

Lab 2: Characterizing UBC Weather Station Climate

Lab Overview

In this lab, you will learn various pre-processing and processing that can be applied to a climate time-series. We will use temperature data from the UBC weather station in order to characterize the climate of the Vancouver area. You will plot the daily mean, monthly mean, and monthly mean anomaly temperature time-series, as well as the seasonal cycle. The seasonal cycle will be characterized by monthly means, standard deviations, and minimum and maximum values for the daily mean temperature.

You should have the data downloaded and ready to go from the pre-lab.

Learning Goals

After completing this lab successfully, you will be able to:

- Inspect a data set to ensure the data has been imported correctly and identify some expected and unexpected signals
- Generate a monthly mean annual cycle from a larger time series, including the monthly standard deviation, and the minimum and maximum values
- reduce a daily time series to a monthly one by calculating monthly averages
- construct an anomaly time series

To hand in

1. One figure containing plots of the daily air temperature, the monthly air temperature, and the monthly air temperature anomaly.
2. A figure showing the monthly means, one standard deviation, minimum and maximum of daily temperature.
3. Captions for each of the figures
4. The worksheet for this lab

1. Load the data

Look at the time series you downloaded during the pre-lab in a text editor: it should start sometime in the late 1950's or early 1960's and have data recorded *once per day*. If this is not the case, you likely downloaded the data incorrectly.

Load the time series into MATLAB using **ONE** of these two options:

-Option 1: open excel, open the .dat files you downloaded in excel, and save them as xlsx (or excel format) file. Then, you can use the command `[num text all]=xlsread('yourfilename.xlsx')` to load the file. num will be a matrix containing all numerical values and text will be a cell array containing all strings. Feel free to check the matlab webpage of the xlsread function!

-Option 2: use the `textscan()` (cf lab 1). As the first and second fields contain strings, not numbers, the appropriate format for each file should be `'%s %s %f'`. The first string is the date, the second the time (which should always be `00:00`), and the third the temperature data.

Your goal is to end up with two vectors, a *date* and *data* pair. Remember that you need to convert the date strings, which you initially loaded into MATLAB into the *'datetime'* format

before they'll be much use to you.

The easiest way to check that your data has loaded correctly is to plot it. You can just call `> plot(temp_date, temp_data)` at the command line to see what it looks like, if vector `temp_date` contains date in datenum format and `temp_data` the corresponding temperature data. Can you think of a way to verify that the plot you see now is correct? Does it make a difference if you remove NaNs before plotting? If yes, is it better to remove them or not?

2. Monthly averages, standard deviations, minimums and maximums.

You will now generate monthly average of the air temperature for each of the 12 month over the entire period. We are asking you to do the following: **1) for each of the twelve months of the year, calculate the average of all the temperatures recorded *during that month* for the entire ~60 years for which you have data. 2) for all the records taken during that specific month, calculate the standard deviation of those values, as well as the minimum recorded value and maximum recorded value.**

What you are doing is collapsing fifty years worth of daily measurements into twelve data points, one for each month. The standard deviation will give you a measure of the variability around each of those data points, and the min and max values will give you an upper and lower bound.

On a single figure, plot the following five curves characterizing the seasonal cycle: the average temperature, the average temperature minus – and plus – one standard deviation, the minimum temperature and the maximum temperature. Each curve should have 12 datapoints (one for each month). Include a legend with the function `legend()` to specify what you are plotting. In the Lab 2 folder on Connect, you will find an example of what *the plot* might look like. The example figure uses soil temperature measurements at UBC from the years 2000 to 2011. Don't forget a caption for your figure - even though we haven't included one in the example plot.

When you are done, please hand in your final figure. ***Make sure you spend a few minutes thinking about your figure:*** does it look correct? Why? Do the result make sense given that the temperature data is recorded in Vancouver?

A few hints that will help you accomplish your task:

1. You will need to put your dates into vector format because you can create a mask from the month information stored in the second column of the resulting array. You can convert numerical dates to vector dates using the function `datevec()`.
2. Try to get the calculations working for one month first. Once you've done this, modify your code to use a for-loop that loops through the 12 months of the year. Perform your data processing within this loop.
3. You will need separate variables to hold your monthly means, standard deviations, etc... Initialize these as empty arrays before the loop begins. What should be the size of each array?
4. Make use of the functions `mean()` or `nanmean()`, `std()` or `nanstd()`, `max()`, `min()`, and `legend()`.
5. If you try to plot 2 time series on the same figure, you may find that Matlab will erase the one plotted first. Use the command `hold on` after plotting the first time-series so that

Matlab keep all time series you plot on the figure.

