adverse weather conditions on autonomous vehicles: How rain, snow, fog, and hail affect the performance of a self-driving car," *IEEE Veh. Technol. Mag.*, vol. 14, no. 2, pp. 103-111, June 2019.

[21] M. Marcus and B. Pattan, "Millimeter wave propagation: spectrum management implications," *IEEE Microwave Magazine*, vol. 6, no. 2, pp. 54-62, June 2005.

[22] Open-Meteo. (n.d.). Historical Weather API. https://open-meteo.com/en/docs/historical-weather-api

[23] Meteorological Service Singapore. (n.d.). Climate of Singapore. https://www.weather.gov.sg/climate-climate-of-singapore/

[24] Null, J. (n.d.). "El Niño and La Niña Years and Intensities," Golden Gate Weather Services. https://ggweather.com/enso/oni.htm

[25] Di Liberto, T. (2014, August 1). The Walker Circulation: ENSO's atmospheric buddy. NOAA Climate.gov. https://www.climate.gov/news-features/blogs/enso/walker-circulation-ensos-atmospheric-buddy

[26] Meteorological Service Singapore. (n.d.). Past Climate Trends. https://www.weather.gov.sg/climate-past-climate-trends/

[27] Qiao, Z., Wang, N., Chen, J., He, T., Xu, X., Liu, L., Sun, Z., & Han, D. (2023). Urbanization accelerates urban warming by changing wind speed: Evidence from China based on 2421 meteorological stations from 1978 to 2017. Environmental Impact Assessment Review, 102, 107189. https://doi.org/10.1016/j.eiar.2023.107189

[28] Effects and Emergency Management of Typhoon "LABUYO." (2013). In Republic of the Philippines | National Disaster Risk Reduction and Management Council. https://ndrrmc.gov.ph/attachments/category/12/FINAL_REPORT_re_Effects_and_Emergency_Management_of_Typhoon_LABUYO_UTOR_09-12AUG2013.pdf

[29] Meng, W., & Wang, Y. (2016). A diagnostic study on heavy rainfall induced by Typhoon Utor (2013) in South China: 1. Rainfall asymmetry at landfall. Journal of Geophysical Research Atmospheres, 121(21), 12781–12802. https://doi.org/10.1002/2015JD024646

[30] Liang, Y. F., Poh, K. L., Chia, V., Boon Hwan, & Koh, B., Geng Hao. (2023). DATA ANALYTICS AND MACHINE LEARNING ON WEATHER IN SINGAPORE AND IMPACT ON SENSORS PERFORMANCE. Retrieved from

https://www.dsta.gov.sg/staticfile/ydsp/projects/files/reports/RYDSP_Report.pdf

[31] National Geographic Education. (n.d.). Air Mass. Retrieved from https://education.nationalgeographic.org/resource/air-mass/

[32] H. Friis, "A Note on a Simple Transmission Formula," Proceedings of the IRE, vol. 34, no. 5, 1 May 1946.

[33] Brown University. (n.d.). Radiation patterns. Antennas II. https://www.brown.edu/research/labs/mittleman/sites/brown.edu.research.labs.mittleman/files/uploads/lecture25.pdf

[34] Electricity - Magnetism (n.d). Directivity of an antenna equation https://www.electricity-magnetism.org/directivity-of-an-antenna-equation/

[35] S. Jones, & P. O'Gorman. (2020, September 30). "How is climate change affecting the weather today?" MIT Climate Portal. https://climate.mit.edu/ask-mit/how-climate-change-affecting-weather-today

[36] National Centers for Environmental Information. (2020, November 18). "A Warming Earth is Also a Wetter Earth," National Centers for Environmental Information (NCEI). https://www.ncei.noaa.gov/news/warming-earth-also-wetter-earth

[37] National Aeronautics and Space Administration. (n.d.). "What is the greenhouse effect? - NASA Science." https://science.nasa.gov/climate-change/faq/what-is-the-greenhouse-effect/

[38] Education, U. C. F. S. (n.d.). Predictions of Future Global Climate | Center for Science Education. UCAR. https://scied.ucar.edu/learning-zone/climate-change-impacts/predictions-future-global-climate

[39] East coast braces for more floods. (1987, December 7). The Straits Times, pp. 14. Singapore.

[40] *1997 Singapore's Hottest, Driest Year*. (1998, January 17). Tehran Times. https://www.tehrantimes.com/news/5228/1997-Singapore-s-Hottest-Driest-Year

[41] Mcbride, J. l., Sahany, S., Hassim, M. E. E., Nguyen, C. M., Lim, S.-Y., Rahmat, R., & Cheong, W.-K. (2015). The 2014 Record Dry Spell at Singapore: An Intertropical Convergence Zone (ITCZ) Drought. *Bulletin of the American Meteorological Society*, *96*(12), S126–S130. https://doi.org/10.1175/bams-d-15-00117.1

[42] Saunders, M., & Lea, A. (2007, January 10). Summary of 2006 NW Pacific Typhoon Season and Verification of Authors' Seasonal Forecasts TropicalStormRisk.com. https://www.tropicalstormrisk.com/docs/TSRNWP2006Verification.pdf

[43] *Singapore experienced its second coolest June in 20 years, wettest in a decade: Met Service*. (2020). TODAY. https://www.todayonline.com/singapore/singapore-experienced-its-coolest-june-20-years-wettest-decade-met-service

[44] (n.d.). 2021 Climate and Weather: The Year in Review [Review of *2021 Climate and Weather: The Year in Review*]. Meteorological Service Singapore. https://www.weather.gov.sg/wp-content/uploads/2022/08/The-Year-in-Review-2021.pdf

[45] Koh, S. (2024, January 16). *Wet and cold weather expected to continue until end of January: Met Service*. The Straits Times. https://www.straitstimes.com/singapore/environment/wet-and-cold-weather-expected-to-continue-until-end-of-january-met-service

[46] *Advisories*. (2024). Nea.gov.sg. https://www.nea.gov.sg/media/news/advisories/index/thundery-showers-in-the-second-week-of-december

APPENDICES

Appendix 1: Fundamentals of Radiofrequency

RF is a range on the electromagnetic (EM) spectrum encompassing frequencies of 3 kilohertz (kHz) to 300 gigahertz (GHz). In most professional applications, calculations involving radio propagation are done in decibel (dB) form.

To convert a linear number *N* to dB form, the following equation is used:

$$N(dB) = 10\log_{10} N$$

And to convert a value in dB form back to linear, the equation is converted to exponential form:

$$N = 10^{\frac{N(dB)}{10}}$$

Free Space Path Loss

Common dB Definitions

dBm	= dB value of Power / 1 milliwatt	Used to describe signal strength
dBW	= dB value of Power / 1 watt	Used to describe signal strength
dBsm	= dB value of Area / 1 meter^2	Used to describe antenna area or radar cross-section
dBi	= dB value of antenna gain relative to the gain of an isotropic antenna	0 dBi is, by definition, the gain of an omnidirectional (isotropic) antenna

Fig. A1-1 Common dB Definitions

The free-space path loss (FSPL) formula is derived from the Friis transmission formula [32]. This states that in a radio system consisting of a transmitting antenna transmitting radio waves to a receiving antenna, the ratio of radio wave power received P_r to the power transmitted P_t is:

$$\frac{P_r}{P_t} = D_t D_r \left(\frac{\lambda}{4\pi d}\right)^2$$

Where

- D_t is the directivity of the transmitting antenna
- D_r is the directivity of the receiving antenna
- λ is the signal wavelength, and
- d is the distance between the antennas

From the equation, as distance between the antennas increase, the power received by receiving antenna decreases. Additionally, the greater the directivity of either antenna, the lower the power loss.

<u>Directivity</u>

The directivity of an antenna is the ratio of the maximum radiated intensity to the average radiated intensity of the antenna. A higher directivity suggests that the radiation pattern is strongly directional, which is advantageous for long-distance communication, as it maximizes the signal strength in the desired direction while minimizing it in others [33].

The directivity of an antennae is mathematically expressed as [34]:

$$D(\theta,\varphi) = \frac{U(\theta,\varphi)}{P_{avg}}$$

Where

- $D(\theta,\varphi)$ is the directivity of the antenna at angles θ (zenith angle) and φ (azimuth)
- $U(\theta,\varphi)$ is the radiation intensity of the antenna in a certain direction defined by angles θ and φ , and
- P_{avg} is the average power radiated by the antenna, which can be calculated by integrating the radiation intensity over all possible directions about the antenna

<u>Zenith angle and azimuth</u>

In a spherical coordinate system (Note: RF signals radiated by antennas propagate outward in a spherical manner), θ represents the zenith angle, while φ represents the azimuth. The zenith angle refers to the angle between the zenith of a spherical system and the direction (which can be represented with a line) which the RF signals propagate outward from an antenna while azimuth refers to the horizontal angle measured clockwise from a reference direction, often north on a plane.

