Time Series

By Celine Sachi and Tracia Pan

DRP Spring 2025

1 Introduction

Time series is used in multiple fields such as examining the stock prices or stock returns over time in Economics, analyzing Covid-19 cases or Influenza hospitalizations over time in Epidemiology, looking at word or token distributions over time in Language modeling...etc. Now one may wonder: what is time series exactly?

Time series consists of data recorded over successive time intervals. Since the data are not i.i.d. (independent and identically distributed), there are correlation induced by the fact that we are making observations over time.

In this write up, you get to see the basic characteristics and visualizations of time series.

2 Characteristics

2.1 Mean and Variance

Given a time series sequence x_t, for $t = 1, 2, 3, \ldots$, we **define**:

- **Mean function:** $\mu_t = \mathbb{E}(x_t)$, representing the expected value over time.
- **Variance function:** $\sigma_t^2 = \mathrm{Var}(x_t) = \mathbb{E}[(x_t - \mu_t)^2]$, representing the expected variability or spread over time.

These functions describe key characteristics of a time series, such as drift and dispersion. However, they are not sufficient to fully characterize the distribution of the process due to the following reasons:

1. **Marginal Limitation:** The mean and variance are not enough to characterize the marginal distribution of a single variate x_t.
2. **Joint Distribution Ignorance:** They provide no information about the joint distribution of values at different time points (e.g., x_s and x_t for $s \neq t$). This includes questions of dependence such as whether the values tend to increase or decrease together.

These limitations highlight the importance of further tools such as auto-covariance and stationarity, particularly in the context of Gaussian processes.

Examples:

- **White Noise**: a collection of points that are uncorrelatedvand identically distributed, has **a mean of 0** and **constant variance.**
- **Moving Average of white noise** with window length 3: has **mean of 0** and **variance of** $\frac{1}{3}\sigma^2$. This shows that the variance is smaller than that of original sequence, which also implies that smoothing reduces variances.

2.2 Auto-Covariance and Auto-Correlation

- The **auto-covariance function** for a time series x_t is defined as

$$\gamma_x(s,t) = \mathrm{Cov}(x_s, x_t)$$

It is symmetric: $\gamma_x(s,t) = \gamma_x(t,s)$, and for $s = t$, it gives the variance: $\gamma_x(t,t) = \sigma_t^2$.

- The **auto-correlation function** is the normalized version of auto-covariance:

$$\rho_x(s,t) = \frac{\gamma_x(s,t)}{\sigma_{x,s}\sigma_{x,t}}$$

Typically written as $\rho(s,t)$ when context is clear. By the Cauchy-Schwarz inequality, which states that:

$$Cov(x,y) \leq \sqrt{Var(x)Var(y)}$$

Then, for any random variable x, y, note that we always have:

$$\rho(s,t) \in [-1, 1]$$

- Auto-correlation measures *linear dependence* across time points. If a series is very smooth, then the auto-covariance will typically be large and positive when the points s, t are closer together. But it may be negative when the points s, t are further apart. And if the time series is choppy, the auto covariance will typically be close to 0. Therefore, auto-covariance captures whether past values help predict the future values.
- **Uncorrelatedness does not imply independence.** However, for Gaussian sequences, uncorrelatedness *does* imply independence.
- For **white noise**, $\gamma(s,t) = 0$ for all $s \neq t$, and thus $\rho(s,t) = 0$ for $s \neq t$.
- For a **moving average** of white noise:

$$y_t = \frac{1}{3}(x_{t-1} + x_t + x_{t+1})$$

The auto-covariance function is:

$$\gamma(s,t) = \begin{cases} \sigma^2 \cdot \frac{1}{3} & s = t \pm 2 \\ \sigma^2 \cdot \frac{2}{9} & s = t \pm 1 \\ \sigma^2 \cdot \frac{1}{3} & s = t \\ 0 & \text{otherwise} \end{cases}$$

and the corresponding auto-correlation function is:

$$\rho(s,t) = \begin{cases} \frac{1}{3} & s = t \pm 2 \\ \frac{2}{3} & s = t \pm 1 \\ 1 & s = t \\ 0 & \text{otherwise} \end{cases}$$

- For a **random walk**, defined as

$$x_t = \delta t + \sum_{i=1}^{t} w_i$$

where w_i is white noise, the auto-covariance function becomes:

$$\gamma(s, t) = \sigma^2 \min\{s, t\}$$

and the auto-correlation function is:

$$\rho(s, t) = \frac{\min\{s, t\}}{\sqrt{st}}$$

Note: Auto-correlation for a random walk decreases as the gap between s and t increases, even though overall correlation remains high for closer points.

