Field of the paper ABC, a black cat; DEF, doesn't ever fret; GHI, goes home immediately.

Author One PhD, Department, Institution, City, State or Province, Postal Code, Country correspondingauthor@email.com Department, Institution, City, State or Province, Postal Code, Country

Funder One, Funder One Department, Grant/Award Number: 123456, 123457 and 123458; Funder Two, Funder Two Department, Grant/Award Number: 123459

Detecting Location Errors with ERA5 Pseudo Stations

Clara Bezeau[1], Elizabeth Geissel[1], Emily Hayes[1], Gertrude Miria[1], Connor Platt[1], William Whelan[1], Teng Yang[1], and Peter Thorne[2]

[1]Maynooth University
[2]Affiliation not available

March 07, 2024

Abstract

Significant efforts are made to eliminate biases from models and observations, especially at operational centres. However, these biases still significantly impact the quality of assimilated data products. In the case of numerical weather prediction, residual biases can result in suboptimal utilization of available data or even render them unusable. In climate research based on re-analyzed datasets, it can be difficult to distinguish between accurate signals and trends from inaccurate ones caused by biases in models and data.This study used a detection algorithm written in the R language to perform statistical computing and data analysis. The algorithm was applied to a synthetic study utilizing pseudo-stations based on ERA5 to simulate and detect instrumental effects. Rather than using observational data from real-world sources, the study generated artificial scenarios to guarantee the quality of the data assessment.ERA5 is a well-known atmospheric reanalysis product that was used to create simulated or pseudo-weather stations. These stations were designed to mimic actual stations but were generated computationally to enable controlled experimentation. The study constructed twenty-five pseudo-stations in Frankfurt, Germany, within the latitude 49–50° and longitude 8–9° in the Northern Hemisphere. The study utilized the ERA5 land surface dataset of hourly 2-m air temperature of September in 2013 and 2014. The study tool significantly improves data quality assessment by evaluating the synthetic dataset's precision, dependability, and general robustness. It introduces a range of factors to assess the degree to which the data quality can be enhanced and maintained, including station movements, errors, and noise.To determine the likelihood of the threshold correlation occurring at our confirmed noise threshold, the correlation values occurring at 1.53 for each locational trial were extracted. Our threshold correlation was evaluated to see if it occurred within a likely range of correlations occurring at 1.53 degrees of noise, where 0.9744052 is less than 0.9744667 but greater than 0.9781093. This process helps improve detection methods for data anomalies, contributing to advancements in data quality assessment.

Keywords: ERA5, Weather Station Validation, Statistical Noise, RStudio, Confidence Threshold

1. Introduction

As time progresses, our society relies more and more heavily on climate data analysis and necessitates reliable weather measurement to create long-term climate models, daily weather forecasts or even vulnerabilities assessments (Jones *et al.*, (2009) ; Rummukainen, (2012)). Weather forecasting and climatology first developed distinct traditions and data sources during the 19th century. This led to the emergence of climate modeling in the 1960s, bringing together the two fields and changing scientists' perspectives from a local to a global perspective (Barry and Chorley (2009); Edwards, (2010); Mauelshagen, (2014); Baker, (2017)). Forecasting the weather, however, was still difficult at the time because sampled weather balloons only began operating in the late 50's and records had poorly and inconsistent surface stations (Edwards, (2010); Kalnay, (2003). In the 1970s, climate modeling laboratories gained interest in energy and environmental policy, leading to an infrastructural overhaul (Edwards, (2010); Maraun and Widmann, (2018)). As the effects of global warming became all too apparent in the 1980s, scientists and policymakers established the Intergovernmental Panel on Climate Change (IPCC) to evaluate scientific data on climate change, its impact, and possible solutions (IPCC, (2009) ; Edwards, (2010) ; Baker, (2017)). In addition, from the 90's emerged a new source of global climate data through weather records reanalysis (Parker, (2016); Trenberth and Olson, (1988) ; Bengtsson and Shukla, (1988)) and nowadays, climate knowledge infrastructure is one of the most reliable source of data, constantly reviewed and reanalyzed with added metadata (Edwards, (2010)).

Climate research and meteorology both rely on station observation and reanalysis techniques. Station observation provides real-world data, while reanalysis techniques provide consistent weather information on a global scale and over continuous time by integrating multiple observational data and numerical models (Edwards, (2010); Rummukainen, (2012); Hersbach *et al.*, (2020)). In scientific research and application, it is often necessary to combine both to obtain more comprehensive meteorological data (Salcedo-Sanz *et al.*, (2020); Schauberger *et al.*, (2020)). Simulation models rely on physical theory while numerical models were developed by weather forecasters to compute large-scale atmospheric movements and anticipate weather patterns (Parker, (2016)). Subsequently, climate scientists adopted similar methodologies to simulate the Earth's climate over extended periods, ranging from years to decades (Pitman, (2003); Jiao *et al.*, (2021)). Additionally, by modifying the simulated variables and conditions, they utilize models to forecast how climate patterns will evolve as human activity affects the composition of the atmosphere and other climate-related systems.

In fact, three types of computer models are now used to understand global climate : simulation, reanalysis and data analysis models, however, this study is mainly focusing on the latest two. Reanalysis models originate from weather forecasting and are widely used datasets in studying weather and climate (Edwards, (2010); Doddy *et al.*, (2021); Jiao *et al.*, (2021)). Unlike pure simulations, these models simulate the weather and blend the results with actual weather observations to produce fully global, uniform data (Gleixner *et al.*, (2020); Ghajarnia *et al.*, (2022)). Reanalyses are valuable datasets for monitoring and comparing past and present climate conditions, testing the accuracy of past forecasts, driving numerical weather prediction (NWP) models, and identifying climate variations and change (Hersbach *et al.*, (2020) ; Jiao *et al.*, (2021)). Unlike data from instruments alone, climate statistics from reanalysis models cover the entire planet at all altitudes (Edwards, (2010)) and are increasingly used in various commercial sectors, including energy, agriculture, water resources, and insurance (Gleixner *et al.*, (2020); Doddy *et al.*, (2021)). On the other hand, data analysis models refer to the techniques, algorithms, and empirically derived adjustments used to process historical weather and climate records. These models are necessary as observing systems have undergone multiple changes over time and combining long-term records is still needed. In addition, data analysis models are employed to account for various factors such as instrument behaviors, data collection practices and weather station site changes and essential to adjust for the unevenness of observations in space and time. All in all, these techniques all are important to our society, being for forecasting, assessing current and future climate change but also mitigation. The data and models obtained can be used for seasonal drought prediction for example, which lead to better assessments, the development of new agricultural and water use policies or the creation of new infrastructures, more suitable or useful to the new climate condition, and so on (Bengtsson *et al.*, (2007); Dee *et al.*, (2014)).

One of the tools using such reanalysis models, is ERA5: in 2010, the European Center for Medium-Range Weather Forecasts (ECMWF) developed it as the fifth-generation Re-Analysis dataset and replaced the ERA-Interim dataset in 2019 (Hoffmann *et al.*, (2019); Jiao *et al.*, (2021); Ghajarnia *et al.*, (2022)). ERA5 is a weather forecasting system that employs advanced techniques like four-dimensional variational data assimilation (4D-Var) and a high-resolution numerical weather model to provide more precise and accurate spatial and temporal resolution. Compared to its predecessors, ERA-Interim, ERA5 has a much higher resolution with 31km and hourly against 79km every 3 hours, making it more reliable (Hersbach et al., (2020); McNicholl *et al.*, (2021); Ghajarnia *et al.*, (2022)). ERA5 uses a sophisticated numerical weather model that assimilates a diverse set of observational data to produce a comprehensive and high-quality representation of global atmospheric conditions (Jiao *et al.*, (2021); Yu *et al.*, (2021)). ERA5 combines observations from different sources such as weather stations, satellites, and ocean buoys, with a numerical weather model to generate a detailed and consistent representation of the Earth's atmosphere (Cucchi *et al.*, (2020)). This process is known as data assimilation, which involves adjusting the initial conditions of the weather model using observations to create a more accurate representation of the atmospheric state (Cucchi *et al.*, (2020); Ghajarnia *et al.*, (2022)). Over the past few decades, advancements in data assimilation techniques have significantly improved the accuracy of NWP forecasts (Kalnay, (2003); Parker, (2016)).

In a recent paper, Velikou *et al.* (2022) conducted an investigation into the ERA5 dataset's reliability in replicating mean and extreme temperatures across Europe. The findings of the study suggest that ERA5 is highly reliable for climate investigation over Europe, as it captures the mean and extreme temperatures very well. The high correlations ranging from 0.995 to 1.000 indicate that ERA5 can capture the annual cycle very well, as supported by previous studies by Doddy *et al.* (2021) and Jiao *et al.* (2021). Furthermore, McNicholl *et al.* (2021) found that satellite temperature performs better in the temperate region compared to the tropical region. This suggests that the accuracy of satellite data is influenced by the time of year and climate region, with milder temperatures producing better estimates.

These last years, ERA5 has become a widely used data source for temperature modeling due to its coverage of large land areas with regular latitude-longitude grids at 0.1° x 0.1° resolution. The reanalysis data also covers a period from 1950 to near-real-time hourly data, making it a valuable resource (Li *et al.*, (2022); Essa *et al.*, (2022)). While the gridded temperature derived from ERA5 reanalysis data provides the opportunity to interpolate temperature at arbitrary locations, this process can introduce errors and uncertainties, as noted in studies by Li *et al.* (2022) and Shi *et al.* (2021). To improve the accuracy of interpolated ERA5 temperature, a refinement method using an ANN model and measured station temperature was used to correct errors, as highlighted in studies by Li *et al.* (2022) and Hoffmann *et al.* (2019). However, the accuracy and biases of reanalysis datasets based on data assimilation continue to affect reanalysis tools, therefore it is essential to evaluate their performance (Yu *et al.*, (2021); Li *et al.*, (2022); Velikou *et al.*, (2022)). In regard to that, this study, thus, aims to evaluate the accuracy of the ERA5 temperature dataset, doing so by analyzing the measurements of twenty-five stations in Frankfurt from September 2013 and September 2014. The main purpose is to identify any potential location errors resulting from incorrect latitude or longitude signs and, if necessary, make the appropriate corrections.

2. Methodology

2.1 Study Location and Air Temperature

Frankfurt, Germany, is situated in Central Europe inland at geographical coordinates of 50.116 degrees latitude and 8.684 degrees longitude and sits at an elevation of 117 meters. The topography within a 3-kilometer radius of Frankfurt am Main city is mostly flat, with a maximum elevation change of 83 meters and an average elevation above sea level of 114 meters. Based on data from 1985–2015, the city experiences a peak in temperature in July and August, then slowly decreases to its minimum in December and January. September, thus, still shows warm weather, and while September 2013 stays quite stable with a peak in the early days, September 2014 is warmer and a bit more variable. However, if we compare the extreme temperatures throughout the year, we can observe similar results. Unstable weather, locally both in time and in topography, could be a slight difficulty to test station accuracy as temperatures can be as much as 4-5°C different from day to day, according to records.

Appropriate data from the .nc file, such as time, coordinates and temperature values, were extracted and converted into .csv format. This was due to the accessible format of .csv files. Csv files are commonly used for the storage and distribution of data. Analysis of the ERA5 was initiated once in this format. Originally two variables were selected for analysis (2m temperature and surface pressure), however upon initiation of analysis, a single variable, temperature, proved to be sufficient.

Each coordinate within a quarter degree of the chosen window (latitude 49–50° and longitude 8–9°) was extracted and assigned to a variable name. This allowed for individual analysis per coordinate or location. The quality and distribution for each location was then assessed. For the purposes of reanalysis in this study, the city of Frankfurt was divided into 25 distinct locations, which are tabulated below:

Table 1: Co-ordinates of each of the 25 locations in nearby regions to Frankfurt

Station location	**Longitude**	**Latitude**	**Name of location**
01	**8°E**	**50°N**	**Oestrich - Winkel**
02	**8.25°E**	**50°N**	**Mainz**
03	**8.5° E**	**50° N**	**Hasslock (East)**
04	**8.75° E**	**50° N**	**Gotzehain**
05	**9° E**	**50° N**	**Zellhausen**
06	**8° E**	**49.75°N**	**Nack (Northwest)**
07	**8.25° E**	**49.75°N**	**Dittelsheim-Hessloch**
08	**8.5° E**	**49.75°N**	**Gernsneim**
09	**8.75° E**	**49.75°N**	**Webern**
10	**9° E**	**49.75°N**	**Nack (North)**
11	**8° E**	**49.5°N**	**Hertlinghausen (West)**
12	**8.25° E**	**49.5°N**	**Weisenheim am Sand**
13	**8.5° E**	**49.5°N**	**Wohlgelegen**
14	**8.75° E**	**49.5°N**	**Lampenhain (West)**
15	**9°E**	**49.5°N**	**Soitzberg**
16	**8°E**	**49.25°N**	**Dernbach**
17	**8.25°E**	**49.25°N**	**Oberlustadt**
18	**8.5°E**	**49.25°N**	**Waghausel**
19	**8.75°E**	**49.25°N**	**Muhlhausen**
20	**9°E**	**49.25°N**	**Ehrstadt (North East)**
21	**8°E**	**49°N**	**Schleithal (North West)**
22	**8.25°E**	**49°N**	**Neuburg am Rhein**
23	**8.5°E**	**49°N**	**Grotzingen**
24	**8.75°E**	**49°N**	**Kleinvillars**
25	**9°E**	**49°N**	**Hohenhaslach**

2.2 ERA5

ERA5 is a dataset created by the European Centre for Medium-Range Weather Forecasts (ECMWF) and managed by Copernicus Climate Change Services (C3S). To produce a more precise spatial and temporal resolution compared to ERA-Interim, ERA5 uses advanced techniques like 4D-Var and a high-resolution numerical weather model (Hersbach *et al.*, (2020)). ERA5 assimilates a broad range of observational data, including satellite measurements, ground-based weather stations, and ocean buoys, thus, improving the accuracy of the initial conditions used in weather models. This dataset plays a significant role in weather forecasting by assimilating observational data, offering high-resolution information, maintaining consistency in data records, providing global coverage, and aiding in model validation. All of these factors contribute to the accuracy and reliability of temperature forecasts (Hersbach *et al.*, (2020) ; Yu *et al.*, (2021) ; McNicholl *et al.*, (2022)).

