Today’s plan….

1. EOF Analysis – What is it?
2. Example of EOF analysis – Sea Level Pressure Northern Hemisphere from Hannachi et. al. 2007
3. Method #1: EOF via Eigenanalysis
4. Method #2: EOF via Singular Value Decomposition
5. Apply EOF analysis to a classic dataset in machine learning and data analysis not from our field.

What is EOF Analysis?

- EOF stands for *Empirical Orthogonal Functions*
- PCA stands for *Principle Component Analysis*
- EOF analysis and PCA are practically the same thing
- PCA is more generally used, and just implies splitting a matrix into sets of linearly uncorrelated variables, called principle components
- EOF analysis normally refers to finding the principle components that maximize the variance explained (so a subset of all PC's)

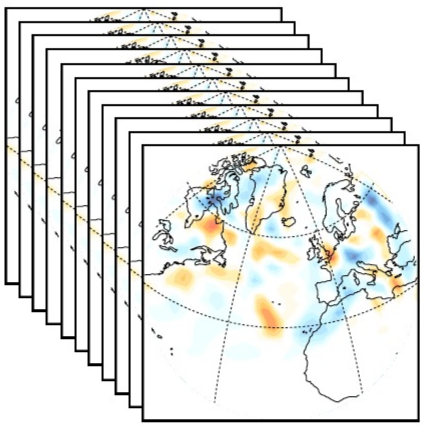
- **EOF analysis seeks structures that explain the maximum amount of variance in a two-dimensional dataset. ONE dimension is the <u>structure dimension</u> and the SECOND dimension is the <u>sampling dimension.</u>**

- **In many ATOC-relevant applications – the structure dimension is a spatial field (e.g., lat,lon collapsed into a single vector), while the sampling dimension is time. EOF analysis produces structures (called "EOFs", e.g., EOF1, EOF2, EOF3 etc.) that directly correspond to the sampling dimension (called the "Principal Components", PC1, PC2, PC3 etc.).**

What is EOF Analysis?

Visualizing EOFs & PC time series

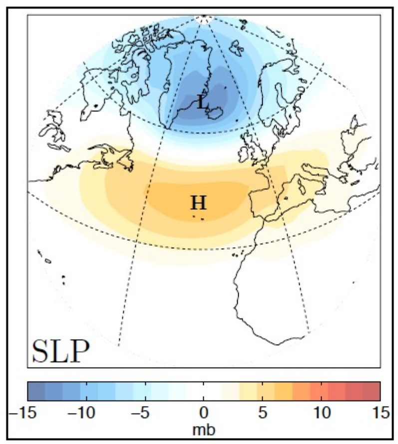
daily maps



X

EOF^T

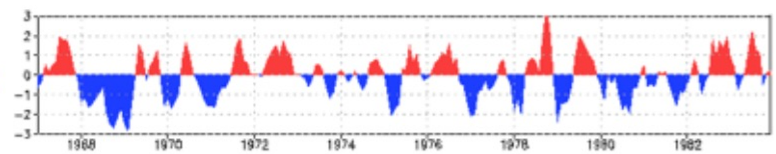
EOF1 pattern explains the most variance



=

PC time series

PC1 = How much each day looks like EOF1?



MxN

Nx1

Mx1

M = sample direction, time, number of days (timeseries)

N = structure dimension, spatial location (map)

Let's Look at a Classic Example: EOF Analysis of Northern Hemisphere Sea Level Pressure During Winter Months

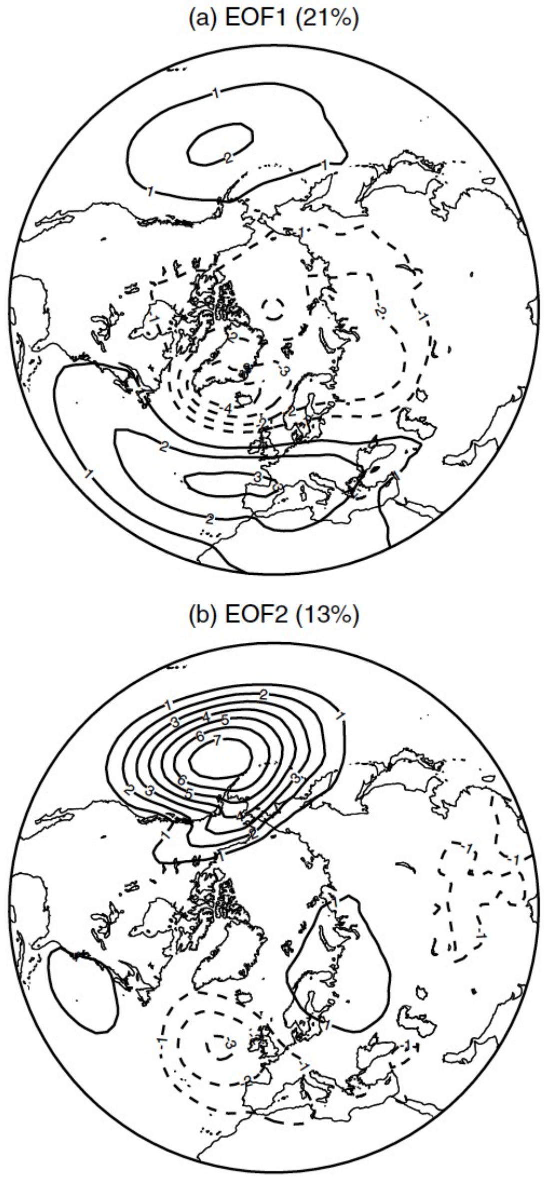


Figure 2. The first (a) and the second (b) EOFs of DJF monthly mean SLP. Positive contours solid, negative contours dashed. EOFs have been multiplied by 100.

