Time Series Analysis
Lecture 2

Regression With Time Series, an Intro to Exploratory Time Series Data Analysis, and Time Series Smoothing

datascience@berkeley

Classical Linear Regression Model for Time Series Data

Classical Linear Regression Revisit

Classical Linear Regression Revisit

Recall that a classical linear regression model takes the following conditional mean functional form with the following assumptions:

$$y_t = \beta_0 + \beta_1 x_{t1} + \beta_2 x_{t2} + \cdots + +\beta_k x_{k1} + \epsilon_t$$

where the β's are (unknown) regression parameters and ϵ_t represents a stochastic (error) process with random variables with mean 0 and variance σ_ϵ^2. If we also assume that the random variables follow the Gaussian distribution, then this setup gives the Classical Normal Linear Regression Model.

- It is a good place to review (again) your notes from w203 on linear regression modeling (regression model setup, , the coefficient estimates, SEs, t-stat, p-value, residual SE and the residual DF, R-squared, Adjusted R-squared, F-stat (and the associated DF), and p-value (of the regression), ANOVA for regression, etc).

Linear Time Trend Regression

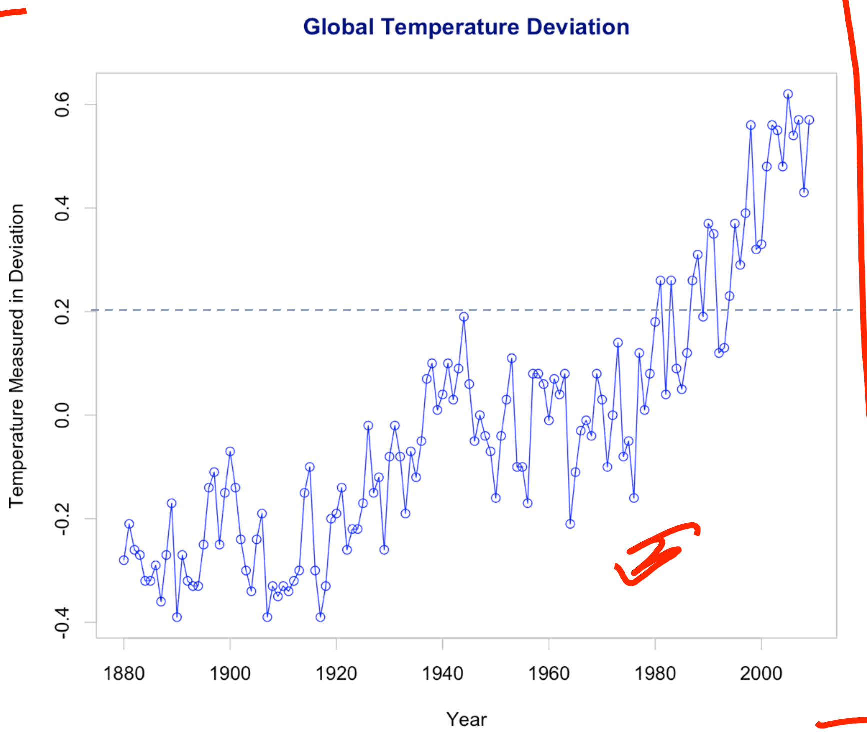
Linear Time Trend Regression

- Use the global temperature (measured in deviation) data for illustration.
- Always a good idea to look at the structure of the data after loading the data.

```
> str(gtemp)
 Time-Series [1:130] from 1880 to 2009: -0.28 -0.21 -0.26 -0.27 -0.32 -0.32 -0.29 -0
.36 -0.27 -0.17 ...
```

```
# Visualize the series
plot(gtemp, type="o", col="blue", fg=16,
     main="Global Temperature Deviation", col.main="Navy",
     xlab="Year",
     ylab="Temperature Measured in Deviation")
```

- Although the (deviation) series fluctuates from year to year, it exhibits an upward trend starting in around 1920, and the temperature has an annual increase of at least 0.2 degree per year since around the mid-1990s.



