Lisanne Blok
Undergraduate Research Opportunity Placement
Supervisor: Yves Plancherel
24 June 2022

Jasmine for beginners

A report of completion of the UROP

Ceda, EFGF, Jasmine and Jupyter notebooks

Before starting, set up:

- Jasmine login
- Jasmine Portal account log in
- CEDA login

Set up SSH key

Go to home directory in terminal and create a new SSH key

```
ssh-keygen -t rsa -b 2048 -C "lisanne.blok19@imperial.ac.uk" -f ~/.ssh/
id_rsa_jasmin
```

Then get the public access key in the Jasmine login account:

```
cat ~/.ssh/id_rsa_jasmin.pub
```

On remote desktop

```
>ssh-keygen -f \\icnas1.cc.ic.ac.uk\lb1519
```

Run Jasmine

To connect Jasmine to the computer terminal, the private key will be used with an authentication agent.

```
eval `ssh-agent -s`
ssh-add ~/.ssh/id_rsa_jasmin
```

wget-20220621160228.sh

wget-20220621160228.sh

Connect to Jasmine to ceda archive by badc (atmospheric data) and neodc for Earth observation data. First like CEDA user account on Jasmin account portal.

```
ssh -A lblok@login1.jasmin.ac.uk
ssh lblok@sci1.jasmin.ac.uk
cd /badc/
```

Time series analysis

Discover patterns in the data using quantitative methods is time series analysis. Important that there is constant spacing (can use interpolation like linear interpolation). Make data stationary (randomness is constant over time, so get rid of trend whose variance changes over time).

Differencing and detrending can make data stationary.

- taking difference between point at time t and data at point t-1. So for periodicity seasonally, can take difference between point on day during year 1 and point at year -1.
- Detrend data: fit low order polynomial through data using regression and taking the difference between regression prediction and point -> new data series of residuals. Use a LOESS to fit the data: locally estimated scatterplot smoothing, so sequential regression on small pieces of data. Don't overfit data!

ADF and ZA statistics

Check for stationarity using Augmented Dicky-Fuller (ADF) or Zivot-Andrews (ZA) test. Use **statsmodel** library in Python, importing awfuller and zivot_adrews.

```
from statsmodels.tsa.stattools import adfuller
t_stat, p_value, _, _, critical_values, _  = adfuller(periodic_walk1,
autolag='AIC')
print(f'ADF Statistic: {t_stat:.2f}')
for key, value in critical_values.items():
     print('Critial Values:')
     print(f'   {key}, {value:.2f}')
print(f'\np-value: {p_value:.2f}')
print("Non-Stationary") if p_value > 0.05 else print("Stationary")
```

When ADF statistics is larger than critical values the random process is stationary. When p value is more than 5% we accept the null hypothesis of a trend -> non stationary.

```
from statsmodels.tsa.stattools import zivot_andrews
t_stat, p_value, critical_values, _, _ = zivot_andrews(periodic_walk1)
print(f'Zivot-Andrews Statistic: {t_stat:.2f}')
for key, value in critical_values.items():
     print('Critial Values:')
     print(f'   {key}, {value:.2f}')

print(f'\np-value: {p_value:.2f}')
print("Non-Stationary") if p_value > 0.05 else print("Stationary")
```

Additive model: Y[t] = T[t] + S[t] + e[t]

Multiplicative model: Y[t] = T[t] * S[t] * e[t]

The zivot andrews model shows much higher p-values in this case.