2.3 Cross-Covariance and Cross-Correlation

- The **cross-covariance function** between two time series x_t and y_t is defined as:

$$\gamma_{xy}(s, t) = \text{Cov}(x_s, y_t)$$

Unlike auto-covariance, cross-covariance is not necessarily symmetric: $\gamma_{xy}(s, t) \neq \gamma_{xy}(t, s)$. Note that $\gamma_{xx}(t, t) = \gamma_x(t)$, i.e., the cross-covariance of a time series with itself is the auto-covariance.

- The **cross-correlation function** normalizes the cross-covariance:

$$\rho_{xy}(s, t) = \frac{\gamma_{xy}(s, t)}{\sigma_{x,s} \cdot \sigma_{y,t}}$$

By the Cauchy-Schwarz inequality, the cross-correlation is bounded:

$$\rho_{xy}(s, t) \in [-1, 1]$$

- **Lag-based Interpretation:** Often, cross-correlation is plotted as a function of lag $h = s - t$, simplifying analysis and visualization. For example, in analyzing Covid-19 data, the cross-correlation between cases x_t and deaths y_t may be maximized at a lag $h = -25$, suggesting that *cases lead deaths*.

- **Terminology:**
  - If cross-correlation is maximized at a negative lag ($h < 0$), then the first time series (e.g., cases) *leads* the second (e.g., deaths).
  - If maximized at a positive lag ($h > 0$), then the second time series *leads* the first.

- Although cross-correlation is a function of two time indices s and t, in practice it is often estimated only as a function of the lag $h = s - t$, due to data limitations. This estimation strategy is further justified by assumptions of stationarity.

2.4 Stationarity

Definition: Stationarity is an important property that determines whether a time series behave consistently over time. This is useful because stationarity affects how we model, forecast, and interpret data.

- Most time series models (like ARIMA) uses assumptions of stationarity.
- Stationary process is more predictable since its behavior is stable over time.
- Estimating parameters is easier and more reliable for stationary series which improves interpretability.

Types of Stationarity

1. Strong Stationarity

   - A time series x_t is said to be **strongly stationary** if:

   $$(x_{t_1}, x_{t_2}, \ldots, x_{t_k}) \stackrel{d}{=} (x_{t_1+\ell}, x_{t_2+\ell}, \ldots, x_{t_k+\ell}) \quad \text{for all } k \geq 1, \text{ all } t_1, \ldots, t_k, \text{ and all } \ell$$

     where $\stackrel{d}{=}$ denotes equality in distribution.
   - This means that any collection of variates along the series has the same joint distribution after we shift the time indices forward or backward.
   - Implications:
     - For $k = 1$: $x_s \stackrel{d}{=} x_t \Rightarrow \mu_{x,s} = \mu_{x,t}$ (mean is constant).
     - For $k = 2$: $(x_s, x_t) \stackrel{d}{=} (x_{s+\ell}, x_{t+\ell}) \Rightarrow \gamma_x(s,t) = \gamma_x(s+\ell, t+\ell)$, meaning the auto-covariance depends only on the lag $h = s - t$.
   - Note: this is a very strong property. In fact, so strong that it is rarely be useful for most applications. For instance, it is not even really possible to assess whether it holds given a single time series.

2. Weak Stationarity

   - A time series is **weakly stationary** if:

   $$\mu_{x,t} = \mu, \quad \text{and} \quad \gamma_x(s,t) = \gamma_x(s+\ell, t+\ell) \quad \text{for all } s, t, \ell$$

   - This implies that the variance $\sigma^2_{x,t} = \sigma^2$ is constant for all t.
   - Relationship to strong stationarity:

   $$\text{Strong stationarity} \Rightarrow \text{Weak stationarity}$$

     but not vice versa, unless the process is Gaussian, in which case the two are equivalent.
   - Since weak stationarity is more practical, we often refer to it simply as stationarity.
   - Under stationarity, the auto-covariance is a function of lag:

   $$\gamma_x(h) := \gamma_x(t, t+h)$$

     and similarly for auto-correlation:

   $$\rho_x(h) := \rho_x(t, t+h)$$

Examples and Notes

- **White noise** is stationary: zero mean, constant variance, and auto-covariance is zero for $s \neq t$.
- **Moving average of white noise** is also stationary. Its auto-covariance is:

$$\gamma(h) = \sigma^2 \cdot \begin{cases} 1/9 & \text{if } h = \pm 2 \\ 2/9 & \text{if } h = \pm 1 \\ 1/3 & \text{if } h = 0 \\ 0 & \text{otherwise} \end{cases}$$

- **Random walk** is not stationary:

  - Variance increases over time: $\sigma_t^2 = \sigma^2 t$
  - Auto-covariance: $\gamma(s,t) = \sigma^2 \min\{s,t\}$, not a function of lag alone

- **Trend Stationarity:** A time series $x_t = \theta_t + w_t$ is trend stationary if θ_t is a fixed nonrandom sequence and w_t is stationary. Then:
$$\gamma_x(s,t) = \gamma_w(s,t)$$
since θ_t does not affect covariance.