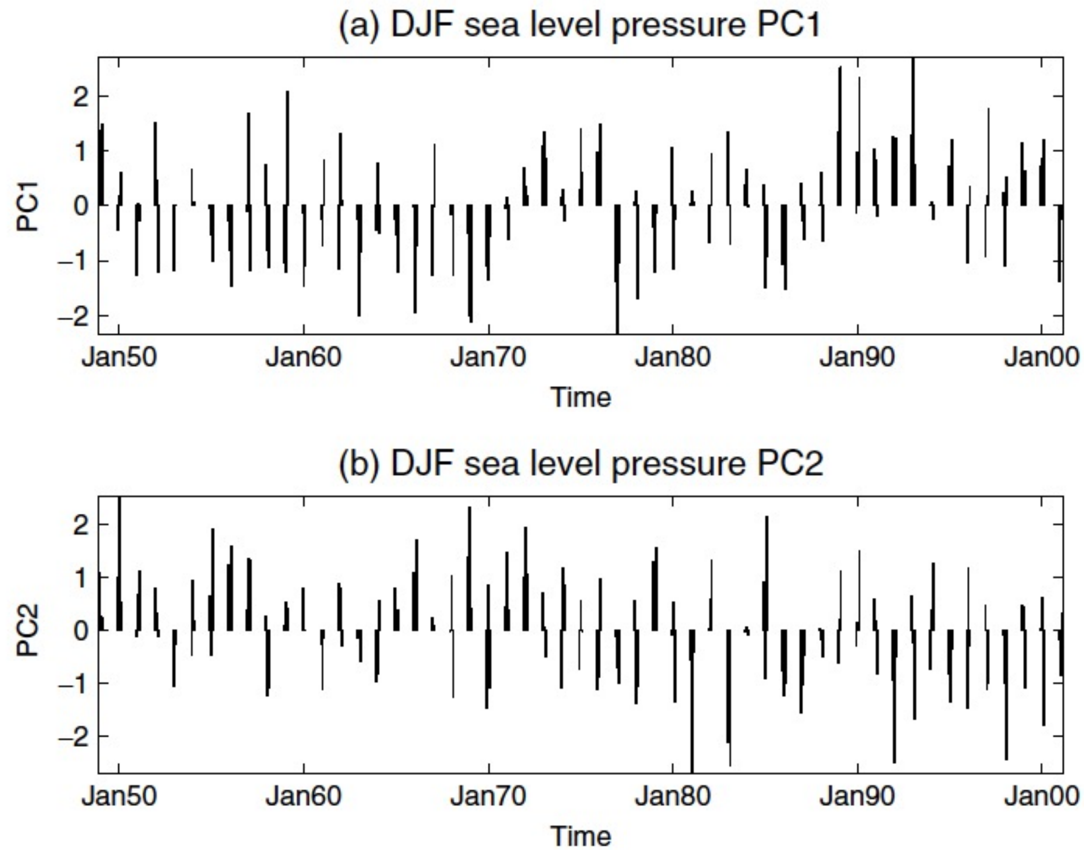


Figure 3. The leading two scaled PCs corresponding to the leading two EOFs of Figure 2.

Figures from Hannachi et al. 2007

Why only plot the first two EOFs? How do I test the statistical significance of the EOF patterns? (more on this next lecture…)

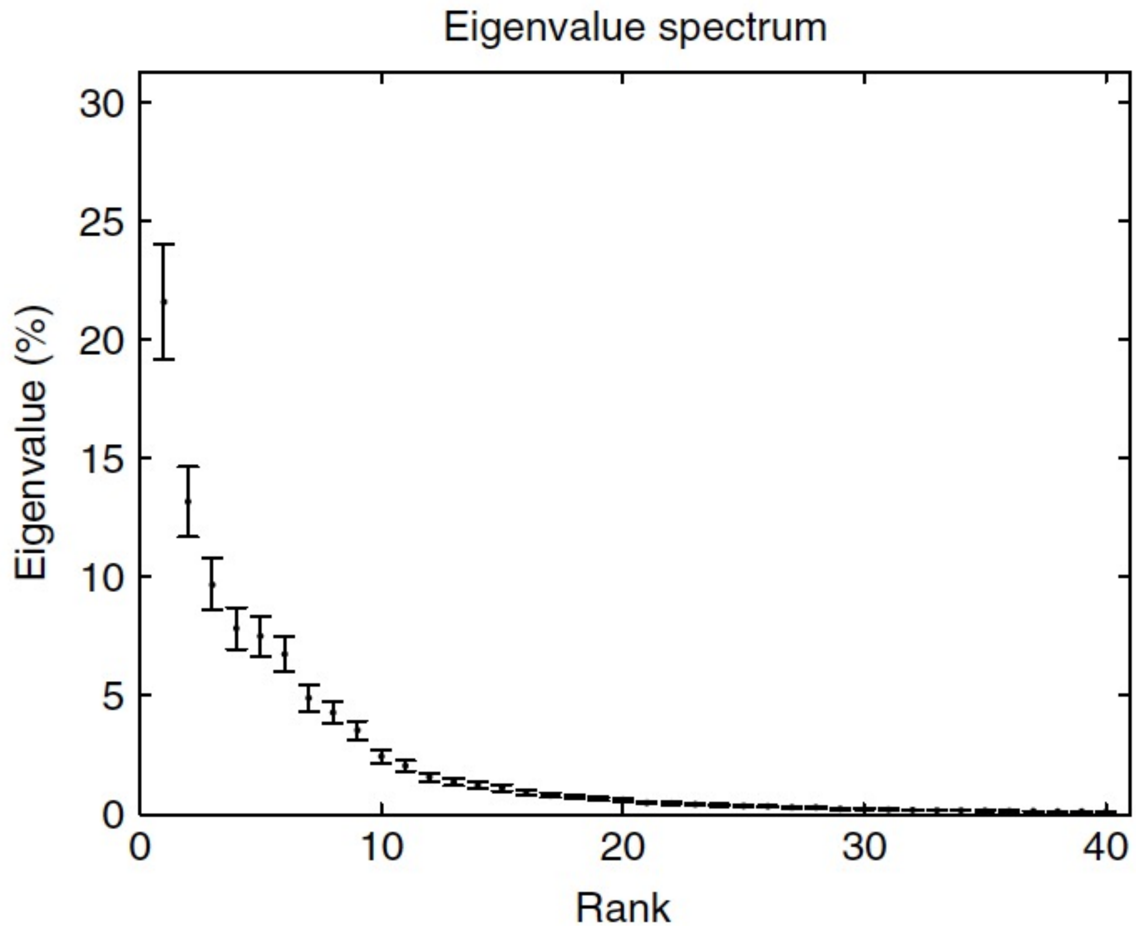


Figure 1. Spectrum, in percentage, of the covariance matrix of winter monthly (DJF) SLP. Vertical bars show approximate 95% confidence limits given by the rule of thumb (18). Only the leading 40 eigenvalues are shown.

Figure from Hannachi et al. 2007 – paper is in class google drive

What are the inputs to EOF analysis?

Very Generally Stated -- Data as Two-Dimensional Matrices

Imagine that you have a data set that is two-dimensional. The easiest example to imagine is a data set that consists of observations of several variables at one instant of time, but includes many realizations of these variable values taken at different times. The variables might be temperature and salinity at one point in the ocean taken every day for a year. Then you would have a data matrix that is 2 by 365; 2 variables measured 365 times. So one dimension is the variable and the other dimension is time. Another example might be measurements of the concentrations of 12 chemical species at 10 locations in the atmosphere. Then you would have a data matrix that is 12x10 (or 10x12). One can imagine several possible generic types of data matrices.

a) A space-time array: Measurements of a single variable at M locations taken at N different times, where M and N are integers.

b) A parameter-time array: Measurements of M variables (e.g. temperature, pressure, relative humidity, rainfall, . . .) taken at one location at N times.

c) A parameter-space array: Measurements of M variables taken at N different locations at a single time.

Reminder about your data matrix ...