Use a Linear Time Trend Regression to Model the Global Temperature (Deviation) Series

Consider the following linear time trend regression model:

$$y_t = \beta_0 + \beta_1 t + \epsilon_t$$

where t in this case takes values in the following set $1880, \ldots, 2009$ and ϵ_t is an $i.i.d.$ normal sequence. Note that this assumption needs to be checked later on.

Note that we can shift the time index to $1, 2, \ldots, 130$ and not affect the interpretation of the slope coefficient β_1; it only affects the intecept β_0.

Linear Time Trend Regression

Keep in mind that we are estimating the following linear regression model:

$$y_t = \beta_0 + \beta_1 t + \epsilon_t$$

```
Call:
lm(formula = gtemp ~ time(gtemp))

Residuals:
     Min       1Q   Median       3Q      Max 
-0.31946 -0.09722  0.00084  0.08245  0.29383 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -1.120e+01  5.689e-01  -19.69   <2e-16 ***
time(gtemp)  5.749e-03  2.925e-04   19.65   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1251 on 128 degrees of freedom
Multiple R-squared:  0.7511,    Adjusted R-squared:  0.7492 
F-statistic: 386.3 on 1 and 128 DF,  p-value: < 2.2e-16
```

Note the use of the time() function.

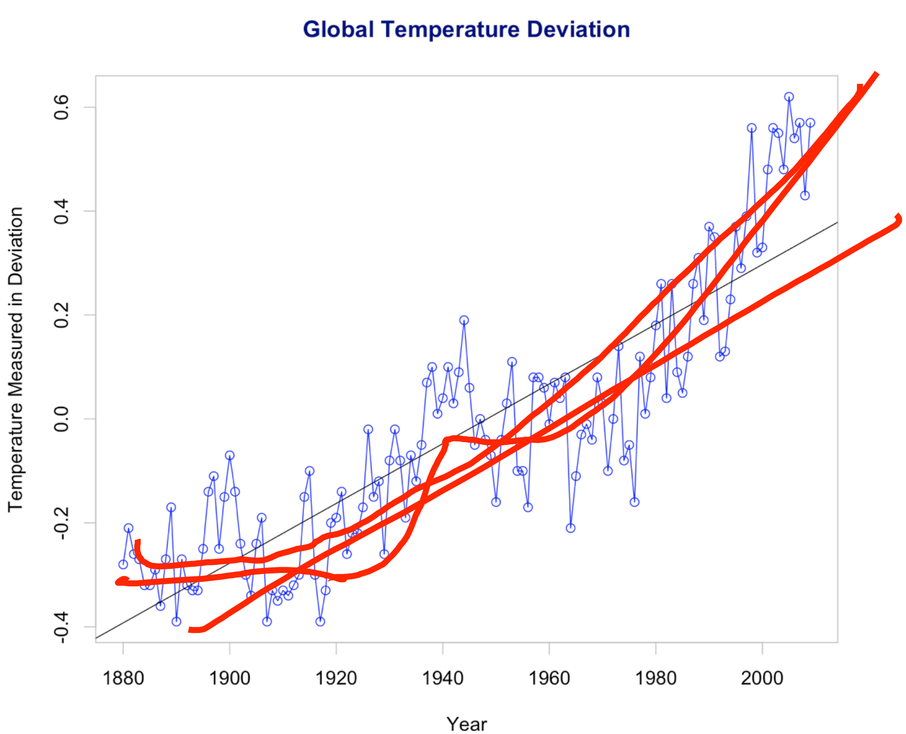
```
> head(time(gtemp))
[1] 1880 1881 1882 1883 1884 1885
```

The estimated regression line takes the following form

$$\hat{y}_t = -11.2 + 0.006t$$

The standard errors are 0.0569 and 0.0003 for $\hat{\beta}_0$ and $\hat{\beta}_1$, respectively.

Importantly, this model, even without verifying the underlying statistical assumption, obviously does not do a good job capturing the pattern of the series, despite the relatively decent R^2.



