Lecture 2

Estimating trend and seasonality

MATH 8090 Time Series Analysis
August 24 & 26, 2021

Whitney Huang
Clemson University

Agenda

1. **The classical decomposition model**

2. **Trend Estimation**

3. **Estimating Seasonality**

The Classical (Additive) Decomposition Model

- The additive model for a time series $\{Y_t\}$ is

$$Y_t = \mu_t + s_t + \eta_t,$$

  where

  - μ_t is the trend component
  - s_t is the seasonal component
  - η_t is the random (noise) component with $\mathbb{E}(\eta_t) = 0$

- Standard procedure:

  (1) Estimate/remove the trend and seasonal components

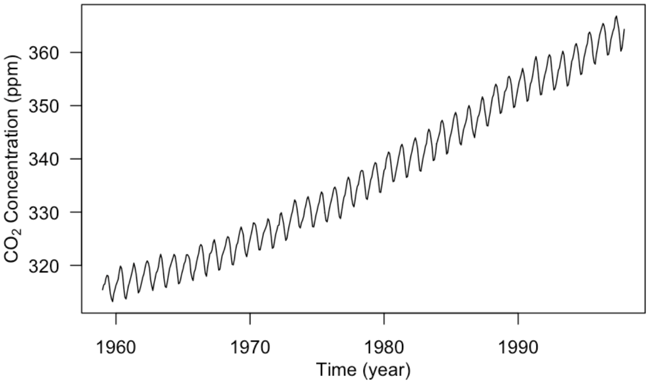
  (2) Analyze the remainder, the residuals $\hat{\eta}_t = y_t - \hat{\mu}_t - \hat{s}_t$

- We will focus on (1) for this week

Mauna Loa Atmospheric CO_2 Concentration Revisited

Monthly atmospheric concentrations of CO_2 at the Mauna Loa Observatory [Source: Keeling & Whorf, Scripps Institution of Oceanography]

```{r}
data(co2)
par(mar = c(3.8, 4, 0.8, 0.6))
plot(co2, las = 1, xlab = "", ylab = "")
mtext("Time (year)", side = 1, line = 2)
mtext(expression(paste("CO"[2], " Concentration (ppm)")), side = 2, line = 2.5)
```



Estimating Trend for Nonseasonal Model

- Assuming $s_t = 0$ (i.e., there is no "seasonal" variation), we have

$$Y_t = \mu_t + \eta_t,$$

with $\mathbb{E}(\eta_t) = 0$

- Methods for estimating trends
  - Least squares regression
  - Smoothing
- Alternatively, one can remove trend by differening time series

Trend Estimation: Linear Regression

- The additive nonseasonal time series model for $\{Y_t\}$ is

$$Y_t = \mu_t + \eta_t,$$

where the trend is assumed to be a linear combination of known covariate series $\{x_{it}\}_{i=1}^p$

$$\mu_t = \beta_0 + \sum_{i=1}^{p} \beta_i x_{it}.$$

- Here we want to **estimate** $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_p)^T$ from the data $\{y_t, \{x_{it}\}_{i=1}^p\}_{t=1}^T$
- You're likely quite familiar with this formulation already ⇒ Regression Analysis

Some Examples of Covariate Series $\{x_{it}\}$

- **Simple linear regression model:**

$$\mu_t = \beta_0 + \beta_1 x_t,$$

for example, the temperature trend at time t could be a constant (β_0) plus a multiple (β_1) of the carbon dioxide level at time t (x_t)

- **Polynomial regression model:**

$$\mu_t = \beta_0 + \sum_{i=1}^{p} \beta_i t^i$$

- **Change point model:**

$$\mu_t = \begin{cases} \beta_0 & \text{if } t \le t^*; \\ \beta_0 + \beta_1 & \text{if } t \ge t^*. \end{cases}$$

Parameter Estimation: Ordinary Least Squares

- Like in the linear regression setting, we can estimate the parameters via ordinary least squares (OLS)

- Specifically, we minimize the following objective function:

$$\ell_{ols} = \sum_{t=1}^{T} (y_t - \beta_0 - \sum_{k=1}^{p} x_{kt}\beta_k)^2.$$

