MNXB11 Project: SMHI OpenData Analysis

Kyra Bohnen, Andrea Delic, Victor Hallberg, Kacper Opas, Lasse Ulbricht

30 October 2025

1 Introduction

"April, April, der macht was er will" is an old german saying about german weather being unpredictable in April. Living in Sweden it feels like one can apply this to the whole year. This of course is not ideal when one wants to have an outdoor party in the sunshine. Where and when should one plan this? A solid answer to this can be deduced from statistics. Analyzing datasets from the Swedish Meteorological and Hydrological Institute (SMHI) gives multiple fun possibilities to compare the Swedish weather in different places over multiple years and decades.

2 Common part

The datasets received from SMHI showed multiple temperature measurements a day at different times, where the quality of the measurements differed. To be able to work with the datasets properly some preprocessing had to be done. This was achieved using `BASH` scripting where a simple script was created to make a preprocessed copy of the given data. In the copy of the data the leading lines as well as the first and last year were deleted due to incomplete sets. Then the files were put into proper csv formatting with year,month,day,temperature. The average temperature per day was calculated using loops in the script. Furthermore, a wrapper script was created to automatically preprocess all the data files according to the format desired. It must be mentioned though, that all given temperatures were used to get the mean value, not considering the quality of the measurement (green and yellow measurements). This is because without the yellow datapoints some datasets would be very small leading to statistically insignificant data.

To receive a nice and uniform code, it was structued with mulitple header files, one for the actual analysis of the different cases `analyses.h` as well as two more headers introducing the class `TemperatureData` and methods that let us perform different actions on the dataobjects such as plotting and filtering out leap years. Further a separate folder for the resulting plot was created as well as a file for writing the datasets into `TTrees` and the `main.cxx` in which the functions for actual analysis are called.

3 Analysis

3.1 Mean Temperature Each Day in Falsterbo

The data was analyzed using the `TemperatureData` class. Each date was grouped by day of the year (1–366), and the mean temperature for each day was calculated over the period 1881–2022. The error for each temperature was determined as the standard deviation of all temperatures recorded

on the same day. Both statistics were computed using the `calculateMeanProfile()` method of the `TemperatureData` class. The results were graphed using the `makeErrorGraph()` function. The resulting plot is shown in **Figure 1** below.

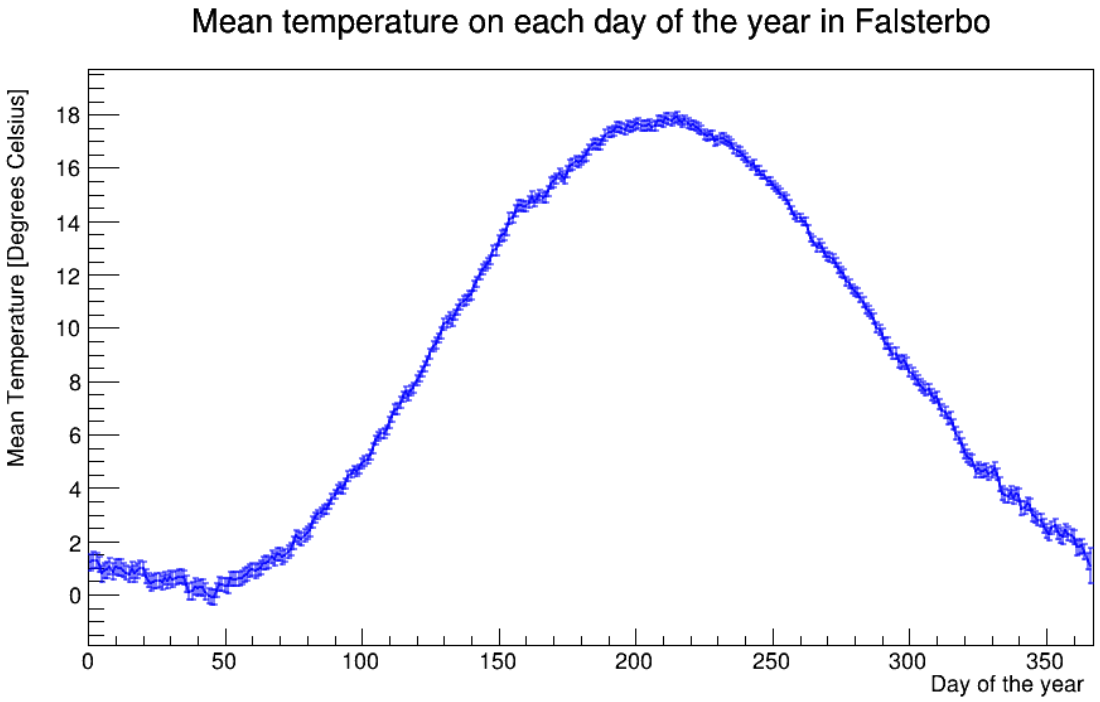

Figure 1: Mean temperature for each day in Falsterbo, calculated over the period 1881–2022.