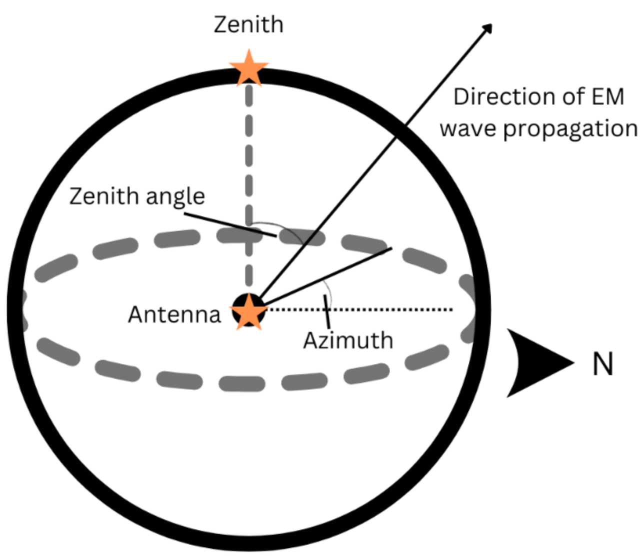


Fig. A1-2 Diagram of a spherical coordinate system

<u>Calculating Directivity into decibel form</u>

Directivity can be expressed in decibel form for easier calculation, via:

$$D(dB) = 10\log_{10}\left[\frac{D}{D_{reference}}\right]$$

The reference antenna can be taken as a theoretical perfect isotropic radiator, which means it radiates uniformly in all directions, and hence has a directivity of 1. This would simplify the equation to:

$$D(dB) = 10\log_{10} D$$

Appendix 2A: Collecting datasets from Open-Meteo.com

Based on the sample code provided by Open-Meteo.com [22], we created the following code to collate the datasets for analysis and machine learning:

```
#code which uses openmeteo api to get weather data

import openmeteo_requests

import requests_cache
import pandas as pd
from retry_requests import retry

# Setup the Open-Meteo API client with cache and retry on error
cache_session = requests_cache.CachedSession('.cache', expire_after = -1)
retry_session = retry(cache_session, retries = 5, backoff_factor = 0.2)
openmeteo = openmeteo_requests.Client(session = retry_session)

# Make sure all required weather variables are listed here
# The order of variables in hourly or daily is important to assign them correctly
below
url = "https://archive-api.open-meteo.com/v1/archive"
params = {
    "latitude": 1.4256, # Coordinates can be obtained by searching for a region
in the site
    "longitude": 103.8161,
    "start_date": "1980-01-01",
    "end_date": "2024-12-10",
    "hourly": ["temperature_2m", "relative_humidity_2m", "dew_point_2m", "rain",
"surface_pressure", "cloud_cover", "cloud_cover_low", "cloud_cover_mid",
"cloud_cover_high", "wind_speed_10m", "wind_speed_100m", "wind_direction_10m",
"wind_direction_100m"],
    "timezone": "auto"
}
responses = openmeteo.weather_api(url, params=params)

# Process first location. Add a for-loop for multiple locations or weather models
response = responses[0]
print(f"Coordinates {response.Latitude()}°N {response.Longitude()}°E")
print(f"Elevation {response.Elevation()} m asl")
print(f"Timezone {response.Timezone()} {response.TimezoneAbbreviation()}")
print(f"Timezone difference to GMT+0 {response.UtcOffsetSeconds()} s")

# Process hourly data. The order of variables needs to be the same as requested.
```

```
hourly = response.Hourly()
hourly_temperature_2m = hourly.Variables(0).ValuesAsNumpy()
hourly_relative_humidity_2m = hourly.Variables(1).ValuesAsNumpy()
hourly_dew_point_2m = hourly.Variables(2).ValuesAsNumpy()
hourly_rain = hourly.Variables(3).ValuesAsNumpy()
hourly_surface_pressure = hourly.Variables(4).ValuesAsNumpy()
hourly_cloud_cover = hourly.Variables(5).ValuesAsNumpy()
hourly_cloud_cover_low = hourly.Variables(6).ValuesAsNumpy()
hourly_cloud_cover_mid = hourly.Variables(7).ValuesAsNumpy()
hourly_cloud_cover_high = hourly.Variables(8).ValuesAsNumpy()
hourly_wind_speed_10m = hourly.Variables(9).ValuesAsNumpy()
hourly_wind_speed_100m = hourly.Variables(10).ValuesAsNumpy()
hourly_wind_direction_10m = hourly.Variables(11).ValuesAsNumpy()
hourly_wind_direction_100m = hourly.Variables(12).ValuesAsNumpy()

hourly_data = {"date": pd.date_range(
    start = pd.to_datetime(hourly.Time(), unit = "s", utc = True),
    end = pd.to_datetime(hourly.TimeEnd(), unit = "s", utc = True),
    freq = pd.Timedelta(seconds = hourly.Interval()),
    inclusive = "left"
)}
hourly_data["temperature_2m"] = hourly_temperature_2m
hourly_data["relative_humidity_2m"] = hourly_relative_humidity_2m
hourly_data["dew_point_2m"] = hourly_dew_point_2m
hourly_data["rain"] = hourly_rain
hourly_data["surface_pressure"] = hourly_surface_pressure
hourly_data["cloud_cover"] = hourly_cloud_cover
hourly_data["cloud_cover_low"] = hourly_cloud_cover_low
hourly_data["cloud_cover_mid"] = hourly_cloud_cover_mid
hourly_data["cloud_cover_high"] = hourly_cloud_cover_high
hourly_data["wind_speed_10m"] = hourly_wind_speed_10m
hourly_data["wind_speed_100m"] = hourly_wind_speed_100m
hourly_data["wind_direction_10m"] = hourly_wind_direction_10m
hourly_data["wind_direction_100m"] = hourly_wind_direction_100m

openmeteo_hourly_sembawang = pd.DataFrame(data = hourly_data)
print(openmeteo_hourly_sembawang) # Variable name is changed dynamically
# DataFrame is saved to csv files using the Data Wrangler extension
```

Appendix 2B: Data Preparation Code

```python
def load_region_data(region_files):
    """
    Load weather data for different regions from CSV files.

    Parameters:
    -----------
    region_files : dict
        Dictionary mapping region names to their respective CSV file paths

    Returns:
    --------
    dict
        Dictionary of DataFrames for each region, indexed by date
    """
    region_data = {}
    for region, file_path in region_files.items():
        df = pd.read_csv(file_path, parse_dates=['date'])
        df.set_index('date', inplace=True)
        region_data[region] = df
    return region_data

# Define region files
region_files = {
    "central": "weather_data_hourly/openmeteo/openmeteo_hourly_amk_park.csv",
    "east": "weather_data_hourly/openmeteo/openmeteo_hourly_changi.csv",
    "west": "weather_data_hourly/openmeteo/openmeteo_hourly_jurong_west.csv",
    "north": "weather_data_hourly/openmeteo/openmeteo_hourly_sembawang.csv",
    "south": "weather_data_hourly/openmeteo/openmeteo_hourly_sentosa.csv",
}

region_data = load_region_data(region_files)

# %%
def add_cyclical_features(df:pd.DataFrame, datetime_column='date'):
    """
    Add sine-cosine transformed cyclical features to the dataset.

    Parameters:
    - df (pd.DataFrame): DataFrame containing the datetime column.
    - datetime_column (str): Name of the datetime column.
```

```
    Returns:
    - pd.DataFrame: Updated DataFrame with cyclical features.
    """
    df = df.copy()

    if isinstance(df.index, pd.DatetimeIndex):
        datetime_column = df.index
    else:
        df[datetime_column] = pd.to_datetime(df[datetime_column])
        datetime_column = df[datetime_column]

    # Extract cyclical components
    df['hour'] = datetime_column.hour
    df['day_of_year'] = datetime_column.dayofyear
    df['month'] = datetime_column.month

    # Hour of the day (0-23)
    df['hour_sin'] = np.sin(2 * np.pi * df['hour'] / 24)
    df['hour_cos'] = np.cos(2 * np.pi * df['hour'] / 24)

    # Day of the year (1-365/366)
    df['day_of_year_sin'] = np.sin(2 * np.pi * df['day_of_year'] / 365)
    df['day_of_year_cos'] = np.cos(2 * np.pi * df['day_of_year'] / 365)

    # Month of the year (1-12)
    df['month_sin'] = np.sin(2 * np.pi * df['month'] / 12)
    df['month_cos'] = np.cos(2 * np.pi * df['month'] / 12)

    df.drop(['hour', 'day_of_year', 'month'], axis=1, inplace=True)

    return df

for region in region_data.keys():
    region_data[region] = add_cyclical_features(region_data[region])

# %%
def create_lagged_features(region_data, targets, lags=24):
    """
    Create lagged features for specified targets, excluding sine-cosine columns
from lagging.

    Parameters:
```

```
    - region_data (dict): Dictionary of DataFrames for each region
    - targets (list): List of column names to lag
    - lags (int): Number of lagged steps to create

    Returns:
    - dict: Dictionary of DataFrames with lagged features for each region
    """
    sine_cosine_columns = ['hour_sin', 'hour_cos', 'day_of_year_sin',
'day_of_year_cos', 'month_sin', 'month_cos']
    lagged_data = {}

    for region, data in region_data.items():
        df = data.copy()

        # Create the lagged features for the target columns
        lagged_columns = {}
        for target in targets:
            if target not in sine_cosine_columns:
                for lag in range(1, lags + 1):
                    lagged_columns[f'{target}_lag_{lag}'] = df[target].shift(lag)

        # Concatenate lagged features with the original DataFrame
        lagged_df = pd.concat([df, pd.DataFrame(lagged_columns, index=df.index)],
axis=1)

        # Add cyclical features
        # Directly add the sine-cosine columns to avoid creating a separate
DataFrame
        lagged_df[sine_cosine_columns] = df[sine_cosine_columns]

        # Drop rows with NaN values after shifting
        lagged_df.dropna(inplace=True)

        lagged_data[region] = lagged_df

    return lagged_data

# Define targets and regions
targets = ["temperature_2m", "relative_humidity_2m", "dew_point_2m",
           "rain", "surface_pressure", "cloud_cover", "cloud_cover_low",
           "cloud_cover_mid", "cloud_cover_high", "wind_speed_10m",
           "wind_speed_100m", "wind_direction_10m", "wind_direction_100m"]
```

```
# Call the function
lagged_region_data = create_lagged_features(region_data, targets, lags=24)
regions = list(lagged_region_data.keys())

# %%
def prepare_regional_model_data(lagged_region_data,
                                 target_region,
                                 train_start='1991-01-01',
                                 train_end='2019-12-31',
                                 test_start='2023-01-01',
                                 test_end='2024-12-31',
                                 ood_start='2020-01-01',
                                 ood_end='2022-12-31'):
    """
    Prepare training, testing, and out-of-distribution data for a specific
region's XGBoost model.

    Parameters:
    - lagged_region_data (dict): Dictionary of DataFrames with lagged features
for each region
    - target_region (str): The region for which the model is being prepared

    Returns:
    - Dictionary containing X_train, y_train, X_test, y_test, X_ood, y_ood
    """
    # Targets to predict (current data columns)
    targets = ["temperature_2m","relative_humidity_2m","dew_point_2m",
               "rain","surface_pressure","cloud_cover","cloud_cover_low",
               "cloud_cover_mid","cloud_cover_high","wind_speed_10m",
               "wind_speed_100m","wind_direction_10m","wind_direction_100m"]

    sine_cosine_columns = ['hour_sin', 'hour_cos', 'day_of_year_sin',
'day_of_year_cos', 'month_sin', 'month_cos']

    target_df = lagged_region_data[target_region]

    train_mask = (target_df.index >= train_start) & (target_df.index <=
train_end)
    test_mask = (target_df.index >= test_start) & (target_df.index <= test_end)
    ood_mask = (target_df.index >= ood_start) & (target_df.index <= ood_end)

    def prepare_data_for_region(mask):
```

```
        y_data = target_df.loc[mask, targets]

        X_data_list = []
        for region, region_df in lagged_region_data.items():
            # Debug: Check if the data is empty after masking
            print(f"Preparing data for region: {region}")
            region_subset = region_df.loc[mask]

            if region_subset.empty:
                print(f"Warning: Data for {region} is empty after masking.")

            # Ensure columns exist
            lagged_columns = [col for col in region_subset.columns if '_lag_' in
col]
            if not lagged_columns:
                print(f"Warning: No lagged columns found for {region}.")

            region_lagged_data = region_subset[lagged_columns].copy()

            # Rename columns to include region prefix
            region_lagged_data.columns = [f'{region}_{col}' for col in
region_lagged_data.columns]

            X_data_list.append(region_lagged_data)

        # Concatenate all lagged data
        X_data = pd.concat(X_data_list, axis=1)

        # Add normalised date/time columns
        sine_cosine_data = region_subset[sine_cosine_columns].copy()
        X_data = pd.concat([X_data, sine_cosine_data], axis=1)

        # Debug: Check data size before returning
        print(f"X_data shape: {X_data.shape}, y_data shape: {y_data.shape}")

        return X_data, y_data

    # Prepare the data for train, test, and ood sets
    X_train, y_train = prepare_data_for_region(train_mask)
    X_test, y_test = prepare_data_for_region(test_mask)
    X_ood, y_ood = prepare_data_for_region(ood_mask)
```

```
    return {
        'X_train': X_train,
        'y_train': y_train,
        'X_test': X_test,
        'y_test': y_test,
        'X_ood': X_ood,
        'y_ood': y_ood
    }

# Example of preparing data for a specific region
regional_data = {}
for region in regions:
    regional_data[region] = prepare_regional_model_data(lagged_region_data,
region)
```

