

**ATOC7500 – Application Lab #2**  
**Regression, Autocorrelation, Red Noise Timeseries**  
**in class Feb. 10/15, 2022**

**Notebook #1 – Autocorrelation and Effective Sample Size using Fort Collins, Colorado weather observations**

[ATOC5860\\_applicationlab2\\_AR1\\_Nstar.ipynb](#)

**LEARNING GOALS:**

- 1) Calculate the autocorrelation at a range of lags using two methods available in python (np.correlate, dot products)
- 2) Estimate the effective sample size ( $N^*$ ) using the lag-1 autocorrelation
- 3) Evaluate the influence of changing the sampling frequency and the specified weather variable on the memory/redness of the data as quantified by the autocorrelation and  $N^*$ .

**DATA and UNDERLYING SCIENCE:**

In this notebook, you will analyze the memory (red noise) in weather observations from Fort Collins, Colorado at Christman Field. The observations are from one year, but are sampled hourly. The default settings for the notebook analyze the air temperature in degrees F sampled once daily (every midnight). But other standard weather variables and sampling frequencies can also be easily analyzed. The file containing the data is called christman\_2016.csv and it is a comma-delimited text file.

**Non-exhaustive Questions to guide your analysis of Notebook #1:**

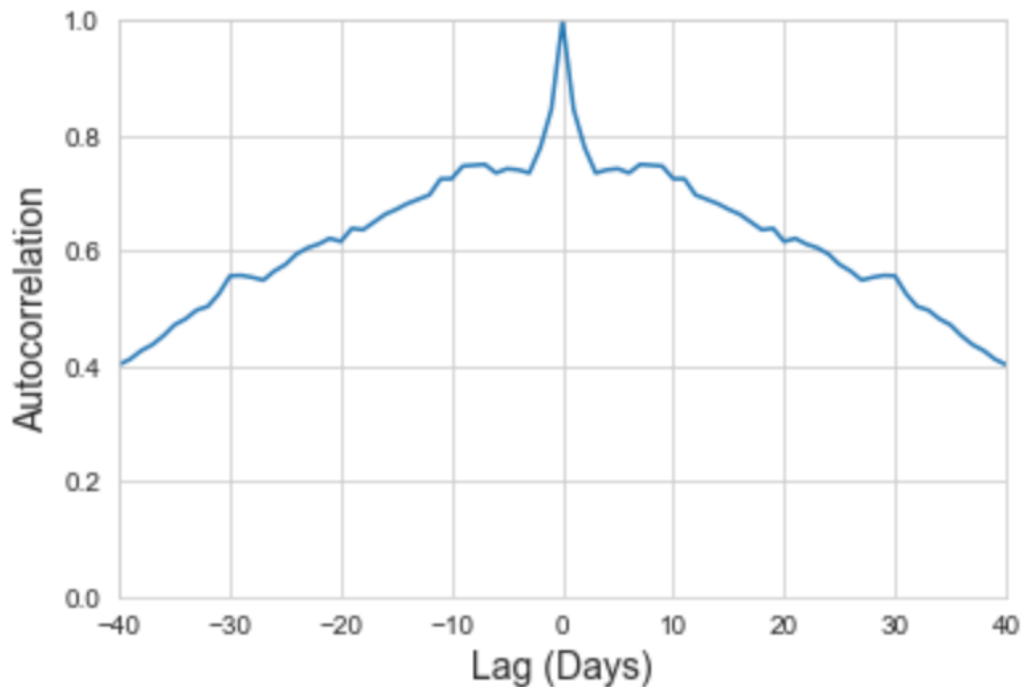
- 1) Start with the default settings in the code. In other words – Read in the data and find the air temperature every 24 hours (every midnight) over the entire year. Calculate the lag-1 autocorrelation using np.correlate and the direct method using dot products. Compare the python syntax for calculating the autocorrelation with the formulas in Barnes. Equation numbers are provided to refer you back to the Barnes Notes. What is the lag-1 autocorrelation?

**The lag-1 autocorrelation is 0.846.**

- 2) Calculate the autocorrelation at a range of lags using np.correlate and the direct method using dot products. Compare the python syntax for calculating the autocorrelation with the formulas in Barnes. Equation numbers are provided to refer you back to the Barnes Notes. How does the autocorrelation change as you vary the lag from -40 days to +40 days?

**The autocorrelation seems to work on two timescales: from zero lags to a lag of a few days, data has very high autocorrelation that drops off until stabilizing around 3 days and decaying more slowly. From ~3-40 days, autocorrelation decays more slowly,**

representing that temperature is autocorrelated on longer timescales up to around a month in length.



3) Calculate the effective sample size ( $N^*$ ) and compare it to your original sample size ( $N$ ). Equation numbers are provided to refer you back to the Barnes Notes. How much memory is there in temperature sampled every midnight?