Seasonal decomposition

Detrend the data by using seasonal_decompose and seasonal_decompose from stats models module. The model option for seasonal decompose can be additive or multiplicative:

T(t) is the trend by applying a convolution filter to the data. The trend is removed from the series and the average of detrended series for each period is the returned seasonal component.

```
# add an exponential, but weakly increasing trend
t_half = 365*1
modulation = 0.01
new_trend0 = modulation*np.exp(np.divide(t,t_half))
trendy_element = pd.DataFrame(index=dti,data=new_trend0)

from statsmodels.tsa.seasonal import seasonal_decompose
sd = seasonal_decompose(trendy_walk1,period=365*24)
sd.plot()
plt.show()
```

Autocorrelation

Non-stationary dataset will show correlation between itself and the lagged version., so temporal patterns -> related. Autocorrelation can show if there are periodic patterns or any memory in the data. Use pandas autocorrelation:

```
pd.plotting.autocorrelation_plot(random_walk1)
plt.show()
```

Modelling time series data in time domain

Data = signal + noise. Use models to present stochastic processes.

- Autoregressive models take previous values to predict current values.
- Integrated models
- Moving average models

Marchov Chains are another model for finite number of states with varying probabilities.

Modelling time series data in frequency domain

Fourier transform and wavelet analysis. Give signal representation as a frequency spectrum. Use FFT (scipy).

frequency and magnitude aspect of change in climate extremes. Stochastic processes are time-dependent random variables representing climate variables with uncertain values. A stochastic process, X(T) is

X(T) = X_trend(T) + X_extr(T) + S(T) x X_noise(T), where we have T is continuous time, X trend and X extreme components and S is variability function scaled by the noise component. Extremes have a large absolute value and are usually rare. For a discrete time, t(i) is a time series. Climate memory can be taken into account, which is autocorrelation.

Models

- Peaks Over Threshold: Variable X(i) is above threshold u. Take event times and variable times. Approach for data whose magnitude is known with good accuracy but not the date. For time-independent threshold, take trend X_trend and S(i) variability into account.

$$\{T_{\text{ext}}(j), X'_{\text{ext}}(j)\}_{j=1}^{m} = \{T(i), X(i) | X(i) > u\}_{i=1}^{n}$$

Challenge is the placement of the threshold, against extreme and noise components. For time-dependent trend and noise variability, can replace time-constant threshold u by time-dependent function. This function can be estimated from calculating a running median. Estimate S(i) by absolute distances from the median (MAD). Thus threshold is median + z * MAD. Select z.

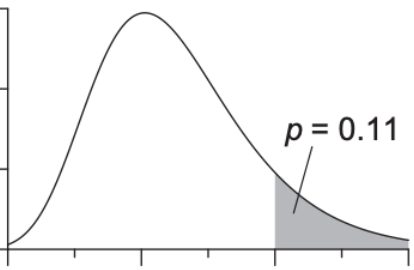
- Block Extremes: X'_ext(j) are input for Generalised Extreme Value distribution, explaining risk that an extreme size or length occurs. Extreme is taken from a block of independent observations (at least 100). Can select blocks over a year.

$$X'_{\text{ext}}(j) = \max\left(\{X(i)\} | T(i) \text{ within } j\text{th year of time series}\right),$$
$$T_{\text{ext}}(j) = j\text{th year of time series.}$$

The correct k depends on data size but the bigger the better (yearly over monthly).

Methods: stationary processes

- Stationary Processes means properties of distribution and persistence are constant over time. Can use GEV distribution in the form of block maxima, GP distribution for POT data.
- Predict risk and probability by consulting data distribution (GEV or GP). p is given by the area under the density function, so p = 1-F_GEV (x_p), where F_GEV is the distribution function depending on shape parameters, location and scale parameters. The value of x_p is the return level. The return period is the expected time span for observing an extreme value that exceeds return level, calculated by 1/p. So when x_p = 2.5, p = 0.11 and the event is a nine-year event.



- Tail behaviour, plying to extremal part of distribution of data, where $p \propto x_{ext}^{-\alpha}$ applying above threshold value $x_{ext} > u > = 0$. The heavy tail distribution model applies to many extreme values, thus also in GEV and GP distributions.