3 Decomposing Time Series

The decomposition of time series is shown by the time series model:

$$x_t = \theta_t + w_t$$

The model is also called the signal plus noise model where noise refers to a sequence $x_t,\ t = 1, 2, 3, ...$ of uncorrelated random variables with zero mean and constant variance, which can also be shown as the following:

$$Cov(x_s, x_t) = 0,\ \text{for all } s \neq t$$
$$\mathbb{E}(x_t) = 0, Var(x_t) = \sigma^2,\ \text{for all } t$$

The signal sequence θ_t can be decomposed into trend and seasonal components:

$$\theta_t = u_t + s_t$$

where trend component u_t shows the general trend of the data and is estimated nonparametrically using smoother and seasonal s_t contains a regular periodic behavior for some fixed behavior.

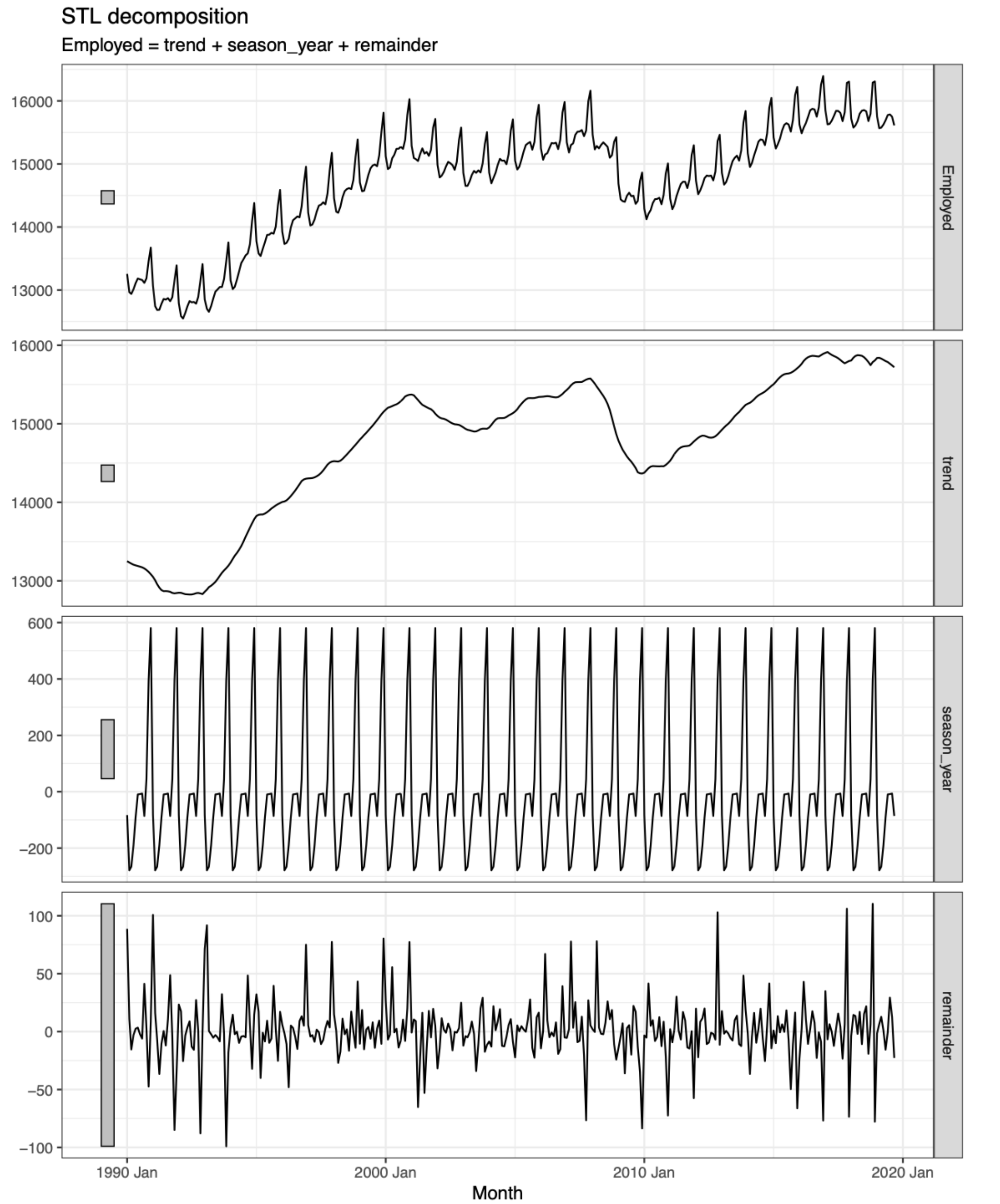

Figure 1: STL decomposition of US retail employment data

This STL decomposition (Seasonal and Trend decomposition using LOESS) here is a great visualization that shows the components of the signal plus noise model. The top box shows the full time series data,and the box below that took out the trend components of the data, outlining the overall motion over time. The third box took out the seasonality component, and it was shown as these identical waves that goes on and on. The last box contains the remaining information of the data that was not defined as trend nor seasonality. In other words, these information are considered noise and identified by the STL decomposition as information that aren't helpful for analyzing the data.