So we can visualize a two-dimensional data matrix **X** as follows:

$$\mathbf{X} = \begin{array}{cc} & N \\ M & \left[\quad\quad\quad \right] \end{array} = X_{i,j} \text{ where } i = 1, M;\ j = 1, N$$

Where M and N are the dimensions of the data matrix enclosed by the square brackets, and we have included the symbolic bold **X** to indicate a matrix, the graphical box that is *MxN* to indicate the same matrix, and finally the subscript notation $X_{i,j}$ to indicate the same matrix. We define the transpose of the matrix by reversing the order of the indices to make it an NxM matrix.

$$\mathbf{X}^T = \begin{array}{cc} & M \\ N & \left[\quad\quad \right] \end{array} = X_{j,i} \text{ where } i = 1, M;\ j = 1, N$$

With a data matrix (X) dimensioned with M rows (sampling dimension) and N columns (structure dimension) -

Typically you will find the covariance matrix (C, N rows by N columns) as follows:

$$\mathbf{C} = \frac{1}{M}\mathbf{X}^{\mathsf{T}}\mathbf{X} \qquad (40)$$

$$\mathbf{X}^T\mathbf{X} = \begin{array}{cccccc} & M & N & & N & \\ N & \begin{bmatrix} \quad & \quad \\ & \\ & \end{bmatrix} & \begin{bmatrix} \quad & \quad & \quad \\ & & \end{bmatrix} & M = & \begin{bmatrix} \quad & \quad & \quad \\ & & \\ & & \end{bmatrix} & N \end{array}$$

Reminder about the covariance matrix (C) of the data matrix (X).

Recall that the covariance matrix of $\mathbf{X}$ can be computed in one of two ways, either,

$$\mathbf{C} = \frac{1}{M}\mathbf{X}^T\mathbf{X} \rightarrow [N \times N] \tag{56}$$

$$\text{or} \tag{57}$$

$$\mathbf{C} = \frac{1}{N}\mathbf{X}\mathbf{X}^T \rightarrow [M \times M] \tag{58}$$

A word to the wise:

Make sure you understand M and N before you start.
Which one is row, which one column?
Which one is sampling dimension,
Which one is structure dimension?
What do you expect your EOF and PC dimensions to be?
(Check them).

<u>EOF Analysis – We will cover two different methods that give identical results. (See Barnes 1.4.1 for the equivalence proof).</u>

Method #1: EOF Analysis via Eigenanalysis of the Covariance Matrix

$$\mathbf{C}\mathbf{e}_i = \lambda_i \mathbf{e}_i$$

$$\mathbf{C}\mathbf{E} = \mathbf{E}\boldsymbol{\Lambda}$$

Method #2: EOF Analysis via Singular Value Decomposition (SVD)

$$\mathsf{X} = \mathsf{U}\mathsf{\Sigma}\mathsf{V}^{\mathsf{T}}$$

Eigenanalysis of the sample covariance matrix

Principle Component Analysis consists of an eigenanalysis of $\mathbf{C}$. That is, any symmetric matrix $\mathbf{C}$ can be decomposed in the following way:

$$\begin{aligned} \mathbf{C}\mathbf{e}_i &= \lambda_i \mathbf{e}_i \qquad (31) \\ \mathbf{C}\mathbf{E} &= \mathbf{E}\mathbf{\Lambda} \qquad (32) \end{aligned}$$

where $\mathbf{E}$ is the matrix with eigenvectors $\mathbf{e}_i$ as its columns and $\mathbf{\Lambda}$ is the matrix with eigenvalues λ_i along its diagonal and zeros elsewhere.

The set of eigenvectors and associated eigenvalues represent a *coordinate transformation* into a coordinate space where $\mathbf{C}$ becomes diagonal.

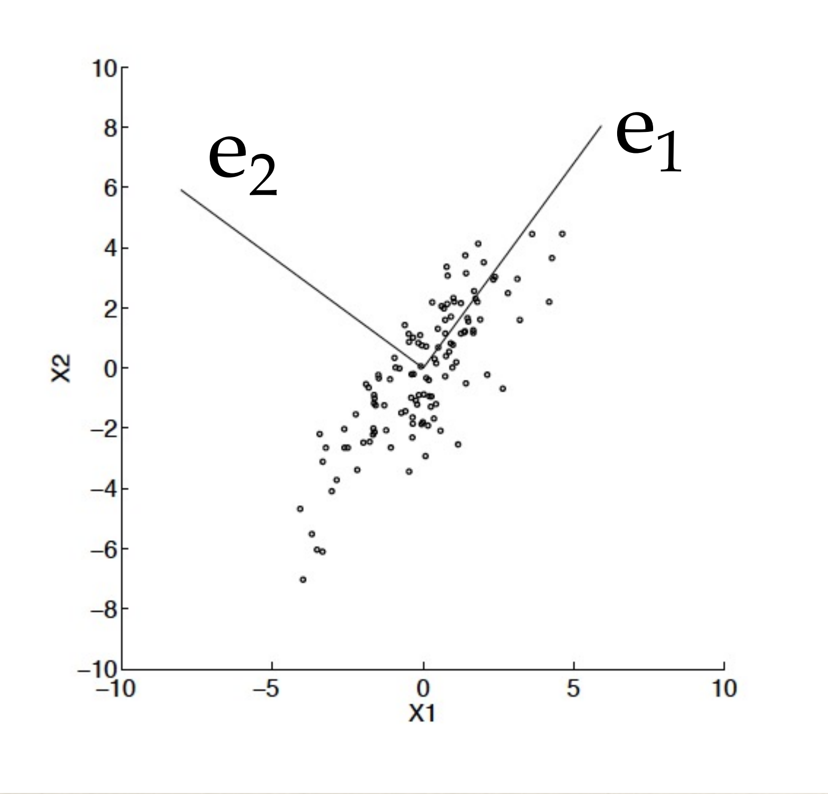
The eigenvectors are solved for in the following way:

$$\begin{aligned} \mathbf{C}\mathbf{e}_i &= \lambda_i \mathbf{e}_i \qquad (34) \\ (\mathbf{C}\mathbf{e}_i - \lambda_i \mathbf{e}_i) &= 0 \qquad (35) \\ (\mathbf{C} - \lambda_i \mathbf{I})\mathbf{e}_i &= 0 \qquad (36) \\ & \qquad (37) \end{aligned}$$

A 2D Example (From Hartmann)

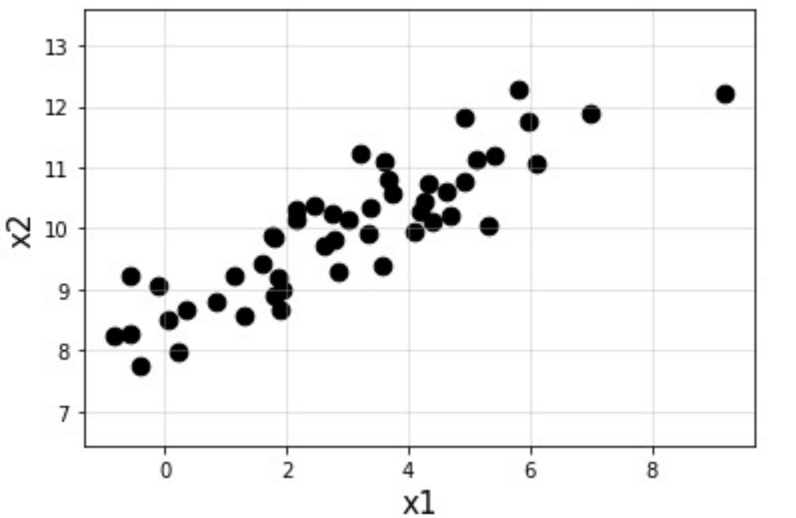
Two-Dimensional Example:

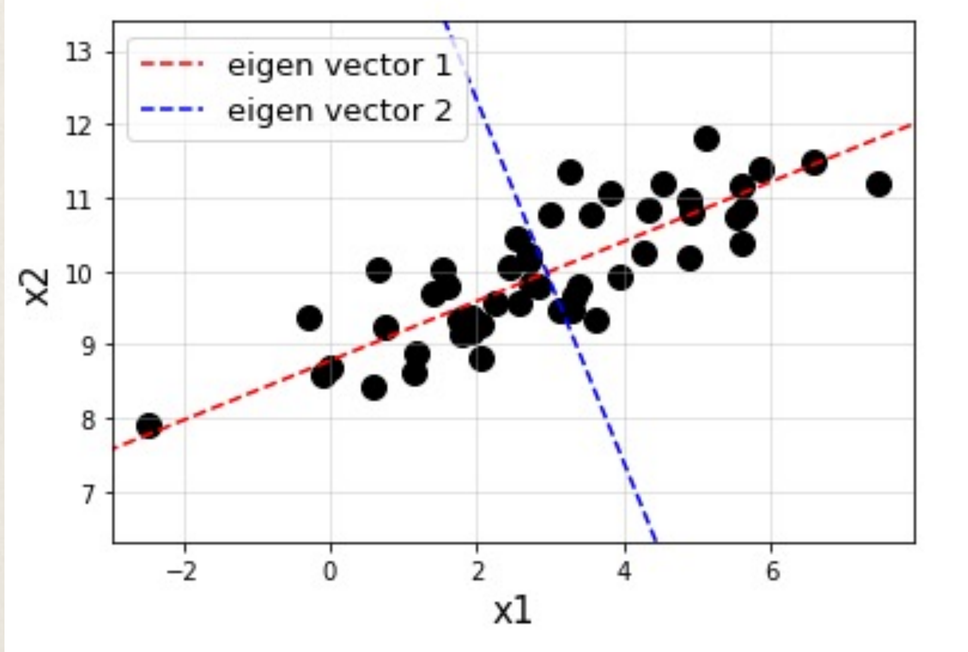
It is simplest to visualize EOFs in two-dimensions as a coordinate rotation that maximizes the efficiency with which variance is explained. Consider the following scatter plot of paired data (x_1, x_2). The eigenvectors are shown as lines in this plot. The first one points down the axis of the most variability, and the second is orthogonal to it.



Why would you do eigenanalysis of the covariance matrix? A 2D example…

```
Covariance Matrix
[[4.   1.6]
 [1.6 1. ]]
```






```
eigen values:
[4.45738215 0.22651675]

eigen vectors:
[[ 0.9266011  -0.37604575]
 [ 0.37604575  0.9266011 ]]
```

Try code in eigenanalysis_example.ipynb

GROUP WORK

- **eigenanalysis_example.ipynb**
- **Discuss – What is the covariance matrix? What are the eigenvalues and the eigenvectors?**

EOF Analysis via Eigenanalysis of the Covariance Matrix

What is the goal?
Find the eigenvector (e_1) that explain most variance

The goal is to decompose $\mathbf{X}$ into orthogonal eigenvectors that explain the most variance of $\mathbf{X}$. That is, we want to *maximize the resemblance* of $\mathbf{e}_1$ (dimensions $[N \times 1]$; that is space) to the data, that is, find the $\mathbf{e}_i$ that explains the most variance.

What are the rules?

1. the projection of the $\mathbf{e}_1$ onto the data is measured by the inner product of $\mathbf{e}_1$ with $\mathbf{X}$ (this is the "resemblance')
2. the projection is squared to ensure that a positive or negative resemblance are counted the same (as in the case of linear regression where we square the errors)
3. to ensure that the projection (measure of resemblance) is independent of the number of observations, we should divide by M
4. to make the projection (measure of resemblance) independent of the magnitude of $\mathbf{e}_1$, we will normalize it to have unit length, that is, $\mathbf{e}_1^T \mathbf{e}_1 = 1$. Otherwise the projection can be made arbitrarily large by increasing the length of $\mathbf{e}_1$

EOF Analysis via Eigenanalysis of the Covariance Matrix

What is the goal?
Find the eigenvector (e_1) that explain most variance

Accomplish the goal by solving the following eigenvalue problem:

$$\mathbf{C}\mathbf{e}_1 = \mathbf{e}_1\lambda_1 \tag{42}$$

Where λ_1 is an eigenvalue (measure of variance explained), e_1 is an eigenvector (direction in which the variance is explained), and C is the covariance matrix of your data matrix X

$$\mathbf{C} = \frac{1}{M}\mathbf{X}^{\mathsf{T}}\mathbf{X} \tag{40}$$

EOF Analysis via Eigenanalysis of the Covariance Matrix

Generalize to find all N eigenvalues and eigenvectors

EOF (or PCA) analysis consists of an eigenvalue analysis of these dispersion matrices. Any symmetric matrix **C** can be decomposed in the following way through a diagonalization, or eigenanalysis.

$$\mathbf{C}\mathbf{e}_i = \lambda_i \mathbf{e}_i \tag{4.1}$$

$$\mathbf{C}\mathbf{E} = \mathbf{E}\mathbf{\Lambda} \tag{4.2}$$

Where **E** is the matrix with the eigenvectors $\mathbf{e}_i$ as its columns, and $\mathbf{\Lambda}$ is the matrix with the eigenvalues λ_i, along its diagonal and zeros elsewhere.

Note: ***E, C, and Λ all have dimensions NxN. C the covariance matrix (also called the dispersion matrix).***

Above from Hartmann -- Barnes goes through the matrix math and arrives at the same equation:

$$\mathbf{E}^{\mathsf{T}}\mathbf{C}\mathbf{E} = \mathbf{\Lambda} \Rightarrow \mathbf{E}\mathbf{E}^{\mathsf{T}}\mathbf{C}\mathbf{E} = \mathbf{E}\mathbf{\Lambda} \Rightarrow \mathbf{C}\mathbf{E} = \mathbf{E}\mathbf{\Lambda} \tag{43}$$

using $\mathbf{E}^{\mathsf{T}}\mathbf{E} = \mathbf{E}\mathbf{E}^{\mathsf{T}} = \mathbf{I}$.

Notes on eigenvectors and eigenvalues obtained by "eigenizing" your covariance matrix...

- the 1st eigenvector corresponds to the vector that explains the most variance in $\mathbf{X}$ and has the largest eigenvalue
- the 2nd eigenvector corresponds to the vector that explains the second most variance in $\mathbf{X}$ and has the 2nd largest eigenvalue
- The eigenanalysis of the covariance matrix transforms $\mathbf{C}$ into a different coordinate system where the "new" dispersion matrix is diagonal ($\mathbf{\Lambda}$).
- In this new coordinate space, all of the variance is along the diagonal since the different vector are orthogonal. Thus, the fraction of variance explained by the j^{th} eigenvector is the corresponding eigenvalue λ_j divided by the sum of all of the eigenvalues, that is $\lambda_j / \sum_i \lambda_i$.