- The estimates $\boldsymbol{\beta} = (\beta_0, \beta_1, \cdots, \beta_p)^T$ minimizing the above objective function are called the OLS estimates of $\boldsymbol{\beta}$ $\Rightarrow$ they are easiest to express in **matrix form**

The Model and Parameter Estimates in Matrix Form

- Matrix representation:

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{\beta} + \boldsymbol{\eta},$$

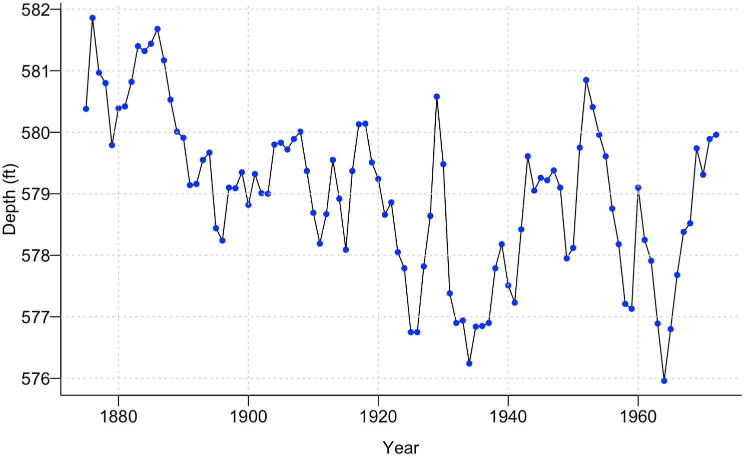
where $\boldsymbol{Y} = \begin{bmatrix} Y_1 \\ \vdots \\ Y_T \end{bmatrix}$, $\boldsymbol{X} = \begin{bmatrix} 1 & x_{11} & x_{12} & \cdots & x_{1p} \\ 1 & x_{21} & x_{22} & \cdots & x_{2p} \\ 1 & \vdots & \cdots & \cdots & \vdots \\ 1 & x_{T1} & x_{T2} & \cdots & x_{Tp} \end{bmatrix}$, and $\boldsymbol{\eta} = \begin{bmatrix} \eta_1 \\ \vdots \\ \eta_T \end{bmatrix}$

- Assuming $\boldsymbol{X}^T\boldsymbol{X}$ is **invertible**, the OLS estimate of $\boldsymbol{\beta}$ can be shown to be

$$\hat{\boldsymbol{\beta}} = (\boldsymbol{X}^T\boldsymbol{X})^{-1}\boldsymbol{X}^T\boldsymbol{Y},$$

and the `lm` function in `R` calculates OLS estimates

Lake Huron Example Revisited



Let's **assume** there is a linear trend in time $\Rightarrow$ we need to estimate the **intercept** β_0 and **slope** β_1

The R Output

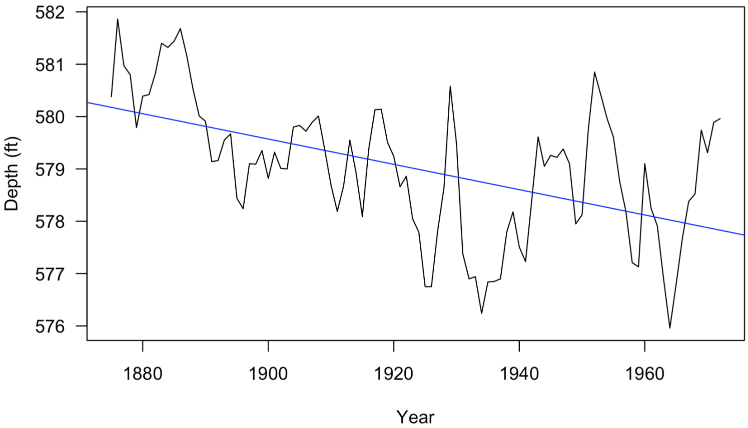
```
Call:
lm(formula = LakeHuron ~ yr)

Residuals:
     Min       1Q   Median       3Q      Max 
-2.50997 -0.72726  0.00083  0.74402  2.53565 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) 625.554918   7.764293  80.568  < 2e-16 ***
yr           -0.024201   0.004036  -5.996 3.55e-08 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

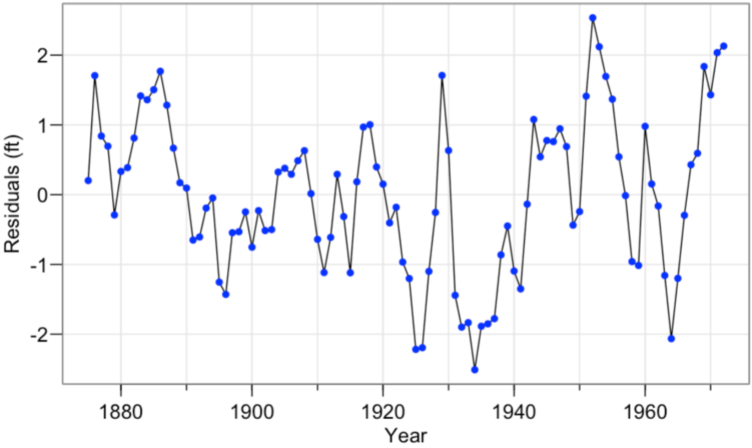
Residual standard error: 1.13 on 96 degrees of freedom
Multiple R-squared:  0.2725,	Adjusted R-squared:  0.2649 
F-statistic: 35.95 on 1 and 96 DF,  p-value: 3.545e-08
```