The observed trend largely agrees with the plot of daily temperatures provided in the lab manual. The main difference is the size of the error bars, which is due to the use of a different method for calculating the error.

3.2 Warmest day in Karlstad

When looking for the warmest day in Karlstad, the function `warmest_day_Karlstad()` is implemented. Here the class `Temperature class` provided preprocessed data of the Temperature in Karlstad. The warmest day during each year was found by looping over the days and adding the highest found temperature to a list. This was done for all years, where the temperature was taken. The amount of times a specific day was the warmest in the year was plotted as a histogram using the function `makeHistogram()` as seen in Fig 2.

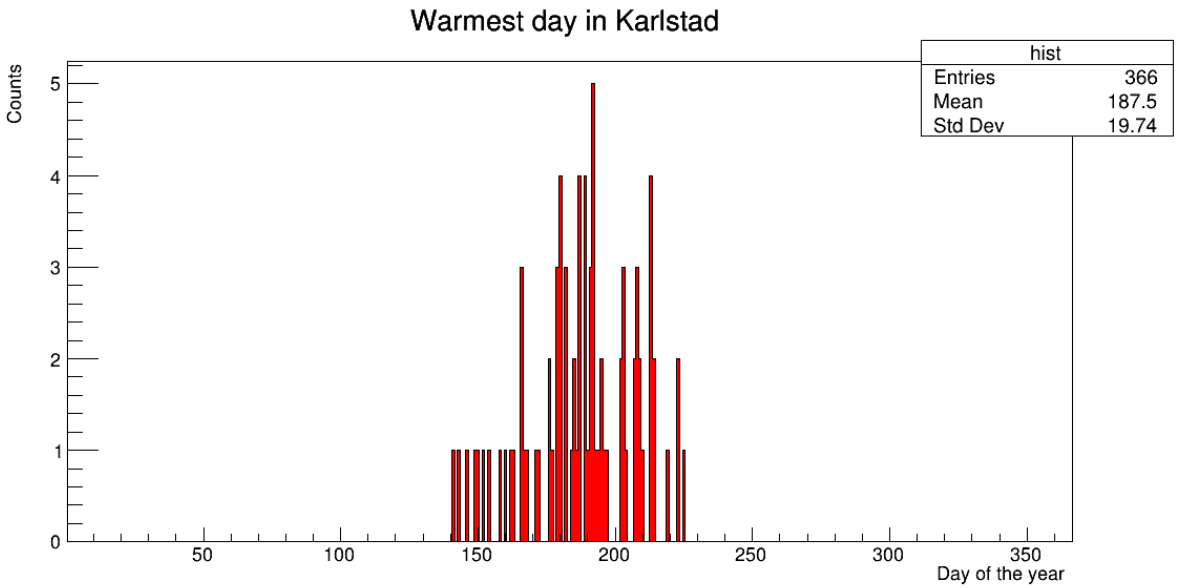

Figure 2: A histogram of the warmest day in Karlstad

3.3 Average January temperature in Luleå

The preprocessed data was further modified by using the `TemperatureData` class. The range of years was defined from 1950 to 2022. Using the member function `calculateMeanProfile`, all months except for January were filtered out and the mean January temperature was calculated. The results were plotted using the `makeBarPlot` function, and are shown in Figure 3 below. A running average trend line is overlayed as well.