Eigenvectors are orthogonal to each other – Barnes proves it!

Finally, to confirm that different eigenvectors are orthogonal, we suppose we have two eigenvectors $\mathbf{e}_j$ and $\mathbf{e}_k$. Thus, we know that

$$\mathbf{C}\mathbf{e}_j = \mathbf{e}_j \lambda_j \tag{44}$$

and

$$\mathbf{C}\mathbf{e}_k = \mathbf{e}_k \lambda_k \tag{45}$$

Multiplying the j^{th} equation by $\mathbf{e}_k^T$ and taking the transpose, then multiplying the k^{th} by $\mathbf{e}_j^T$ and subtracting the two equations leads to

$$\mathbf{e}_j^T \mathbf{C}^T \mathbf{e}_k - \mathbf{e}_j^T \mathbf{C}\mathbf{e}_k = (\lambda_j - \lambda_k)\mathbf{e}_j^T \mathbf{e}_k \tag{46}$$

Since $\mathbf{C}$ is symmetric, $\mathbf{C} = \mathbf{C}^T$ and so the left-hand side is zero. Thus,

$$\mathbf{e}_j^T \mathbf{e}_k = 0 \tag{47}$$

$$\text{unless} \tag{48}$$

$$\lambda_j = \lambda_k \tag{49}$$

Therefore, the eigenvectors are orthogonal if the eigenvalues are distinct.

Finding the principal components (z_1, z_2, … in the matrix Z).

We define

$$\mathbf{Z} = \mathbf{X}\mathbf{E} \tag{50}$$

Then it follows that

$$\mathbf{X} = \mathbf{Z}\mathbf{E}^T \tag{51}$$

$$\text{since} \tag{52}$$

$$\mathbf{E}\mathbf{E}^T = \mathbf{I} \tag{53}$$

Thus, we can get our original $\mathbf{X}$ data back by multiplying $\mathbf{Z}\mathbf{E}^T$.

$\mathbf{Z}$ are known as the principle components (PC's), and are length M. These values tell you how much a given sample looks like a particular EOF structure.

Projection 101: How do we get the first principal components (z_1 length M) from the first eigenvectors (e_1 length N) and the original data (matrix X – dimensioned M rows by N columns)?

As an example, suppose you have a set of orthogonal and normalized eigenvectors. The first one might looks like this:

$$\mathbf{e}_1 = \begin{bmatrix} e_{11} \\ e_{21} \\ e_{31} \\ e_{41} \\ \cdots \end{bmatrix}$$

Next, *project* $\mathbf{e}_1$ onto the original data to get the amplitude of this eigenvector at each time step. This is done in the following way:

$$\begin{bmatrix} x_{11} & x_{12} & x_{13} & \dots & x_{1n} \\ x_{21} & x_{22} & x_{23} & \dots & x_{2n} \\ x_{31} & x_{32} & x_{33} & \dots & x_{3n} \\ \vdots & \vdots & \vdots & & \vdots \\ x_{m1} & x_{m2} & x_{m3} & \dots & x_{mn} \end{bmatrix} \begin{bmatrix} e_{11} \\ e_{21} \\ e_{31} \\ e_{41} \\ \dots \\ e_{n1} \end{bmatrix} = \begin{bmatrix} z_{11} \\ z_{21} \\ z_{31} \\ z_{41} \\ \dots \\ z_{m1} \end{bmatrix}$$

That is

$$\mathbf{XE} = \mathbf{Z} \tag{55}$$

Note, you do not want to project $\mathbf{e}$ onto the covariance matrix $\mathbf{C}$, but rather, you want to project it onto the original data.

Getting “back to your original data”

One reconstructs the original $\mathbf{X}$ data from this new coordinate system by combining the eigenvectors in linear combinations with amplitudes of the PCs. That is, a given observation at time t and location i, written as $\mathbf{x}_i(t)$ can be found using a linear combination of the EOFs multiplied by their PC time series value

$$x_i(t) = \sum_{j=1}^{N} e_{ij} z_j(t) \tag{54}$$

Summary

1.3.5 General summary of finding EOFs by eigenanalyzing the covariance matrix for $\mathbf{X} = [\mathrm{M} \times \mathrm{N}]$

1. subtract the mean values along the sampling dimension of $\mathbf{X}$
2. unless there are issues with data size and computation time, define $\mathbf{C}$ along the sampling dimension so you are left with a matrix that is the space dimensions, i.e.,

$$\mathbf{C} = \frac{1}{\mathrm{M}}\mathbf{X}^{\mathrm{T}}\mathbf{X} \rightarrow [\mathrm{N} \times \mathrm{N}], \text{ if M is the sampling dimension} \tag{62}$$

$$\text{or} \tag{63}$$

$$\mathbf{C} = \frac{1}{\mathrm{N}}\mathbf{X}\mathbf{X}^{\mathrm{T}} \rightarrow [\mathrm{M} \times \mathrm{M}], \text{ if N is the sampling dimension} \tag{64}$$

3. eigenanalyze $\mathbf{C}$ by diagonalizing the matrix (e.g. "eig" in Matlab, "EIGENQL" in IDL)
4. the first EOF is the eigenvector corresponding to the largest eigenvalue
5. to find the PC's, project the data onto the set of eigenvectors (PC's should be the length of your sampling dimension)
6. the fraction of variance explained by the i^{th} EOF pair is $\frac{\lambda_i}{\sum_{j=1} \lambda_j}$

EOF Analysis via Singular Value Decomposition (SVD)

It can be shown that any $\mathsf{M} \times \mathsf{N}$ matrix can be decomposed into the product of 3 matrices each with special qualities:

$$\mathbf{X} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^{\mathsf{T}} \tag{65}$$

where

- $\mathbf{U}$ is $\mathsf{M} \times \mathsf{M}$
- $\mathbf{\Sigma}$ is $\mathsf{M} \times \mathsf{N}$
- $\mathbf{V}$ is $\mathsf{N} \times \mathsf{N}$
- $\mathbf{U}$ and $\mathbf{V}$ are orthogonal
- $\mathbf{\Sigma}$ is a diagonal matrix

EOF Analysis via Singular Value Decomposition (SVD)

It can be shown that any $M \times N$ matrix can be decomposed into the product of 3 matrices each with special qualities:

$$\mathbf{X} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}^{\mathsf{T}} \tag{65}$$

If we assume the M dimension is space and the N dimension is time (this is to stay consistent with Prof. Hartmann's notes), something even more exciting follows:

- the columns of $\mathbf{U}$ are the *eigenvectors of* $\mathbf{X}\mathbf{X}^{\mathsf{T}}$
- the columns of $\mathbf{V}$ (rows of $\mathbf{V}^{\mathsf{T}}$) are the *eigenvectors of* $\mathbf{X}^{\mathsf{T}}\mathbf{X}$, or, in other words, are the corresponding PCs
- the r *singular values* along the diagonal of $\boldsymbol{\Sigma}$ are proportional to the square roots of the nonzero eigenvalues of both $\mathbf{X}\mathbf{X}^{\mathsf{T}}$ and $\mathbf{X}^{\mathsf{T}}\mathbf{X}$, in other words, $s_i = \sqrt{\lambda_i}$
- to obtain the variance explained by each EOF/PC pair, just square the elements of $\boldsymbol{\Sigma}$

Note! If you have time along the M dimension and space along the N dimension, this just means that the columns of $\mathbf{V}$ are the EOFs and the columns of $\mathbf{U}$ are the PCs. Everything else remains identical.