2.3 Air Temperature and ERA5

Several studies have assessed ERA5 efficiency both in terms of air temperature data and air temperature trends (Almeida and Coelho, (2023) ; Yilmaz, (2023)). According to them, ERA5 has a tendency to slightly underestimate air temperature in some regions, possesses a greater accuracy with simulations across flatter areas in contrast to locations of high altitude and complex, uneven terrain patterns (Almeida and Coelho, (2023)). While it may be best to be cautious for short term environmental studies, it is overall really effective to describe air temperature in Europe (Almeida and Coelho, (2023)). Focusing more on temperature trends, ERA5 is shown to be consistent with observed trends with a better accuracy over long term period, its trends can be on average slightly higher than observed but to a negligible level of difference (Yilmaz, (2023)). Factors such as time period, location of study, biases in ground

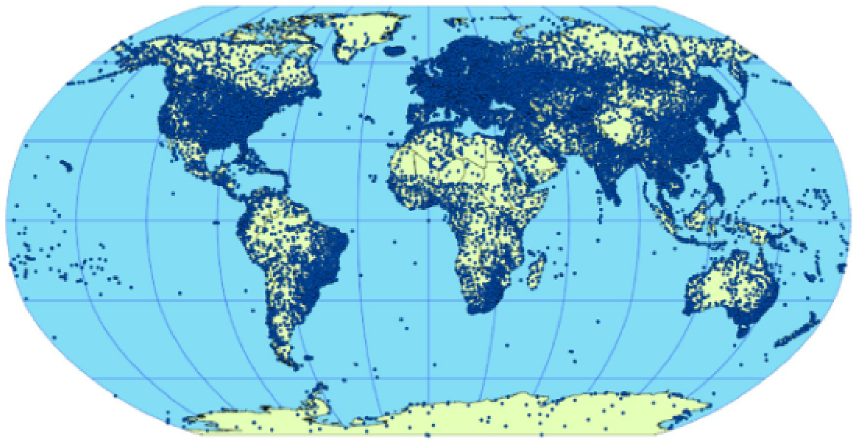

Figure 1: Map showing the global distribution of all known weather stations. Note the relatively high station density in Europe, hence our choice of study in Frankfurt.

observation and inhomogeneities can introduce trends and variability in the dataset that are inconsistent with observed values (Almeida and Coelho, (2023)). In light of these points, Almeida and Coelho (2023) suggest carrying out assessments of reanalysis datasets under different climatic conditions to eliminate as much uncertainty as possible, however, all in all, studies still agree that ERA5 can be highly trusted with air temperature. Therefore the data must be simulated to determine if an outlying data point is truly incorrect (whether from an alternate data set or a typo), if it is genuinely a novel change in data (e.g., freak events, creating a novel area for research), or if it is due to an unrecorded change in station location. A step-by-step approach used in the data manipulation, calibration and verification of this process is shown below.

2.4 Analysis Method

We used the Copernicus Climate Change Service (C3S) Climate Data Store (`https://climate.Copernicus.eu/climate-reanalysis`) to obtain hourly 2-m air temperature from ERA5-Land surface, which the European Centre provides for Medium-Range Weather Forecasts (ECMWF). The data was downloaded on December 7, 2023, in NetCDF format (CDS, n.d.) for a single month in September 2013 and September 2014; it was available at 0.250 (31 km) gridded resolution in the latitude 49–50 degrees and longitude 8–9 degrees.

The R software (version 4.1.2) and the packages "devtools" and GitHub ("ProcessMiner/nlcor") were used for the analyses in this study. The R package ggplot2 was utilised to generate the diagrams in the study. The following section of the report contains all scriptable R codes for the analyses performed in this investigation. We began by importing NetCDF as an NC file and converted it to CSV format. This is to improve the compatibility and ease in opening and manipulating CSV files in RStudio using the R programming language.

2.5 Statistical Method and Data Analysis

A statistical summary, including the minimum value, maximum value, quartiles, median and mean, was performed, allowing for trend comparison across all locations. The inter quartile range (IQR) was also calculated, providing a confidence interval for 50% of distributed data. Annual temperature follows a normal distribution. A Shapiro Wilk test was used to test the null hypothesis that temperature data is also normally distributed in our sample data for September. Contrary to expectation, the p-value was not insignificant, and so it was accepted that the data was more likely not to be normally distributed.

As the data did not follow a confident normal distribution, future consideration must be applied. Hence the 'rnorm()'

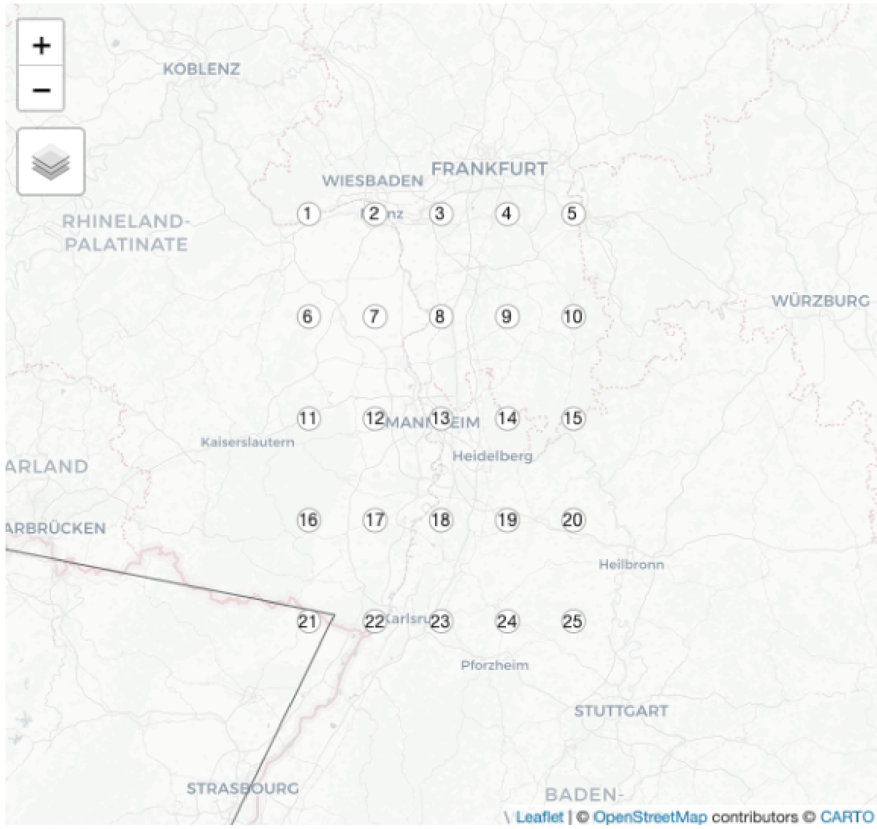

Figure 2: Map of the area surrounding Frankfurt, including the 25 locations examined in this study indicated by dots (refer to Table 1 for precise identification of these locations).

function commonly used in R to simulate normally distributed values, cannot be applied in this instance. Therefore an alternative method to simulate our data is required in the determination of false coordinate values. A location was chosen at random to test possible alternate methods.

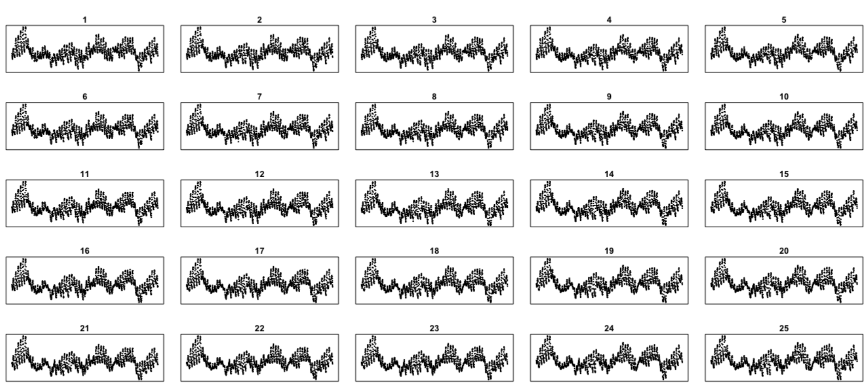

Figure 3: Original data distributions per location. Note the degree of similarity between individual graphs prior to subsequent analysis

Location 7 showed to have the highest correlation with location 8 and then 6, and the lowest correlation with location

25. If a confidence threshold is produced, it may be possible to statistically determine when a value from location 7 is truly from location 8, and not just highly correlated. Therefore various methods were assessed in their confidence to accurately reproduce the distribution of data from location 7.

3. Results

The function 'descdist()' was performed to estimate kurtosis and skew in location 7. Kurtosis indicates the length of a skew tail, whereas the resulting skew output indicates the skew bias. The function showed a positive skew and a kurtosis not far from three. Therefore three common right-skewed distributions could be considered for fit: Weibull, gamma and lognormal distributions. As the skew is very short tailed, a possible normal distribution could be accepted upon rejection of the other distributions, even with the previously rejected Shapiro-Wilks test.

The function 'fitdistr()' was used to assess each chosen possible distribution (Weibull, gamma, lognormal and normal). The resulting value provides a mean as the maximum likelihood parameter. Therefore whichever distribution is closest in result to the true mean is selected. The distribution which provided the most promising value was normal distribution.

Transformation of location 7 to a true normal distribution was considered, but given the above results the data was assumed to be nearest in distribution to a normal distribution. Sample data was then generated using the aforementioned normal distribution simulation function and was applied to the distribution of location 7. The resulting sample spread ,however, showed to have a very low correlation with the data, showing that reproducing sample data for location 7 with a normal distribution was still inappropriate.

An alternative idea was then produced: generating sample data under the same distribution curve of location 7. With values fitted to a graphical distribution, what amount of adjustment is required for correlation to no longer occur? To assess this, it must first be possible to regenerate the sample data to have a correlation of near 1 to the original observed values. First the density from the observed values was evaluated and used to create the model of observed distribution. The 'adjust' function was applied to shift data values in order to create our sample values. Correlation, with minimal adjustment, appeared to be close to zero, showing this method to be equally unusable as the previous.

It was then realised that the function 'cor()' was being used to measure the correlation, however this measures linear correlation, with our data being non-linear. Therefore 'nlcor()' was introduced and previous correlation calculations were reassessed non-linearly. The correlation values improved slightly, but not significantly enough to be accepted. The need for a probability matrix for each temperature was then considered, however deemed to introduce unwanted bias.

As the previous methods had provided little progression, an entirely different approach was deliberated. Rather than creating sample data through the generation of individual values, the existing values could themselves shift slightly, therefore, overtime, the data will eventually deviate in correlation from observed location values. The R function used to achieve this is the base function 'jitter()'. If 0.1% of change can be added to each value in progression, at what amount of change in jitter (or in this case 'noise') can we say that the value no longer belongs to, or correlated with the original dataset? If correlation occurs with surrounding data, at what stage or amount of noise does this occur? The default amount of jitter created is the factor by one fifth of the smallest difference between observed values. Therefore a minimal amount of noise (within realistic values) is applied. However temperature values recorded can be greater than 5 significant figures, so the noise applied would be too insignificant. Therefore our chosen noise value is set to amount and not factor within the function. This resulted in extremely high correlation, near 1 (0.99999998) with $p<0.05$, so the resulting sample dataset is accepted. This was then applied to location 7 in varying degrees of noise, from 0.01 to 4.00 in increments of 0.01, and repeated five times so as to gain more accurate value estimates. Further trials would be carried out, however, this test was extensive and required an extended duration. This was also carried out on the additional 24 locations.

Once 5 trials were run for every location, the max correlation for each row within a trial was calculated. This indicated the amount of noise added where the sample data no longer had the highest correlation with observed data for the location, and instead had a higher correlation with a nearby location. The degree of noise and correlation were extracted for these threshold points.