**The effective sample size is 31, which is a little less than  $1/10^{\text{th}}$  the size of the full dataset. This indicates that data has memory on the timescale of around 12 days.**

4) Now you are ready to tinker ... i.e., make minor adjustments to the code with the parameters set in the code to see how your results change. Suggestion: Make a copy of the notebook for your tinkering so that you can refer back to your original answers and the unmodified original code. For example: Repeat steps 1-3) above with a different variable (e.g., relative humidity (RH), wind speed (wind\_mph)). Repeat steps 1-3) above with a different temporal sampling frequency (e.g., every 12 hours, every 6 hours, every 4 days). How do you answers change?

**Temperature with higher frequency (12 hours):**

Lag-1 autocorrelation: 0.593

How does the autocorrelation change as you vary the lag from -40 days to +40 days?:

There is periodicity representing the diurnal cycle

Compare  $N$  and  $N^*$ :  $N = 732$ ,  $N^* = 187$ , which means that TEMP has relatively memory, but this is broken because of the diurnal periodicity.

**RH:**

Lag-1 autocorrelation: 0.298

How does the autocorrelation change as you vary the lag from -40 days to +40 days?:

After the first 2 lags, there is effectively zero autocorrelation in the data.

Compare N and N\*: N = 366, N\* = 198, which means that RH has very little memory.

**wind\_mph:**

Lag-1 autocorrelation: -0.045

How does the autocorrelation change as you vary the lag from -40 days to +40 days?:

There is no autocorrelation after the first lead-time!

Compare N and N\*: N = 366, N\* = 335, which means that wind has very little memory.

**Precipitation:**

Lag-1 autocorrelation: -0.001

How does the autocorrelation change as you vary the lag from -40 days to +40 days?: It drops off and there is no autocorrelation

Compare N and N\*: N = 366, N\* = 366, so there is no memory

**Notebook #2 – Red noise time series generation, Regression, and Statistical Significance**  
**Testing While Regressing**  
[ATOC5860\\_applicationlab2\\_AR1\\_regression\\_AO.ipynb](#)

**LEARNING GOALS:**

- 1) Calculate and analyze the autocorrelation at a range of lags using output from an EOF analysis (the Arctic Oscillation Index).
- 2) Generate a red noise time series with equivalent memory as an observed time series (i.e., given lag-1 autocorrelation).
- 3) Correlate two time series and calculate the statistical significance.
- 4) Evaluate the statistical significance obtained in the context of the number of chances provided for success. What happens when you go “fishing” for correlations and give yourself lots of opportunity for success? Can you critically evaluate the chances that your regression is statistically different than 0 just by chance?

**DATA and UNDERLYING SCIENCE:**

In this notebook, you will analyze the monthly Arctic Oscillation (AO) timeseries from January 1950 to present. The AO timeseries comes from an Empirical Orthogonal Function (EOF) analysis. We will implement EOFs in the next application lab so in this lab we are actually using multiple analysis methods introduced in this class, some that you have learned and some that you are still yet to learn 😊.

How do you find the AO value each month? To identify the atmospheric circulation patterns that explain the most variance, NOAA regularly applies EOF analysis to the monthly mean 1000-hPa height anomalies poleward of 20° latitude for the Northern Hemisphere. The AO spatial pattern (Figure 1 below) emerges as the first EOF (explaining the most variance, 19%). The AO timeseries we will analyze is a measure of the amplitude of the pattern in Figure 1 in a given month. In other words – the AO timeseries is the first principal component (a timeseries) associated with the first EOF (a spatial structure). More information on the EOF analysis here:

[http://www.cpc.ncep.noaa.gov/products/precip/CWlink/daily\\_ao\\_index/history/method.shtml](http://www.cpc.ncep.noaa.gov/products/precip/CWlink/daily_ao_index/history/method.shtml)

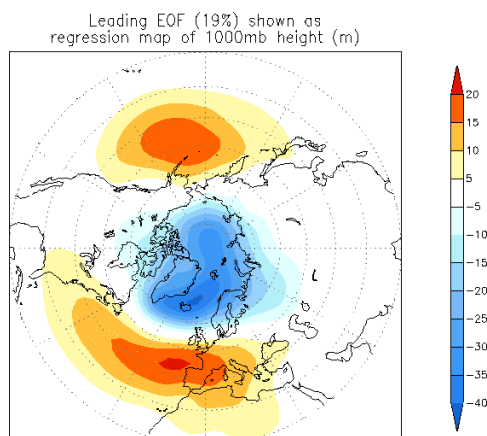


Figure 1. The loading pattern of the Arctic Oscillation (AO), i.e., the structure explaining the most variance of monthly mean 1000mb height during 1979-2000 period. In other words – this is the first EOF.