Methods: non-stationary processes

Non-stationarity means that properties of the data generating process vary over time. Describe processes by occurrence rate, lambda, defined as number of independent events per time unit. Write time dependence as lamda(T), called a non-stationary Poisson process. Estimation can be done by:

The mathematical formula for the kernel estimator is

$$\hat{\lambda}(T) = h^{-1} \sum_{j=1}^{m} K\left(\left[T - T_{\text{ext}}(j)\right]/h\right), \tag{3.16}$$

where h is the bandwidth and K is a kernel function. We take the Gaussian kernel, $K(y) = (2\pi)^{-1/2} \exp(-y^2/2)$.

- Kernel Occurrence Rate Estimation. Count number of events in shifted time window. Full curve to estimate occurrence rate.
- Bandwidth selection, h. Large means many data points contribute to estimation, so low standard errors. But increase of estimation bias. Small bandwidth leads to reduced bias but larger standard errors. This is the soothing problem: solve by using cross-validation bandwidth selector valled the minimiser of the function.
- Boundary bias correction. Observation interval at [t(1);t(n)], but value at interval boundaries. But half of kernel window cannot be collected. Can create pseudo data by extrapolation, extension outside by three bandwidths.
- Measure uncertainty by occurrence rate estimation. employ percentile-t confidence band, 9
-
- Parametric model for occurrence rate is Cox-Lewis Test. $\lambda(T) = exp(\beta_0 + \beta_1 T)$, a monotonic function. Increasing or decreasing trend of occurrence rate. Less flexible so use for increase or decrease not actual value.

Floods and droughts

Runoff is inferred via water stage and runoff calibration. Look at alteration of river geometry for calibration curve.

- Reservoirs
- Land use
- Evaluate average occurrence rate to look at risk and the 90% confidence band.
- Evaluate trends: regional warming, salt concentrations?

Estimate occurrence time by choosing a band width. Plot trend in occurrence.

Assess space, time and resolution (spatial and temporal).

Heatwaves and cold spells

Look at duration of exceedances and threshold. Calculate anomalies via day-wise subtraction of day-wise averages. Calculate temperature average over a time window of 5 days and 30 years. Define warm days (YX90p) as the seasonal/annual count of days when the calendar day 90th percentile is exceeded, but consider also absolute value. Also consider duration of minimum 6 days, by WSDI.

Action measure combines magnitude with duration. Integral of threshold exceedance curve over duration of three or more days.

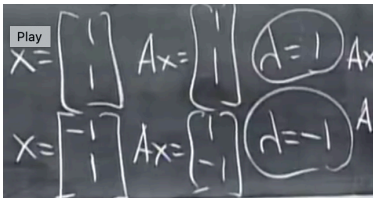
Can also include apparent temperature including humidity.

Index	Description	Definition	Unit
TXx	Warmest T_{max}	Seasonal/annual maximum of T_{max}	°C
TNx	Warmest T_{min}	Seasonal/annual maximum of T_{min}	°C
TX90p	Warm days	Seasonal/annual count of days when T_{max} > calendar day 90th percentile	d
TN90p	Warm nights	Seasonal/annual count of days when T_{min} > calendar day 90th percentile	d
TR	Tropical nights	Seasonal/annual count of days when T_{min} > 20 °C	d
WSDI	Warm spell duration index	Seasonal/annual count of days when T_{max} > 90th percentile on ≥ 6 consecutive days	d
ATX|90p|	Action measure for warm days	Integral of exceedance (T_{max} − 90th percentile) over duration on ≥ 3 consecutive days	°C · d
ATN|90p|	Action measure for warm nights	Integral of exceedance (T_{min} − 90th percentile) over duration on ≥ 3 consecutive days	°C · d

Calculate action measure by detecting the upper threshold and minimum duration into single impact number. Action index is in the form of POT data, so nonstationary analysis framework in the form of event magnitudes and magnitudes. This allows to estimate time-dependent occurrence rates and thus to quantify the risk. Using TX90p misses aspects of absolute values, but can illuminate relative aspects.

Linear algebra

Eigenvectors: where Ax comes out parallel to x so $Ax = \lambda x$. Lambda is the eigenvalue.



If A is singular, then λ=0 is an eigenvalue.