4 Seasonality

4.1 Definition

Seasonality is a regular, repeating patterns or cycles in time series that occur at fixed intervals due to seasonality factors (time of year, month, week, day, etc).

To be clear, seasonality and trend are two different things.

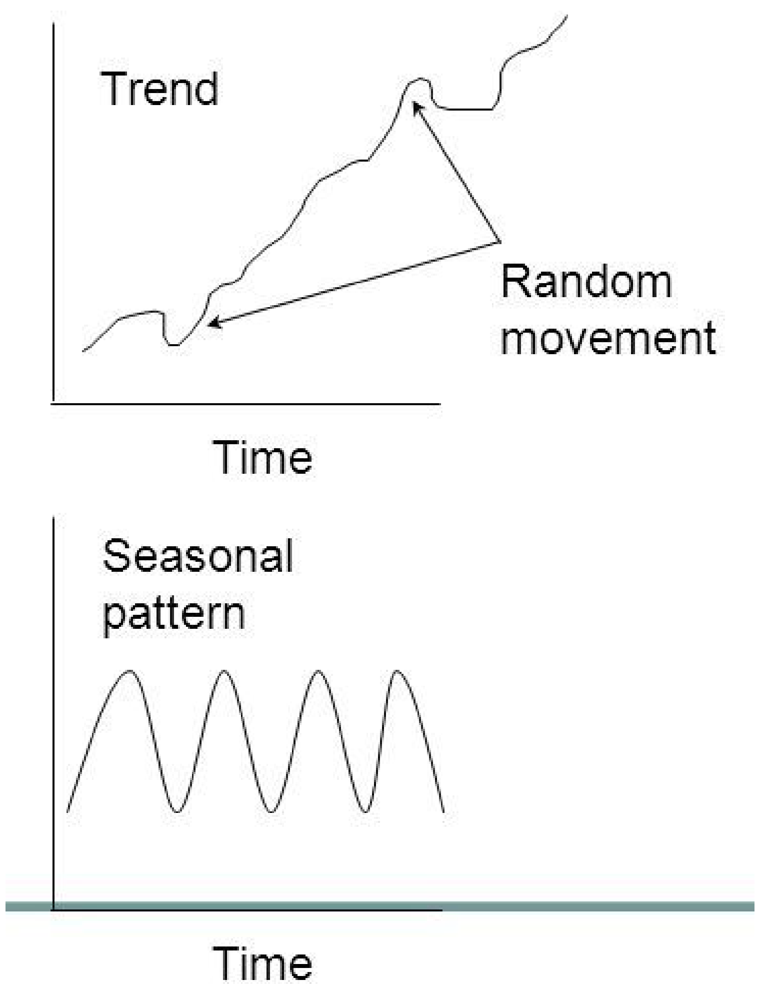

Figure 2: Trend vs Seasonality

As we can see from the plots above, trend is a long-term upward/downward movement. Meanwhile, seasonality is short-term, cyclic, periodic variation

4.2 Methods

To detect Seasonality, there are several methods, such as:

1. Fourier Decomposition
2. Discrete Fourier Transform
3. Seasonal-Trend Decomposition

4.2.1 Fourier Decomposition

- Given a time series x_t, for $t = 1, \ldots, n$, we seek a decomposition into cosine and sine components:

$$x_t \approx a_0 + \sum_{j=1}^{p} (a_j c_{tj} + b_j s_{tj}), \quad t = 1, \ldots, n$$

  where the basis functions are:

$$c_{tj} = \cos\left(\frac{2\pi jt}{n}\right), \quad s_{tj} = \sin\left(\frac{2\pi jt}{n}\right)$$

- The coefficients a_j and b_j represent the amplitudes of the corresponding cosine and sine components.
- To perfectly reconstruct any time series of length n, we only need $p = \frac{n-1}{2}$ (assuming n is odd), making the decomposition exact.
- This decomposition can be computed via linear regression. Defining the matrix $Z \in \mathbb{R}^{n \times n}$ whose columns are the basis functions:

$$Z = \begin{bmatrix} \frac{1}{\sqrt{2}} & \cos\left(\frac{2\pi \cdot 1 \cdot 1}{n}\right) & \sin\left(\frac{2\pi \cdot 1 \cdot 1}{n}\right) & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ \frac{1}{\sqrt{2}} & \cos\left(\frac{2\pi \cdot (n-1) \cdot 1}{n}\right) & \sin\left(\frac{2\pi \cdot (n-1) \cdot 1}{n}\right) & \cdots \end{bmatrix}$$