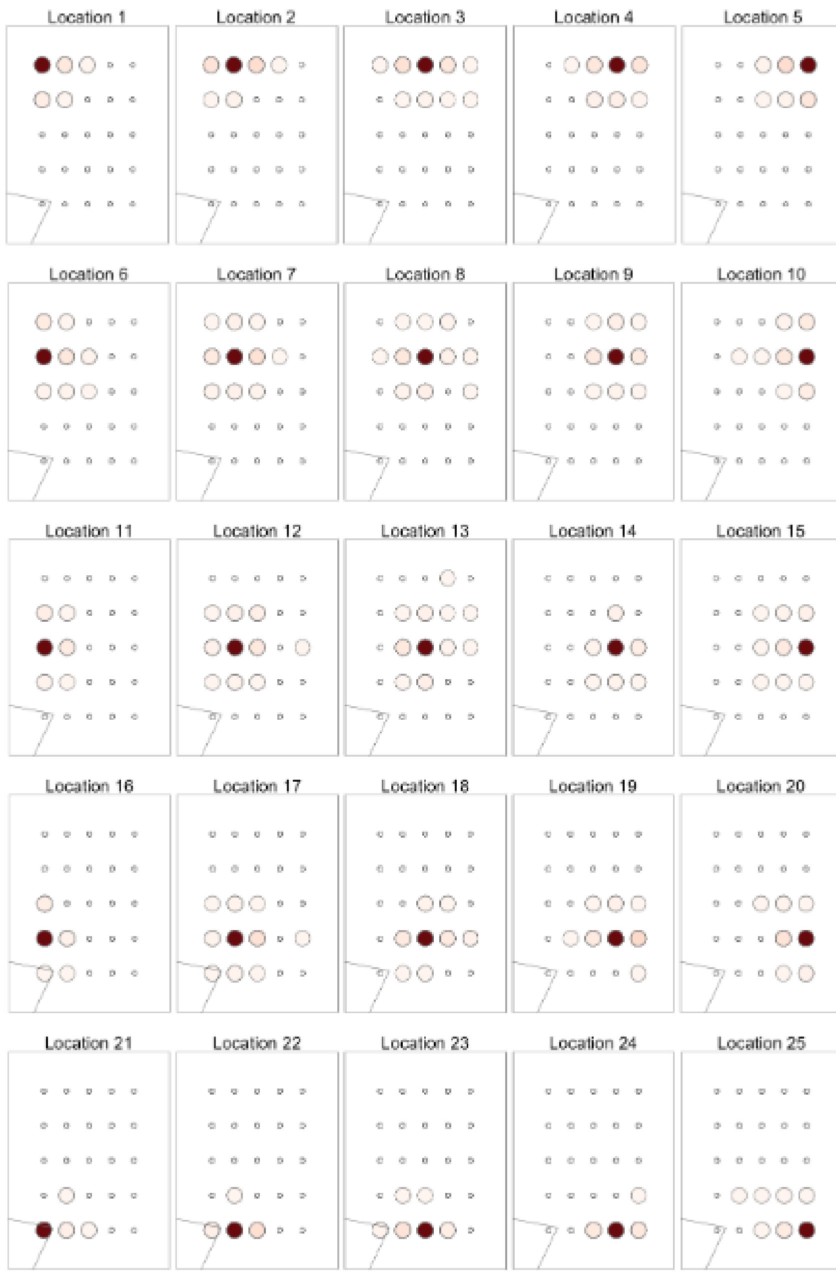

Figure 4: R elative correlation values between locations. Darkest coloured dots equate to greater correlation frequency.

Each location was then plotted by highest correlation frequency, allowing for visualisation in the spread of locational correlations for each location. It was hence decided that locations that did not have all neighbouring locations present were excluded, as this lack in data could cause a potential skew in our results, as potential correlations could not occur. Therefore locations 7, 8, 9, 12, 13, 14, 17, 18, 19 were chosen to continue with analysis.

Sample data for the spread of values in correlations and degree of noise were created to generate larger datasets, ensuing confidence in our resulting data. These were validated using a t-test. A Shapiro-Wilks test was performed to assess distribution. Both variables indicated normal distribution, however original correlation values, when plotted, showed to have a slight skew to the left, and so the degree of noise was accepted as our variable for analysis, due to its visual confidence and clear statistical normal distribution.

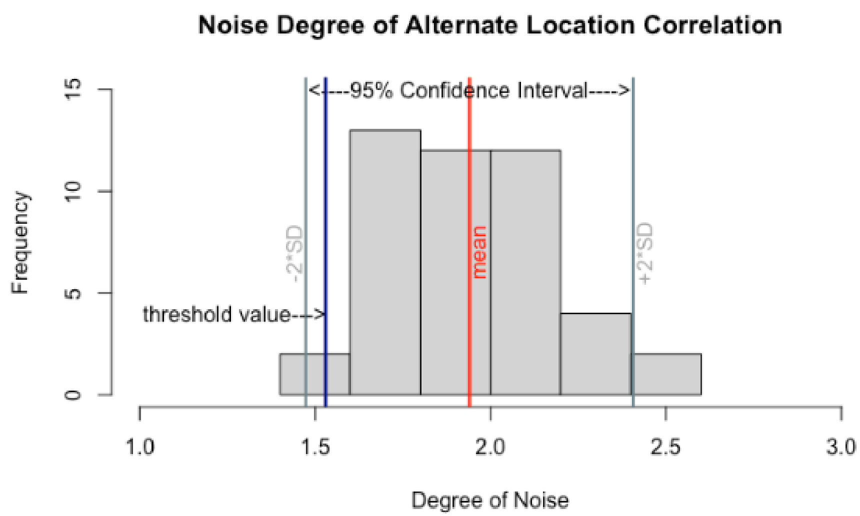

Figure 5: Histogram of original locational noise values. Threshold value indicating correlation deviance shown in blue

A confidence interval was then created within two standard deviations of the mean. This was to ensure that our lowest threshold value () was confidently within expected or possible threshold values. 1.481252 <1.53 >2.427654

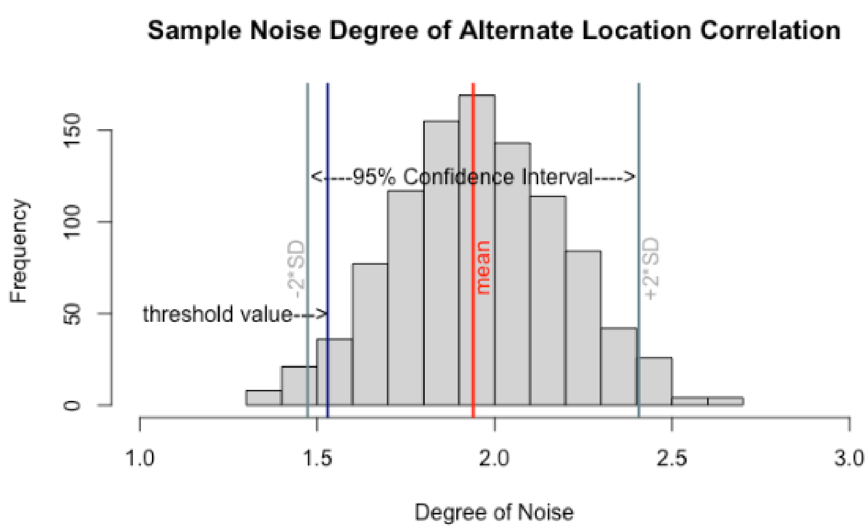

Figure 6: Sampled noise with threshold value and confidence interval included

To assess the likelihood of the threshold correlation occurring at our confirmed noise threshold, the correlation values occurring at 1.53 for each locational trial were extracted. Our threshold correlation was assessed to see if it occurred within a likely range of correlations occurring at 1.53 degrees of noise. $0.9744052 < 0.9744667 > 0.9781093$

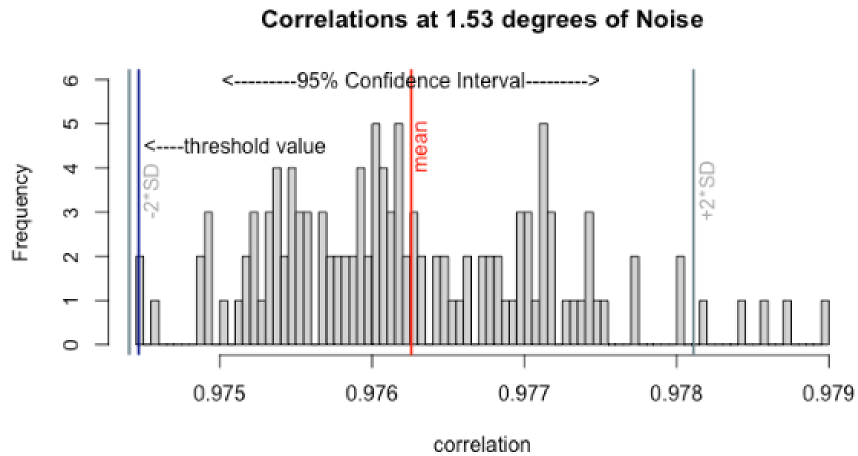

Figure 7: Precise frequency of correlation values between all 25 locations given a value of simulated noise of 1.53 degrees

4. Discussion

4.1 Location Errors in the ERA5 datasets

The distribution of temperature based on satellite measurements, on a given time and date, can be seen for the 25 stations in Fig. 3. From this graph, it is clear that in the surrounding areas of Frankfurt, the temperature values are spread evenly. The even spread of temperatures shows that the temperature data produced by ERA5 is potentially reliable, as opposed to a dataset with large fluctuations between data points. The high spatial resolution of ERA5 means that temperatures can potentially be mapped out accurately over relatively small geographical areas. This would make it a valuable dataset, which has been noted in Figure 4. It is clear that for each year, the temperature difference varies in a cyclic pattern. This can be observed when following the median of each location as the maximum likelihood parameter. Location 7 , 8, 9, 12, 13, 14, 17, 18, 19 were chosen to continue further analysis of true mean with promising value for a normal distribution with the result shown in Figure 5.

The accuracy of results reflected within ERA5 datasets has been shown to be significantly dependent on the location of the area being studied, with some areas known for having less reanalysis potential in comparison to others. Antarctica is a notable example of this, owing its lack of study potential to a lack of long-term direct observations, most of which are largely constrained to coastal areas (Tetzner *et al.*, 2019). Therefore, reanalysis is often used as the sole means to obtain reliable estimates of atmospheric structure through time by constraining a physical atmospheric model by use of what few observational records exist (Bracegirdle, 2013). Though we do not anticipate a lack of observations to be a significant issue in this study, there still remains some potential sources for significant biases and errors in our measurements, such as land use cover and change, which can have notable implications for factors such as albedo and rates of radiative surface temperature change (Li *et al.*, 2023). From this observation, it stands to reason that heavily urbanised areas would frequently display higher temperature measurements due to the presence of the urban island heating effect, creating a potential positive skew of measurements unrepresentative of true temperature values. Furthermore, as explained previously, there were a significant number of cases in which certain locations displayed particularly strong correlations with others. Given the lack of geological complexity within the study area, it stands to reason that there was noticeable error in the data, and that it was likely influenced by potential misreporting of station location.

Errors in station location can arise due to several factors, including inaccuracies in GPS signals, interference caused by multipath or atmospheric conditions, poor satellite geometry, errors with receiver clocks, or even deliberate interference. These variables can result in inaccuracies when calculating the precise position of a GPS receiver.

For location coordinates, noise can contribute to inaccuracies, causing slight variations or shifts in the reported location as shown in Figure 7. In the case of 2-meter surface temperature data, noise can introduce errors or biases into temperature readings, making it difficult to identify accurate temperature patterns and trends. This can impact the reliability of weather forecasts, climate studies, and other applications that rely on precise temperature data.

4.2 Data Reliability and ERA Datasets

Owing to the relatively flat gradient of the locations of study surrounding Frankfurt and the high number of available weather stations there, we managed to obtain surface temperature measurements of relatively high accuracy within the city. This network of numerous, interconnected weather monitoring stations operate in a manner that optimises economic and social benefit that stands as a testament to the careful consideration given to their establishment (Amorim *et al.*, 2012). In anticipation of the inherently random nature of temperature variations, the simulated noise levels were deemed necessary in accounting for this unpredictability, and indeed we were able to generate correlation values for each location at a given level of noise. However, as we extracted our chosen data solely from the ERA5 dataset, the accuracy of our analysis may have had the opportunity to improve in significant degrees if the dataset's land-based counterpart, ERA5-Land (Gleixner *et al.*, 2020), was used in conjunction with the data obtained in this study. This would have allowed for the potential identification of any non-surface variables, such as cloud cover, for any potential influence on surface temperature. This also may have potentially mitigated any significant lack of clarity in the data resulting from the coarse resolution of the ERA5 dataset (Gleixner *et al.*, 2020).