Appendix 2C: XGBoost Model Training Code

```
def train_xgboost_model_with_tscv(X, y, n_splits=8, target_column=None):
    """
    Train an XGBoost model using Time Series Cross-Validation.

    Parameters:
    - X (pd.DataFrame): Input features
    - y (pd.DataFrame): Target values
    - n_splits (int): Number of splits for time series cross-validation
    - target_column (str, optional): Specific target column to predict.
                                     If None, trains a model for each target.

    Returns:
    - Dictionary of trained models, cross-validation results, and overall
performance
    """
    if target_column is None:
        models = {}
        cv_results = {}
        for target in y.columns:
            print(f"Training for {target}...")
            result = train_xgboost_model_with_tscv(X, y[target], n_splits,
target)
            models[target] = result['model']
            cv_results[target] = result['cv_results']
        return {
            'models': models,
            'cv_results': cv_results
        }

    tscv = TimeSeriesSplit(n_splits=n_splits, gap=24)

    cv_scores_mse = []
    cv_scores_mae = []
    models = []

    for fold, (train_index, test_index) in enumerate(tscv.split(X), 1):
        # Split the data
        X_train, X_test = X.iloc[train_index], X.iloc[test_index]
        y_train, y_test = y.iloc[train_index], y.iloc[test_index]

        # Prepare the data for XGBoost
        dtrain = xgb.DMatrix(X_train, label=y_train)
```

```
    dtest = xgb.DMatrix(X_test, label=y_test)

    # Define XGBoost parameters
    params = {
        'objective': 'reg:squarederror',
        'eval_metric': 'rmse',
        'learning_rate': 0.1,
        'max_depth': 6,
        'subsample': 0.8,
        'colsample_bytree': 0.8,
        'seed': 42,
        'device': 'cuda'
    }

    model = xgb.train(
        params,
        dtrain,
        num_boost_round=100,
        evals=[(dtest, 'eval')],
        early_stopping_rounds=10,
        verbose_eval=False
    )

    y_pred = model.predict(dtest)

    mse = mean_squared_error(y_test, y_pred)
    mae = mean_absolute_error(y_test, y_pred)

    print(f"Fold {fold} - Train MSE: {mse:.6f}, Train MAE: {mae:.6f}")

    # Store results
    cv_scores_mse.append(mse)
    cv_scores_mae.append(mae)
    models.append(model)

# Compute average cross-validation scores
cv_results = {
    'MSE': {
        'scores': cv_scores_mse,
        'mean': np.mean(cv_scores_mse),
        'std': np.std(cv_scores_mse)
    },
    'MAE': {
```

```
            'scores': cv_scores_mae,
            'mean': np.mean(cv_scores_mae),
            'std': np.std(cv_scores_mae)
        }
    }

    # Train final model on entire dataset
    final_model = xgb.train(
        params,
        xgb.DMatrix(X, label=y),
        num_boost_round=100
    )

    return {
        'model': final_model,
        'cv_results': cv_results
    }
```

Appendix 3A: Monthly Total Rainfall and Mean Temperature Graphs

<u>Sembawang (North)</u>

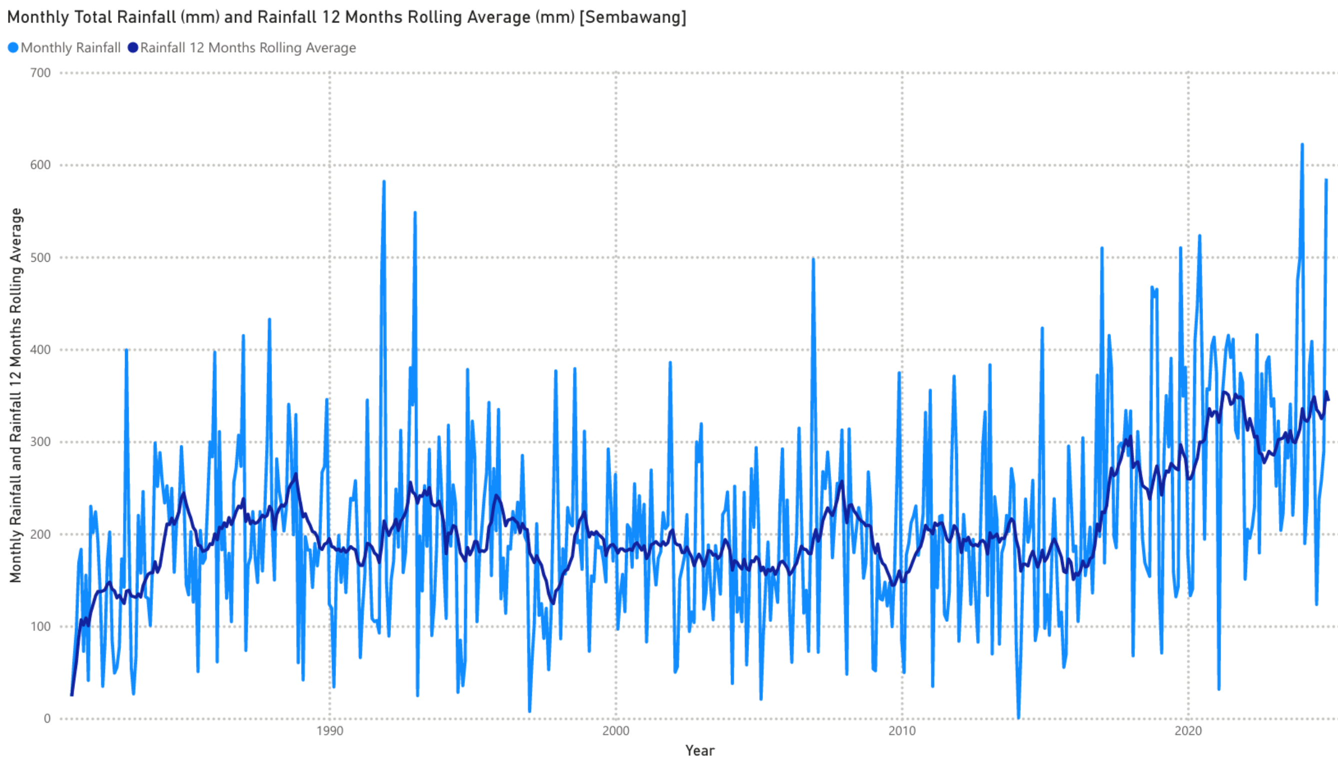


Fig. A3A-1 Graph of Monthly Total Rainfall (mm) and Rainfall 12 Months Rolling Average (mm) at Sembawang from 1981 January - 2024 November

From the 1980s to 2010s, monthly total rainfall at Sembawang was quite consistent, with around 200mm of rainfall per month, with some fluctuations and occasional spikes or dips. Over the past 8 years, however, rainfall totals have been increasing, with total monthly rainfall consistently hovering above 250mm per month. This trend can be attributed to the exacerbation of global warming, resulting in wetter weather in Singapore [35] [36].