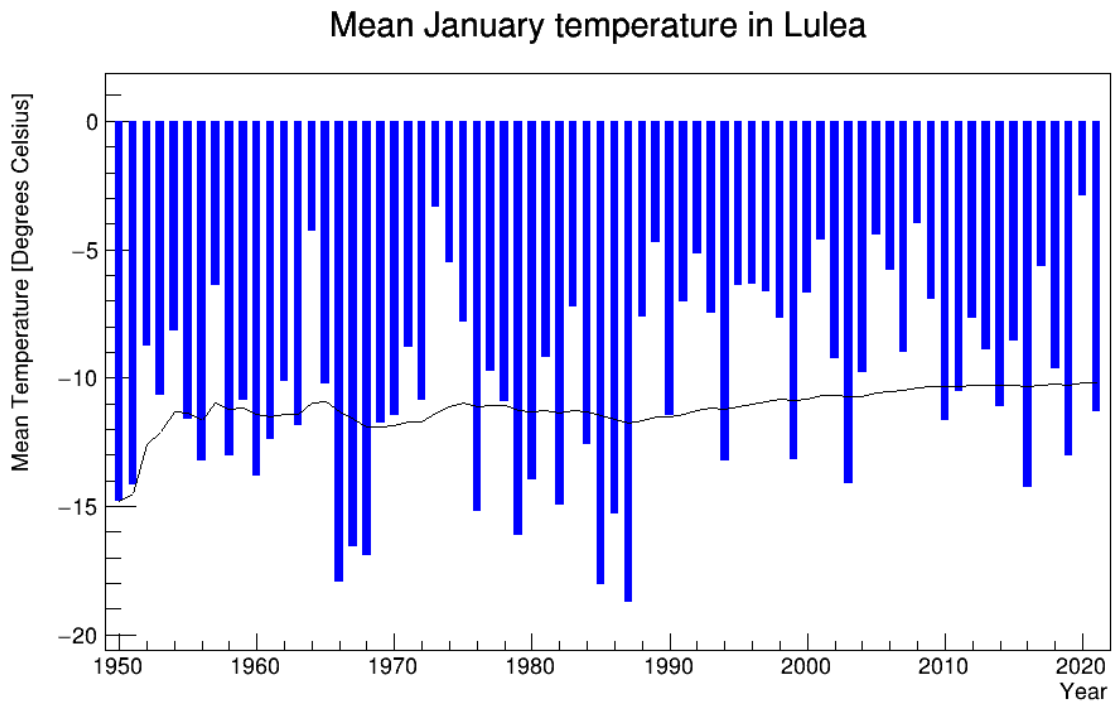

Figure 3: Mean January temperature for Luleå over the years 1950 to 2022.

One can observe a slight trend upwards in the running average trend line, which is to be expected due to global warming.

3.4 Difference in yearly average temperature between Luleå and Falsterbo

Here the `TemperatureData` class was used to create one object each for Luleå and Falsterbo. The `calculateMeanProfile` member function was then used to extract the mean values for each year and then vectors were used to calculate the difference in temperature each year using loops. Finally the `makeBarPlot` member function was used to create a histogramm showing the temperature difference over the years.