- The coefficients are found via least squares:

$$(Z^T Z)^{-1} Z^T x$$

  but since $Z^T Z = (n/2)I$, the coefficients reduce to scaled inner products:

$$\hat{a}_j = \frac{2}{n} \sum_{t=1}^{n} x_t \cos\left(\frac{2\pi jt}{n}\right), \quad \hat{b}_j = \frac{2}{n} \sum_{t=1}^{n} x_t \sin\left(\frac{2\pi jt}{n}\right)$$

  and the mean term is:

$$\hat{a}_0 = \bar{x}$$

- Hence, the exact reconstruction becomes:

$$x_t = \bar{x} + \sum_{j=1}^{(n-1)/2} \left(\hat{a}_j \cos\left(\frac{2\pi jt}{n}\right) + \hat{b}_j \sin\left(\frac{2\pi jt}{n}\right)\right), \quad t = 1, \ldots, n$$

- This works because Z is an orthogonal matrix whose columns have norm $\sqrt{n/2}$. Hence, regression onto its columns is equivalent to projection onto an orthogonal basis.
- The computational cost is $O(n^2)$ in general, but with the **Fast Fourier Transform (FFT)**, this can be reduced to $O(n \log n)$.
- Example:

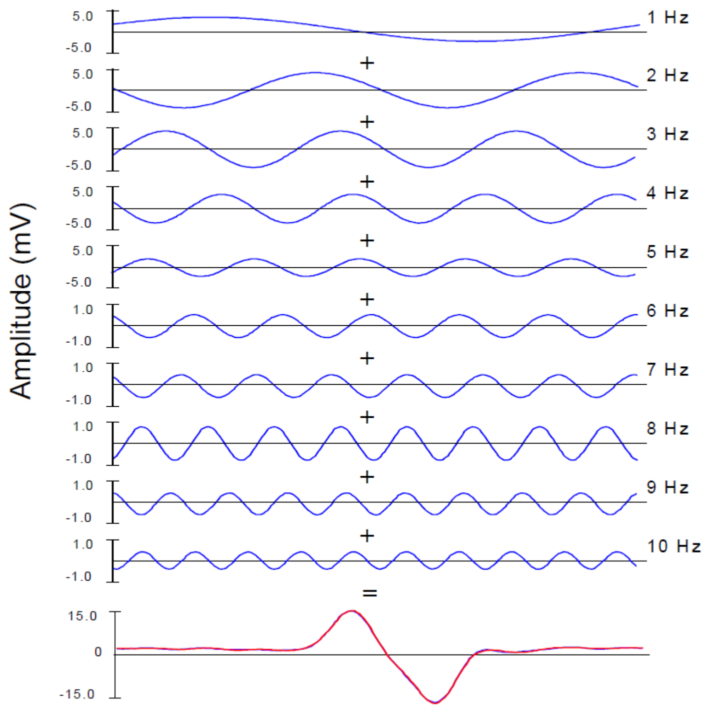

Figure 3: Example of FD

The blue plots above show individual sine waves and as we move downward, the frequency increases from 1 to 1 Hz and the amplitude slightly decreases. The red plot shows the sum of all individual sine, but not a pure sine wave. It is more complex with sharp rise and fall. The shape is smooth but clearly not periodic in the same way the individual sine waves are. Overall, Fourier Decomposition method works by breaking down a complex signal into simple sine and cosine components, and adding those components together to reconstruct the original signal.

4.2.2 Discrete Fourier Transform

- The coefficients in the Fourier decomposition can be computed efficiently using the **Discrete Fourier Transform (DFT)**.
- The DFT of a time series x_t, $t = 1, \dots, n$, is denoted d_x and defined as:

$$d_x(j/n) = \frac{1}{\sqrt{n}} \sum_{t=1}^{n} x_t \exp\left(-2\pi i j t/n\right), \quad j = 0, \dots, n-1$$

where i is the imaginary unit ($i^2 = -1$). The DFT is thus complex-valued.

- Using Euler's formula, we have:

$$d(j/n) = \frac{1}{\sqrt{n}} \sum_{t=1}^{n} x_t \cos\left(\frac{2\pi jt}{n}\right) - i \cdot \frac{1}{\sqrt{n}} \sum_{t=1}^{n} x_t \sin\left(\frac{2\pi jt}{n}\right)$$

- From this, the Fourier coefficients can be recovered as:

$$\hat{a}_j = \frac{2}{\sqrt{n}} \Re\{d(j/n)\}, \quad \hat{b}_j = -\frac{2}{\sqrt{n}} \Im\{d(j/n)\}$$

  where $\Re\{\cdot\}$ and $\Im\{\cdot\}$ denote the real and imaginary parts, respectively.