We assess the synthetic dataset's precision, dependability, and general robustness to improve the data quality assessment. A range of factors were considered to evaluate the degree to which the data quality can be enhanced and maintained, including station movements, errors, and noise. The analysis in Figure 3 helps us pinpoint which locations were most sensitive or responsive to the introduction of noise, providing insights into how noise affects the correlation between temperature data and location coordinates for different weather stations. Noise, in this sense, refers to the unpredictable variations in the temperature data. For example, if location 1 has a correlation of 0.9633642, it means there is a strong positive variable (temperature) being measured at location 1 with low noise or random variability in the data.

4.3 Meteorological data loss

Weather radars often suffer from data loss issues, which limits their data quality and applications. The traditional weather radar missing data completion method based on radar physics and statistics has defects in various aspects (Gong, *et al.*, 2023).Modern weather radars are powerful tools in today's real-time weather monitoring. Thanks to their high spatial resolution and short scanning interval, radars can usually obtain more comprehensive and finer-grained observations in regions than rain gauges and satellites. Despite the advantages of radars, they suffer from the data-missing problem that limits their data quality. A significant cause of radar missing data is beam blockage, which occurs when radar beams are obstructed by terrain objects like mountains and buildings, resulting in wedge-shaped blind zones behind the objects. Some data is missing. This may also cause abnormal temperature data (Gong, *et al.*, 2023). [?]Besides beam blockage, other equally significant factors include the phenomenon of attenuation, whereby radar signals are weakened as they pass through intense rainfall, which leads to underestimations of rainfall and linked temperature data (Fabry, 1996). [?][?]These restrictions in radar technology can cause gaps in meteorological data, which could lead to inaccurate temperature results. [?]

4.4 Local factors

Due to the largely spherical shape of the Earth, it stands to reason that it receives unequal amounts of heat energy from the Sun across such a large spatial scale. However, the global-scale temperature regime is made even more nonlinear and inconsistent across several regions due to the influence of local meteorological and climatological controls over smaller-scale areas.

Local variations in topography are well known to exert significant control over, and bring about distortions in, small-scale temperature regimes over given locations (Zhu *et al.*, 2021), which presents an obstacle in calculating the true values for surface air temperature. This observational gap in data may be evidently shown by separate stations as far apart as 3km given sufficient altitudinal differences (Zhu *et al.*, 2021). The potential for trees to influence air flow and precipitation patterns brings to attention the land-use cover and change (LUCC) regime of the specified area. Research conducted by Li *et al.* (2023) demonstrates the cooling effect of reforestation efforts, with areas of grassland-to-forest conversion displaying lower daily maximum surface temperatures in summer and autumn over reforested areas of southern China. The degree of continentality (distance from the sea or ocean) of a given area must also be considered. Locations at a closer proximity to the coast are shown to experience variations in temperature in lesser magnitude than locations found in inland environments, due to the faster rate of temperature change observed in continental rock in comparison to the ocean, resulting in general decrease in land-surface temperature in areas closer to the ocean (Ning *et al.*, 2018). This factor can result in temperature regimes that are inconsistent with the latitudinal location of a given region: for example, the cities of Glasgow and Moscow are located at similar latitudes, but the location of the former city closer to the coast results in milder, warmer winters than that of the latter (BBC, n.d.).

4.5 Future Consideration

ERA5 reanalysis studies are often hindered by a similar set of obstacles, such as complex terrain and a lack of in situ observations (Gleixner *et al.*, 2020). And in the case of the ERA5 model itself, its resolution value of 0.25 degrees is considered too coarse for small-scale regional modelling and impact models (Gleixner *et al.*, 2020) (though its land-only counterpart, ERA5-Land, is often used instead to counteract this limitation (Gleixner *et al.*, 2020)). Nevertheless, ERA5 is widely agreed to be a vast improvement upon its predecessor, the ERA-interim dataset, on the grounds of precipitation measurements, as well as those of temperature (Gleixner *et al.*, 2020). This will ultimately prove essential when observational values are needed in conjunction with multiple climate variables in order to, for example, model the natural variability of coupled systems (Trenberth *et al.*, 2008). Whether or not any improvements in ERA5 will prove significant will depend on the outcome of future studies, which often test such newfound capabilities in regions whose climate is difficult to analyse, e.g. East Africa (Gleixner *et al.*, 2020), which features complex terrain and frequently heavy cloud cover (Holmes *et al.*, 2016) in addition to a sparsity of in situ measurements (Gleixner *et al.*, 2020). A wealth of advantages obtained in any reanalysis study therefore allows for additional statistical experimentation to be performed, as is the case with our study, in which sufficient data was made available for the assimilation of random variation of surface temperature in our calculations. Due to this, we can state with more confidence that shifts in station location remain one of the most likely sources of error or bias in the data. Though another method to consider is one suggested by Almeida and Coelho (2023), involving the simulation of different climatic conditions in a study area to eliminate further uncertainties. In the case of this study, it may have proved useful in identifying further potential sources of skew in location correlation data.

5. Conclusion

The study yielded valuable insights into the strengths and limitations of the ERA5 temperature dataset, especially in data quality assessment. The findings are significant in advancing the methodologies used to evaluate reanalysis products and underscore the need to consider the dataset's limitations when interpreting climate research outcomes.

ERA5 reanalysis data is highly reliable and provides detailed information on global atmospheric conditions at high spatial (up to 0.25 degrees) and temporal (hourly) resolutions. It is possible to conduct comprehensive climate studies by considering various atmospheric variables, such as wind, humidity, precipitation, and temperature. ERA5 employs advanced data assimilation techniques, combining observational data with model outputs to represent atmospheric conditions more accurately.

Biases, also known as systematic errors, are commonly found in data-assimilation systems. All system components, including the forecast model, boundary conditions, observations, observation operators, and covariance models, can introduce, extrapolate, or amplify biases. To detect biases, differences between observations and their model-predicted equivalents can be monitored on the input side. At the same time, systematic features of the analysis increments can be examined on the output side. Identifying different sources of bias requires additional information, such as independent observations, knowledge of underlying causes, or hypotheses about the error characteristics of possible sources.

Most data assimilation systems do not correct biases during the analysis step, although developing bias-aware assimilation methods is conceptually straightforward. The main challenge is correctly attributing detected biases to their sources and developing applicable models for them. Assimilation may correct the wrong source when multiple sources produce similar biases. This risk increases when more degrees of freedom are added to the system. For example, in a weak-constraint variational analysis, parameters for radiance bias correction support the model-error correction. It is still being determined whether constraints on the correction terms can be designed to ensure that model and observation biases can always be correctly and simultaneously identified in the analysis.

A bias-aware analysis scheme designed to correct bias in either the background or the observations will reduce mean analysis increments by construction, but not necessarily for the correct reason. It is necessary to test whether the analysis has improved by verifying that the bias attribution is accurate. Figure 7 illustrates how a successful bias correction of the background during assimilation should lead to better analysis and reduced forecast errors. However, reducing the bias in the initial conditions may only improve the forecast in practice if the model itself is changed.

Model bias correction is particularly challenging because it is difficult to develop valuable representations for the biases or the mechanisms that cause them. Intermittent bias correction of background estimates in a sequential estimation scheme does not prevent the generation of bias during the integration of the model. Incremental bias correction schemes, which use bias estimates to correct model tendencies, may be more effective in guiding the model to an unbiased forecast, provided the corrections are physically meaningful.

6. Bibliography

Almeida, M. and Coelho, P. (2023). A first assessment of ERA5 and ERA5-Land reanalysis air temperature in Portugal. International Journal of Climatology, 43(14), pp.6643-6663.

Amorim, A.M., Gonçalves, A.B., Nunes, L.M. and Sousa, A.J. (2012). Optimizing the location of weather monitoring stations using estimation uncertainty. International Journal of Climatology, 32(6), pp.941-952.

Anon, (2019). How Atmospheric Sounding Transformed Weather Prediction - NASA. [online] Available at: https://www.nasa.gov/history/how-atmospheric-sounding-transformed-weather-prediction-jdzlx/ [Accessed 5 Feb. 2024].

Auffhammer, M., Hsiang, S.M., Schlenker, W. and Sobel, A., 2013. Using weather data and climate model output in economic analyses of climate change. Review of Environmental Economics and Policy.

Baker, Z. (2017). Climate state: Science-state struggles and the formation of climate science in the US from the 1930s to 1960s. Social Studies of Science, 47(6), 861–887.

Baker, N. and Phillips, K. (2022). 'Mares' tails and mackerel scales': The many bizarre ways humans have tried to predict the weather. (2022). ABC News. [online] 6 Jul. Available at: https://www.abc.net.au/news/2022-07-07/the-history-of-weather-forecasting/101181492.

Barry, R.G. and Chorley, R.J., 2009. Atmosphere, weather and climate. Routledge.

BBC (n.d.). Factors Affecting Climate - Factors Affecting Climate - CCEA - GCSE Geography Revision - CCEA. [online] BBC Bitesize. Available at: https://www.bbc.co.uk/bitesize/guides/zgyw4qt/revision/2.

Bengtsson, L. and Shukla, J., 1988. Integration of space and in situ observations to study global climate change. Bulletin of the American Meteorological Society, 69(10), pp.1130-1143.

Bengtsson, L., Arkin, P., Berrisford, P., Bougeault, P., Folland, C.K., Gordon, C., Haines, K., Hodges, K.I., Jones, P., Kallberg, P. and Rayner, N., 2007. The need for a dynamical climate reanalysis. Bulletin of the American Meteorological Society, 88(4), pp.495-501.

Bowman, D.M.J.S., Balch, J.K., Artaxo, P., et al. (2009). Fire in the Earth system. Science, 324(5926), 481-484

Bouktif, S., Fiaz, A., Ouni, A. and Serhani, M.A., 2018. Optimal deep learning lstm model for electric load forecasting using feature selection and genetic algorithm: Comparison with machine learning approaches. Energies, 11(7), p.1636.

Bracegirdle, T.J. (2013). Climatology and recent increase of westerly winds over the Amundsen Sea derived from six reanalyses. International Journal of Climatology, 33, pp. 843-851

Che-Castaldo C, Crisafulli CM. Long-term monitoring of Mount St. Helens micrometeorology. Ecology. 2023 Mar;104(3):e3950. doi: 10.1002/ecy.3950. Epub 2023 Jan 6. PMID: 36484720.

Chen, G., Shi, Y., Wang, R., Ren, C., Ng, E., Fang, X. and Ren, Z., 2022. Integrating weather observations and local-climate-zone-based landscape patterns for regional hourly air temperature mapping using machine learning. Science of The Total Environment, 841, p.156737.

Cucchi, M., Weedon, G.P., Amici, A., Bellouin, N., Lange, S., Müller Schmied, H., Hersbach, H. and Buontempo, C., 2020. WFDE5: bias-adjusted ERA5 reanalysis data for impact studies. Earth System Science Data, 12(3), pp.2097-2120.

Dee, D.P., Balmaseda, M., Balsamo, G., Engelen, R., Simmons, A.J. and Thépaut, J.N., 2014. Toward a consistent reanalysis of the climate system. Bulletin of the American Meteorological Society, 95(8), pp.1235-1248.

Doddy Clarke, E.; Griffin, S.; McDermott, F.; Monteiro Correia, J.; Sweeney, C. Which Reanalysis Dataset Should We Use for Renewable Energy Analysis in Ireland? Atmosphere 2021, 12, 624.

Edwards P.N (2010). A VAST MACHINE: Computer models, climate data, and the politics of global warming. The MIT Press Cambridge, Massachusetts London, England.

Essa, Y.H., Cagnazzo, C., Madonna, F., Cristofanelli, P., Yang, C., Serva, F., Caporaso, L. and Santoleri, R., 2022. Intercomparison of Atmospheric Upper-Air Temperature From Recent Global Reanalysis Datasets. Frontiers in Earth Science, 10, p.935139.

Fabry, F. (1996). Radar Meteorology: Principles and Practice. Cambridge University Press.

Ghajarnia, N., Akbari, M., Saemian, P., Ehsani, M.R., Hosseini-Moghari, S.M., Azizian, A., Kalantari, Z., Behrangi, A., Tourian, M.J., Klove, B. and Haghighi, A.T., 2022. Evaluating the evolution of ECMWF precipitation products using observational data for Iran: From ERA40 to ERA5. Earth and Space Science, 9(10), p.e2022EA002352.

Gleixner, S., Demissie, T. and Diro, G.T., 2020. Did ERA5 improve temperature and precipitation reanalysis over East Africa?. Atmosphere, 11(9), p.996.

Gong, A.; Chen, H.; Ni, G. Improving the Completion of Weather Radar Missing Data with Deep Learning. Remote Sens. 2023, 15, 4568.

Hersbach, H., Bell, B., Berrisford, P., Hirahara, S., Horanyi, A., Munoz-Sabater, J., Nicolas, J., Peubey, C., Radu, R., Schepers, D. and Simmons, A., 2020. The ERA5 global reanalysis. Quarterly Journal of the Royal Meteorological Society, 146(730), pp.1999-2049.

Hoffmann, L., Gunther, G., Li, D., Stein, O., Wu, X., Griessbach, S., Heng, Y., Konopka, P., Muller, R., Vogel, B. and Wright, J.S., 2019. From ERA-Interim to ERA5: the considerable impact of ECMWF's next-generation reanalysis on

Lagrangian transport simulations. Atmospheric Chemistry and Physics, 19(5), pp.3097-3124.