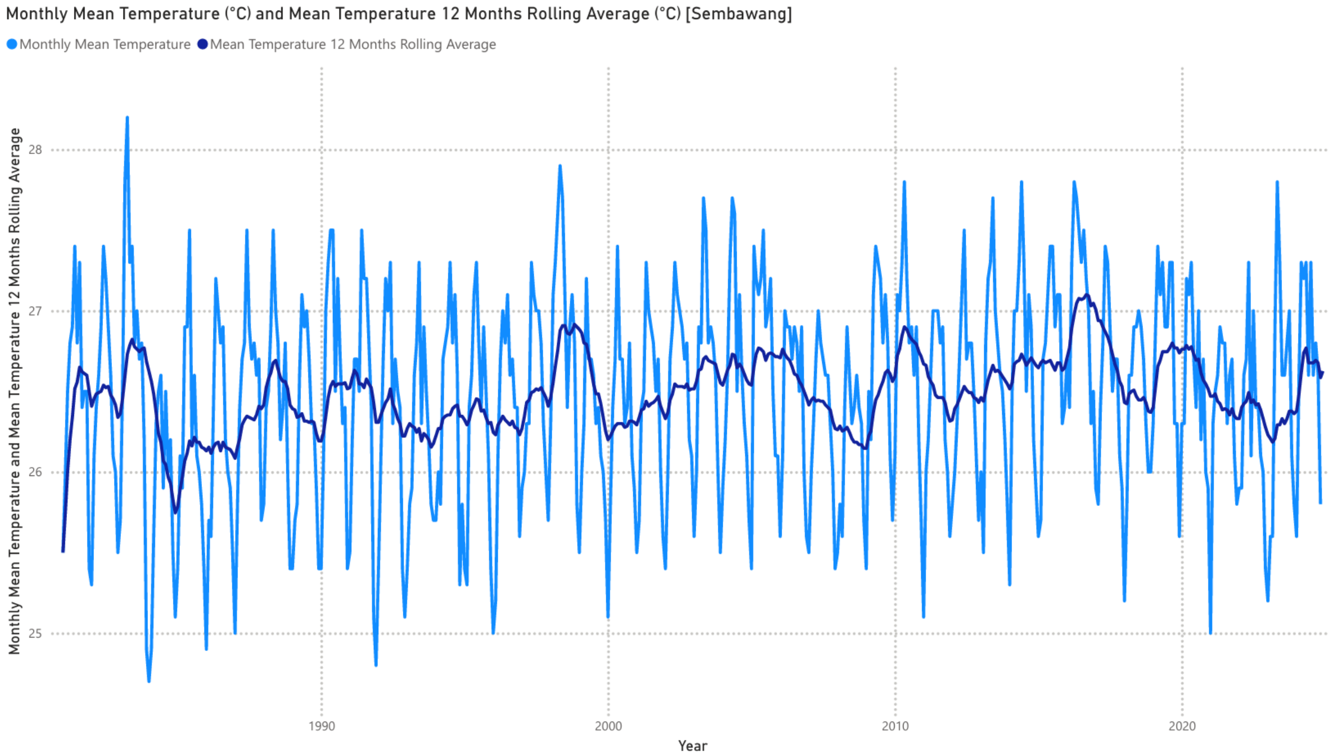


Fig. A3A-2 Graph of Monthly Mean Temperature (°C) and Mean Temperature 12 Months Rolling Average (°C) at Sembawang from 1981 January - 2024 November

Since the 1980s, temperatures at Sembawang have been fluctuating between 25°C and 28°C, but there is not an obvious general increasing or decreasing trend. Temperatures typically drop to a low during the first and final few months of the years, which is largely due to the Northeast Monsoon, and rise to a peak in the middle (around May and June). El Niño and La Niña events also contribute to the fluctuations in temperature, with El Niño causing slightly higher temperatures, and La Niña causing slightly lower temperatures than average.

Sentosa (South)

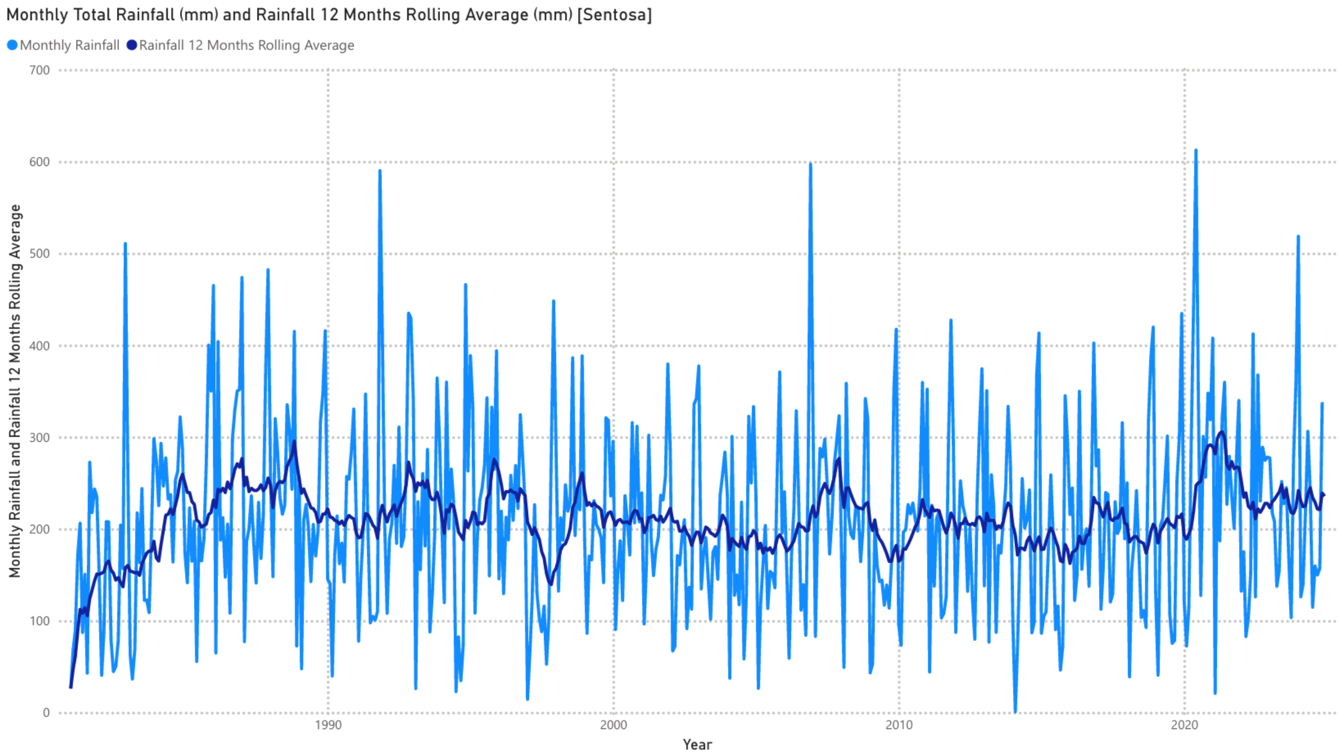


Fig. A3A-3 Graph of Monthly Total Rainfall (mm) and Rainfall 12 Months Rolling Average (mm) at Sentosa from 1981 January - 2024 November

Since the 1980s, monthly total rainfall at Sentosa has remained consistent, typically ranging between 100mm and 400mm, with only occasional spikes and dips caused by environmental factors such as typhoon remnants, monsoon surges, El Niño or La Niña events. Monthly total rainfall typically peaks from November to January, but that is not always the case, due to the volatility aforementioned factors.

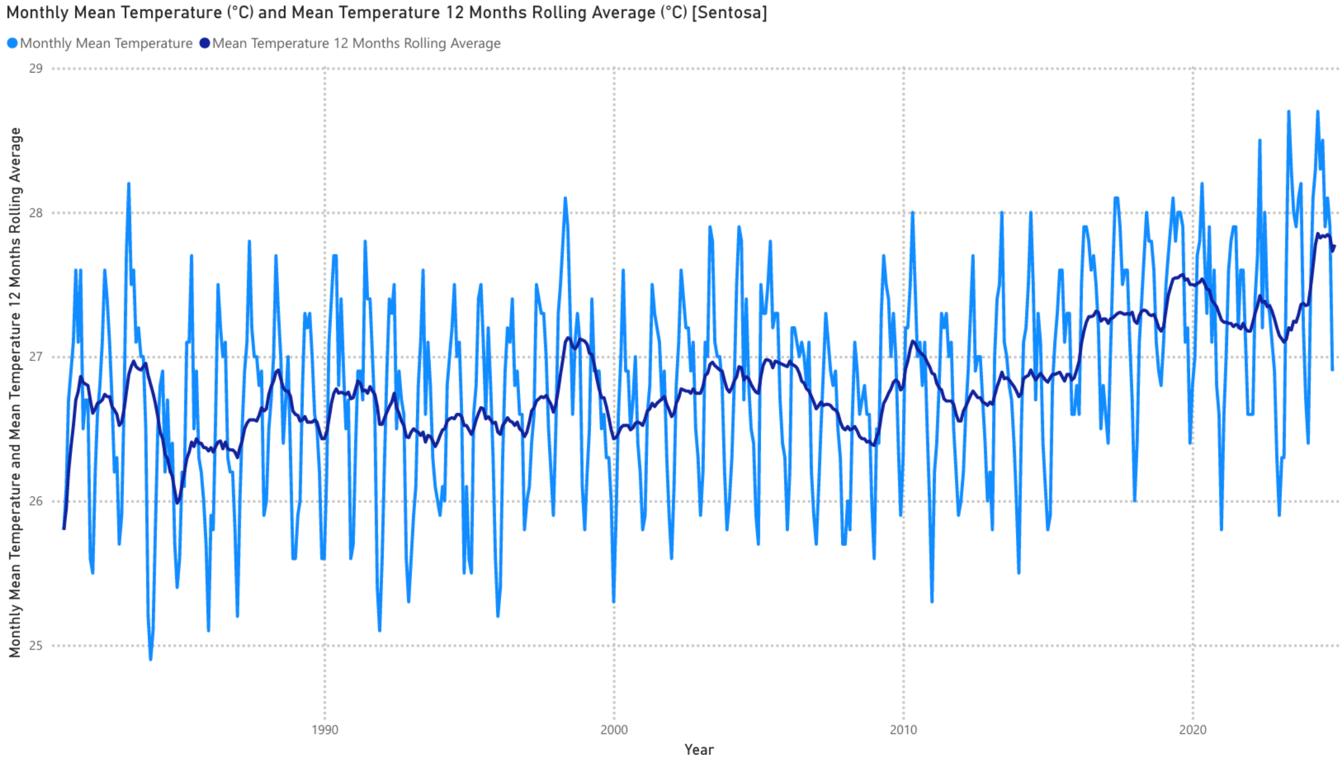


Fig. A3A-4 Graph of Monthly Mean Temperature (°C) and Mean Temperature 12 Months Rolling Average (°C) at Sentosa from 1981 January - 2024 November

From the 1980s to 2010s, temperatures at Sentosa fluctuated between 25°C and 28°C, and there is no clear increasing or decreasing trend. Over the past 8 years, mean temperatures have been steadily increasing, with temperatures hitting over 28.5°C and consistently staying above 26°C. This is largely thanks to the exacerbation of global warming, resulting in rising temperatures. Temperatures reach a peak during the middle of the year from May to July and typically reach a low from December to January.