- **Periodogram:** The periodogram $P(j/n)$ is defined as the squared modulus of the DFT:

$$P(j/n) = |d(j/n)|^2 = \Re\{d(j/n)\}^2 + \Im\{d(j/n)\}^2$$

  Using the cosine and sine coefficient expressions, this can also be written as:

$$P(j/n) = \frac{n}{4}(\hat{a}_j^2 + \hat{b}_j^2)$$

- **Fast Fourier Transform (FFT):** The DFT can be computed efficiently in $O(n \log n)$ time using the FFT algorithm, especially effective when n is a highly composite number (like a power of 2). The modern FFT is attributed to Cooley and Tukey (1960s).

- Example:

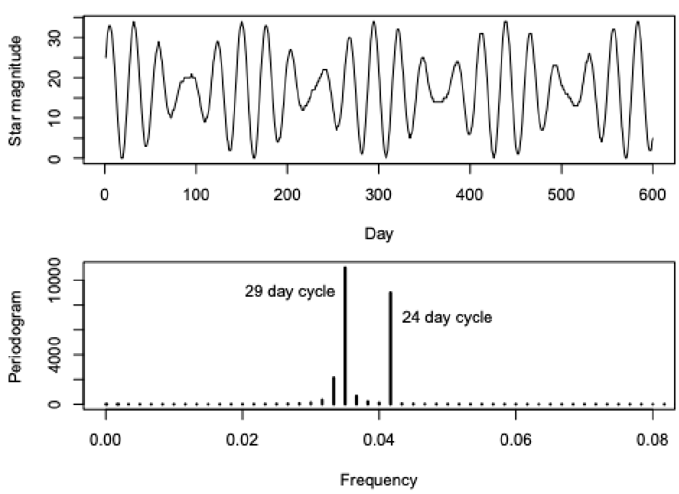

Figure 4: Example of DFT

The graph shows the Discrete Fourier Transform (DFT) applied to a time series of star brightness (or magnitude) measurements taken over 600 days.

The top plot shows a raw signal of Time Series (Star Magnitude vs. Day) — how the brightness of a star changes over time. The pattern appears cyclical, with repeating waves, but it is not easy to tell exactly how many cycles or their lengths just from the plot. Meanwhile, the bottom plot is a Periodogram (Frequency Domain Representation). This is the result of applying DFT to the time series above. The x-axis shows frequency, and the y-axis shows power (strength of each frequency).The taller the spike, the stronger that frequency is in the original signal.

There are two dominant peaks in the frequency domain: one at 1/29 and one at 1/24. These spikes tell us the star's brightness is influenced by two strong, regular cycles: one repeating every 29 days, and another every 24 days.

Note: Even though the top plot looks messy or complicated, the DFT reveals clear periodic components. This suggests that the star may be undergoing regular physical processes (e.g., rotation, orbit, or pulsation) that repeat with those two cycles.

Overall, the DFT breaks the star's brightness pattern into wave-like components. Then, it finds two major repeating signals — 29-day and 24-day cycles — which dominate the observed variations.

4.2.3 Seasonal Trend Decomposition

- Definition: a statistical method to detect the presence of seasonality in time series by separating into 3 components, such that:

$$\text{Time Series} = \text{Trend} + \text{Seasonality} + \text{Noise}$$

- **Generic procedure for ST decomposition** of a time series y_t, $t = 1, \ldots, n$:
  1. Use a smoother to estimate the trend $\hat{\theta}_t$, intentionally oversmoothing to avoid capturing seasonal components.
  2. Compute residuals: $r_t = y_t - \hat{\theta}_t$.
  3. Compute the **periodogram** of r_t, without assuming any specific seasonal period (e.g., weekly, quarterly).
  4. Identify prominent peaks in the periodogram; interpret them as indicative of seasonal components.
  5. Fit and add these seasonal components back into the model.
- **Limitations:**
  - This method assumes seasonal components have constant amplitude over time.
  - More advanced techniques (such as adaptive models) are needed when seasonality varies.
- Example:

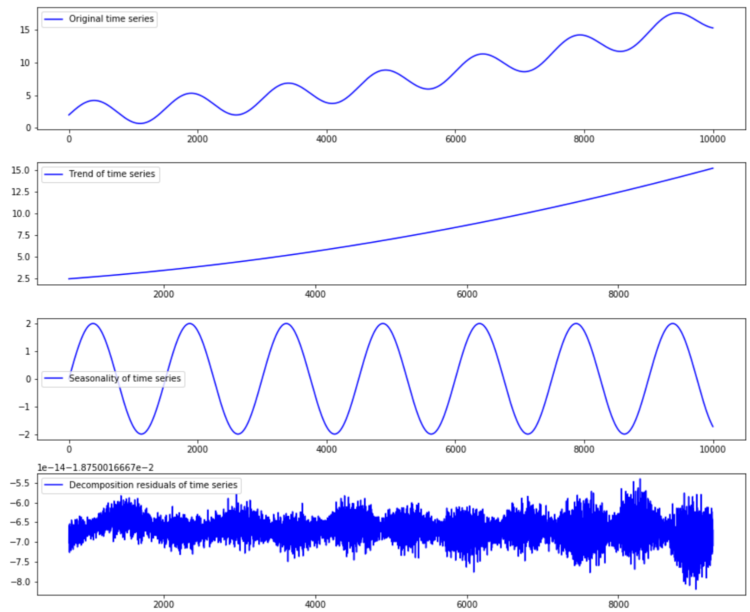

Figure 5: Example of ST Decomposition

Interpretation of the plot (from top to bottom order):