Holmes, T.R., Hain, C.R., Anderson, M.C. and Crow, W.T. (2016). Cloud tolerance of remote-sensing technologies to measure land surface temperature. Hydrology and Earth System Sciences, 20(8), pp.3263-3275.

Houghton, J., 2005. Global warming. Reports on progress in physics, 68(6), p.1343.

Hu, Q., Li, Z., Wang, L., Huang, Y., Wang, Y. and Li, L., 2019. Rainfall spatial estimations: A review from spatial interpolation to multi-source data merging. Water, 11(3), p.579

Huang, X., Han, S. and Shi, C., 2021. Multiscale assessments of three reanalysis temperature data systems over China. Agriculture, 11(12), p.1292.

Jiao, D., Xu, N., Yang, F. and Xu, K., 2021. Evaluation of spatial-temporal variation performance of ERA5 precipitation data in China. Scientific Reports, 11(1), p.17956.

Jones, P.G. and Thornton, P.K., 2013. Generating downscaled weather data from a suite of climate models for agricultural modelling applications. Agricultural Systems, 114, pp.1-5.

Kalnay, E., 2003. Atmospheric modeling, data assimilation, and predictability. Cambridge University Press.

Karl, T.R., Tarpley, J.D., Quayle, R.G., Diaz, H.F., Robinson, D.A. and Bradley, R.S., 1989. The recent climate record: What it can and cannot tell us. Reviews of Geophysics, 27(3), pp.405-430.

Kirsanov, A.; Rozinkina, I.; Rivin, G.; Zakharchenko, D.; Olchev, A. Effect of Natural Forest Fires on Regional Weather Conditions in Siberia. Atmosphere 2020, 11, 1133.

Li, Z., Gong, X., Liu, M., Tang, H., Yao, Y., Liu, M. and Yang, F., 2022. A Temperature Refinement Method Using the ERA5 Reanalysis Data. Atmosphere, 13(10), p.1622.

Li, X., Chen, H., Hua, W., Ma, H., Li, X., Sun, S., Lu, Y., Pang, X., Zhang, X. and Zhang, Q. (2023). Modeling the effects of realistic land cover changes on land surface temperatures over China. Climate Dynamics, 61(3), pp.1451-1474.

Maraun, D. and Widmann, M., 2018. Statistical downscaling and bias correction for climate research. Cambridge University Press.

Mauelshagen, F. (2014). Redefining historical climatology in the Anthropocene. The Anthropocene Review, 1(2), 171-204.

McNicholl, B., Lee, Y.H., Campbell, A.G. and Dev, S., 2021. Evaluating the reliability of air temperature from ERA5 reanalysis data. IEEE Geoscience and Remote Sensing Letters, 19, pp.1-5.

Munoz-Sabater, J., Dutra, E., Agusti-Panareda, A., Albergel, C., Arduini, G., Balsamo, G., Boussetta, S., Choulga, M., Harrigan, S., Hersbach, H. and Martens, B. (2021). ERA5-Land: A state-of-the-art global reanalysis dataset for land applications. Earth system science data, 13(9), pp.4349-4383.

Ning, J., Gao, Z., Meng, R., Xu, F. and Gao, M. (2018). Analysis of relationships between land surface temperature and land use changes in the Yellow River Delta. Frontiers of earth science, 12, pp.444-456.

Oke, T.R. (1982). The energetic basis of the urban heat island. Quarterly Journal of the Royal Meteorological Society, 108(455), 1-24.

Parker, W.S., 2016. Reanalyses and observations: What's the difference?. Bulletin of the American Meteorological Society, 97(9), pp.1565-1572.

Pitman, A.J., 2003. The evolution of, and revolution in, land surface schemes designed for climate models. International Journal of Climatology: A Journal of the Royal Meteorological Society, 23(5), pp.479-510.

Ramirez-Villegas, J. and Challinor, A., 2012. Assessing relevant climate data for agricultural applications. Agricultural and forest meteorology, 161, pp.26-45

Robock, A., 2000. Volcanic eruptions and climate. Reviews of geophysics, 38(2), pp.191-219.

Rummukainen, M., 2012. Changes in climate and weather extremes in the 21st century. Wiley Interdisciplinary Reviews: Climate Change, 3(2), pp.115-129.

Schauberger, B., Jagermeyr, J. and Gornott, C., 2020. A systematic review of local to regional yield forecasting approaches and frequently used data resources. European Journal of Agronomy, 120, p.126153.

Salcedo-Sanz, S., Ghamisi, P., Piles, M., Werner, M., Cuadra, L., Moreno-Martinez, A., Izquierdo-Verdiguier, E., Munoz-Mari, J., Mosavi, A. and Camps-Valls, G., 2020. Machine learning information fusion in Earth observation: A comprehensive review of methods, applications and data sources. Information Fusion, 63, pp.256-272.

Shi, H., Cao, X., Li, Q., Li, D., Sun, J., You, Z. and Sun, Q., 2021. Evaluating the accuracy of ERA5 wave reanalysis in the water around China. Journal of Ocean University of China, 20, pp.1-9.

Tetzner, D., Thomas, E. and Allen, C. (2019). A validation of ERA5 reanalysis data in the Southern Antarctic Peninsula—Ellsworth land region, and its implications for ice core studies. Geosciences, 9(7), p.289.

Trenberth, K.E. and Olson, J.G., 1988. An evaluation and intercomparison of global analyses from the National Meteorological Center and the European Centre for Medium Range Weather Forecasts. Bulletin of the American Meteorological Society, 69(9), pp.1047-1057.

Trenberth, K.E., Koike, T. and Onogi, K. (2008). Progress and prospects for reanalysis for weather and climate. Eos, Transactions American Geophysical Union, 89(26), pp.234-235.

Velikou, K., Lazoglou, G., Tolika, K. and Anagnostopoulou, C., 2022. Reliability of the ERA5 in replicating mean and extreme temperatures across Europe. Water, 14(4), p.543.

Wolfe, J., Mar 5,2000. www.earthdata.nasa.gov. [online]

[date of visit: 24 01 2024].

Yilmaz, M., 2023. Accuracy assessment of temperature trends from ERA5 and ERA5-Land. Science of The Total Environment, 856, p.159182.

Yu, Y., Xiao, W., Zhang, Z., Cheng, X., Hui, F. and Zhao, J., 2021. Evaluation of 2-m air temperature and surface temperature from ERA5 and ERA-I using buoy observations in the Arctic during 2010–2020. Remote Sensing, 13(14), p.2813.

Zhu, J., Xie, A., Qin, X., Wang, Y., Xu, B. and Wang, Y. (2021). An assessment of ERA5 reanalysis for Antarctic near-surface air temperature. Atmosphere, 12(2), p.217.