As many eigenvectors as dimensions. Matrix **trace**: sum of all eigenvalues is 0.

Singular matrix: determinant is zero, but x is not zero.

Triangular matrix: eigenvalues are on the diagonal as 0 outside.

Degenerate matrix: repeated eigenvalue means a shortage of eigenvectors.

Singular Value Decomposition (SVD)

Factorisation of a matrix into an orthogonal matrix, diagonal matrix and orthogonal matrix again: $A = U \sum V^T$. A can be any matrix whatsoever, so any matrix can be SVD'ed. For a symmetric matrix, the eigenvectors are orthogonal, so can produce an orthogonal matrix. Positive definite matrix, ordinary eigenvalues become a positive lambda.

The goal of SVD, is to find an orthogonal basis in the row space (R^N) that can have an orthogonal basis in the column space (R^M) too when multiplied by a value. It is a special set up. So, how about we now make unit vectors into multiples of these unit vectors. The multiple is sigma, the stretching number. In maxtrix language:

$$A[v_1 v_2 v_3 .. v_r] = [u_1 u_2 u_3 .. u_r] \begin{bmatrix} \sigma_1 & 0 & 0 \\ 0 & \sigma_2 & 0 \\ 0 & 0 & \dots \end{bmatrix}$$

v's are the basis vectors in the row space, u's are the basis vectors in the column space and sigmas are the multiplying factors. So, $AV = U \sum$. So matrix A is getting converted to diagonal matrix sigma.

Can write: $AV = U \sum$ as $A = U \sum V^{-1}$ which is the same as $A = U \sum V^T$. To solve it, I don't want to solve U and V at once, so I want the U's to disappear to solve V first. Multiplying A by the transpose will make it symmetric. $A^T = V \sum^T U^T$. So $A^T A = V \sum^2 V^T$, since U * UT is the identity matrix and the product of two diagonal matrices is the squared. So, V will be eigenvectors and sigma are the eigenvalues. The Us are the eigenvectors of A A^T, using the same method.

Example1: $A = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix}$ and use v1 and v2, orthonormal v $A^T A =$ ectors in row space where R^2, u1 and u2, orthonormal vectors in column space where R^2 and find σ_1>0 and σ_2>0, which are the scaling factors. A is not orthogonal/symmetric but I want $Av_1 = \sigma_1 u_1$ & $Av_2 = \sigma_2 u_2$.

First step: compute $A^T A = \begin{bmatrix} 4 & -3 \\ 3 & 4 \end{bmatrix} \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = \begin{bmatrix} 25 & 7 \\ 7 & 25 \end{bmatrix}$, so becomes a symmetric matrix. Eigenvectors are $\begin{bmatrix} 1 \\ 1 \end{bmatrix} and \begin{bmatrix} 1 \\ -1 \end{bmatrix}$. Eigenvalue is the square root of 32 for eigenvector 1 and the square root of 18. Need to normalise eigenvectors by dividing by their length of sqrt(2). We now know that:

$A = U \sum V^T = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = [] \begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix} \begin{bmatrix} \sqrt{2} & \sqrt{2} \\ \sqrt{2} & -\sqrt{2} \end{bmatrix}$. Now find U's.

Look at A A^T = U sigma sigma^T U^T.

$AA^T = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} \begin{bmatrix} 4 & -3 \\ 4 & 3 \end{bmatrix} = \begin{bmatrix} 32 & 0 \\ 0 & 18 \end{bmatrix}$. Eigenvectors are

$AA^T \begin{bmatrix} 1 \\ 0 \end{bmatrix} = 32 \begin{bmatrix} 1 \\ 0 \end{bmatrix} and AA^T \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 18 \begin{bmatrix} 0 \\ 1 \end{bmatrix}$. Eigenvalues of Aˆ T A and A A^T are the same if order of multiplication has changed. But have new eigenvectors as identity matrix.

