**ATOC7500 – Application Lab #3**

**Empirical Orthogonal Function (EOF) Analysis**

**Note: This application lab requires netcdf4 and cartopy packages. Also – The 2020 python environment is provided – That one works on my computer. The 2022 python environment does not work (I think because xarray has been downgraded). Please check for updates on the python environment. I am providing these files early…**

**A reminder of the EOF/PCA Analysis Recipe – 5 steps**

**1) Prepare your data for analysis. Examples might include:**

**a) sub-setting the global data to a smaller domain**

**b) subtract the mean**

**b) standardizing the data (divide by the standard deviation)**

**d) cosine weighting (Account for the decrease in grid-box area as one approaches the pole (i.e. weight your data by the cosine of latitude)**

**e) detrend the data**

**f) remove the seasonal or diurnal cycle**

**g) remove NaN – EOF analysis does not work with missing data.**

**2) Calculate the EOFs and PCs using one of the two methods discussed in class: a) Eigenanalysis of the covariance matrix**

**b) Singular Value Decomposition (SVD).**

**3) Plot the first 10 eigenvalues (scaled as the percent variance explained) in order of variance explained. Add error bars following North et al. 1982. Describe how you determined the effective degrees of freedom N\*. How many statistically significant EOFs are there?**

**4) Plot EOF patterns and PC timeseries (usually just the first three or so unless you want to look at more).**

**5) Regress the data (unweighted data if applicable) onto standardize values of the 3 leading PCs. In other words, project the standardized principal component onto the original anomaly data X to get the EOF in pjysical units. You should have one regression pattern for each PC – i.e., the EOF pattern associated with a 1 standard deviation anomaly of the PC. *Note: The resulting patterns will be similar to the EOFs but not identical.***

**Notebook #1 – EOF analysis using images of people**

**ATOC5860\_applicationlab3\_eigenfaces.ipynb**

**LEARNING GOALS:**

1) Complete an EOF analysis using Singular Value Decomposition (SVD).

2) Provide a qualitative description of the results. What are the eigenvalues, the eigenvectors, and the principal components? What do you learn from each one about the space-time structure of your underlying dataset?

**DATA and UNDERLYING SCIENCE:**

In this notebook, you apply EOF analysis to a standard database for facial recognition: the At&t database.

<https://www.cl.cam.ac.uk/research/dtg/attarchive/facedatabase.html>

*“Our Database of Faces, (formerly 'The ORL Database of Faces'), contains a set of face images taken between April 1992 and April 1994 at the lab. The database was used in the context of a face recognition project carried out in collaboration with the Speech, Vision and Robotics Group of the Cambridge University Engineering Department.*

*There are ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement).”*

The goal is to think a bit “out of the box” of Atmospheric and Oceanic Sciences about potential applications for the methods you are learning in this class for other applications.

**Questions to guide your analysis of Notebook #1:**

1. **Execute all code without making any modifications. What do the EOFs (spatial patterns) tell you? What do the PCs tell you? How do you interpret what you are finding?**

*The EOFs appear to tell us about basic facial features, such as: eyes, nose, mouths, etc. The PCs tell us how well the reconstructed face compares to each eigenface from the sample. The more PCs you include in the reconstruction, the more the variance will be explained. The fewer PCs you use, the more it will look like the first few eigenfaces.*

1. **Reconstruct a face. How many EOFs do you need to reconstruct a face from the database? Does it depend on the face that it used?**

*Through some trial and error, it rough takes about 50 EOFs for me to begin recognizing the target face. At roughly 300 EOFs, almost all the variance is explained and the reconstructed face looks pretty similar to the original.*

**Graphical user interface, application

Description automatically generated**

**Face 120, # PCs: 100**

**3) Food for thought: The database contains 75% white men (**[**https://www.cl.cam.ac.uk/research/dtg/attarchive/facesataglance.html**](https://www.cl.cam.ac.uk/research/dtg/attarchive/facesataglance.html)**). How do you think this database limitation impacts the utility of the database for subjects who are not white men? What are some parallels that you might draw when analyzing atmospheric and oceanic sciences datasets? *Hint: Think about the limitations of extrapolation beyond the domain where you have data.***