Summary

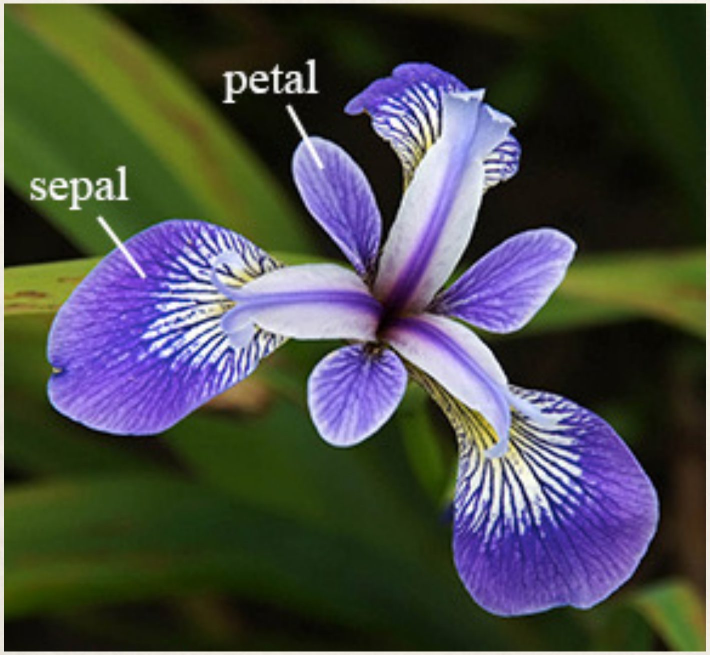
1.4.2 General summary of finding EOFs with SVD for $\mathbf{X} = [M \times N]$

1. subtract the mean values along the sampling dimension of $\mathbf{X}$
2. decompose $\mathbf{X}$ using built-in SVD routine
3. if M is the sample space (time), the EOFs (spatial patterns) correspond to the columns of $\mathbf{V}$ and the PCs (time series) to the columns of $\mathbf{U}$
4. if N is the sample space (time), the EOFs (spatial patterns) correspond to $\mathbf{U}$ and the PCs (time series) to $\mathbf{V}$
5. the fraction of variance explained by the i^{th} EOF pair is $\frac{\sigma_i^2}{\sum_{j=1} \sigma_j^2}$

Let’s do an example using an ipython notebook. It’s pretty easy to implement because python does all the matrix math for you ☺

We are going to use a “classic dataset”, not from our field but from biology and machine learning.

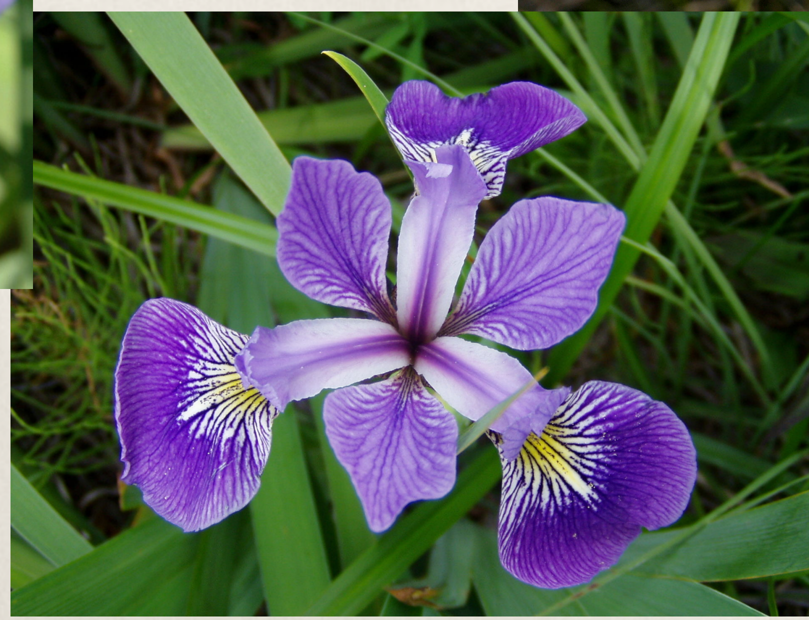
The data are 50 samples each from 3 species of Iris (Iris setosa, Iris virginica and Iris versicolor). Four traits were measured for every flower: the length and the width of the sepals and petals, in centimetres. Based on these 4 traits, the scientist working with the data (Fisher, 1936) investigated if it was possible to distinguish the Iris species from each other.



Let’s see what we find in an iypthon notebook using EOF/PCA analysis!!

Null Hypothesis – All three Iris species are the same based on measurements of the length and the width of their sepals and petals.