$A = U \sum V^T = \begin{bmatrix} 4 & 4 \\ -3 & 3 \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \sqrt{32} & 0 \\ 0 & \sqrt{18} \end{bmatrix} \begin{bmatrix} 1/\sqrt{2} & 1/\sqrt{2} \\ 1/\sqrt{2} & -1/\sqrt{2} \end{bmatrix}$

(He made a mistake but couldn't explain).

Example 2: With a singular matrix, using a null space. So key values are in row and column space.

$A = \begin{bmatrix} 4 & 3 \\ 8 & 6 \end{bmatrix}$, rank one matrix with one dimensional row space and column space and a null space. 2 x 2. Row spaces is all multiples of $\begin{bmatrix} 4 \\ 3 \end{bmatrix}$. In column space, is multiples of $\begin{bmatrix} 4 \\ 8 \end{bmatrix}$. V1 is unit vector, since one dimension in row space, so $\begin{bmatrix} 4 \\ 3 \end{bmatrix}$ vector

in a unit vector so $\begin{bmatrix}0.8\\0.6\end{bmatrix}$. V2 is in null space direction so $\begin{bmatrix}0.6\\-0.8\end{bmatrix}$, knowing it is an orthonormal vector. U1 will be $1/\sqrt{5}\begin{bmatrix}1\\2\end{bmatrix}$ and the orthogonal for U2. To do the SVD:

Calculate sigmas by $A^TA = \begin{bmatrix}4&8\\3&6\end{bmatrix}\begin{bmatrix}4&3\\8&6\end{bmatrix} = \begin{bmatrix}80&60\\60&45\end{bmatrix}$ is it a rank 1 matrix as are multiples or [4 3]^T. Rank 1 so one eigenvalue is 0 and other eigenvalue is 125.

$$A = \begin{bmatrix}4&3\\8&6\end{bmatrix} = 1/\sqrt{5}\begin{bmatrix}1&2\\2&-1\end{bmatrix}\begin{bmatrix}\sqrt{125}&0\\0&0\end{bmatrix}\begin{bmatrix}0.8&0.6\\0.6&-0.8\end{bmatrix}.$$

So V1, .., Vr are orthonormal basis for the row space. U1, .., Ur are orthonormal basis for column space. Vr+1 , .. , Vn are orthonormal basis for the null space. Ur+1, .., UM are orthonormal basis for null space of A^T.

Dimension of row space is rank r, dimension of column space is r. Dimension of null space is n-r, same for n(A^T).

When we choose a V1 in the direction of a corresponding U.

EOF analysis

Independent variables that convey as much information as possible: explore structure of the variability in a dataset and analyse relationships between the variables. Also called principal component analysis.

$$Z(x,y,t) = \sum_{k=1}^{N} PC(t) \cdot EOF(x,y)$$

Z(x,y,t) is the original time series as a function of time (t) and space (x, y).

EOF(x, y) show the spatial structures (x, y) of the major factors that can account for the temporal variations of Z.

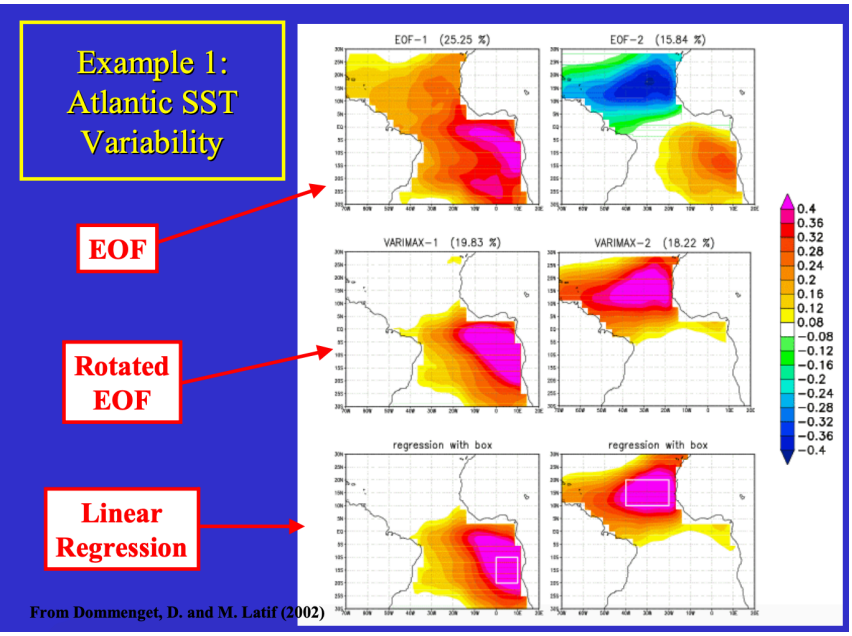
PC(t) is the principal component that tells you how the amplitude of each EOF varies with time.

