Manipulating Time-Series Data

To manipulate the time-series data for the Southern Oscillation Index (SOI), a document class labelled `LoadSOI` was developed. This Object-oriented programming approach was selected as it enables a simpler user interface for data analysis.

At first glance of the time-series of SOI over the period 1950–2019, the signal appears to solely consist of noise. Large or long-term fluctuations are obscured by this high-frequency noise. To observe long-term trends/ processes that occur over longer time-scales this noise needs to be removed from the signal. Additionally, high-frequency noise tends to be random and not useful.

A relatively simple and effective way to remove noise is by a low-pass filter, which removes high-frequency fluctuations that are contained within the signal. These high-frequency fluctuations are not linked to the El Niño Southern Oscillation This can be done through a frequency-based approach (Fourier Transform) or a running mean (Convolution).

For signal de-noising, I use a 'running mean' approach, which takes the convolution of a box-kernel with the original signal. Alternative approaches include frequency-based low-pass filtering and using different kernels for the 'running mean' approach.

In addition to noise, atmospheric measurements often require cleaning or effective data management. For example, in remote sensing, there are 'bad pixels' which are often masked and affect data processing such as averaging. Data Scientists have to ensure that the signal only contains values within an 'expected range'.

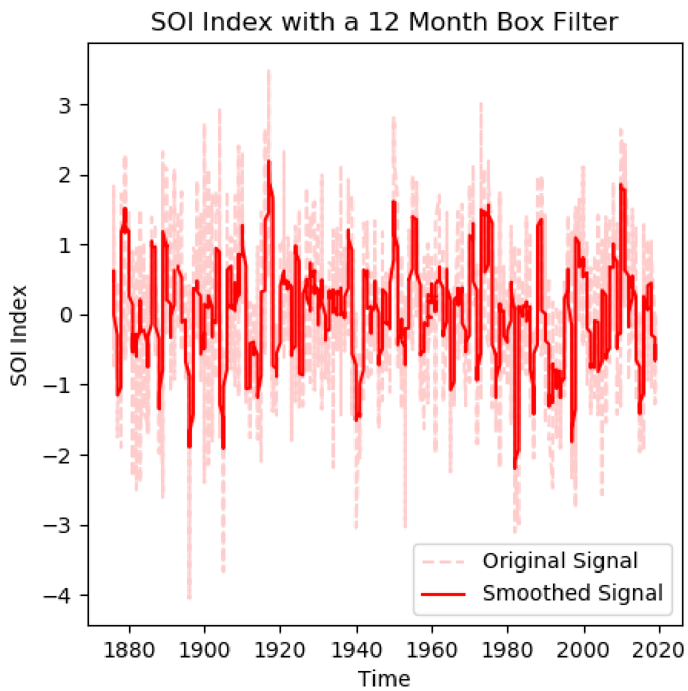

Figure 1: A comparison between the original SOI signal and a 12-month smoothed SOI.

Investing the Correlation between two time-series

The SOI is solely a measure of the atmospheric component of ENSO. A limitation of the SOI is that it is computed from two locations south of the Equator–where the majority of the ENSO action occurs. The Nino 3.4 Index has been deemed the most ENSO representative. Various other Nino indexes have also been developed to incorporate both an atmospheric and oceanic contribution of ENSO.