Plot the (Estimated) Trend $\hat{\mu}_t = \hat{\beta}_0 + \hat{\beta}_1 t$



$\hat{\beta}_1 = -0.0242$ (ft/yr) $\Rightarrow$ there seems to be a decreasing trend

Plot the Residuals $\{\hat{\eta}_t = y_t - \hat{\beta}_0 - \hat{\beta}_1 t\}$

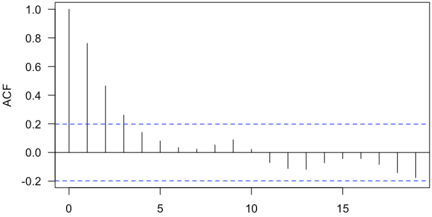


$\{\hat{\eta}_t\}$ seems to exhibit some temporal dependence structure, should we worry about the results we have (recall OLS makes an i.i.d. assumption)?

Statistical Properties of the OLS Estimates with Correlated Errors

- Assume the components of $\boldsymbol{X}$ are not random, the OLS estimates $\hat{\boldsymbol{\beta}}$ are unbiased for $\boldsymbol{\beta}$
  Proof:

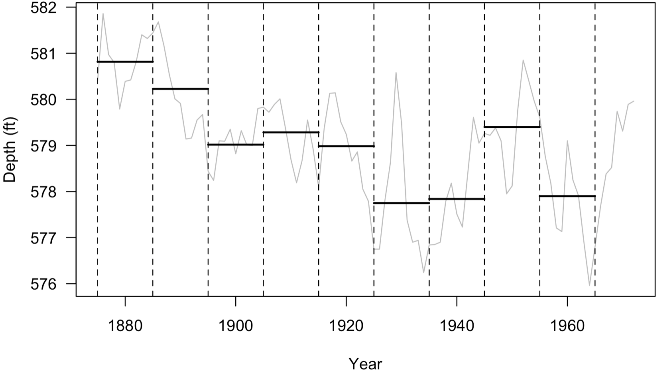
- Since $\{\eta_t\}$ is typically not an i.i.d. process (see the acf plot below), statistical inferences regarding $\boldsymbol{\beta}$ will be invalid



Smoothing or Local Averaging

In certain situations, we may want to relax the assumption on the trend ⇒ “non-parametric” approach

Here, we break the time series up into “small” blocks (each with 10 years of data) and average each block

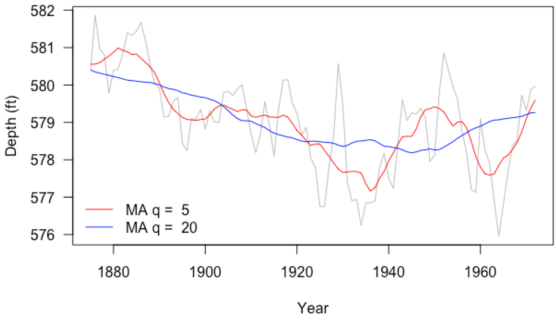


Doing this gives a very rough estimate of the trend. **Can we do better?**

Moving Average Smoother

- A moving average smoother estimates the trend at time t by averaging the current observation and the q nearest observations from either side. That is

$$\hat{\mu}_t = \frac{1}{2q+1} \sum_{j=-q}^{q} y_{t-j}$$



- q is the "smoothing" parameter, which controls the smoothness of the estimated trend $\hat{\mu}_t$