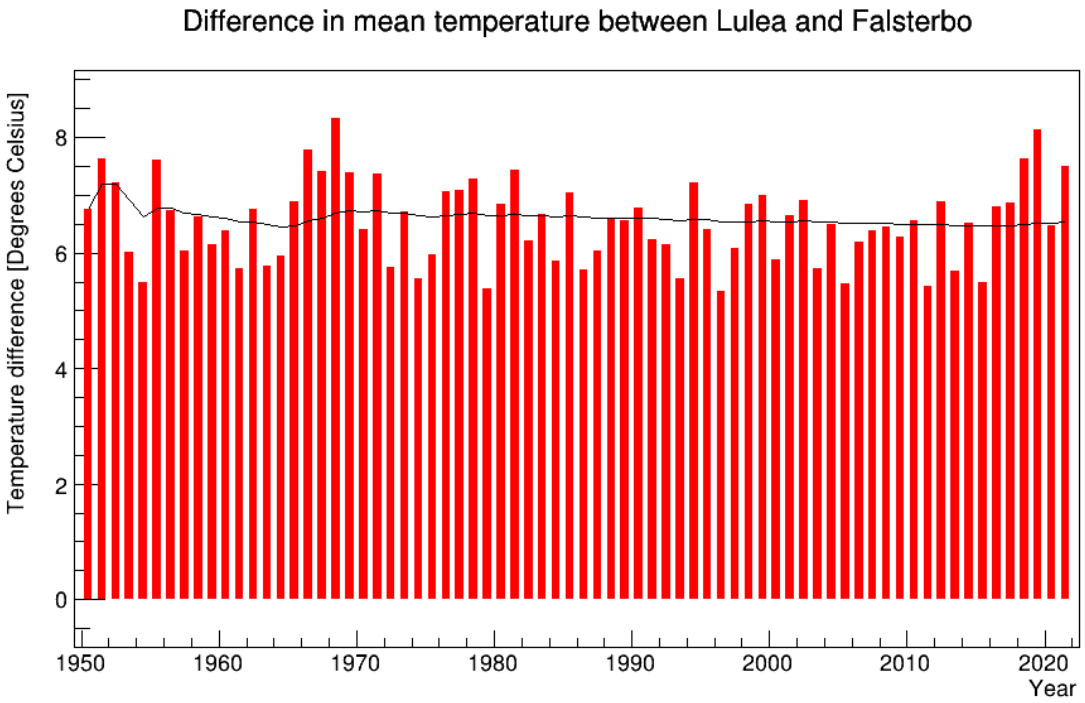

Figure 4: Difference in yearly average temperature between Luleå and Falsterbo between 1950 and 2022 with a trendline

As seen from the graph and general trendline there are no significant changes over the years which could be due to the small size of the dataset or due to the fact that the difference is fairly contant due to both being coastal cities.

3.5 Warmest vs Coldest day in Karlstad

In order to calculate the difference in temperature between the warmest and coldest day each year we apply the **TemperatureData** class. The amount of bins necessary to have one for each year was found by having the program subtract the first year from the last. The program then loops over all the years and selecting the lowest and highest recorded temperatures in the list, calculating the difference between them and adding them to a list. The results are then plotted using **makeBarPlot**, measured against a mean, as seen in Figure 5.

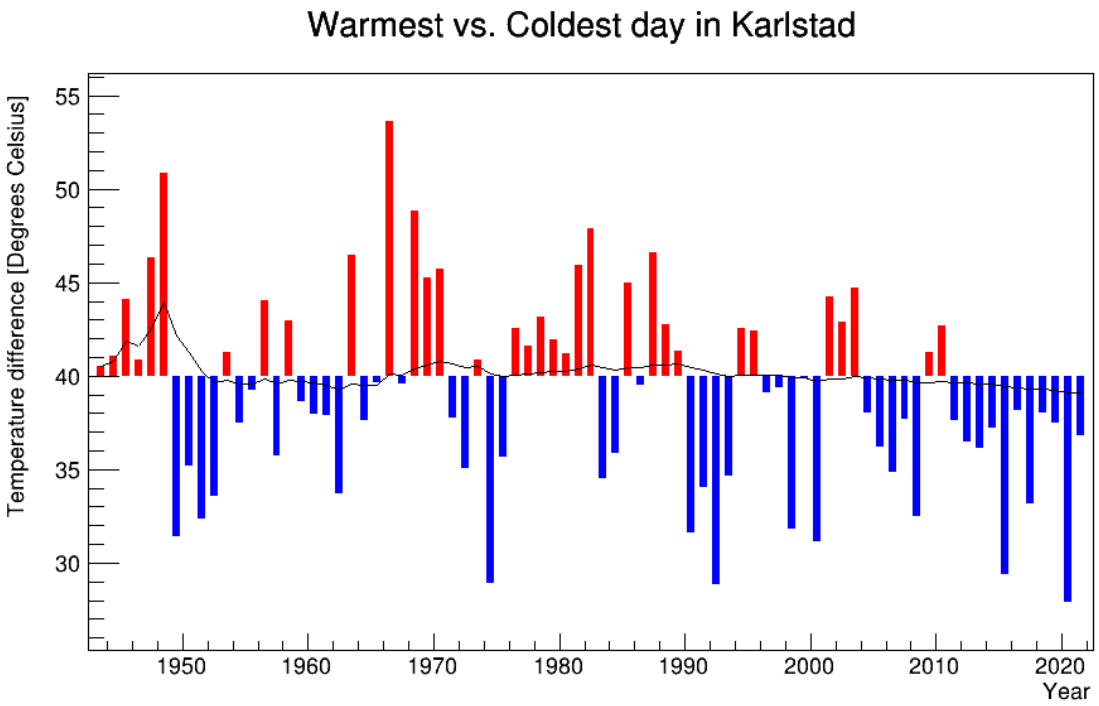

Figure 5: Difference in highest and lowest recorded temperature in Karlstad for each year between 1943 and 2022.

There does not seem to be any real insights to be gained due to lack of data, though there does seem to be a slight downward trend in temperature difference over time.

4 Conclusion

Generally this project shows that many different results can be produced. The use of automation of dataprocessing using `BASH` as well as the use of the class `TemperatureData` made it easy to produce different results on different datasets. Overall collaborating via github was effective in creating an environment in which everyone could contribute and structure a bigger project via many small contributions.