- First Plot: Original Time series. The plot shows the raw data combining all patterns — long-term trend, repeated seasonal variation, and irregular noise. Overall, there is a rising pattern, with regular wave-like oscillations.
- Second Plot: Trend. The line represents the long-term direction of the data, which is clearly increasing, and slightly nonlinear (curving upward), suggesting steady growth. Overall, trend captures the broad shape of data over long range, ignoring smaller cyclical patterns.
- Third Plot: Seasonality. The plot shows repeating patterns that occur at fixed intervals (like monthly or yearly cycles). We can see there are clear sine-wave-like oscillations with a consistent period and amplitude.
- Fourth Plot: Residuals (Noise). This plot is a leftover (remainder) after removing the trend and seasonal components from the original data. The residuals appear to center around zero and increasing variance towards the end.

5 Smoothing

Smoothing is used to eliminate noise and irregularities to reveal underlying trend and patterns in the data and is mainly used for retrospective estimation, meaning that it is used for time series data that we already obtain.

To do so, we will specifically focus on the signal component of the signal plus noise model. The two broad class of smoothers of interests are linear filters and penalized least squares.

5.1 Linear Filters

- Moving average is one of the most common examples of a linear filter of the form:

$$\hat{\theta}_i = \sum_{j=-k}^{k} a_j y_{i-j}, i = 1, ..., n$$

for some weights $a_j, j = -k...k$

- Centered MA smoother uses weights $a_{-k} = ...a_0 = ... = a_k = \frac{1}{m}$
- Trailing MA smoother uses weights $a_{-k} = ... = a_{-1} = 0$ and $a_0 = ... = a_k = \frac{1}{m}$
- To get even smoother-looking estimates, we use Kernel smoothing where it uses $k = n$ and adds a smoother weight sequence:

$$a_j = \frac{K(j/b)}{\sum_{i=-n}^{n} K(i/b)}, j = 1, ..., n$$

for kernel function $K : \mathbb{R} \to \mathbb{R}$ and *bandwidthb* where *bandwidth* can be think as λ in ridge and lasso.

- Gaussian kernel, $K(u) = e^{-u^2/2}$ is a commonly chosen kernel that is also shown in the example below on Southern Oscillation Index.

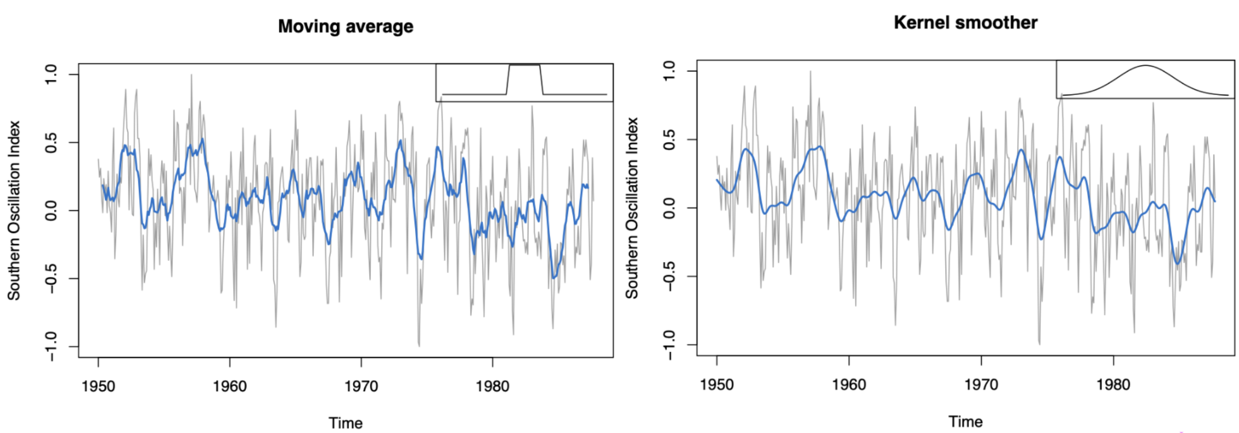

Figure 6: Moving average and Gaussian kernel smoothing estimates fit to the Southern Oscillation Index

In this side by side comparison it is shown that kernel smoother has much smoother curve lines compared to moving average.