*The fact that this database only contains 75% white men severely limits the ability of EOFs to successfully capture someone of color and gender. This is a common issue in a lot of facial recognition technology today.*

*Some parallels in our field would be satellite data. We’ve only had satellite data for a 4 or so decades, and so we only have a small subset of data capture the Earth from space, compared to the geological timespan of Earth’s existence.*

**Notebook #2 – EOF analysis of Observed North Pacific Sea Surface Temperatures**

**ATOC5860\_applicationlab3\_eof\_analysis\_cosineweighting\_cartopy.ipynb**

**LEARNING GOALS:**

1) Complete an EOF analysis using the two methods discussed in class: eigenanalysis of the covariance matrix, Singular Value Decomposition (SVD).

2) Assess the statistical significance of the results, including estimating the effective sample size.

3) Provide a qualitative description of the results. What are the eigenvalue, the eigenvector, and the principal component? What do you learn from each one about the space-time structure of your underlying dataset?

4) Assess influence of data preparation on EOF results. What happens when you remove the seasonal cycle? What happens when you detrend? What happens when you cosine weight by latitude? What happens when you standardize your data (divide by standard deviation)? What happens when you compute anomalies?

**DATA and UNDERLYING SCIENCE:**

In this notebook, you will analyze observed monthly sea surface temperatures from HadISST (http://www.metoffice.gov.uk/hadobs/hadisst/data/download.html). The data are in netcdf format in a file called HadISST\_sst.nc. *Note that this file is ~500 MB so it might take a bit of time to download.* You will subset the data to only look at the North Pacific. Depending on how you prepare your data for analysis – you might expect to see different spatial patterns (eigenvectors) and different time series (principal components). Some things you might look for in your results are the Pacific Decadal Oscillation, “global warming”, the seasonal cycle, …. Depending on your data preparation – your hypothesis for what you should see in your EOF analysis should change. Note: In this dataset - land is NaN, sea ice is -999 – the notebook sets all values over land and sea ice to 0 for the EOF analysis.

**Questions to guide your analysis of Notebook #1:**

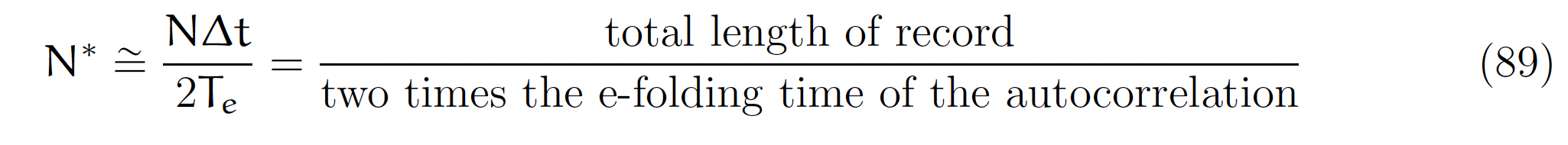
1. **Your first time through the notebook – Execute all code without making any modifications. Provide a physical interpretation for at least the first two EOFs and principal components (PC). What do the EOFs (spatial patterns) tell you? What do the PC time series for the EOFs tell you? What do you think of the method for estimating the effective sample size (Nstar)? Can you propose an alternative way to estimate Nstar? Do you get the same results using eigenanalysis and SVD? If you got a different sign do you think that is meaningful?**

*The EOFs tell us that despite removing the mean and seasonal cycle, we still get some very strong patterns showing up. The first EOF shoes a solid warming signal along the North American coast, and an opposite signal in the middle of the Pacific. This signal seems to represent the Pacific Decadal Oscillation (PDO), specifically a positive PDO. The second EOF may be the North Pacific Gyre Oscillation (NPGO) or the Victoria mode.*

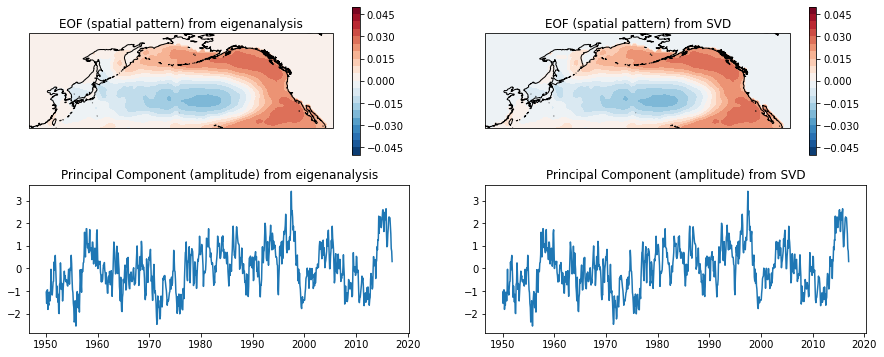
*The PC time series tell us that this is a red-noise time series, with a predictable nature.*