If we want to get the principal component, we project a single eigenvector onto the data and get an amplitude of this eigenvector at each time, $e^T X$:

$$\begin{bmatrix} e_{11} & e_{21} & e_{31} & \cdots & e_{M1} \end{bmatrix} \begin{bmatrix} x_{11} & x_{12} & x_{13} & \cdots & x_{1N} \\ x_{21} & x_{22} & x_{23} & \cdots & x_{2N} \\ x_{31} & x_{32} & x_{33} & \cdots & x_{3N} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{M1} & x_{M2} & x_{M3} & \cdots & x_{MN} \end{bmatrix} = \begin{bmatrix} z_{11} & z_{12} & z_{13} & \cdots & z_{1N} \end{bmatrix}$$

For example, the amplitude of EOF-1 at the first measurement time is calculated as the following:

$$z_{11} = e_{11}x_{11} + e_{21}x_{21} + e_{31}x_{31} + \ldots + e_{M1}x_{M1}$$



Uses orthogonal functions to represent time series. Model tells us the pattern and the principal component tells us which year and how strong events were.

EOF analysis

PCs are orthogonal in time, so no simultaneous temporal correlation between any two principal components.

EOFs are orthogonal in space. The Empirical Orthogonal Functions (EOFs) of a time series are the eigenvectors of the covariance matrix of a time series. The eigenvalues of the covariance matrix tells us the fraction of variance explained by each EOF.

Use Singular Value Decomposition (SVD) to get EOFs, eigenvalues and PCs directly from the data without a covariance matrix. This is more efficient for small datasets. An m by n matrix A can be factored into: $A = U \sum V^T$ where U (m by m) are the EOFs for spatial data, V (n by n) are the normalised PCs for time data. The diagonal values of the sum are the eigenvalues representing the amplitude of the EOFs, NOT the variance. The first EOF in U and the first EOF in V explain most covariance (correlation) between two variables.

Present EOFs by: take PCs of time series then normalise to variance and then regress. Can also correlate the PCs with the original time series for each data point -> what is the co-variying part of the variable in spatial domain.

The amount of eigenvalues: Look at 95% significance error or the slope of the eigenvalue spectrum.

https://www.ess.uci.edu/~yu/class/ess210b/lecture.5.EOF.all.pdf

Download ESGF data

```
sh wget-GOGH.sh -H -i
```

```
open ID: https://esgf-node.llnl.gov/esgf-idp/openid/lisanne.blok
```

CMIP6.HighResMIP.MOHC.HadGEM3-GC31-HM.highres-future.r1i1p1f1.day.tas.gn Data Node: esgf.ceda.ac.uk Version: 20190301 Total Number of Files (for all variables): 36	Remove
CMIP6.HighResMIP.MOHC.HadGEM3-GC31-HM.hist-1950.r1i1p1f1.day.tas.gn Data Node: esgf.ceda.ac.uk Version: 20180730 Total Number of Files (for all variables): 65	Remove
Full Dataset Services:	[Show Metadata] [List Files] [WGET Script] [PID] [Show Citation] [Globus Download] [Further Info]
Full Dataset Services:	[Show Metadata] [List Files] [WGET Script] [PID] [Show Citation] [Globus Download] [Further Info]