Exponential Smoothing

- Let $\alpha \in [0,1]$ be some fixed constant, defined

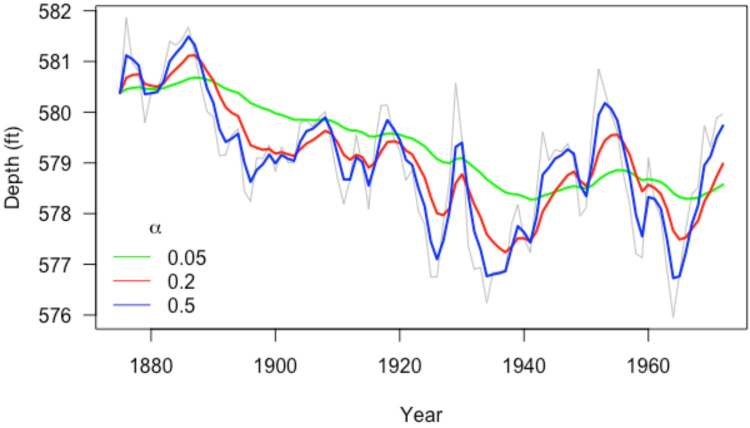
$$\hat{\mu}_t = \begin{cases} Y_1 & \text{if } t = 1; \\ \alpha Y_t + (1-\alpha)\hat{\mu}_{t-1} & t = 2, \cdots T. \end{cases}$$

- For $t = 2, \cdots, T$, we can rewrite $\hat{\mu}_t$ as

$$\sum_{j=0}^{t-2} \alpha(1-\alpha)^j Y_{t-j} + (1-\alpha)^{t-1} Y_1.$$

$\Rightarrow$ it is a one-sided moving average filter with exponentially decreasing weights. One can alter α to control the amounts of smoothing (see next slide for an example)

α is the Smoothing Parameter for Exponential Smoothing



The smaller the α, the smoother the resulting trend

Differencing

- We define the first order difference operator ∇ as

$$\nabla Y_t = Y_t - Y_{t-1} = (1 - B)Y_t,$$

where B is the **backshift operator** and is defined as $BY_t = Y_{t-1}$.

- Similarly the general order difference operator $\nabla^q Y_t$ is **defined recursively** as $\nabla[\nabla^{q-1} Y_t]$

- The backshift operator of power q is defined as $B^q Y_t = Y_{t-q}$

In next slide we will see an example regarding the relationship between ∇^q and B^q

Difference and Backshift Notation

The second order difference is given by

$$\nabla^2 Y_t = \nabla[\nabla Y_t]$$

Difference and Backshift Notation

The second order difference is given by

$$\begin{aligned}\nabla^2 Y_t &= \nabla[\nabla Y_t] \\ &= \nabla[Y_t - Y_{t-1}]\end{aligned}$$

Difference and Backshift Notation

The second order difference is given by

$$
\begin{aligned}
\nabla^2 Y_t &= \nabla[\nabla Y_t] \\
&= \nabla[Y_t - Y_{t-1}] \\
&= (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2})
\end{aligned}
$$

Difference and Backshift Notation

The second order difference is given by

$$\begin{aligned}
\nabla^2 Y_t &= \nabla[\nabla Y_t] \\
&= \nabla[Y_t - Y_{t-1}] \\
&= (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2}) \\
&= Y_t - 2Y_{t-1} + Y_{t-2}
\end{aligned}$$

Difference and Backshift Notation

The second order difference is given by

$$
\begin{aligned}
\nabla^2 Y_t &= \nabla[\nabla Y_t] \\
&= \nabla[Y_t - Y_{t-1}] \\
&= (Y_t - Y_{t-1}) - (Y_{t-1} - Y_{t-2}) \\
&= Y_t - 2Y_{t-1} + Y_{t-2} \\
&= (1 - 2B + B^2)Y_t
\end{aligned}
$$