Code Used:

```
setwd("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_ERA
5_Analysis")
#install.packages("ncdf4")
library(ncdf4)
ERA5_F_2013_14_cdf<- nc_open("ERA5_F_2013_14.nc")
print(ERA5_F_2013_14_cdf)
names(ERA5_F_2013_14_cdf$var)
ncvar_get(ERA5_F_2013_14_cdf, varid="t2m")
attributes(ERA5_F_2013_14_cdf)
attributes(ERA5_F_2013_14_cdf$var)
attributes(ERA5_F_2013_14_cdf$dim)
lat<- ncvar_get(ERA5_F_2013_14_cdf, "latitude")
dim(lat)
lon<- ncvar_get(ERA5_F_2013_14_cdf, "longitude")
dim(lon)
print(c(dim(lon), dim(lat)))
tim<- ncvar_get(ERA5_F_2013_14_cdf, "time")
head(tim)
dim(tim)
t2m_array<- ncvar_get(ERA5_F_2013_14_cdf,"t2m")
fillvalue<- ncatt_get(ERA5_F_2013_14_cdf,"t2m","_FillValue")
dim(t2m_array)
t2m_array[t2m_array==fillvalue$value]<- NA
t2m_array
#install.packages("anytime")
library(anytime)
mins<-tim*60
secs<-mins*60
time_units2<- as.POSIXct(secs, origin = "1900-01-01 00:00:00.0", tz ="GMT")
dim(time_units2)
range(time_units2)
lonlattime <- as.matrix(expand.grid(lon,lat,time_units2))
head(lonlattime)
t2m_vector<- as.vector(t2m_array)
length(t2m_vector)
head(t2m_vector)
t2m_df<- data.frame(cbind(lonlattime, t2m_vector))
colnames(t2m_df)<-c("longitude", "latitude", "time", "tempK")
head(t2m_df)
write.csv(t2m_df, "ERA5_F_2013_14.csv", row.names=T)
######################################################################
############
ERA5_F_2013_14<-read.csv("ERA5_F_2013_14.csv")
table(ERA5_F_2013_14[,2])
long_coords<-c(8, 8.25, 8.5, 8.75, 9)
table(ERA5_F_2013_14[,3]) #switched log/lat by accident (no effect on output)
lat_coords<-c(49, 49.25, 49.5, 49.75, 50)
E<-ERA5_F_2013_14
```

```
station_coords = NULL
for (i in long_coords) { for (j in lat_coords) {
  station = E[(E$longitude==i) & (E$latitude==j), 1:3]
  station_row = c(station[,1])
  station_lat = c(station[,2])
  station_lon = c(station[,3])
  station_coords0 = data.frame(station_row, station_lat, station_lon)
  station_coords = rbind(station_coords, station_coords0)}}
colnames(station_coords)<-c("row","latitude","longitude")
head(station_coords)
station_coords<-data.frame(station_coords)
S<- station_coords
row_num = c()
for (i in long_coords) { for(j in lat_coords){
  row_sample = S[(S[,2]==i) & (S[,3]==j), 1]
  row_sample0 = data.frame(rep("row",6),row_sample)
  row_num = cbind(row_num, row_sample0[,2])}}
head(row_num)
location_01<-E[(E$longitude==8) & (E$latitude==50), ]
location_02<-E[(E$longitude==8.25) & (E$latitude==50), ]
location_03<-E[(E$longitude==8.5) & (E$latitude==50), ]
location_04<-E[(E$longitude==8.75) & (E$latitude==50), ]
location_05<-E[(E$longitude==9) & (E$latitude==50), ]
location_06<-E[(E$longitude==8) & (E$latitude==49.75), ]
location_07<-E[(E$longitude==8.25) & (E$latitude==49.75), ]
location_08<-E[(E$longitude==8.5) & (E$latitude==49.75), ]
location_09<-E[(E$longitude==8.75) & (E$latitude==49.75), ]
location_10<-E[(E$longitude==9) & (E$latitude==49.75), ]
location_11<-E[(E$longitude==8) & (E$latitude==49.5), ]
location_12<-E[(E$longitude==8.25) & (E$latitude==49.5), ]
location_13<-E[(E$longitude==8.5) & (E$latitude==49.5), ]
location_14<-E[(E$longitude==8.75) & (E$latitude==49.5), ]
location_15<-E[(E$longitude==9) & (E$latitude==49.5), ]
location_16<-E[(E$longitude==8) & (E$latitude==49.25), ]
location_17<-E[(E$longitude==8.25) & (E$latitude==49.25), ]
location_18<-E[(E$longitude==8.5) & (E$latitude==49.25), ]
location_19<-E[(E$longitude==8.75) & (E$latitude==49.25), ]
location_20<-E[(E$longitude==9) & (E$latitude==49.25), ]
location_21<-E[(E$longitude==8) & (E$latitude==49), ]
location_22<-E[(E$longitude==8.25) & (E$latitude==49), ]
location_23<-E[(E$longitude==8.5) & (E$latitude==49), ]
location_24<-E[(E$longitude==8.75) & (E$latitude==49), ]
location_25<-E[(E$longitude==9) & (E$latitude==49), ]
station_sums = c()
for (i in long_coords) { for (j in lat_coords) {
  station_sum0 = E[(E$longitude==i) & (E$latitude==j), 5]
  station_sum = c(i,j,summary(station_sum0))
  station_sums = rbind(station_sums, station_sum)}}
station_sums<- data.frame(station_sums)
onetwentyfive<-seq(from = 1, to = 25, length.out =25)
```

```
station_summaries<-cbind(onetwentyfive, station_sums)
colnames(station_summaries)<- c("station", "longitude", "latitude", "Min", "Q1",
"Med", "Mean", "Q3", "Max")
rownames(station_summaries)<-NULL
station_summaries
location_list <-
list(location_01,location_02,location_03,location_04,location_05,location_06,
                    location_07,location_08,location_09,location_10,location_11,
location_12,
                    location_13, location_14, location_15, location_16, location_17,
location_18,
                    location_19, location_20, location_21, location_22, location_23,
location_24, location_25)
location_shapiro<- c()
for (i in location_list) {result = shapiro.test(i[,5])
result = c(i[1,1],result)
location_shapiro = rbind(location_shapiro, result)}
location_shapiroW<-location_shapiro[,1:3]
row.names(location_shapiroW)<- NULL
colnames(location_shapiroW)<- c("location", "SW-statistic", "p-value")
location_shapiroW
location_IQR<- c()
for (i in location_list) {result = IQR(i[,5])
result = c(i[1,1],result)
location_IQR = rbind(location_IQR, result)}
colnames(location_IQR)<-c("location","IQR")
row.names(location_IQR)<-NULL
location_IQR
location_stats<-cbind(location_shapiroW, location_IQR[,2])
colnames(location_stats)<- c("location", "SW-statistic", "p-value", "IQR")
location_stats
loc_cor00 = NULL
loc_cor0 = NULL
loc_cor = NULL
for (i in location_list) {for(j in location_list){
  loc_cor00 = c(cor(i[,5],j[,5]),i[1,1],j[1,1])
  loc_cor0 = rbind(loc_cor0,loc_cor00)}}
loc_cor000<-loc_cor0[order(loc_cor0[,3]),]
loc_cor<- matrix(data = loc_cor000[,1], nrow = 25, ncol = 25, byrow = TRUE)
colnames(loc_cor)<-c("1":"25")
location_stats<-cbind(location_stats, loc_cor)
head(location_stats)
head(E)
#install.packages("sf")
library(sf)
#install.packages("rjson")
library(rjson)
europe<-st_read("europe.geo.json")
EJSON<- st_as_sf(x = E, coords = c("longitude", "latitude"), crs = st_crs(europe))
#install.packages("tmap")
```

```
library(tmap)
tmap_mode('view')
base_map <-leaflet::providers$CartoDB.Positron
tm_basemap(base_map)+ tm_shape(EJSON) + tm_bubbles(col = "pink4", size =
0.01) +
  tm_shape(europe[europe$sov_a3=="DEU",]) + tm_borders()
kelvin<- c(E[,5])
kelvin -273.15 ->celcius
celcius
E.C<- cbind(E, celcius)
location_07.stats<- location_stats[7,]
location_07.stats
summary(location_07)
mock_07d<-rnonnorm(1440, mean = 288.5 , sd = 3.913184, skew = 0.55883625, kurt
= 0.72209601)
mock_07<-data.frame(mock_07d)
plot(location_07[,5])
plot(mock_07[,1])
cor(location_07[,5],mock_07[,1])
mock_n_07<-rnorm(location_07[,5])
plot(mock_n_07)
cor(location_07[,5], mock_n_07)
mock_iqr_07d<-rnonnorm(1440, mean = 288.5 , sd = 4.771247, skew = 0.55883625,
kurt = 0.72209601)
mock_iqr_07d<-data.frame(mock_iqr_07d)
plot(mock_iqr_07d[,1])
cor(location_07[,5], mock_iqr_07d[,1])
t2m_07<-location_07[,5]
mock_07dd<-rnonnorm(1440, mean = 288.5 , sd = 3.913184, skew = 0.55883625,
kurt = 0.72209601)$t2m_07
#install.packages("mnonr")
library(mnonr)
location_07m<-data.matrix(location_07[,5])
mardia_07<-mardia(location_07m, na.rm = TRUE)
mardia_07
#install.packages("fitdistrplus")
library(fitdistrplus)
t2m_07<-location_07[,5]
descdist(t2m_07)
descdist(t2m_07, boot= 1440)
#install.packages("MASS")
library(MASS)
fitdistr(t2m_07,"weibull")
fitdistr(t2m_07,"gamma")
fitdistr(t2m_07,"lognormal")
fitdistr(t2m_07,"normal")
hist(t2m_07)
set.seed(00)
sample1_07<-rnorm(1440, 288.5, 3.9)
head(sample1_07)
```

```
summary(sample1_07)
sd(sample1_07)
cor(t2m_07,sample1_07)
plot(sample1_07)
plot(t2m_07)
set.seed(01)
sample2_07<-rnorm(1440, 288.5, 3.5)
summary(sample2_07)
sd(sample2_07)
cor(t2m_07,sample2_07)
plot(sample2_07)
barplot(sample2_07)
barplot(t2m_07)
dens_07a = density(t2m_07, adjust=0.8)
set.seed(03)
sample2a_07 = sample(dens_07a$x, 1440, replace=TRUE, prob=dens_07a$y)
summary(sample2a_07)
sd(sample2a_07)
cor(sample2a_07, t2m_07)
par()
plot(sample2a_07)
plot_sept_loc=function(x){
  par(mfrow=c(5,5), mar = c(1, 1, 1, 1))
  plot_draw=lapply(x, function(x)  plot(x[,5], cex=0.2))}
plot_sept_loc(location_list)
#install.packages("devtools")
library(devtools)
#install_github("ProcessMiner/nlcor")
library(nlcor)
cor(t2m_07, sample1_07)
cor(t2m_07, sample2_07)
cor(t2m_07, sample2a_07)
nlcor(t2m_07, sample1_07, plt = T)
nlcor(t2m_07, sample2_07, plt = T)
nlcor(t2m_07, sample2a_07, plt = T)
set.seed(NULL)
dens_07.1 = density(t2m_07, adjust=1)
set.seed(1)
sample_07.1 = sample(dens_07.1$x, 1440, replace=TRUE, prob=dens_07.1$y)
nlcor(t2m_07, sample_07.1, plt = T)
summary(t2m_07)
individual_07t<- seq(278.3,302.7, by=0.1)
rounded_07.1<- round(t2m_07, digits = 1)
rounded_07.2<- round(t2m_07, digits = 2)
#install.packages("plyr")
library(plyr)
count(rounded_07.1)
freq_mat_07<- as.matrix(count(rounded_07.2))
nlcor(t2m_07, sample_07.1, refine = 0.95, plt = T)
set.seed(NULL)
```

```
set.seed(01)
noise_07.1<-jitter(t2m_07, factor=0.01)
nlcor(t2m_07, noise_07.1, refine = 0.95, plt = T)
noise_factor<-seq(0.01, 1, by= 0.01)
noise_07 = NULL
for (i in noise_factor) {
  a = jitter(t2m_07, factor=i)
  b = nlcor(t2m_07, a, refine = 0.95, plt = T)
  c = b$cor.estimate
  d = b$adjusted.p.value
  e = c(i,c,d)
  noise_07 = rbind(noise_07, e)}
tail(noise_07)
noise_factor2<-seq(3, 6, by= 0.01)
noise_07.2 = NULL
for (i in noise_factor2) {
  a = jitter(t2m_07, factor=i)
  b = nlcor(t2m_07, a, refine = 0.95, plt = T)
  c = b$cor.estimate
  d = b$adjusted.p.value
  e = c(i,c,d)
  noise_07.2 = rbind(noise_07.2, e)}
tail(noise_07.2)
noise_factor2<-seq(3, 6, by= 0.01)
noise_07.2 = NULL
for (i in noise_factor2) {
  a = jitter(t2m_07, amount = i)
  b = nlcor(t2m_07, a, refine = 0.95, plt = T)
  c = b$cor.estimate
  d = b$adjusted.p.value
  e = c(i,c,d)
  noise_07.2 = rbind(noise_07.2, e)}
tail(noise_07.2)
head(noise_07.2)
noise_factor<-seq(0.01, 3, by= 0.01)
noise_07 = NULL
set.seed(300)
for (i in noise_factor) {
  a = jitter(t2m_07, amount = i)
  b = nlcor(t2m_07, a, refine = 0.95, plt = T)
  c = b$cor.estimate
  d = b$adjusted.p.value
  e = c(i,c,d)
  noise_07 = rbind(noise_07, e)}
noise_07<- data.frame(noise_07)
colnames(noise_07)<- c("noise factor", "correlation", "p-value") #analyse then in tab?
rownames(noise_07)<-NULL
set.seed(NULL)
set.seed(42)
noise_07.42<-jitter(t2m_07, amount =0.42)
```

```
noise_07.42_cor<-nlcor(t2m_07, noise_07.42, refine = 0.95, plt = T)
print(noise_07.42_cor$cor.estimate, digits = 9)
print(nlcor(location_08[,5], noise_07.42, refine = 0.95, plt = T), digits = 9)
par()
plot(noise_07.42)
plot(t2m_07)
noise_factor_small<-seq(0.01, 1.5, by= 0.01)
loc_noise_cor = NULL
for (j in location_list) { for(i in noise_factor_small){
  a = jitter(t2m_07, amount = i)
  b = nlcor(j[,5], a, refine = 0.95, plt = T)
  c = b$cor.estimate
  d = b$adjusted.p.value
  e = c(i,j[1,1],c,d)
  loc_noise_cor = rbind(loc_noise_cor, e)}}
head(loc_noise_cor) #19mins
loc_noise_cor<- loc_noise_cor[,1:3]
rownames(loc_noise_cor)<- NULL
noise_cor_list<-loc_noise_cor[,3]
loc_noise_cor0<- matrix(data = noise_cor_list,ncol = 25, byrow = F)
colnames(loc_noise_cor0)<-c("1":"25")
rownames(loc_noise_cor0)<-seq(0.01, 1.50, by = 0.01)
noise_cor_w_loc<- data.frame(loc_noise_cor0)
head(noise_cor_w_loc)
write.csv(noise_cor_w_loc, "noise_cor_w_loc.csv", row.names=T)
max_cor_w_07 = NULL
for (i in 1:150) {
  m = max(noise_cor_w_loc[i,])
  col_plus_max = c(i, m)
  max_cor_w_07 = rbind(max_cor_w_07, col_plus_max)}
colnames(max_cor_w_07)<-c("factor","max cor per f.")
rownames(max_cor_w_07)<-seq(0.01, 1.50, by = 0.01)
max_cor_w_07
noise_factor_med<-seq(1.51, 4, by= 0.01) #15.42 to 16.20
loc_noise_cor2 = NULL
for (j in location_list) { for(i in noise_factor_med){
  a = jitter(t2m_07, amount = i)
  b = nlcor(j[,5], a, refine = 0.95, plt = T)
  c = b$cor.estimate
  d = b$adjusted.p.value
  e = c(i,j[1,1],c,d)
  loc_noise_cor2 = rbind(loc_noise_cor2, e)}}
head(loc_noise_cor2)
loc_noise_cor2<- loc_noise_cor2[,1:3]
rownames(loc_noise_cor2)<- NULL
noise_cor_list2<-loc_noise_cor2[,3]
loc_noise_cor02<- matrix(data = noise_cor_list2,ncol = 25, byrow = F)
colnames(loc_noise_cor02)<-c("1":"25")
rownames(loc_noise_cor02)<-seq(1.51, 4, by = 0.01)
noise_cor_w_loc2<- data.frame(loc_noise_cor02)
```

```
head(noise_cor_w_loc2)
write.csv(noise_cor_w_loc2, "noise_cor_w_loc2.csv", row.names=T)
max_cor2_w_07 = NULL
for (i in 1:250) {
  m = max(noise_cor_w_loc2[i,])
  col_plus_max = c(i, m)
  max_cor2_w_07 = rbind(max_cor2_w_07, col_plus_max)}
colnames(max_cor2_w_07)<-c("factor","max cor per f.")
rownames(max_cor2_w_07)<-seq(1.51, 4, by = 0.01)