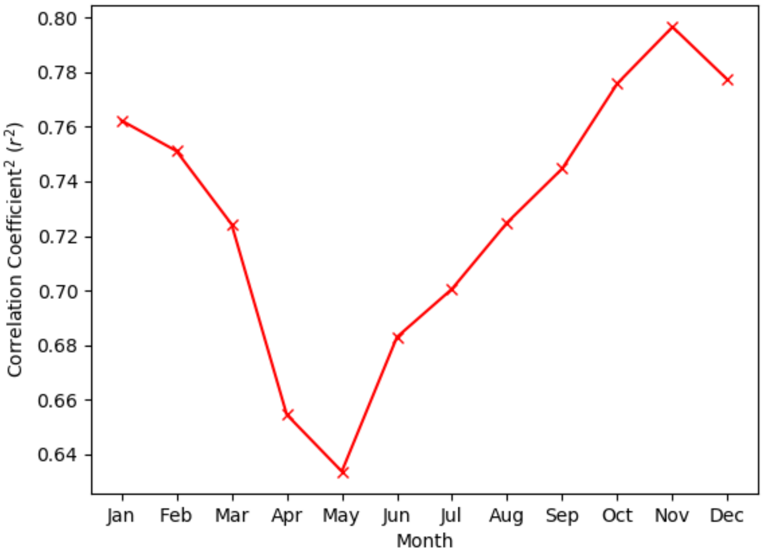

Figure 2: The correlation coefficient^2 between the SOI and the Nino 3.4 Index as a function of month.

The magnitude of the correlation coefficient follows a distinct seasonal cycle, with the largest values during the Austral summer. A wide variety of literature has suggested that ENSO activity is strongest in the Austral summer, and thus likely why the Nino 3.4 and SOI are more correlated during the Austral Summer.

In Figure 2, the r^2 values are computed which are always positive. However, the SOI and Nino 3.4 Index are negatively correlated as the SOI is pressure-measured.

Read and perform simple manipulations on spatio - temporal datasets

The methods outlined in the ”LoadNetcdf” class are aimed to solve the questions outlined for this section. Rather than using `xarray` to read the NetCDF files we have used h5py, which has relatively similar functionality and interface to xarray.

For computational efficiency, NumPy linear algebra functions are used preferentially when computing anomalies. Some questions/computations are not explicitly computed in the code and are methods/functions only. Figure 3 shows examples of the climatological mean air temperature (1981–2010) for various months throughout the year.

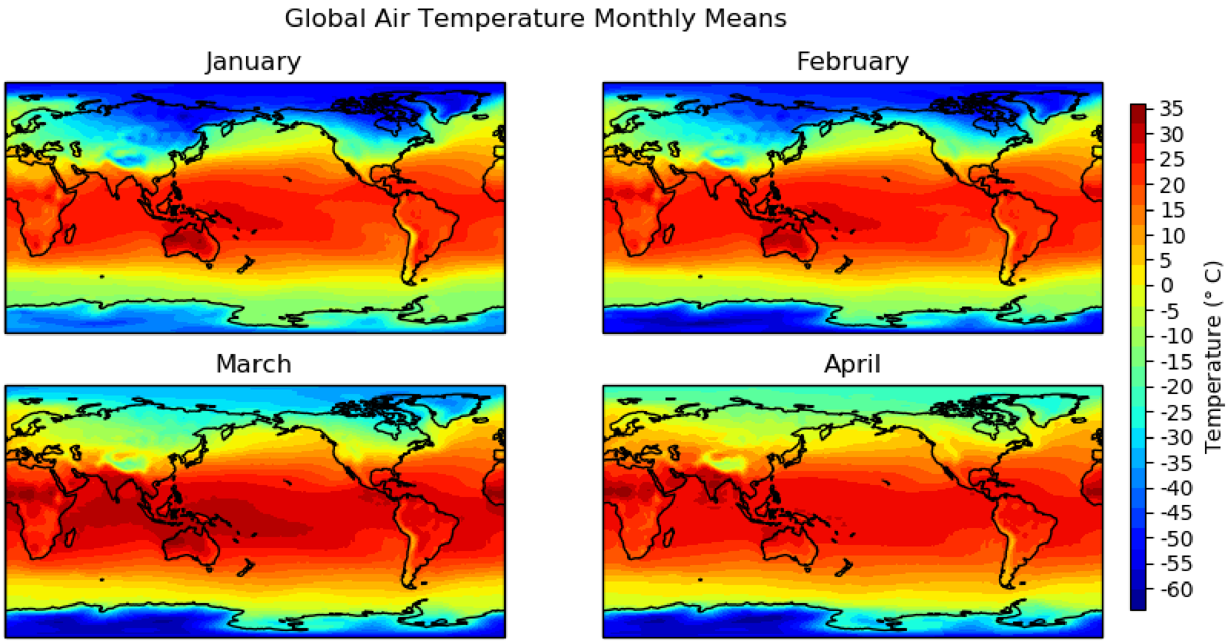

Figure 3: The mean global average air-temperature for various months.