Changi (East)

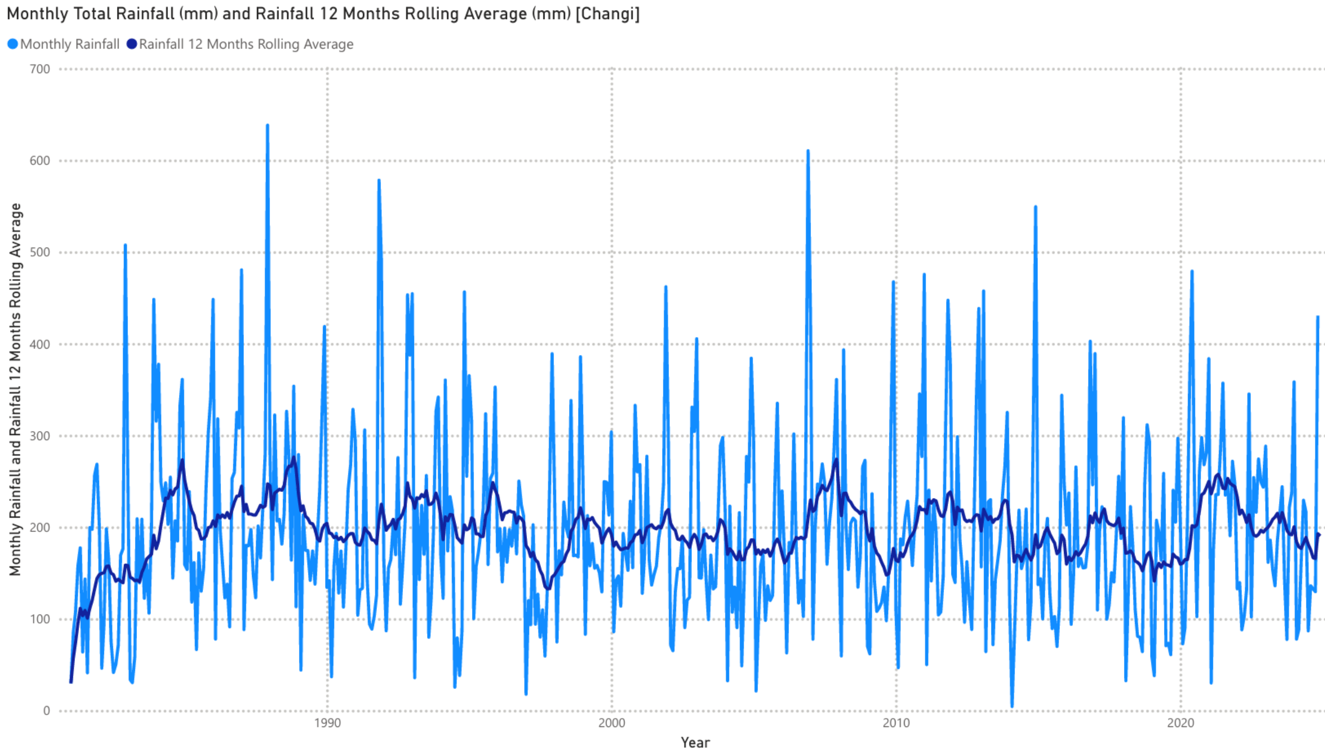


Fig. A3A-5 Graph of Monthly Total Rainfall (mm) and Rainfall 12 Months Rolling Average (mm) at Changi from 1981 January - 2024 November

Since the 1980s, monthly rainfall totals have been very consistent at Changi, ranging between 100mm and 400mm, with some variation due to environmental factors, such as typhoon remnants, monsoon surges, El Niño or La Niña events. Monthly total rainfall typically peaks from November to January, but that is not always the case, due to the volatility of the aforementioned factors.

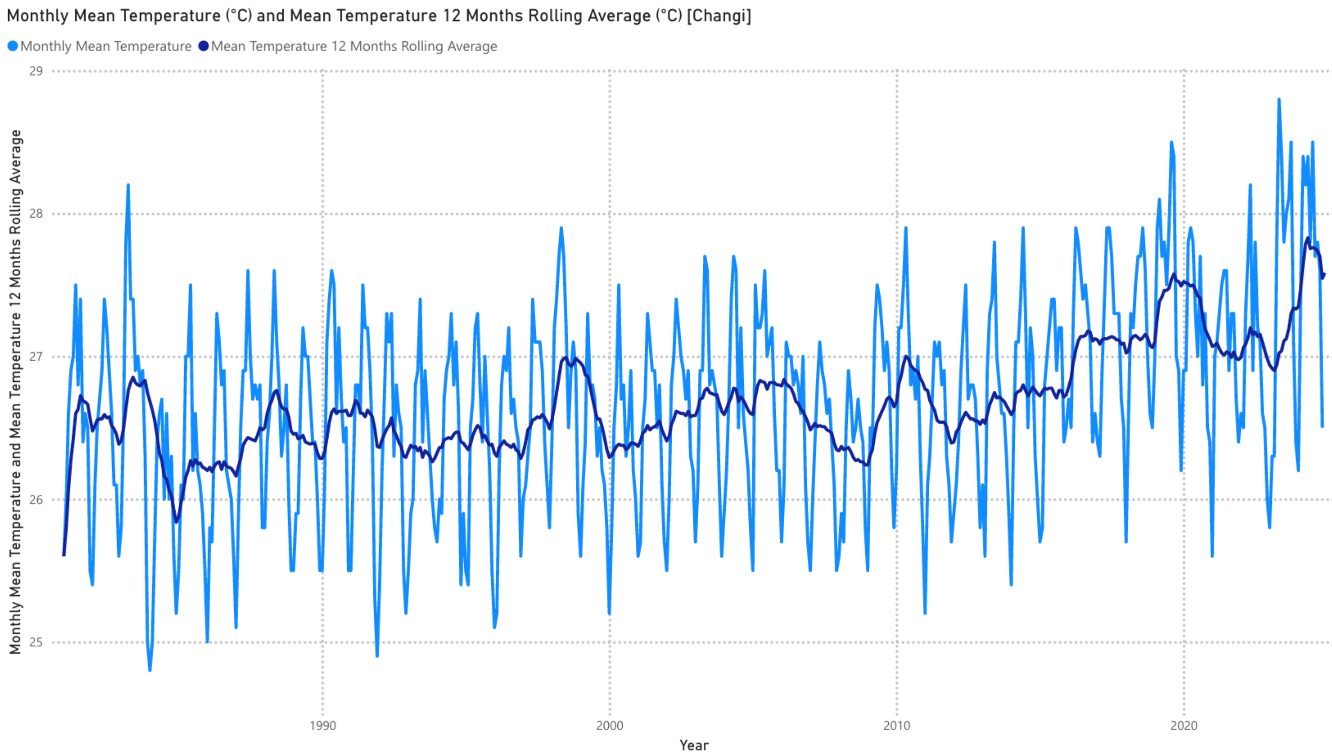


Fig. A3A-6 Graph of Monthly Mean Temperature (°C) and Mean Temperature 12 Months Rolling Average (°C) at Changi from 1981 January - 2024 November

From the 1980s to 2010s, monthly mean temperatures at Changi fluctuated between 25°C and 28°C, with significant variations during strong El Niño events, such as 1998 to 1999 and 2009 to 2010. Over the past 8 years, however, temperatures have been rising, with a notable spike in the last 2 years. Monthly mean temperatures now consistently exceed 26°C, reaching record highs at Changi not seen in the past 44 years.

Jurong West (West)

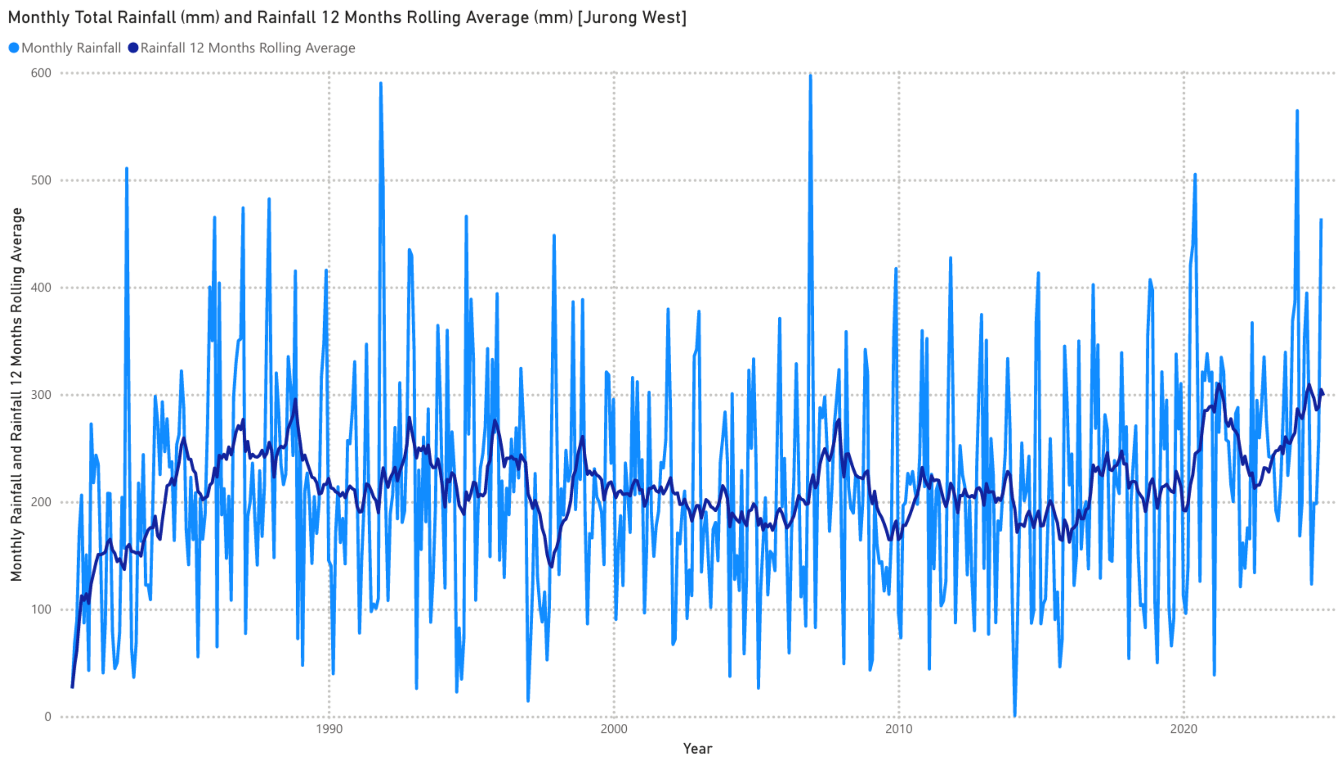


Fig. A3A-7 Graph of Monthly Total Rainfall (mm) and Rainfall 12 Months Rolling Average (mm) at Jurong West from 1981 January - 2024 November

Since the 1980s, monthly rainfall totals at Jurong West have remained quite consistent. However, over the past 3 years, an increasing trend has emerged. It is uncertain whether this trend will persist, or if it is a temporary variation caused merely by environmental factors, such as typhoon remnants, monsoon surges, or El Niño events. Monthly total rainfall typically peaks from November to January, but that is not always the case, due to the volatility of the aforementioned factors.