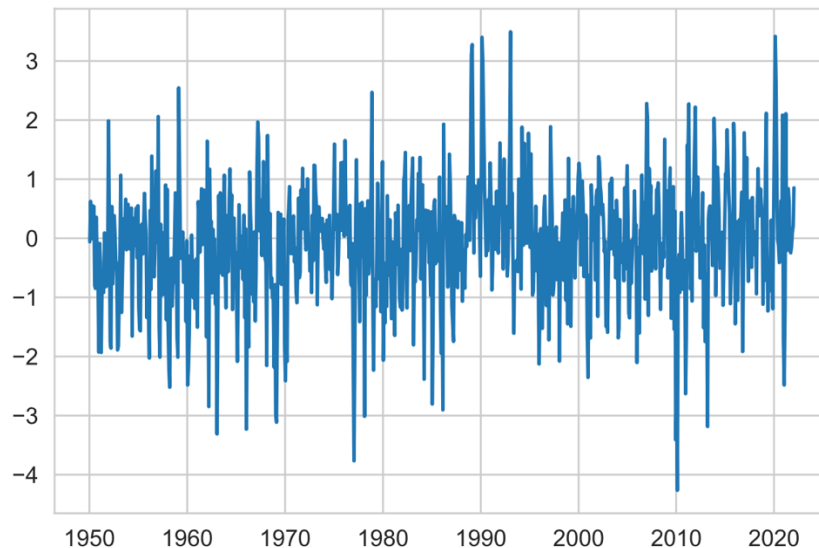
The data are available and regularly updated here:

<http://www.cpc.ncep.noaa.gov/products/precip/CWlink/pna/norm.nao.monthly.b5001.current.ascii>

You can work with the data directly on the web (assuming you have an internet connection). I have also downloaded the data and made them available – The name of the data file is “monthly.ao.index.b50.current.ascii”.

### Questions to guide your analysis of Notebook #2:

1) Start with the default settings in the code. First read in the Arctic Oscillation (AO) data. Look at your data!! Plot it as a timeseries. Save the timeseries plot as a postscript file and put it in this document.



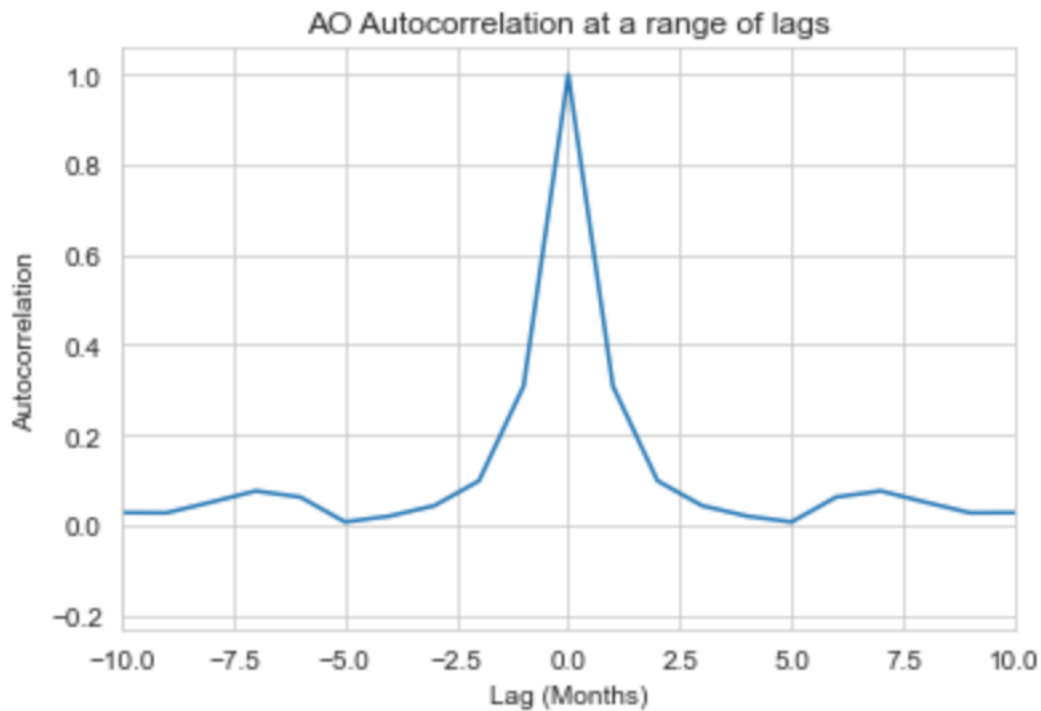
2) Calculate the lag-one autocorrelation (AR1) of the AO data and record it here. Use two methods (np.correlate, dot products). Check that they give you the same result. Interpret the value. How much memory (red noise) is there in the AO from month to month?