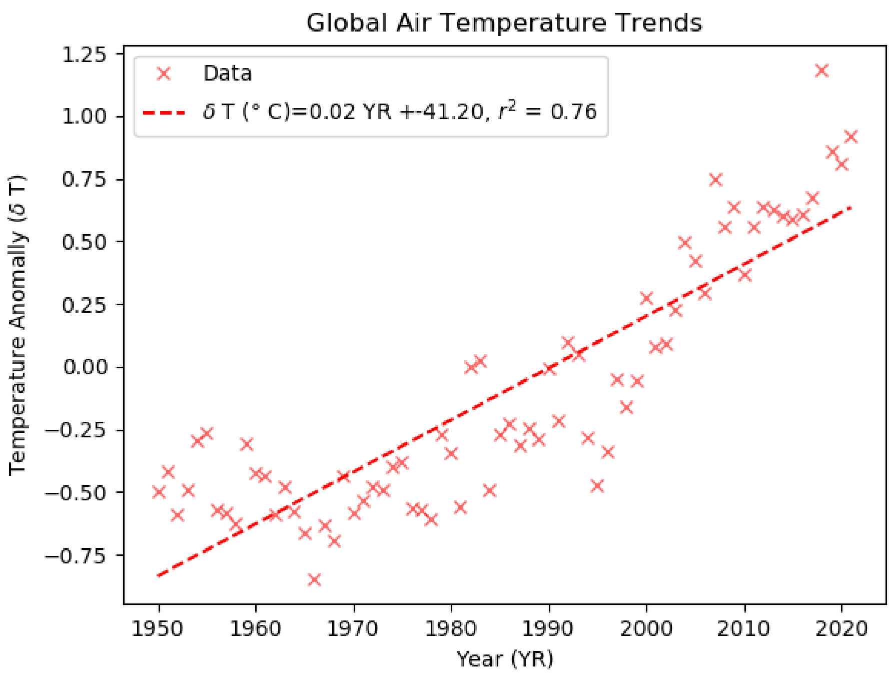

Figure 4: The global annual temperature anomaly at a given year relative to the mean global temperature over the period 1981–2010.

