**Application Lab 2**

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

**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?

From the code, it looks like np.correlate does a similar thing to np.dot.

The lag-1 autocorrelation is the correlation of a data set with itself when it is lagged by 1 timestep. Both methods give a lag-1 autocorrelation of 0.846 which is pretty high. This suggests the data are fairly red. A high/strong lag-1 autocorrelation suggests that samples are not completely independent.

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?

Note: the code says you cannot make the lag negative but the autocorrelation is symmetric about 0 at all lags so it should be the same/very similar for -40 and 40. If the lag were negative, the equation in the code with “n-lag” would do the wrong thing (it would add and make the denominator larger).

|  |  |
| --- | --- |
| Lag | Autocorrelation |
| -2, 2 | 0.779 |
| -5, 5 | 0.743 |
| -10, 10 | 0.725 |
| -15, 15 | 0.672 |
| -20, 20 | 0.616 |
| -25, 25 | 0.576 |
| -30, 30 | 0.557 |
| -35, 35 | 0.472 |
| -40, 40 | 0.403 |

The autocorrelation peaks at a lag of 0 and decreases as the lag moves in the positive or negative direction, symmetric about 0. The decrease is fairly gradual which tells us there is some memory in the data.

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 and the original sample size is 366. Thus, there is a lot of memory in the temperature sampled every midnight since 366 samples is effectively only 31 samples, which is an order of magnitude smaller.

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?

Repeating steps 1-3 with daily windgust:

The lag-1 autocorrelaiton is -0.024. As lag goes from -40 to 40, the autocorrelation peaks at lag 0 and drops off very quickly after that, then bounces around on the sides, going towards both -40 and 40. The independent sample size becomes 349 from 366, suggesting the data is very independent with little memory. If you instead do hourly windgust, the lag-1 autocorrelation becomes 0.849 and the autocorrelation still peaks at 0 but falls off more slowly and has many peaks around 24 and -24. The effective sample size goes from 8784 to 715, meaning there is a lot of memory in the data, which is consistent with the fact that the lag-1 autocorrelation is so high.

**Notebook #2 – Red noise time series generation, Regression, and Statistical Significance Testing While Regressing**

**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.

Chart, line chart

Description automatically generated

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?

The lag-1 autocorrelation from both methods is 0.30855. This implies that there is some memory in the AO from month to month.

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 autocorrelation peaks at 0 and drops off gradually as the lag moves to -10 and 10. There is a little increase in the autocorrelation around 6-7 months and then it levels off again. For the timeseries shown in the plot in problem 1, I would expect some lag-1 autocorrelation but nothing too strong since the data seem to be a little bit random with sections consistently going up or down. For an AR1 value of 0.30855, I would expect the same kind of gradual drop off from 0 as we see in the plot with different lags.

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.

A picture containing text, lined, line, day

Description automatically generated

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 do not expect to find a strong correlation between the two datasets because while they do have the same lag-1 autocorrelation, this autocorrelation is based on the datasets compared with themselves and does not relate to their relationship with each other. The correlation between the two datasets is 0.0577% which is extremely small, suggesting they are not correlated. The regression coefficient is 0.000577.

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 I obtained is 0.59 and the largest variance explained is 34.59% which are both fairly strong, especially given that we are just comparing random datasets.

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.

Both methods give the same values: slope = 0.612, intercept = 0.098, and the r value = 0.588. The confidence interval is 0.16 < r < 0.83.

The steps for hypothesis testing:

1. State the significance level (alpha)
   1. For this, our significance level is alpha = 0.05, so a 95% confidence level.
2. State the null hypothesis H0 and the alternative H1
   1. H0: The correlation is zero.
   2. H1: The correlation is not zero.
3. State the statistic to be used, and the assumptions required to use it
   1. We will use a t-statistic since the number of samples is 20. This assumes that the underlying distribution is normal.
4. State the critical region
   1. The critical value of t is 2.0860. Thus, we can reject the null hypothesis if t > 2.0860. The critical region is -2.0860 < t < 2.0860 so if our t value is outside of this region, we can reject the null hypothesis.
5. Evaluate the statistic and state the conclusion
   1. We find a t-statistic of 2.110, which is outside of the critical region, meaning we can reject the null hypothesis that the correlation is zero. The confidence interval is 0.16 < r < 0.83. This confidence interval does not contain 0, further confirming that 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”?

Yes! I definitely think I found a false positive. The maximum correlation value I found previously was statistically significant, yet this has no physical meaning because the datasets are completely unrelated. The only thing in common with the two datasets are their lag-1 autocorrelations, which represents how much they correlate with themselves. I think the false positive showed up because we compared the datasets 200 times which could easily allow for the randomness in each dataset to eventually line up in a way that is statistically significant, yet this does not have any physical meaning. From this activity, I learned that one should be careful when looking for correlations because randomness can eventually correlate significantly and this would make you think you found something interesting when you did not.

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