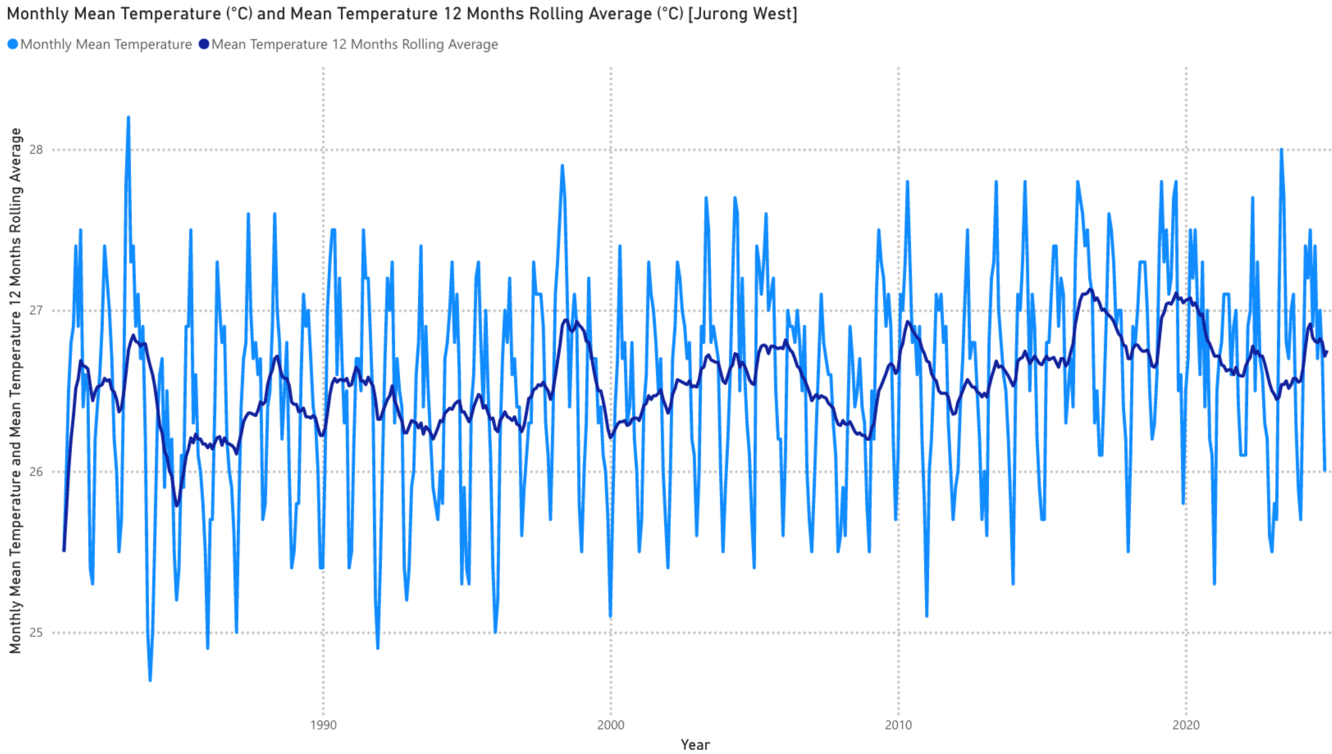


Fig. A3A-8 Graph of Monthly Mean Temperature (°C) and Mean Temperature 12 Months Rolling Average (°C) at Jurong West from 1981 January - 2024 November

From the 1990s to the present, monthly mean temperatures at Jurong West have generally increased, with temperatures ranging from 25°C to 27.5°C in the past, and temperatures ranging from 25.5°C to 28°C in the present. This is a 0.5°C increase in monthly mean temperatures on average.

<u>Ang Mo Kio Park (Central)</u>

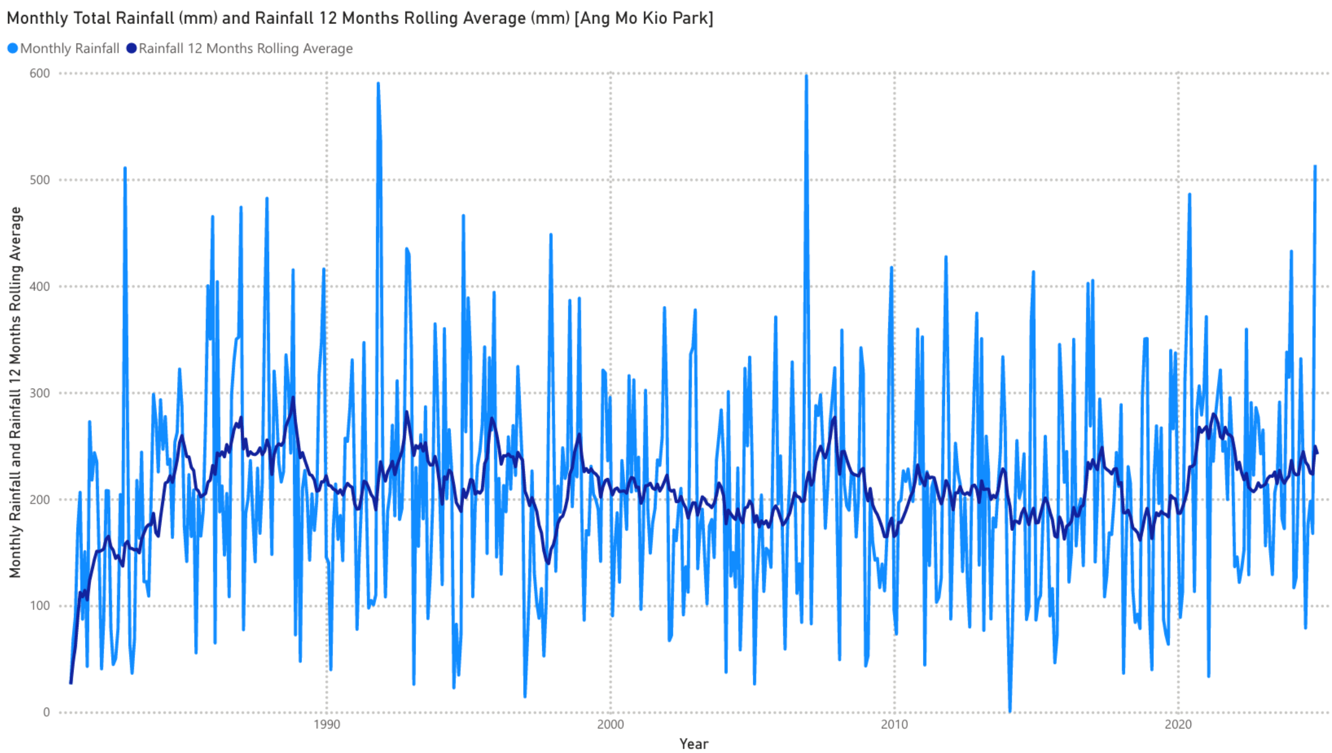


Fig. A3A-9 Graph of Monthly Total Rainfall (mm) and Rainfall 12 Months Rolling Average (mm) at Ang Mo Kio Park from 1981 January - 2024 November

Since the 1980s, monthly rainfall totals at Ang Mo Kio Park have been very consistent, typically ranging between 100 and 400mm, with some variation due to environmental factors, such as typhoon remnants, monsoon surges, El Niño or La Niña events. Monthly total rainfall typically peaks from November to January, but that is not always the case, due to the volatility of the aforementioned factors.

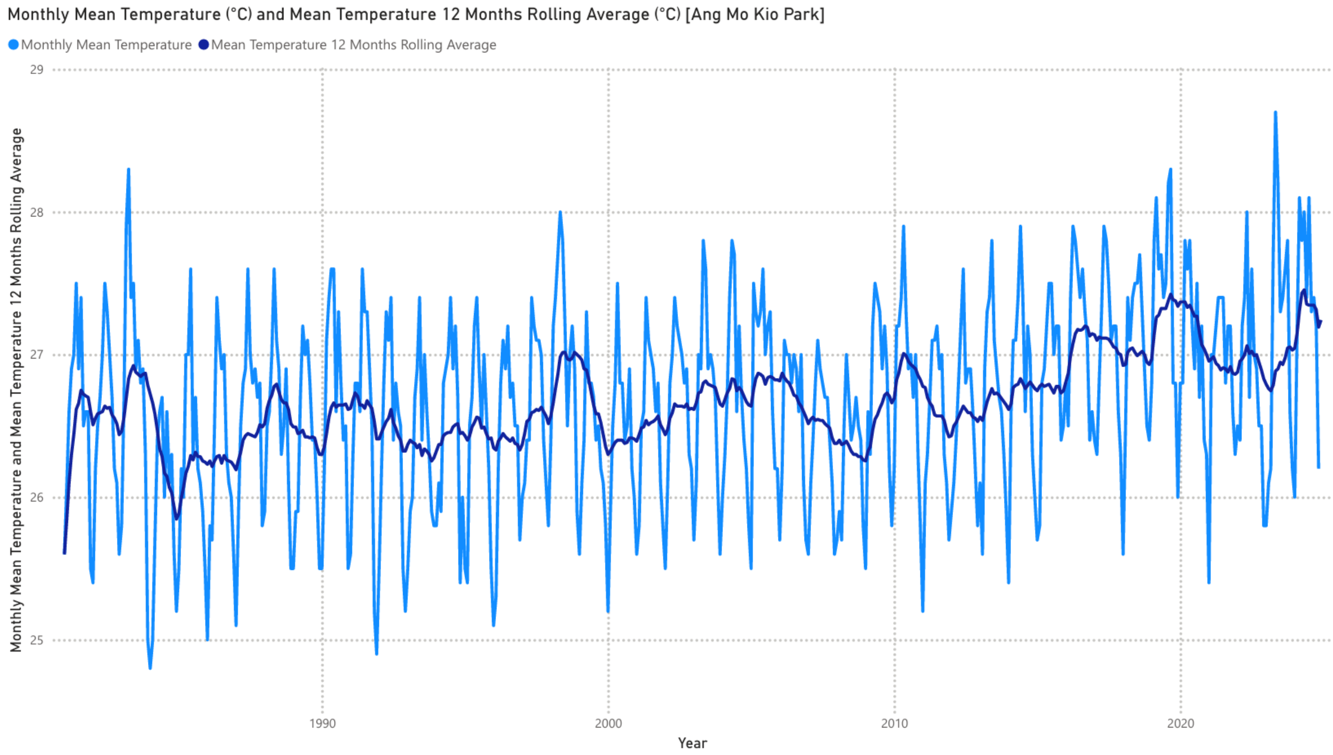


Fig. A3A-10 Graph of Monthly Mean Temperature (°C) and Mean Temperature 12 Months Rolling Average (°C) at Ang Mo Kio Park from 1981 January - 2024 November

From 1980s to the present, monthly mean temperatures at Ang Mo Kio Park have generally increased, with large increases in the last 8 years. This can be attributed to global warming, which has caused rising temperatures all across the world. Another possible factor that has contributed to the rise is El Niño events. Temperatures typically reach a peak during the middle of the year from May to July and typically reach a low from December to January.