Both methods give a lag-1 autocorrelation: 0.30855. Thus, our  $R^2$  says that 9.5% of the variance is explained by the lag-1 autocorrelation. This means that there ain't a lot of memory.

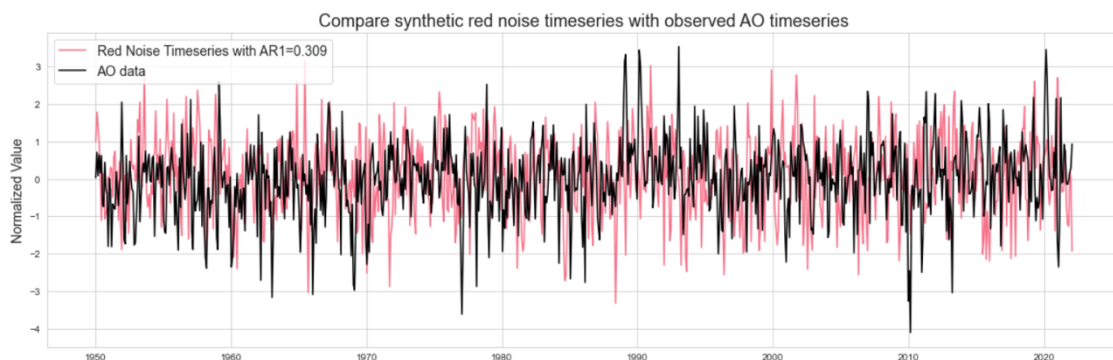
3) Calculate and plot the autocorrelation of the AO data at all lags. Describe your results. How red are the data at lags other than lag=1? Is there any interesting behavior

of the autocorrelation as a function of lag? What would you expect for red noise timeseries with an  $AR1$ =value reported in 2)?

The data as displayed below are not very red, as the autocorrelation falls off super fast. This matches what we would expect because there was a low lag-1 autocorrelation!



4) Generate a synthetic red noise time series with the same lag-1 autocorrelation as the AO data. Your synthetic dataset should have different time evolution but the same memory as the AO. Plot the AO timeseries and the synthetic red noise time series. Put the plot below.



5) Do you expect to find any correlation between the two datasets, i.e., the synthetic red noise and the actual AO data? What is the correlation between the synthetic red

noise and the actual AO data? Calculate a regression coefficient and other associated regression statistics.

I would expect very little correlation between the two datasets, as they both have little autocorrelation (inherently). Our regression coefficient is 0.018, so there is very little correlation there.

6) Next -- Have some fun and go “fishing for correlations”. What happens if you try correlating subsets of the two datasets many times? When you try 200 times -- what is the maximum correlation/variance explained you can obtain between the synthetic red noise and the actual data? *Note: you are effectively searching for a high correlation with no a priori reason to do so.... THIS IS NOT good practice for science but we are doing it here because it is instructive to see what happens :)*

The maximum correlation is 0.52 and variance explained is 27.25%.

7) Calculate the correlation statistics for the highest correlation obtained in question 6). Two methods are provided - they should give you the same answers. Place a confidence interval on your correlation. Because you have found a correlation that is not equal to 0, use the Fisher-Z Transformation. Did your "fishing" for a statistically significant correlation work? Is your highest correlation statistically significant (i.e., can you reject the null hypothesis that the correlation is zero)? Write out the steps for hypothesis testing and use the values you calculate to formally assess.

Our highest correlation values has a 95% confidence interval of 0.07-0.8. This is not equal to 0, so the fishing worked and we can reject the null hypothesis!

1. 95% Confidence ( $\alpha = 0.05$ )
2.  $H_0: r = 0$ ,  $H_1: r \neq 0$
3. We used a fisher transform to then use a t-test on our correlation values (assume normality as  $N < 30$ )
4. Reject the null hypothesis if our confidence intervals are not  $=0$ , with a critical value of 2.11
5. We can reject the null hypothesis. Our highest correlation values has a 95% confidence interval of 0.07-0.8. This is not equal to 0, so the fishing worked and we can reject the null hypothesis!

8) You went searching for correlations, you searched long and hard (200 times!) You should have been concerned that the largest correlation you found would be a false positive. Do you think you found a false positive? Explain what you found and potentially why you think it is important statistically but not physically. What lessons did you learn by “fishing for correlations”?

I think we likely found a **false correlation**, as we had no a priori reason to assume a non-zero correlation. Also, with 95% confidence interval, there is a 5% of a false correlation, which increases dramatically since we did this 200 times (dramatically increasing our odds).

FOR FUN: Check out - <https://www.tylervigen.com/spurious-correlations>