In the next slide we will see an example of using differening to remove the trend

Removing Trend via Differening

Consider a time series data with a linear trend (i.e., $\{Y_t = \beta_0 + \beta_1 t + \eta_t\}$) where η_t is a stationary time series. Then first order differencing results in a stationary series with no trend. To see why

$$
\begin{aligned}
\nabla Y_t &= Y_t - Y_{t-1} \\
&= (\beta_0 + \beta_1 t + \eta_t) - (\beta_0 + \beta_1 (t-1) + \eta_{t-1}) \\
&= \beta_1 + \eta_t - \eta_{t-1}
\end{aligned}
$$

This is the sum of a stationary series and a constant, and therefore we have successfully remove the linear trend.

Notes on Differening

- A polynomial trend of order q can be removed by q-th order differencing
- By q-th order differencing a time series we are shortening its length by q
- Differencing does not allow you to estimate the trend, only to remove it. *Therefore it is not appropriate if the aim of the analysis is to describe the trend*

Seasonal Component Estimation

- Let's consider the situation that a time series consists of seasonal component only (assuming the trend has been estimated/removed), that is,

$$Y_t = s_t + \eta_t,$$

with $\{s_t\}$ having period d (i.e., $s_{t+jd} = s_t$ for all integers j and t), $\sum_{t=1}^{d} s_t = 0$ and $\mathbb{E}(\eta_t) = 0$

- Two methods to estimate $\{s_t\}$
  - Harmonic regression
  - Seasonal mean model
- A method to remove $\{s_t\} \Rightarrow$ Lag differencing

Harmonic Regression

- A harmonic regression model has the form

$$s_t = \sum_{j=1}^{k} A_k \cos(2\pi f_j + \phi_j).$$

For each $j = 1, \cdots, k$:

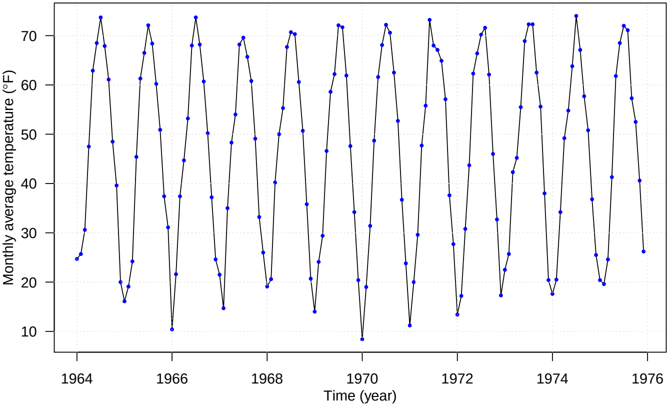
- $A_j > 0$ is the amplitude of the j-th cosine wave
- f_j controls the the frequency of the j-th cosine wave (how often waves repeats)
- $\phi_j \in [-\pi, \pi]$ is the phase of the j-th wave (where it starts)

- The above can be expressed as

$$\sum_{j=1}^{k} \left(\beta_{1j} \cos(2\pi f_j) + \beta_{2j} \sin(2\pi f_j)\right),$$

where $\beta_{1j} = A_j \cos(\phi_j)$ and $\beta_{2j} = A_j \sin(\phi_j) \Rightarrow$ if $\{f_j\}_{j=1}^{k}$ are known, **we can use regression techniques to estimate the parameters** $\{\beta_{1j}, \beta_{2j}\}_{j=1}^{k}$

Monthly Average Temperature in Dubuque, IA [Cryer & Chan, 2008]



Let's assume there is trend in this time series. Here we want to estimate s_t, the seasonal component

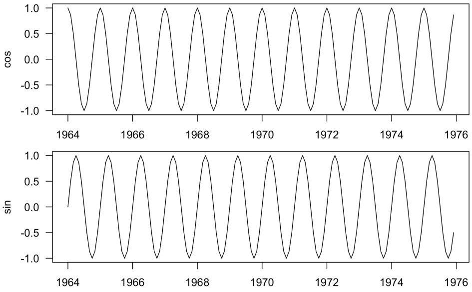
Use a Harmonic Regression to Model Annual Cycles