Appendix 3B: Average Monthly Rainfall and Mean Temperature Trend Graphs

Just as the average monthly rainfall trends are plotted, the mean temperature trends across the months are too for the same span of time (1981 January - 2024 November). A dual axis graph of the average rainfall and mean temperature trends is plotted, to help identify possible correlation between the two different variables.

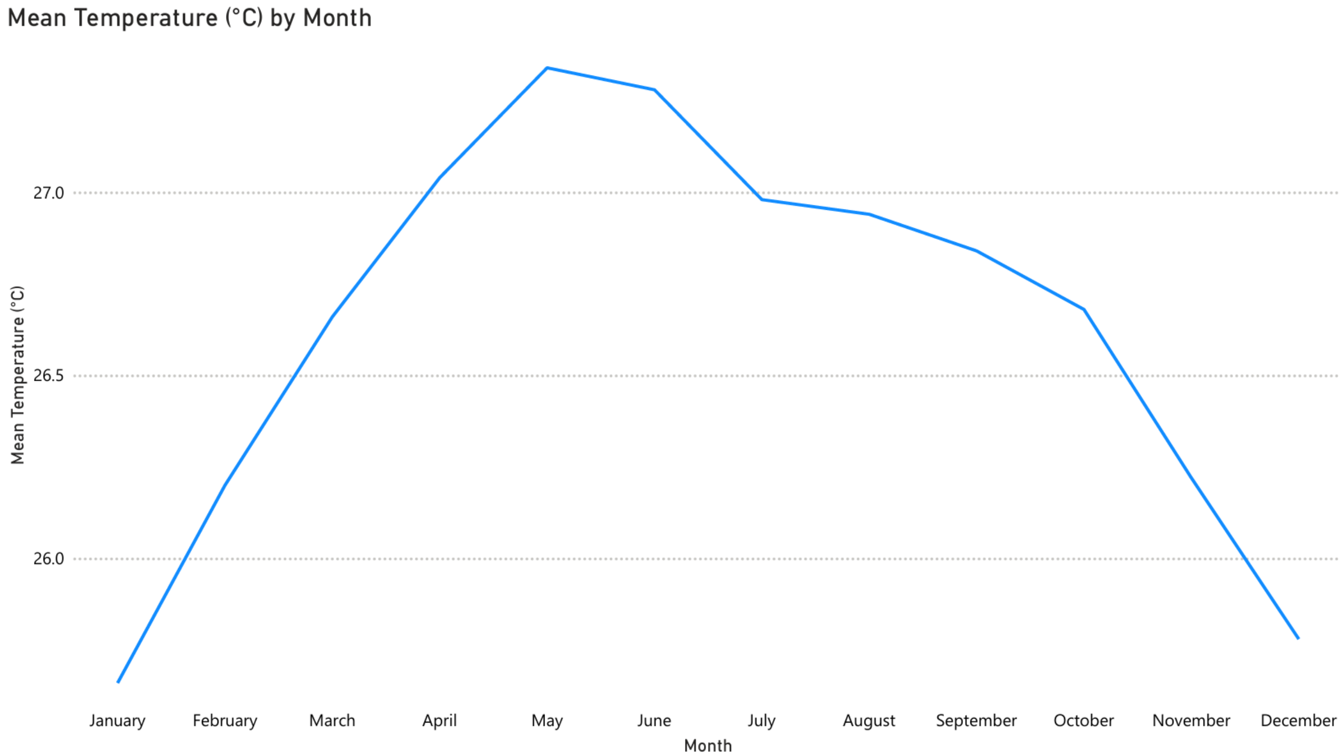


Fig. A3B-1 Graph of Mean Temperature (°C) Trend across Ang Mo Kio Park, Changi, Jurong West, Sembawang and Sentosa (1981 January - 2024 November)

During the Northeast Monsoon (from December to early March), temperatures are slightly lower compared to the other months of the year, primarily due to monsoon surges that bring cooler air from the northern hemisphere to Singapore. In contrast, during the Southwest Monsoon (from June to September), especially between July and September, there is minimal variation in temperature, as fewer external wind patterns impact Singapore. Singapore's tropical climate, characterised by consistent solar radiation and high humidity, results in minimal seasonal temperature variation, except when influenced by external factors like monsoon surges.

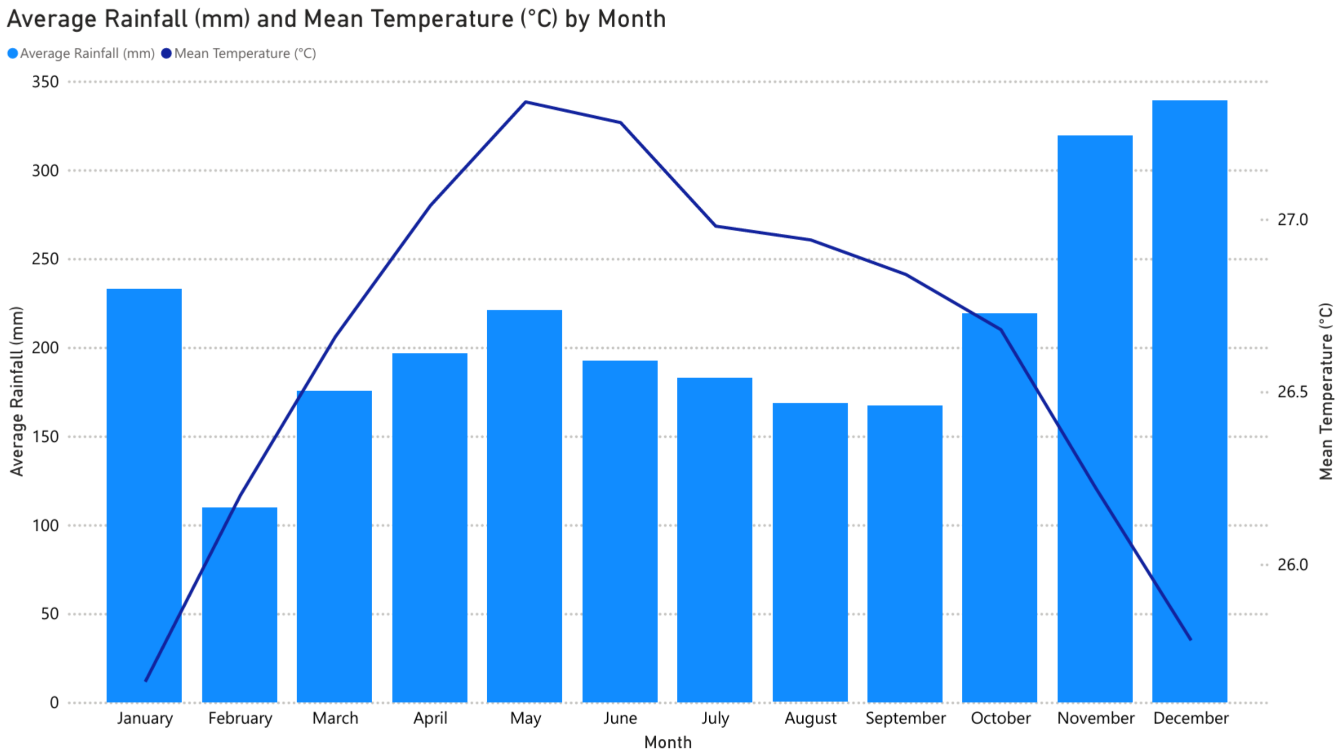


Fig. A3B-2 Dual axis graph of Average Monthly Rainfall (mm) and Mean Temperature (°C) Trends across Ang Mo Kio Park, Changi, Jurong West, Sembawang and Sentosa (1981 January - 2024 November)

Generally, when the mean temperature is lower, the average rainfall is higher, and vice versa. However, there are exceptions, due to the Southwest Monsoon affecting rainfall and temperature between June to September. Additionally, Singapore experiences a tropical climate, and hence, has consistently warm temperatures and minimal seasonal variation, thus the average rainfall from March to September is rather consistent.

Appendix 3C: Enhanced Greenhouse Effect and its impacts on Singapore's weather and sensors

The Greenhouse Effect is the process by which thermal energy is trapped near Earth's surface by greenhouse gases, such as carbon dioxide, nitrous oxide, and methane, which helps to maintain a warmer surface temperature than would otherwise occur [37]. The Enhanced Greenhouse Effect occurs when the production of greenhouse gases largely exceeds their depletion, leading to rising global temperatures.

In Singapore, a rise in temperature would increase the rate of evaporation, which would in turn speed up the water cycle. More water vapour would enter the atmosphere which could lead to more precipitation. A 2°C increase in temperature is predicted to make heavy rain events 1.7 times more likely, and 14% more intense [38]. This could increase Singapore's monthly total rainfall by a substantial amount.

Additionally, an increase in atmospheric water vapour could also result in more clouds being formed; however, scientists have yet to determine the influence of these clouds on Singapore's climate.