5.2 Penalized Least Squares

The general form of penalized least squares smoothers is given by solving:

$$\min_{\theta \in \mathbb{R}^{\ltimes}} \sum_{i=1}^{n} (y_i - \theta_i)^2 + \lambda P(\theta)$$

for some penalty function $P : \mathbb{R}^n \to \mathbb{R}_+$ and tuning parameter $\lambda \geq 0$

- Hodrick-Prescott filter, also known as HP filter, is defined by:

$$\min_{\theta \in \mathbb{R}^{\ltimes}} \sum_{i=1}^{n} (y_i - \theta_i)^2 + \lambda \sum_{i=1}^{n-2} (\theta_i - 2\theta_{i+1} + \theta_{i+2})^2$$

where the first portion is to minimize the sum of squares and the second portion creates the smoothness on the curves.

- HP filter decomposes a time series into a smooth trend component and a cyclical component where the cyclical component captures the short-term fluctuations or deviations from the trend. Residuals from HP filter are used to estimate cyclic component.

- ℓ_1 trend filter is another type of penalized estimator defined by:

$$\min_{\theta \in \mathbb{R}^{\ltimes}} \sum_{i=1}^{n} (y_i - \theta_i)^2 + \lambda \sum_{i=1}^{n-2} |\theta_i - 2\theta_{i+1} + \theta_{i+2}|$$

- The use of ℓ_1 penalty leads to sparsity of knots, and these knots served as special points where the smoother "trend" changes. As a result, it forms a piecewise linear structure.

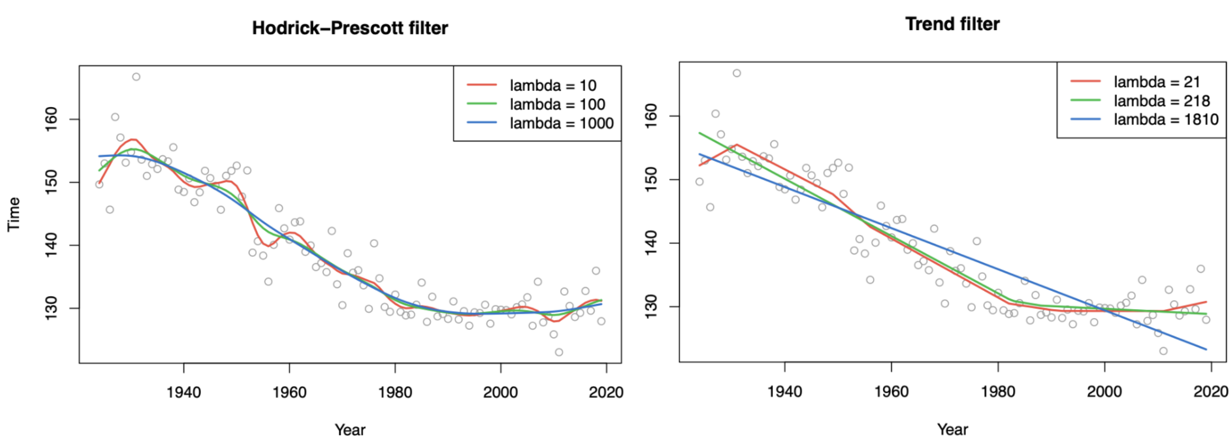

Figure 7: HP and trend filter estimates fit to the winning men's Boston marathon times

In the trend filter graph, we can identify these abrupt points where the slope of the line changes every time it hits a knot. As seen in both graphs, we can also observe that as λ increases, the curves and lines become smoother. In extremely large λ, we can observe that the curve and line were so smooth that it was only able to provide general information of the patterns.

6 References

University of California, Berkeley. (2024). Characteristics of time series [Lecture notes]. https://stat153.berkeley.edu/fall2024/lectures/characteristics/characteristics.pdf

University of California, Berkeley. (2024). Smoothing [Lecture notes]. https://stat153.berkeley.edu/fall-2024/lectures/smoothing/smoothing.pdf

University of California, Berkeley. (2024). Dependence [Lecture notes]. https://stat153.berkeley.edu/fall-2024/lectures/dependence/dependence.pdf

University of California, Berkeley. (2024). Spectral analysis [Lecture notes]. https://stat153.berkeley.edu/fall-2024/lectures/spectral/spectral.pdf

10. Decomposition of Time-Series in Python: Trend and Seasonal (Periodic) Components – Fusion of Engineering, Control, Coding, Machine Learning, and Science. (2021, March 28). Aleksandarhaber.com. https://aleksandarhaber.com/decomposition-of-time-series-in-python-trend-and-seasonal-periodic-components/

Howard, R. (2017). The application of data analysis methods for surface electromyography in shot putting and sprinting [Master's thesis]. https://doi.org/10.13140/RG.2.2.15907.04640

Components of time series (trend, seasonality, cycles) — Data, Inference, and Decisions Class Notes — Fiveable. (2018). Fiveable. https://library.fiveable.me/data-inference-and-decisions/unit-9/components-time-series-trend-seasonality-cycles/study-guide/ZAXNLWZlyAsiKWMu