Model: $s_t = \beta_0 + \beta_1 \cos(2\pi t) + \beta_2 \sin(2\pi t)$

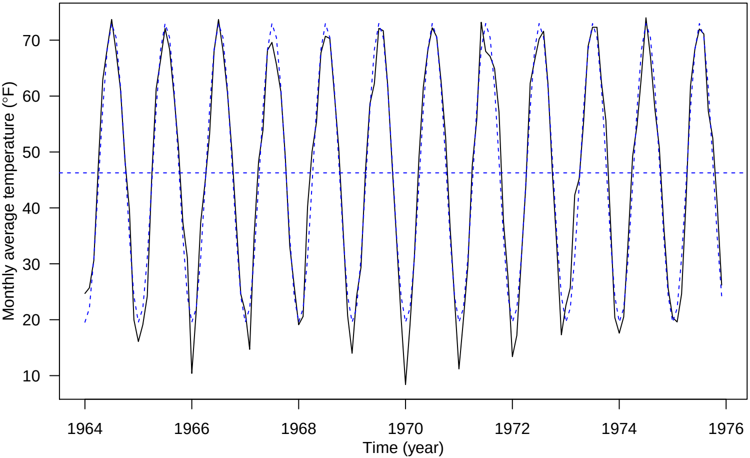
$\Rightarrow$ annual cycles can be modeled by a linear combination of $\cos$ and $\sin$ with 1-year period.

In R, we can easily create these harmonics using the `harmonic` function in the `TSA` package

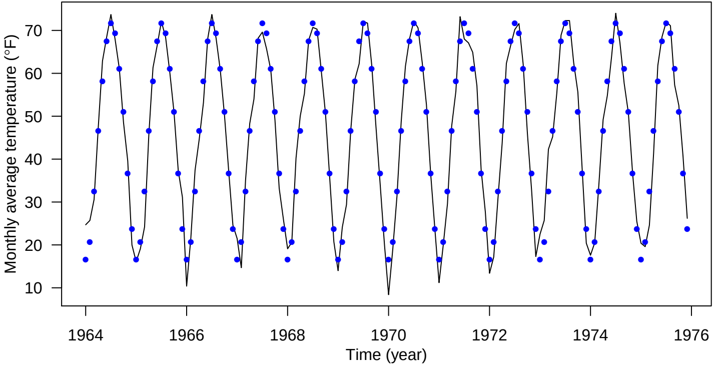
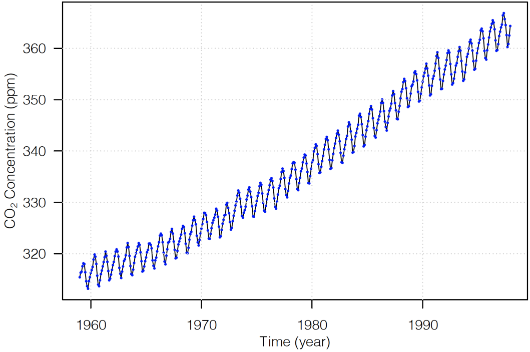
```
harmonics <- harmonic(tempdub, 1)
```



```
```{r}
harReg <- lm(tempdub ~ harmonics)
summary(harReg)
```

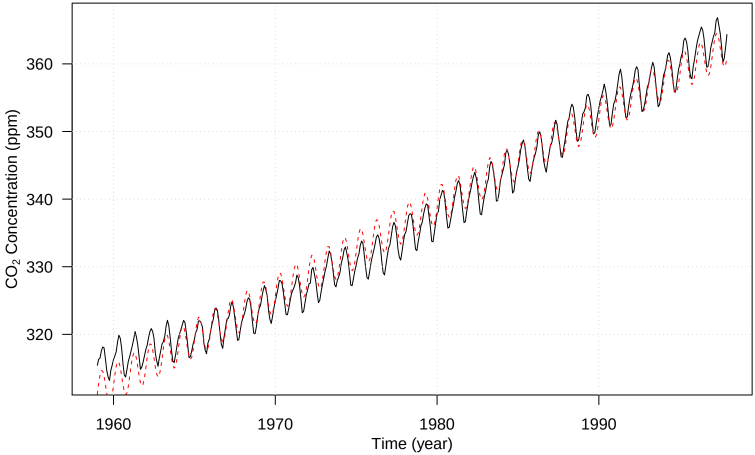
```
Call:
lm(formula = tempdub ~ harmonics)