As the amount of rainfall increases, sensors in Singapore face higher risk of being affected by attenuation. This will most likely hinder the effectiveness of sensors, especially those working on radiofrequencies of 10GHz and higher. On the other hand, temperature changes have minimal, if any, impact on the transmission of radiofrequencies

Appendix 3D: Examining Anomalous Data

1982 December

There was a significant jump in rainfall in this month across all the regions in Singapore, likely due to the Northeast Monsoon.

1987 December

This month saw the highest amount of rainfall for Changi (East), but not in the other regions. An article from The Straits Times on 7th December this year confirms that the East Coast of Johor was facing flooding this month, stating that Malaysia's Meteorological Department warned of its flooding throughout the month until the next January [39]. Since Johor is just North of Singapore, it can be assumed that rainfall amounts in Singapore were similarly high.

1991 November-December

These two months saw one of the highest amounts of rain in our data timeframe. While it was not record-breaking, newspaper reports published by The Straits Times in this month mentioned flooding in Singapore as well. This was likely exacerbated by the Northeast Monsoon during these months.

1997 January

1997 was, at the time, said to be Singapore’s hottest, driest year on record, with a mean temperature of 28.2°C [40]. This can be seen in Singapore’s rainfall in January 1997, which was in the driest three months from 1981-2024. On top of January being right before the dry phase of the Northeast Monsoon that year, a very strong El Niño hit in 1997, bringing especially dry weather to Singapore.

2005 February (all regions)

The February of 2005, while not as dry as January 1997, is also among the driest months in Singapore’s history. Also, in the dry phase of the Northeast Monsoon, it is expected that this month would hold some of Singapore’s lowest rainfall amounts. However, the leading cause of its lack of rainfall was likely the dry spell that overcame Singapore in 2005, which is said to have lasted 40 days from January to February 2005 [41].

2006 December to 2007 January

From 2006 December to 2007 January, Singapore saw an abnormally high amount of rainfall of up to 350 mm. This was attributed to Typhoon Utor, a category 3 typhoon, which also caused floodings within Southeast Asia. These rain conditions also caused a significant decrease in temperature, with temperatures falling to 25°C. This typhoon occurred during the 2006 Pacific typhoon season which correspond to the period of greatest frequency for the formation of typhoons [42].

2014 February

This month was hit by a record-breaking dry spell, becoming the second driest calendar month in Singapore's history [41]. It is also in the dry phase of the Northeast Monsoon and is the driest month from 1981-2024 within the five selected regions.

2020 June

June 2020 was the second coolest June in two decades, and the wettest in a decade [43]. This is attributed to a moderate La Niña phenomenon during the year, and frequent Sumatra Squalls bringing early morning rain to the country. It has also been suggested that this was exacerbated by climate change, which has been increasing the severity of extreme weather events.

2021 February

This month was Singapore's second driest February [44]. According to our plotted graphs, it was the third driest month from 1981 to 2024.

2024 January

Across Singapore, the total rainfall was higher than average in January this year; the first fortnight saw a total rainfall that was 184% higher than average in Kranji [45], and generally heavy rain over the whole island. The month of January is near the end of the Northeast Monsoon, seeing a lot of rain every year.

2024 November (most regions)

As usual, the Northeast Monsoon brings significantly high total rainfall to Singapore, as seen in these months being some of the highest in the graphs, across all five regions. According to NEA, the latter half of November saw 185% above average total rainfall in Admiralty and Pasir Ris [46].

Appendix 3E: Graphs of hourly rainfall prediction

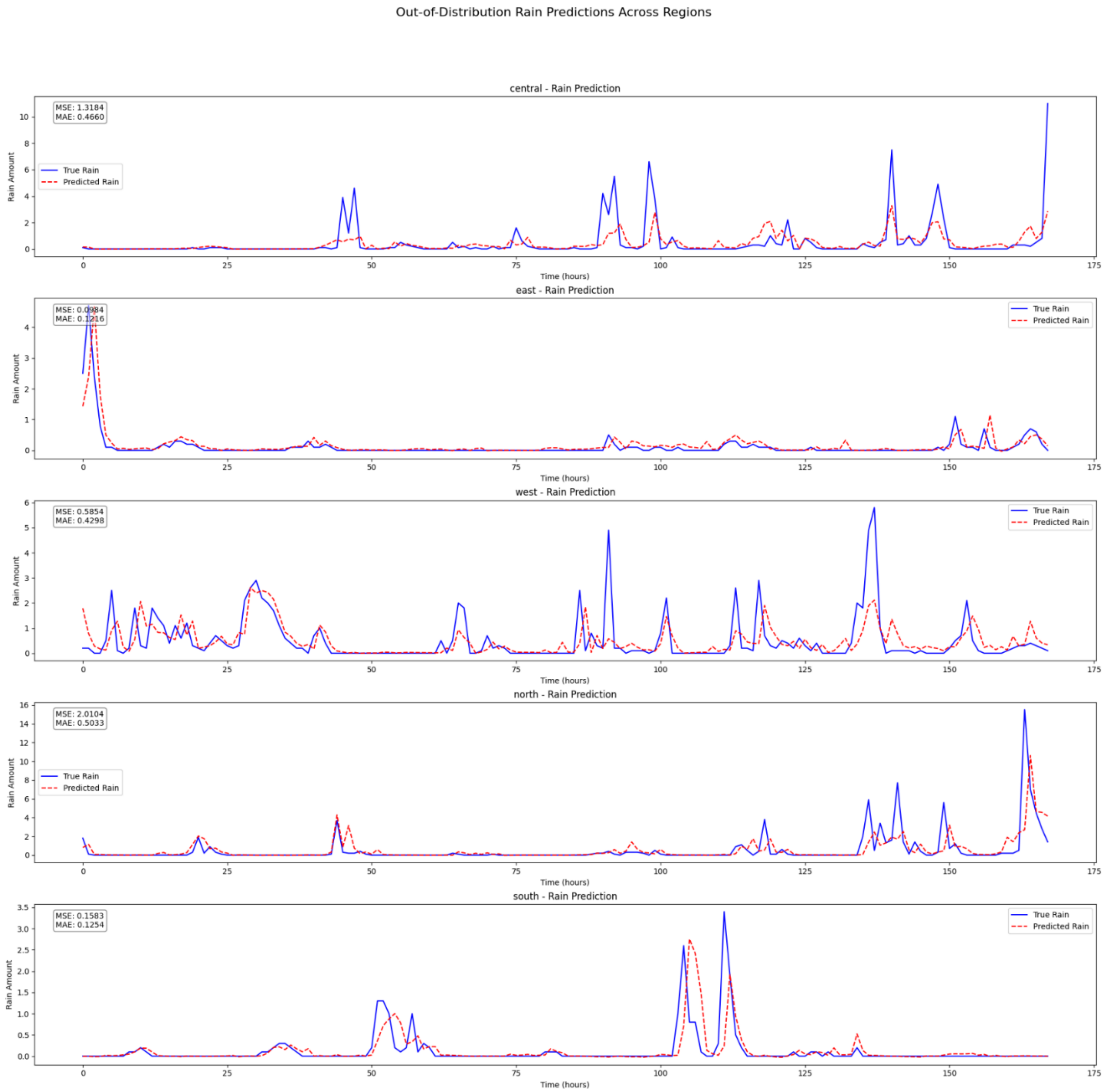


Fig. A3E-1 Graphs of hourly rainfall prediction against actual hourly rainfall for 175 hours, by region

Appendix 3F: Accuracy of Predictions

```
# Focus only on rain predictions for out-of-distribution data

plt.figure(figsize=(20, 4*len(regional_models)))
plt.suptitle('Out-of-Distribution Rain Predictions Across Regions', fontsize=16)

for region_idx, (region, region_data) in enumerate(regional_models.items()):
   X_ood = region_data['original_data']['X_ood']
   y_ood = region_data['original_data']['y_ood']

   # Randomly select a 1-week period
   start_idx = random.randint(0, len(X_ood) - 1750)  # 168 hours in a week
   end_idx = start_idx + 1750

   X_ood_subset = X_ood.iloc[start_idx:end_idx]
   y_ood_subset = y_ood.iloc[start_idx:end_idx]

   # Create subplot for rain prediction
   plt.subplot(len(regional_models), 1, region_idx + 1)

   # Get rain model and make predictions
   rain_model = region_data['results']['models']['rain']
   dood = xgb.DMatrix(X_ood_subset)
   y_pred = rain_model.predict(dood)

   # Plot true vs predicted rain
   plt.plot(y_ood_subset['rain'].values, label='True Rain', color='blue')
   plt.plot(y_pred, label='Predicted Rain', color='red', linestyle='--')

   plt.title(f'{region} - Rain Prediction')
   plt.xlabel('Time (hours)')
   plt.ylabel('Rain Amount')
   plt.legend()

   mse = np.mean((y_ood_subset['rain'].values - y_pred)**2)
   mae = np.mean(np.abs(y_ood_subset['rain'].values - y_pred))
   plt.text(0.02, 0.95, f'MSE: {mse:.4f}\nMAE: {mae:.4f}',
           transform=plt.gca().transAxes, verticalalignment='top',
           bbox=dict(boxstyle='round', facecolor='white', alpha=0.5))

# Adjust layout
plt.tight_layout()
```

```
plt.subplots_adjust(top=0.9)  # Adjust for suptitle

plt.show()

print("\nDetailed Rain Prediction Metrics:")
for region, region_data in regional_models.items():
   print(f"\nRegion: {region}")

   X_ood = region_data['original_data']['X_ood']
   y_ood = region_data['original_data']['y_ood']

   # Randomly select a 1-week period
   start_idx = random.randint(0, len(X_ood) - 1750)
   end_idx = start_idx + 1750

   X_ood_subset = X_ood.iloc[start_idx:end_idx]
   y_ood_subset = y_ood.iloc[start_idx:end_idx]

   # Focus only on rain prediction
   rain_model = region_data['results']['models']['rain']
   dood = xgb.DMatrix(X_ood_subset)
   y_pred = rain_model.predict(dood)

   print("  Rain Target:")
   print(f"    MSE: {np.mean((y_ood_subset['rain'].values - y_pred)**2):.4f}")
   print(f"    RMSE: {np.sqrt(np.mean((y_ood_subset['rain'].values -
y_pred)**2)):.4f}")
   print(f"    MAE: {np.mean(np.abs(y_ood_subset['rain'].values - y_pred)):.4f}")
   print(f"    Correlation: {np.corrcoef(y_ood_subset['rain'].values, y_pred)[0,
1]:.4f}")
```