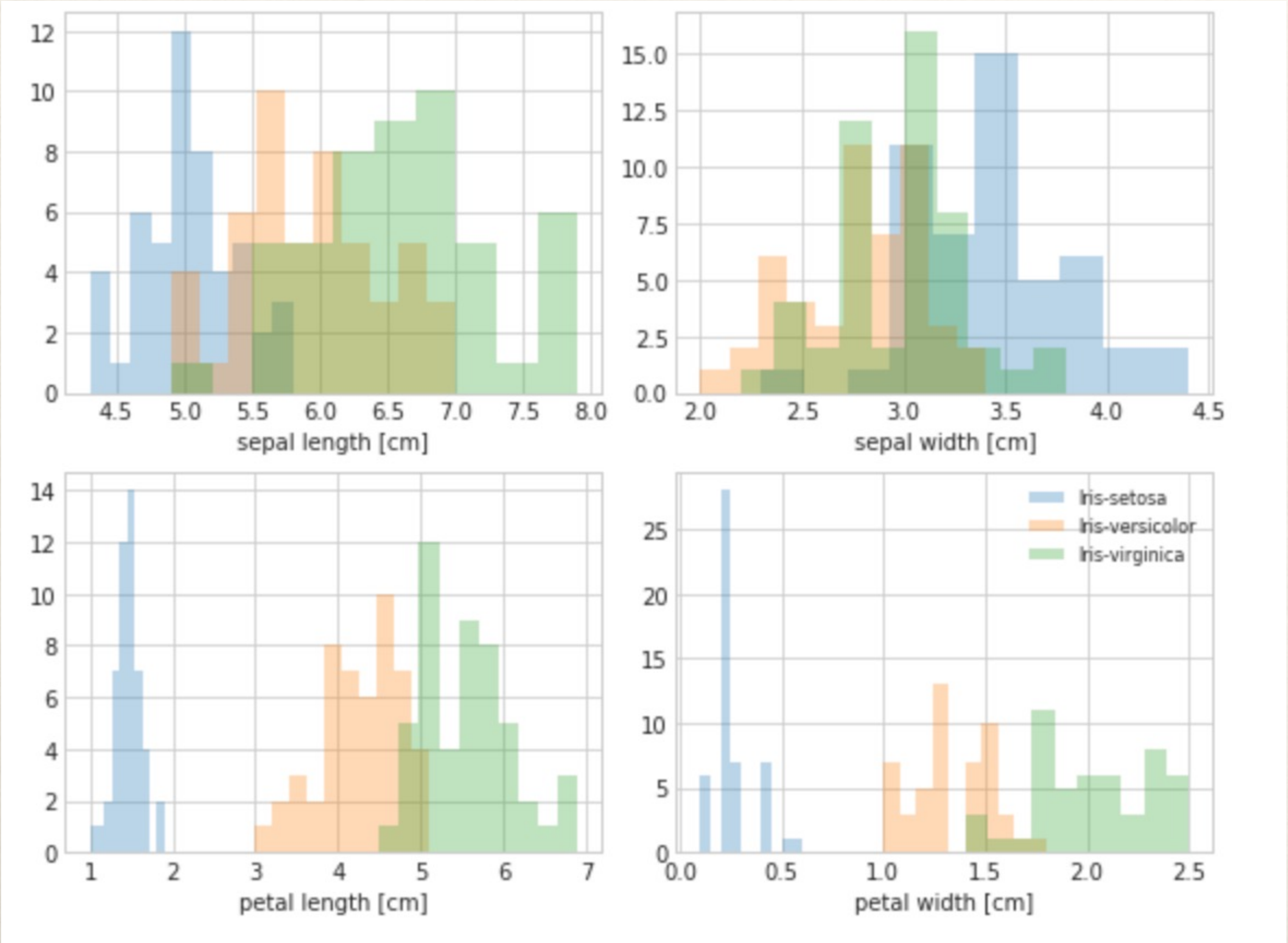
Iris setosa



Iris versicolor

Iris virginica

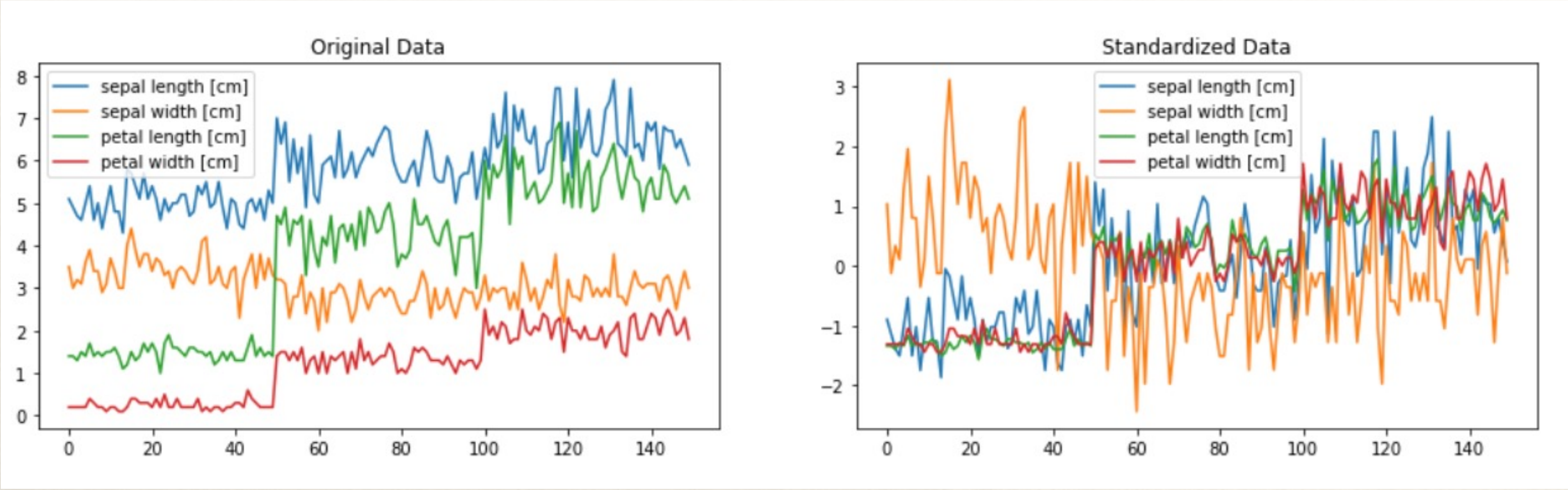
LOOK AT YOUR DATA
Histograms of Flower Measurements for the Different Iris



LOOK AT YOUR DATA

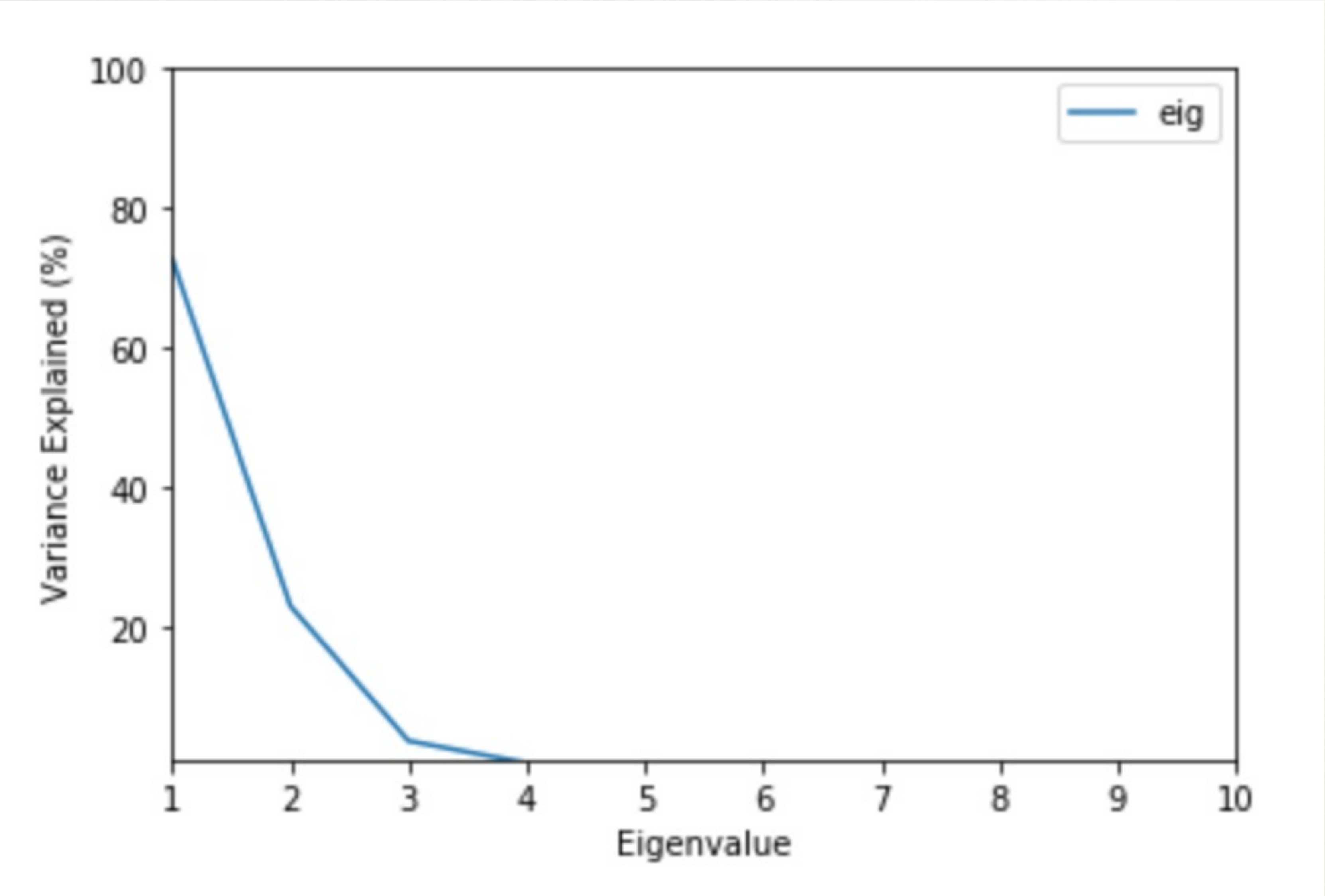
Sampling dimension = Iris flower (150) – first 50 samples from Iris setosa, next 50 from Iris virginica and final 50 from Iris versicolor