Residuals:
     Min       1Q   Median       3Q      Max 
-11.1580  -2.2756  -0.1457   2.3754  11.2671 

Coefficients:
                      Estimate Std. Error t value Pr(>|t|)    
(Intercept)            46.2660     0.3088 149.816  < 2e-16 ***
harmonicscos(2*pi*t) -26.7079     0.4367 -61.154  < 2e-16 ***
harmonicssin(2*pi*t)  -2.1697     0.4367  -4.968 1.93e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```
```

The Harmonic Regression Model Fit



Seasonal Means Model

- **Harmonics regression** assumes the seasonal pattern has a regular shape, i.e., the height of the peaks is the same as the depth of the troughs

- A less restrictive approach is to model $\{s_t\}$ as

$$s_t = \begin{cases} \beta_1 & \text{for } t = 1, 1+d, 1+2d, \cdots & ; \\ \beta_2 & \text{for } t = 2, 2+d, 2+2d, \cdots & ; \\ \vdots & \vdots & ; \\ \beta_d & \text{for } t = d, 2d, 3d, \cdots & . \end{cases}$$

- This is the seasonal means model, the parameters $(\beta_1, \beta_2, \cdots, \beta_d)^T$ can be estimated under the linear model framework (think about ANOVA)

R Output

```
Call:
lm(formula = tempdub ~ month - 1)

Residuals:
    Min      1Q  Median      3Q     Max 
-8.2750 -2.2479  0.1125  1.8896  9.8250 

Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
monthJanuary     16.608      0.987   16.83   <2e-16 ***
monthFebruary    20.650      0.987   20.92   <2e-16 ***
monthMarch       32.475      0.987   32.90   <2e-16 ***
monthApril       46.525      0.987   47.14   <2e-16 ***
monthMay         58.092      0.987   58.86   <2e-16 ***
monthJune        67.500      0.987   68.39   <2e-16 ***
monthJuly        71.717      0.987   72.66   <2e-16 ***
monthAugust      69.333      0.987   70.25   <2e-16 ***
monthSeptember   61.025      0.987   61.83   <2e-16 ***
monthOctober     50.975      0.987   51.65   <2e-16 ***
monthNovember    36.650      0.987   37.13   <2e-16 ***
monthDecember    23.642      0.987   23.95   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The Seasonal Means Model Fit



- The lag-d difference operator, ∇_d, is defined by

$$\nabla_d Y_t = Y_t - Y_{t-d} = (1 - B^d) Y_t.$$

Note: This is NOT ∇^d!

- **Example**: Consider data that arise from the model $Y_t = \beta_0 + \beta_1 t + s_t + \eta_t$, which has a linear trend and seasonal component that repeats itself every d time points. Then by just seasonal differencing (lag-d differening here) this series becomes stationary.

$$\begin{aligned}
\nabla_d Y_t &= Y_t - Y_{t-d} \\
&= [\beta_0 + \beta_1 t + s_t + \eta_t] - [\beta_0 + \beta_1 (t-d) + s_{t-d} + \eta_{t-d}] \\
&= d\beta_1 + \eta_t - \eta_{t-d}
\end{aligned}$$

Estimating the Trend and Seasonal variation Together



Let's perform a regression analysis to model both μ_t (assuming a linear time trend) and s_t (using $\cos$ and $\sin$)

```
```{r}
time <- as.numeric(time(co2))
harmonics <- harmonic(co2, 1)

lm_trendSeason <- lm(co2 ~ time + harmonics)
summary(lm_trendSeason)
```
```

The Regression Fit



Seasonal and Trend decomposition using Loess [Cleveland, et. al., 1990]

```{r}
# Seasonal and Trend decomposition using Loess (STL)
par(mar = c(4, 3.6, 0.8, 0.6))
stl <- stl(co2, s.window = "periodic")
plot(stl, las = 1)
```