Environment Canada publishes normal climate values for Vancouver international airport (VANCOUVER INT'L A) on their website; you can use this data to check if your seasonal values looks correct.

http://climate.weather.gc.ca/climate_normals/index_e.html

3. Create a monthly time-series

Daily time series are of course useful if one want to learn about weather variability, but characterizing climate generally involves a look at what happens at longer timescales, typically from seasonal timescale to hundreds of million years. We will thus calculate the average air temperature for each month of each year for which we have daily measurement available. The next step will be to calculate monthly temperature anomalies, i.e., calculate the difference between this monthly mean time-series and the monthly average, or normal, calculated across all years for which we have data in part 2.

To calculate the monthly mean temperature time series, you will need to use a loop that cycles through each month of each of the ~60 years. You can also use the same loop to create an appropriate date vector which you can use to represent the dates for your time series (like we did in lab 1 part 2). The outside of the loop will look something like this:

```
for year=firstyear:lastyear
    for month=1:12
        %Your averaging algorithm goes here
    end
end
```

A couple of tips that may help you:

- Convert your datenumber arrays to datevector arrays, and use these to create a mask which will isolate the year AND month from which you want to average data.
- You'll need a counter inside the loop for correctly indexing into the new date array which you are creating. You can use this same counter to also index into the new monthly-temperature that you have created.
- It's not a problem if a given month doesn't contain any data at all, since taking the mean of an empty array simply returns a NaN.

Again, the easiest way to check that your data has loaded correctly is to plot it. Can you think of a way to verify that the plot you see now is correct? What are the main differences between the plot of the daily mean and the monthly mean temperature at UBC weather station? Based on these time series, would you say that there is any trend in surface temperature recorded at UBC? Try to discuss any feature you find surprising in this plot with your classmates and/or with me.

4. Generate anomaly time series

Temperature variations in the daily mean and monthly mean temperature time series generated in part 1 and 3 are largely dominated by the seasonal cycle, characterized in part 2. Thus, these two time-series do not tell much about climate variations. They simply indicate that temperature variations, in Vancouver, are dominated by the orbit of the Earth, which repeats every year in (almost) the same way. Therefore, to examine variability in the climate, we are now going to remove the seasonal cycle. A long time series of data which has the annual cycle removed is known as an *anomaly time series*, since it measures how anomalous a given month is compared to the average annual cycle.

To remove the mean annual cycle from a monthly time series, you have to first calculate the mean annual cycle -- you did this in part 2 when you looped through the twelve months of the year. Once you have the mean annual cycle, subtract it from the monthly data.

The easiest way to accomplish this might be to create an annual cycle vector as long as your monthly mean temperature time-series, in which the 12 month annual cycle repeats as many times as the number of years in the monthly mean time-series. This can be done using the function `repmat()`. You can use the help documentation if you need to, but in general, the call `repmat(A,M,N)` tiles the array `A` vertically `M` times and horizontally `N` times. Once you have this annual cycle vector, you can simply subtract it from your monthly time series using `B-A` since the two vectors have the same dimension. I've used dummy variable names here; you'll be using your own variable names.

Once again, plot the monthly mean temperature anomaly time-series that you obtained and verify that the data looks correct. What are the main differences between the plot of the monthly mean temperature and the monthly mean temperature anomaly at UBC weather station? Based on the temperature anomaly time-series, is there any trend in monthly mean temperature anomalies recorded at UBC? Try to discuss any feature you find surprising in this plot with your classmates and/or with me.

5. Plot the data using *subplot()*

During this lab, you generated 3 different temperature time series: the daily mean temperature, the monthly mean temperature, and the monthly mean temperature anomaly. You will now produce the second figure required for this lab by plotting each of the three time-series *in a separate subplot of one figure*. Use the `subplot()` command to do this. You

should make a proper figure for submission, with appropriate axes labels, a title, easy to read font-size, etc... You will have to decide whether it is more logical to place the individual subplots side-by-side (horizontally) or one on top of the other (vertically). Remember to include a figure caption.

To begin using the subplot command, you can modify the basic example template below. In this example, I have chosen to make a figure with six subplots, three rows and two columns.

```
%First initialize/open a new figure
figure(1)
%Then split the figure window into six fields, three rows,
%two columns. Make the first field active.
subplot(3,2,1)
%Make your first plotting call and add any labels, etc...
plot(x1,y1)
xlabel('place label here')
%Make the second field active. Make second plotting call.
subplot(3,2,2)
plot(x2,y2)
%Etcetera
```

You can find more information by entering `help subplot` at the command prompt or looking in the MATLAB help browser.