max_cor2_w_07
n2<-noise_cor_w_loc2
cor_point = NULL
for (i in noise_cor_list2){
  indice = which(n2==i,arr.ind=TRUE)
  cor_point = rbind(cor_point, indice)}
head(cor_point )
noise_cor_w_loc3<- rbind(noise_cor_w_loc,noise_cor_w_loc2) #now shows corrs for
stats 0.01 to 4.00
max_cor3_w_07<- rbind(max_cor_w_07, max_cor2_w_07)
head(max_cor3_w_07)
tail(max_cor3_w_07)
n3<-noise_cor_w_loc3
cor_point = NULL
for (i in max_cor3_w_07){
  indice = which(n3==i,arr.ind=TRUE)
  cor_point = rbind(cor_point, indice)}
head(cor_point)
tail(cor_point)
cor_point<-data.frame(cor_point)
write.csv(noise_cor_w_loc3, "noise_cor_w_loc3.csv", row.names=T)
###########################Reproducible#################################
######################
E <- read.csv("ERA5_F_2013_14.csv") #a) tell R to read this file in (it has to be in
# your working directory first)

#b) group the temperature information per station:
location_01<-E[(E$longitude==8) & (E$latitude==50), ]
location_02<-E[(E$longitude==8.25) & (E$latitude==50), ]
location_03<-E[(E$longitude==8.5) & (E$latitude==50), ]
location_04<-E[(E$longitude==8.75) & (E$latitude==50), ]
location_05<-E[(E$longitude==9) & (E$latitude==50), ]
location_06<-E[(E$longitude==8) & (E$latitude==49.75), ]
location_07<-E[(E$longitude==8.25) & (E$latitude==49.75), ]
location_08<-E[(E$longitude==8.5) & (E$latitude==49.75), ]
location_09<-E[(E$longitude==8.75) & (E$latitude==49.75), ]
location_10<-E[(E$longitude==9) & (E$latitude==49.75), ]
location_11<-E[(E$longitude==8) & (E$latitude==49.5), ]
location_12<-E[(E$longitude==8.25) & (E$latitude==49.5), ]
location_13<-E[(E$longitude==8.5) & (E$latitude==49.5), ]
location_14<-E[(E$longitude==8.75) & (E$latitude==49.5), ]
```

```
location_15<-E[(E$longitude==9) & (E$latitude==49.5), ]
location_16<-E[(E$longitude==8) & (E$latitude==49.25), ]
location_17<-E[(E$longitude==8.25) & (E$latitude==49.25), ]
location_18<-E[(E$longitude==8.5) & (E$latitude==49.25), ]
location_19<-E[(E$longitude==8.75) & (E$latitude==49.25), ]
location_20<-E[(E$longitude==9) & (E$latitude==49.25), ]
location_21<-E[(E$longitude==8) & (E$latitude==49), ]
location_22<-E[(E$longitude==8.25) & (E$latitude==49), ]
location_23<-E[(E$longitude==8.5) & (E$latitude==49), ]
location_24<-E[(E$longitude==8.75) & (E$latitude==49), ]
location_25<-E[(E$longitude==9) & (E$latitude==49), ]

#c) list each location to be accessible under one function:
location_list <- list(location_01,location_02,location_03,location_04,location_05,
                      location_06,location_07,location_08,location_09,location_10,
                      location_11, location_12,location_13, location_14, location_15,
                      location_16, location_17, location_18,location_19, location_20,
                      location_21, location_22, location_23, location_24, location_25)

t2m_0<-location_07[,5] #d) extract only the temperature information

# (do this for your chosen location only, i.e. location 7 is used in this example,
# but if you are working on e.g. location 25, change this to: t2m_0<-location_25[,5]
etc.)
# (don't forget to change any names in the code below if they were changed in this
section)

noise_factor <- seq(0.01, 4, by= 0.01) #e) create a vector of the range of noise added

################ 2: Correlate Generated Noise to all Location Co-ords
############

#f) print all the correlations per location whene correlated with the noise added
# to our chosen location:
loc_noise_cor1 = NULL
for (j in location_list) { for(i in noise_factor){
  a1 = jitter(t2m_0, amount = i)
  b1 = nlcor(j[,5], a1, refine = 0.95, plt = T)
  c1 = b1$cor.estimate
  d1 = b1$adjusted.p.value
  e1 = c(i,j[1,1],c1,d1) #11.48:12.39
  loc_noise_cor1 = rbind(loc_noise_cor1, e1)}}
head(loc_noise_cor1) # view the first six lines to make sure you have 4 columns:
# noise factor, location, correlation, p-value (p-value will be 0)

#g) give these columns names and remove p-value (as is 0)
loc_noise_cor1<- loc_noise_cor1[,1:3]
rownames(loc_noise_cor1)<- NULL
noise_cor_list1<-loc_noise_cor1[,3]
loc_noise_cor_01<- matrix(data = noise_cor_list1, ncol = 25, byrow = F)
```

```
colnames(loc_noise_cor_01)<-c("1":"25")
rownames(loc_noise_cor_01)<-seq(0.01, 4, by = 0.01)

#h) make this data into an extractable dataframe and save as a csv to your computer:
noise_cor_w_loc1<- data.frame(loc_noise_cor_01)
write.csv(noise_cor_w_loc1, "noise_cor_w_loc_7_2.csv", row.names=T)
head(noise_cor_w_loc1)

trial07_2<- read.csv("noise_cor_w_loc_7_2.csv")
trial07_2<- data.frame(trial07_2[,2:26])
max_cor_trial07_2 = NULL
for (i in 1:400) {
  m = max(trial07_2[i,])
  col_plus_max = c(i, m)
  max_cor_trial07_2 = rbind(max_cor_trial07_2, col_plus_max)}
colnames(max_cor_trial07_2)<-c("factor","max cor per f.") # name the rows and cols
rownames(max_cor_trial07_2)<-seq(0.01, 4, by = 0.01)
head(max_cor_trial07_2)
tail(max_cor_trial07_2)
max_cor_trial07_2<- max_cor_trial07_2[,2]
cor_point_t2 = NULL
#for i in the list of max value per row:
for (i in max_cor_trial07_2){
  indice = which(trial07_2==i, arr.ind=TRUE)
  cor_point_t2 = rbind(cor_point_t2, indice)}
head(cor_point_t2)
write.csv(cor_point_t2, "cor_point_7_1.csv", row.names=T)

setwd("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_ERA
5_Analysis/trials/7")
y<-read.csv("cor_point_7_1.csv")
x<- read.csv("noise_cor_w_loc_7_1.csv")
x<- data.frame(x[,2:26])
max_cors = NULL
for (i in 1:400) {
  m = max(x[i,])
  col_plus_max = c(i, m)
  max_cors = rbind(max_cors, col_plus_max)}
colnames(max_cors)<-c("factor","corr")
rownames(max_cors)<-seq(0.01, 4, by = 0.01)
max_cors<-data.frame(max_cors)
cor_w_max<-cbind(y,max_cors$corr)
colnames(cor_w_max)<-c("noise","row","location","max_corr_per_row")
E2<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/ERA5_F_2013_14.csv")
Elong<-E2[c(1:25),2]
Elat<-E2[c(1:25),3]
long<- NULL
for (i in cor_w_max$location) {
```

```
  z <- Elong[i]
  long<- rbind(long,z)}
cor_w_max$long<-long
lat<- NULL
for (i in cor_w_max$location) {
  z <- Elat[i]
  lat<- rbind(lat,z)}
cor_w_max$lat<-lat
head(cor_w_max)
write.csv(cor_w_max, "cor_w_max_7_2.csv", row.names=T)

max_1<-read.csv("cor_w_max_7_2.csv")
freq_0= NULL
for (i in c(1:400)) {
  x = max_1[i,4]
  y = nrow(max_1[max_1$location== x,])
  z = c(x,y)
  freq_0<- rbind(freq_0, z)}
freq_0<-data.frame(freq_0)
max_1$freq<- freq_0[,2]
head(max_1)
write.csv(max_7,
"/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_ERA5_Anal
ysis/trials/7/cor_w_max_7_2.csv", row.names=T)

setwd("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_ERA
5_Analysis")
CORR<- read.csv("CORR.csv")
c_7<-CORR[CORR$Location==7,]
c_8<-CORR[CORR$Location==8,]
c_9<-CORR[CORR$Location==9,]
c_12<-CORR[CORR$Location==12,]
c_13<-CORR[CORR$Location==13,]
c_14<-CORR[CORR$Location==14,]
c_17<-CORR[CORR$Location==17,]
c_18<-CORR[CORR$Location==18,]
c_19<-CORR[CORR$Location==19,]
summary(CORR$Correlation)
sd(CORR$Correlation)
summary(CORR$Noise)
sd(CORR$Noise)
summary(c_7$Correlation) #0.9677  0.9678  0.9693  0.9694  0.9699  0.9724
sd(c_7$Correlation) #0.001927486
summary(c_8$Correlation) #0.9611  0.9676  0.9720  0.9696  0.9726  0.9745
sd(c_8$Correlation) #0.005359999
summary(c_9$Correlation) #0.9554  0.9569  0.9587  0.9595  0.9623  0.9642
sd(c_9$Correlation) #0.003667868
summary(c_12$Correlation) #0.9600  0.9604  0.9655  0.9646  0.9656  0.9715
sd(c_12$Correlation) #0.004715054
summary(c_13$Correlation) #0.9528  0.9552  0.9567  0.9590  0.9616  0.9686
```

```
sd(c_13$Correlation) #0.006251089
summary(c_14$Correlation) #0.9376  0.9442  0.9505  0.9489  0.9521  0.9603
sd(c_14$Correlation) #0.008546863
summary(c_17$Correlation) #0.9524  0.9541  0.9586  0.9594  0.9607  0.9710
sd(c_17$Correlation) #0.007308597
summary(c_18$Correlation) #0.9483  0.9530  0.9599  0.9595  0.9672  0.9691
sd(c_18$Correlation) #0.00893114
summary(c_19$Correlation) #0.9573  0.9655  0.9660  0.9655  0.9686  0.9700
sd(c_19$Correlation) #0.00492992
m_iCorr<-c(mean(c_7$Correlation), mean(c_8$Correlation), mean(c_9$Correlation),
           mean(c_12$Correlation), mean(c_13$Correlation), mean(c_14$Correlation),
           mean(c_17$Correlation), mean(c_18$Correlation), mean(c_19$Correlation))
mean(m_iCorr)
sd(m_iCorr)
sd_iCorr<-c(sd(c_7$Correlation), sd(c_8$Correlation), sd(c_9$Correlation),
sd(c_12$Correlation),
            sd(c_13$Correlation), sd(c_14$Correlation), sd(c_17$Correlation),
sd(c_18$Correlation),
            sd(c_19$Correlation))
mean(sd_iCorr)
sd(sd_iCorr)
summary(c_7$Noise) #  1.62    1.71    1.71    1.71    1.75    1.76
sd(c_7$Noise) #0.05522681
summary(c_8$Noise) #  1.530   1.590   1.620   1.678   1.710   1.940
sd(c_8$Noise) #0.1602186
summary(c_9$Noise) #  1.870   1.930   2.050   2.002   2.070   2.090
sd(c_9$Noise) #0.09654015
summary(c_12$Noise) # 1.640   1.810   1.840   1.846   1.960   1.980
sd(c_12$Noise) #0.1366748
summary(c_13$Noise) #   1.73    1.92    2.09    2.00    2.09    2.17
sd(c_13$Noise) #0.1763519
summary(c_14$Noise) #  2.020   2.180   2.210   2.278   2.410   2.570
sd(c_14$Noise) #0.2141728
summary(c_17$Noise) #  1.680   2.000   2.050   2.024   2.170   2.220
sd(c_17$Noise) #0.211731
summary(c_18$Noise) #  1.760   1.820   2.010   2.028   2.220   2.330
sd(c_18$Noise) #0.2467185
summary(c_19$Noise) #  1.760   1.790   1.850   1.892   1.930   2.130
sd(c_19$Noise) #0.148054
m_iNoise<-c(mean(c_7$Noise), mean(c_8$Noise), mean(c_9$Noise),
mean(c_12$Noise), mean(c_13$Noise),
            mean(c_14$Noise), mean(c_17$Noise), mean(c_18$Noise),
mean(c_19$Noise))
mean(m_iNoise)
sd(m_iNoise)
sd_iNoise<-c(sd(c_7$Noise), sd(c_8$Noise), sd(c_9$Noise), sd(c_12$Noise),
sd(c_13$Noise),
             sd(c_14$Noise), sd(c_17$Noise), sd(c_18$Noise), sd(c_19$Noise))
mean(sd_iNoise)
sd(sd_iNoise)
```

```
set.seed(1)
sample_Corr<- rnorm(1000, 0.96170, 0.008261583)
set.seed(2)
sample_Noise<- rnorm(1000, 1.94, 0.2331161)
t.test(sample_Corr, CORR$Correlation)
t.test(m_iCorr, CORR$Correlation)
t.test(sample_Noise, CORR$Noise)
t.test(m_iNoise, CORR$Noise)
hist(CORR$Correlation)
hist(CORR$Noise)
hist(sample_Noise)
sd(CORR$Correlation)/sqrt(length(CORR$Correlation))
sd(CORR$Noise)/sqrt(length(CORR$Noise))
qt(p=0.05/2, df=(length(CORR$Correlation)-1),lower.tail=F)
ME_Corr<-2.015368*0.001231564
ME_Noise<-2.015368*0.0347509
CI_Corr<- c(mean(CORR$Correlation)-ME_Corr, mean(CORR$Correlation) +
ME_Corr)
CI_Noise<- c(mean(CORR$Correlation)-ME_Noise, mean(CORR$Correlation) +
ME_Noise)
shapiro.test(CORR$Correlation)
shapiro.test(CORR$Noise)
hist(CORR$Noise, xlim = c(1,3), ylim = c(0,15),
     main= "Noise Degree of Alternate Location Correlation",
     xlab= "Degree of Noise")
abline(v=mean(CORR$Noise),col='red', lwd=2)
text(x=1.97, y=7,'mean', col='red', srt=90)
abline(v=mean(CORR$Noise)-(2*sd(CORR$Noise)), col='lightblue4', lwd=2)
text(x=1.44, y=7,'-2*SD', col='darkgrey', srt=90)
abline(v=mean(CORR$Noise)+(2*sd(CORR$Noise)), col='lightblue4', lwd=2)
text(x=2.44, y=7,'+2*SD', col='darkgrey', srt=90)
text(x=1.94, y=15,'<----95% Confidence Interval---->', col='black')
abline(v=1.53, col='darkblue', lwd=2)
text(x=1.27, y=4,'threshold value--->', col='black')
hist(sample_Noise, xlim = c(1,3),
     main= "Sample Noise Degree of Alternate Location Correlation",
     xlab= "Degree of Noise")
abline(v=mean(sample_Noise),col='red', lwd=2)
text(x=1.98, y=75,'mean', col='red', srt=90)
abline(v=mean(sample_Noise)-(2*sd(sample_Noise)), col='lightblue4', lwd=2)
text(x=1.45, y=75,'-2*SD', col='darkgrey', srt=90)
abline(v=mean(sample_Noise)+(2*sd(sample_Noise)), col='lightblue4', lwd=2)
text(x=2.46, y=75,'+2*SD', col='darkgrey', srt=90)
text(x=1.95, y=125,'<----95% Confidence Interval---->', col='black')
abline(v=1.53, col='darkblue', lwd=2)
text(x=1.27, y=50,'threshold value--->', col='black')
setwd("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_ERA
5_Analysis")
noise_1.53<- read.csv("1_53corrs.csv")
summary(noise_1.53$max_corr_per_row)
```

```
sd(noise_1.53$max_corr_per_row)
hist(noise_1.53$max_corr_per_row)
wo_outliers_153<- noise_1.53[c(1:85,87:125),6]
hist(wo_outliers_153, breaks = 124, ylim = c(0,6),
     main = "Correlations at 1.53 degrees of Noise", xlab = "correlation")
abline(v=mean(wo_outliers_153),col='red', lwd=2)
text(x=0.976325, y=4.5,'mean', col='red', srt=90)
abline(v=0.9744667, col='darkblue', lwd=2)
text(x=0.9751, y=4.5,'<----threshold value', col='black')
abline(v=mean(wo_outliers_153)-(2*sd(wo_outliers_153)), col='lightblue4', lwd=2)
text(x=0.97455, y=3.5,'-2*SD', col='darkgrey', srt=90)
abline(v=mean(wo_outliers_153)+(2*sd(wo_outliers_153)), col='lightblue4', lwd=2)
text(x=0.9782, y=3.5,'+2*SD', col='darkgrey', srt=90)