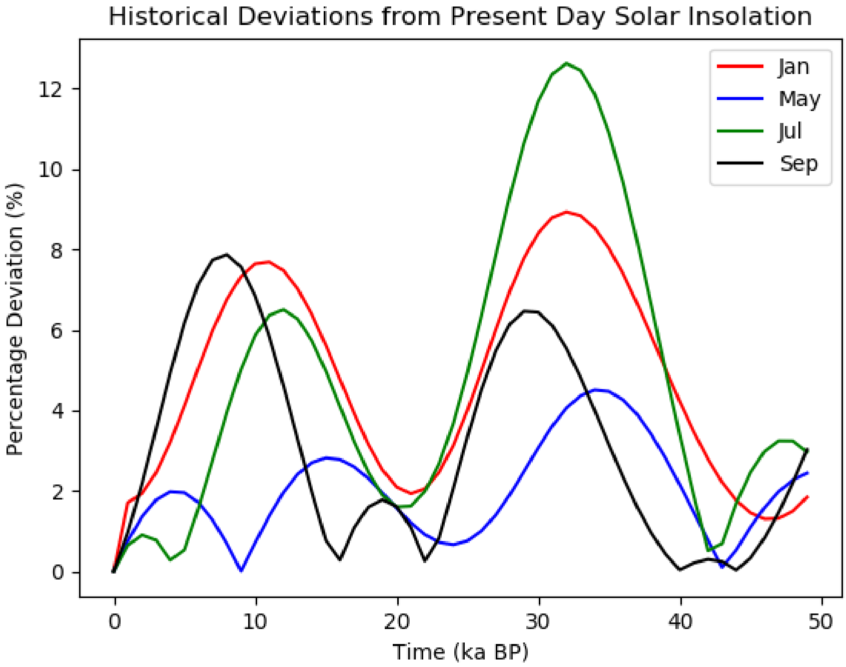

Figure 5: Percentage deviations in Solar Insolation from present-day insolation for selected months.

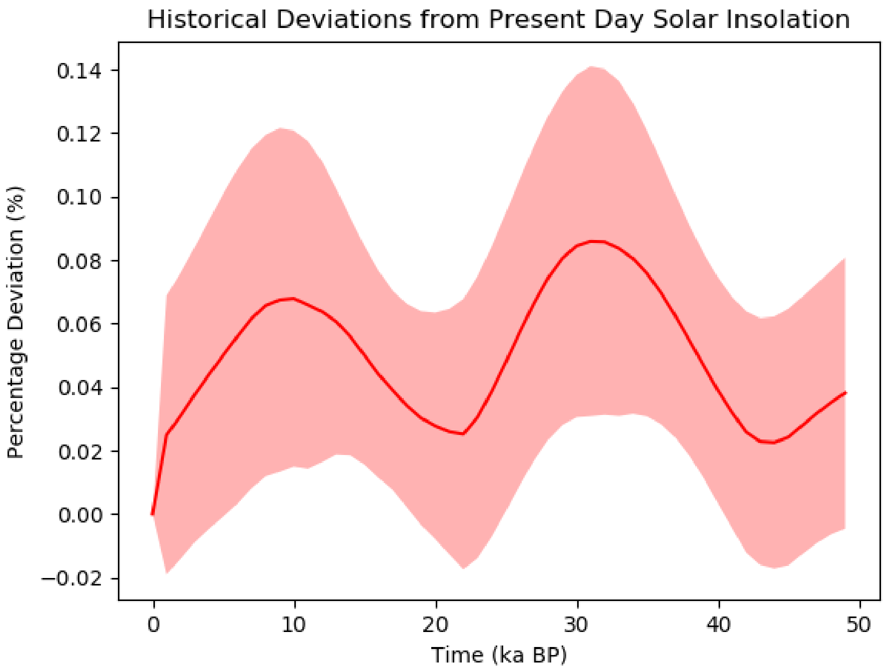

Figure 6: Percentage deviations in Solar Insolation from present-day insolation for all months

.

Further manipulation on some paleoclimatological 'data'

For this section, a small class was developed to perform small computations regarding the paleoclimatological data.

(a) To compute the mid-month insolation, linear interpolation using `scipy.interpolate`. This is outlined as "linear spline" in Task4.

(b) To compute the 3-month running averages, the signal was first oversampled (to include mid-month interpolation) and then was convoluted with a box-kernel (running mean). However, after the box-kernel operation, the signal is still over-sampled (as the running mean is applied over the mid-month points). The signal is then re-evaluated at the end/start month centre points (commented in the code). This function is easily reproducible to a 6-month running mean filter also.

(c) The signal is further interpolated to have a finer temporal resolution (100 years) using the "linear spline" function.

(d) Figure 5 and 6 illustrate the percentage deviation from present-day solar insolation as a function of time. Figure 6, illustrates the average for all months, with an outlined standard deviation. Both Figures illustrate the periodicity that exists in for the deviation in solar insolation, with an approximate period of 20 ka Bp. This percentage variability in insolation appears to be strongest in January and July from our primitive analysis.