*As for the method to estimate the effective sample size, it should be sufficient. The equation used (Barnes Chapter 2, eq. 88) is a first-order autoregressive process, so this assumes that the time series is red. Here that is true, especially for the first two EOFs.*

*An alternative way to estimate Nstar would be to assume all the data is independent, but this is a bad assumption in this case. You could also use the Leith method shown below, which incorporates an e-folding time, but is essentially the same as eq. 88.*

**

*Regarding the difference between the eigenanalysis and SVD, they are nearly the same for the first EOF, but given opposite signs for the second EOF (same magnitude). The sign difference is likely not meaningful, as we have the same magnitude, so the differences come down to the math.*

**

*Graphical user interface

Description automatically generated*

*Graphical user interface

Description automatically generated*

*Graphical user interface

Description automatically generated*

**2) Save a copy of the notebook, rename it. Repeat the analysis but this time do not remove the seasonal cycle. What do you think you will see? Discus your results with your neighbor. How do the EOFs and PC change? Was removing the seasonal cycle from the data useful? What impacts does removing the seasonal cycle have on your analysis?**

*I think I will see the seasonal cycle play a very large role, and dwarf the other signals.*

*After not removing the seasonal cycle, my guess above was correct. The seasonal cycle explains roughly 90% of the variability, while the second EOF explains a mere 3%. The PCs look very cyclical (seasons). In all though, removing the season was useful because it gave us an idea of how large the seasonal cycle does play compared to all the other modes of variance. We may end up removing it anyway, but it’s good to known its influence.*

*Graphical user interface

Description automatically generated with low confidence*

*A picture containing text

Description automatically generated*

*A picture containing text, screen

Description automatically generated*

*Graphical user interface

Description automatically generated with medium confidence*

**3) Save a copy of the notebook, rename it. Repeat the analysis but this time detrend the data. Discus your results. How do the EOFs and PC change? Was detrending the data useful? What impacts does detrending have on your analysis?**

*Removing the trend and the seasonality, we do get some signals. There are at least 10 EOFS with at least 1% explained variability. The EOF results tamper off from the first (~26%) to the 10th (~1.7%), a lot less extreme compared to leaving the seasons in. Essentially here, we truly isolate the PDO and NPGO as the trend from (potentially) climate change is removed.*

*A picture containing text, picture frame

Description automatically generated*

*Graphical user interface

Description automatically generated*

*Graphical user interface

Description automatically generated*

*Graphical user interface

Description automatically generated*

**4) Save a copy of the notebook, rename it. Repeat the analysis but this time do not apply the cosine weighting. Discus your results. How do the EOFs and PC change? Was cosine weighting the data useful? What impacts does cosine weighting have on your analysis? What are examples of analyses where cosine weighting would be more/less important to do?**

*We get essentially the same results as the previous notebook. This is due to the cosine weighting occurring before the standardization, so the effect is unseen. It would definitely be useful in an analysis where standardization isn’t necessary, or if there weas a way to cosine weight after you standardize, but I’m not sure a method to do this exists.*

**4) Save a copy of the notebook, rename it. Repeat the analysis but this time do not standardize the data (i.e., comment out dividing by standard deviation). Discus your results. How do the EOFs and PC change? Was standardizing the data useful? What impacts does standardizing the data have on your analysis?**

*Overall, not standardizing the data just makes the values/plots in the actual physical units (unless I am doing something wrong). I am not seeing a large difference in the actual EOF/PC plots, and I think that is okay since we aren’t really comparing this data to anything else. If we were, we would definitely want to standardize it.*

*I think it was useful overall to do the standardization process, and it was helpful in the other scenarios above, but here I still get the gist of what we are looking at.*