text(x=mean(wo_outliers_153), y=6,'<---------95% Confidence Interval--------->',
     col='black')
mean(sample_Noise)-(2*sd(sample_Noise)) #1.481252 vs##1.53
mean(sample_Noise)+(2*sd(sample_Noise)) #2.427654
mean(wo_outliers_153)-(2*sd(wo_outliers_153)) #0.9744052 vs##0.9744667
mean(wo_outliers_153)+(2*sd(wo_outliers_153)) #0.9781093
setwd("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_ERA
5_Analysis")
library(sf)
#install.packages("rjson")
library(rjson)
#install.packages("tmap")
library(tmap)
#install.packages("geojson")
library(geojson)
E<-read.csv("ERA5_F_2013_14.csv")
europe<-st_read("europe.geo.json")
germany<-europe[europe$sovereignt== "Germany",]
EJSON<- st_as_sf(x = E, coords = c("longitude", "latitude"), crs = st_crs(europe))
tmap_mode('view')
base_map <-leaflet::providers$CartoDB.Positron
tm_basemap(base_map)+ tm_shape(EJSON) + tm_bubbles(col = "pink4", size =
0.01) +
  tm_shape(europe[europe$sov_a3=="DEU",]) + tm_borders()
tmap_mode('plot')
tm_shape(germany) + tm_borders() + tm_shape(EJSON) + tm_bubbles(col = "pink4",
size = 0.01)
st_bbox(EJSON) #find bounding box coordinates: (x-min: 8, y-min: 49, x-max: 9, y-
max: 50)
bbox_new<- st_bbox(EJSON)
bbox_new[1] <- bbox_new[1] - 0.25
bbox_new[2] <- bbox_new[2] - 0.25
bbox_new[3] <- bbox_new[3] + 0.25
bbox_new[4] <- bbox_new[4] + 0.25
tmap_mode('plot')
tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "pink4", size = 0.1) +
  tm_shape(germany) + tm_borders()
```

```
long_coords<-c(8, 8.25, 8.5, 8.75, 9)
lat_coords<-c(49, 49.25, 49.5, 49.75, 50)
station_coords = NULL
for (i in long_coords) { for (j in lat_coords) {
  station = E[(E$longitude==i) & (E$latitude==j), 1:3]
  station_row = c(station[,1])
  station_lat = c(station[,2])
  station_lon = c(station[,3])
  station_coords0 = data.frame(station_row, station_lat, station_lon)
  station_coords = rbind(station_coords, station_coords0)}}
colnames(station_coords)<-c("row","latitude","longitude")
head(station_coords)
station_coords<-data.frame(station_coords)
S<- station_coords
row_num0 = c()
for (i in long_coords) { for(j in lat_coords){
  row_sample = S[(S[,2]==i) & (S[,3]==j), 1]
  row_sample0 = data.frame(rep("row",6),row_sample)
  row_num0 = cbind(row_num0, row_sample0[,2])}}
head(row_num0)
row_num<-
row_num0[,c(5,10,15,20,25,4,9,14,19,24,3,8,13,18,23,2,7,12,17,22,1,6,11,16,21)]
head(row_num)
EJSON$loc<- EJSON$X
for (i in c(1:25)) {for (j in row_num[,i]) {
  EJSON$loc[EJSON$loc == j] <- i}}
tmap_mode('view')
base_map <-leaflet::providers$CartoDB.Positron
tm_basemap(base_map)+ tm_shape(EJSON[c(1:25),], bbox = bbox_new) +
  tm_bubbles(col = "white", size = 0.75) +
  tm_text("loc", size = 1, col = "black", shadow = TRUE) +
  tm_shape(germany) + tm_borders()
tmap_mode('plot')
tm_shape(EJSON[c(1:25),], bbox = bbox_new) +
  tm_text("loc", size = 1, col = "black", shadow = TRUE) +
  tm_shape(germany) + tm_borders()
max_1<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/1/cor_w_max_1_2.csv")
head(max_1)
max_2<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/2/cor_w_max_2_5.csv")
head(max_2)
max_3<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/3/cor_w_max_3_1.csv")
head(max_3)
```

```
max_4<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/4/cor_w_max_4_2.csv")
head(max_4)
max_5<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/5/cor_w_max_5_3.csv")
head(max_5)
max_6<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/6/cor_w_max_6_4.csv")
head(max_6)
max_7<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/7/cor_w_max_07_5.csv")
head(max_7)
max_8<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/8/cor_w_max_8_4.csv")
head(max_8)
max_9<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/9/cor_w_max_9_2.csv")
head(max_9)
max_10<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/10/cor_w_max_10_3.csv")
head(max_10)
max_11<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/11/cor_w_max_11_5.csv")
head(max_11)
max_12<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/12/cor_w_max_12_3.csv")
head(max_12)
max_13<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/13/cor_w_max_13_5.csv")
head(max_13)
max_14<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/14/cor_w_max_14_3.csv")
head(max_14)
max_15<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/15/cor_w_max_15_1.csv")
head(max_15)
```

```
max_16<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/16/cor_w_max_16_3.csv")
head(max_16)
max_17<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/17/cor_w_max_17_4.csv")
head(max_17)
max_18<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/18/cor_w_max_18_4.csv")
head(max_18)
max_19<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/19/cor_w_max_19_4.csv")
head(max_19)
max_20<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/20/cor_w_max_20_2.csv")
head(max_20)
max_21<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/21/cor_w_max_21_1.csv")
head(max_21)
max_22<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/22/cor_w_max_22_4.csv")
head(max_22)
max_23<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/23/cor_w_max_23_2.csv")
head(max_23)
max_24<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/24/cor_w_max_24_3.csv")
head(max_24)
max_25<-
read.csv("/Users/elizabeth/Documents/GY652_Applied_Climate_Sciences/Group_E
RA5_Analysis/trials/25/cor_w_max_25_3.csv")
head(max_25)
l_1<- st_as_sf(x = max_1, coords = c("long", "lat"), crs = st_crs(europe))
l_2<- st_as_sf(x = max_2, coords = c("long", "lat"), crs = st_crs(europe))
l_3<- st_as_sf(x = max_3, coords = c("long", "lat"), crs = st_crs(europe))
l_4<- st_as_sf(x = max_4, coords = c("long", "lat"), crs = st_crs(europe))
l_5<- st_as_sf(x = max_5, coords = c("long", "lat"), crs = st_crs(europe))
l_6<- st_as_sf(x = max_6, coords = c("long", "lat"), crs = st_crs(europe))
l_7<- st_as_sf(x = max_7, coords = c("long", "lat"), crs = st_crs(europe))
l_8<- st_as_sf(x = max_8, coords = c("long", "lat"), crs = st_crs(europe))
l_9<- st_as_sf(x = max_9, coords = c("long", "lat"), crs = st_crs(europe))
l_10<- st_as_sf(x = max_10, coords = c("long", "lat"), crs = st_crs(europe))
```

```
l_11<- st_as_sf(x = max_11, coords = c("long", "lat"), crs = st_crs(europe))
l_12<- st_as_sf(x = max_12, coords = c("long", "lat"), crs = st_crs(europe))
l_13<- st_as_sf(x = max_13, coords = c("long", "lat"), crs = st_crs(europe))
l_14<- st_as_sf(x = max_14, coords = c("long", "lat"), crs = st_crs(europe))
l_15<- st_as_sf(x = max_15, coords = c("long", "lat"), crs = st_crs(europe))
l_16<- st_as_sf(x = max_16, coords = c("long", "lat"), crs = st_crs(europe))
l_17<- st_as_sf(x = max_17, coords = c("long", "lat"), crs = st_crs(europe))
l_18<- st_as_sf(x = max_18, coords = c("long", "lat"), crs = st_crs(europe))
l_19<- st_as_sf(x = max_19, coords = c("long", "lat"), crs = st_crs(europe))
l_20<- st_as_sf(x = max_20, coords = c("long", "lat"), crs = st_crs(europe))
l_21<- st_as_sf(x = max_21, coords = c("long", "lat"), crs = st_crs(europe))
l_22<- st_as_sf(x = max_22, coords = c("long", "lat"), crs = st_crs(europe))
l_23<- st_as_sf(x = max_23, coords = c("long", "lat"), crs = st_crs(europe))
l_24<- st_as_sf(x = max_24, coords = c("long", "lat"), crs = st_crs(europe))
l_25<- st_as_sf(x = max_25, coords = c("long", "lat"), crs = st_crs(europe))
tmap_mode('plot')
p1<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_1, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 1", main.title.position = "center")
p2<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_2, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 2", main.title.position = "center")
p3<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_3, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 3", main.title.position = "center")
p4<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_4, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 4", main.title.position = "center")
p5<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_5, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 5", main.title.position = "center")
p6<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_6, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 6", main.title.position = "center")
p7<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_7, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
```

```
  tm_layout(legend.show = F, main.title = "Location 7", main.title.position = "center")
p8<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_8, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 8", main.title.position = "center")
p9<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1) +
  tm_shape(l_9, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 9", main.title.position = "center")
p10<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_10, bbox = bbox_new) + tm_bubbles(col = "freq",
                                               size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 10", main.title.position = "center")
p11<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_11, bbox = bbox_new) + tm_bubbles(col = "freq",
                                               size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 11", main.title.position = "center")
p12<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_12, bbox = bbox_new) + tm_bubbles(col = "freq",
                                               size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 12", main.title.position = "center")
p13<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_13, bbox = bbox_new) + tm_bubbles(col = "freq",
                                               size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 13", main.title.position = "center")
p14<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_14, bbox = bbox_new) + tm_bubbles(col = "freq",
                                               size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 14", main.title.position = "center")
p15<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_15, bbox = bbox_new) + tm_bubbles(col = "freq",
                                               size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 15", main.title.position = "center")
p16<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_16, bbox = bbox_new) + tm_bubbles(col = "freq",
```

```
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 16", main.title.position = "center")
p17<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_17, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 17", main.title.position = "center")
p18<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_18, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 18", main.title.position = "center")
p19<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_19, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 19", main.title.position = "center")
p20<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_20, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 20", main.title.position = "center")
p21<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_21, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 21", main.title.position = "center")
p22<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_22, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 22", main.title.position = "center")
p23<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_23, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 23", main.title.position = "center")
p24<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_24, bbox = bbox_new) + tm_bubbles(col = "freq",
                                       size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
```

```
  tm_layout(legend.show = F, main.title = "Location 24", main.title.position = "center")
p25<- tm_shape(EJSON, bbox = bbox_new) + tm_bubbles(col = "white", size = 0.1)
+
  tm_shape(l_25, bbox = bbox_new) + tm_bubbles(col = "freq",
                                              size = 1.25, style = "cont", palette = "Reds") +
tm_shape(germany) + tm_borders() +
  tm_layout(legend.show = F, main.title = "Location 25", main.title.position = "center")
tmap_mode('plot')
tmap_arrange(p1,p2,p3,p4,p5, nrow=1)
tmap_arrange(p6,p7,p8,p9,p10, nrow=1)
tmap_arrange(p11,p12,p13,p14,p15, nrow=1)
tmap_arrange(p16,p17,p18,p19,p20, nrow=1)
tmap_arrange(p21,p22,p23,p24,p25, nrow=1)
plot_sept_loc=function(x){
  par(mfrow=c(5,5), mar = c(2, 1, 1.5, 1))
  plot_draw=lapply(x, function(x)  plot(x[,5], cex=0.2, main= x[1,1], yaxt="n",
xaxt="n"))}
plot_sept_loc(location_list)
```