Structure dimension = Flower measurement (4)



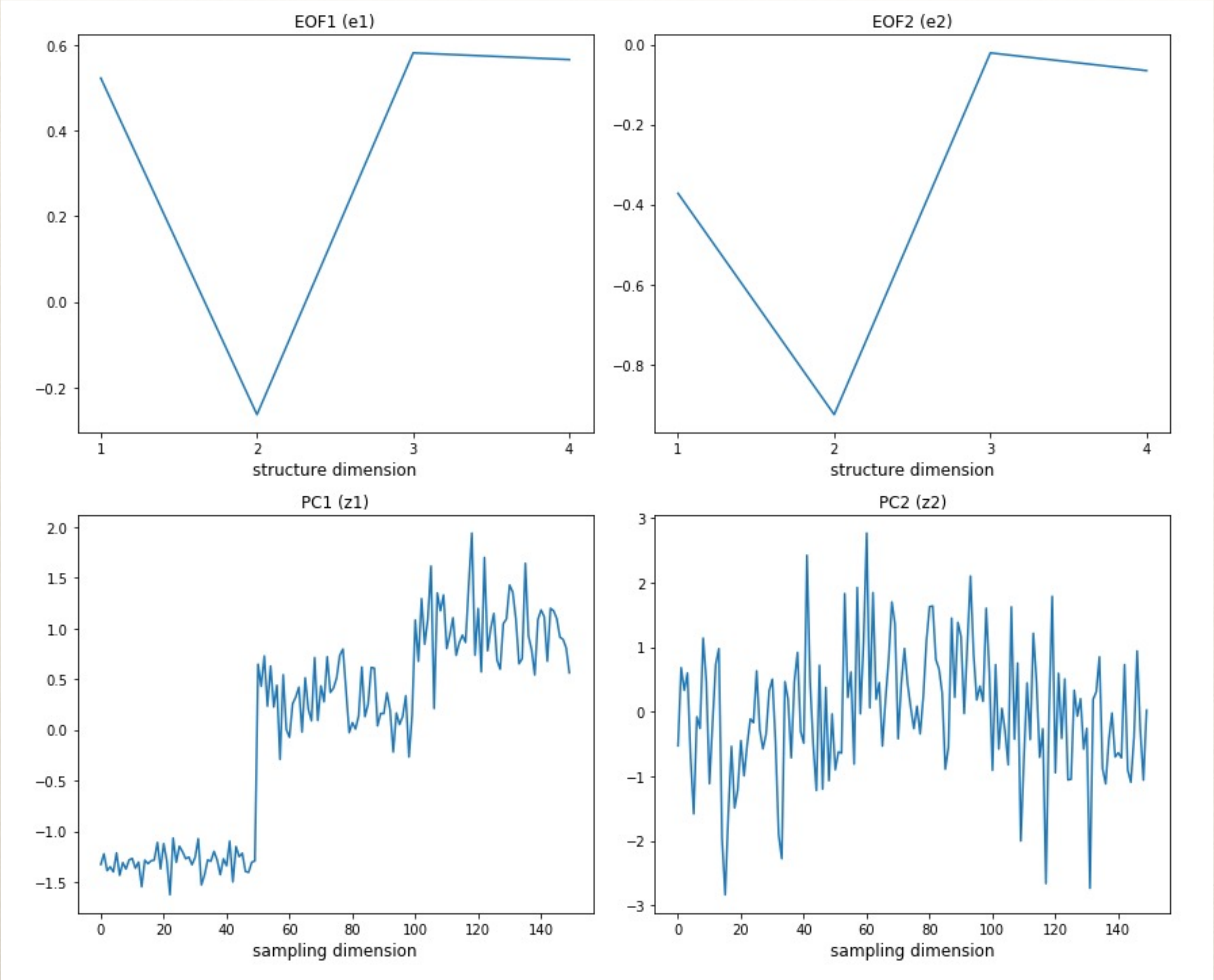
Notes: 1) Standardized data has been divided by the standard deviation across the structure dimension. 2) The mean across the structure dimension has also subtracted (i.e., we are working with the anomaly).

Eigenvalues and Variance Explained



Variance explained by first two eigenvalues: 95.8%

Plot the first two eigenvectors and principal components, Discuss.
Remember: first 50 samples from Iris setosa, next 50 from Iris virginica and final 50 from Iris versicolor.

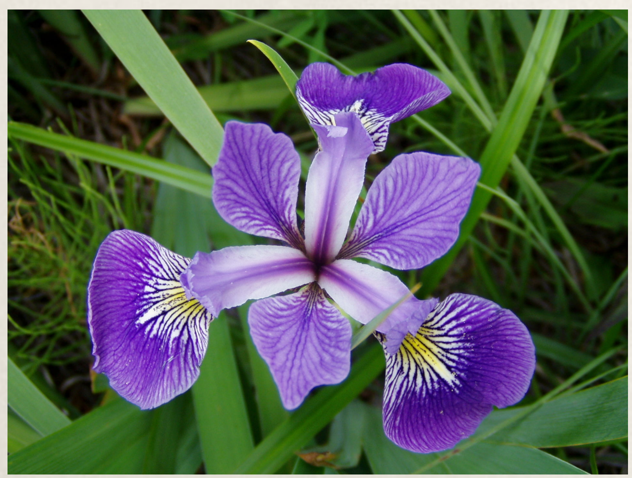


GROUP WORK

Check out the ipython code: eof_example_iris.ipynb
Code employs both eigenanalysis of the covariance matrix and SVD!

Iris virginica

Iris setosa



Iris versicolor