Goodness of Fit Measures (for Time Series Models)

Goodness of Fit Measure: AIC, AICc, BIC

Akaike's Information Criterion (AIC)

$$\mathrm{AIC} = \log\ \widehat{\sigma}_k^2 + \frac{n+2k}{n} \qquad \left(\widehat{\sigma}_k^2 = \frac{SSE_k}{n}\right) \qquad \left[SSE = \sum_{t=1}^{n}(x_t - \widehat{\beta}' z_t)^2\right]$$

where **k** is the number of parameters in the model and **n** denotes the sample size.

Biased-Corrected Akaike's Information Criterion (AICc)

$$\mathrm{AICc} = \log\ \widehat{\sigma}_k^2 + \boxed{\frac{n+k}{n-k-2}}$$

Bayesian's Information Criterion (BIC)

$$\mathrm{BIC} = \log\ \widehat{\sigma}_k^2 + \left(\frac{k \log n}{n}\right)$$

Time Series Regression: Example 2

Time Series Regression: Example 2

- One could regress one time series on another. For a pure illustration purpose, we use the Southern Oscillation Index (SOI) and Recruitment series as an example. This dataset is used in a few examples in Chapters 1 and 2 in our textbook.

- For instance, it is shown in Chapter 1 that the lag values of SOI are correlated to the current value of Recruitment. One model we can entertain is the following simple mode:

S_t

$$R_t = \beta_1 + \beta_2 S_{t-6} + w_t,$$

where R_t denotes the Recruitment for month t and S_{t-6} denotes the SOI six months prior.

- The estimated model is

$$\widehat{R}_t = 65.79 - 44.28_{(2.78)} S_{t-6}$$

with $\widehat{\sigma}_w = 22.5$ on 445 degrees of freedom

Time Series Regression: Example 2

- One could regress one time series on another. For a pure illustration purpose, we use the Southern Oscillation Index (SOI) and Recruitment series as an example. This dataset is used in a few examples in Chapters 1 and 2 in our textbook.

- For instance, it is shown in Chapter 1 that the lag values of SOI are correlated to the current value of Recruitment. One model we can entertain is the following simple mode:

$$R_t = \beta_1 + \beta_2 S_{t-6} + w_t,$$

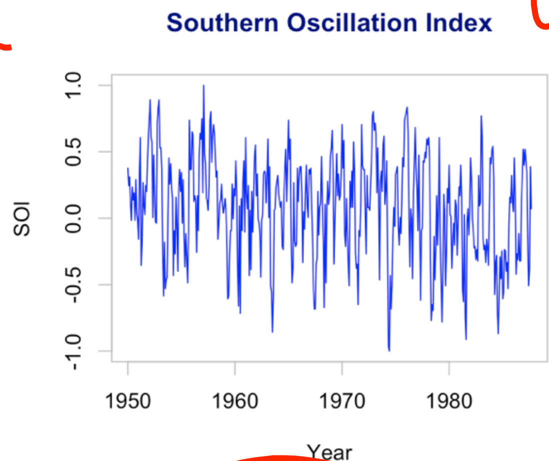
where R_t denotes the Recruitment for month t and S_{t-6} denotes the SOI six months prior.

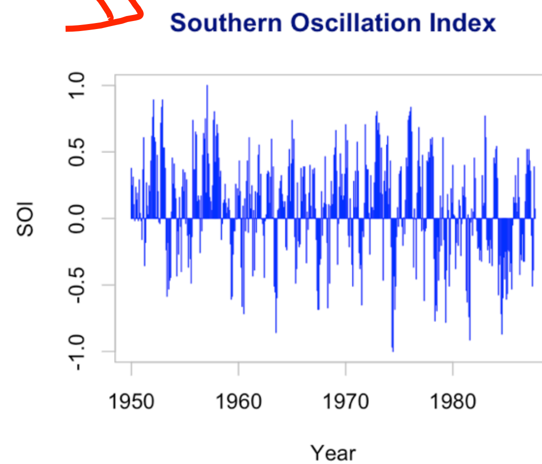
- The estimated model is

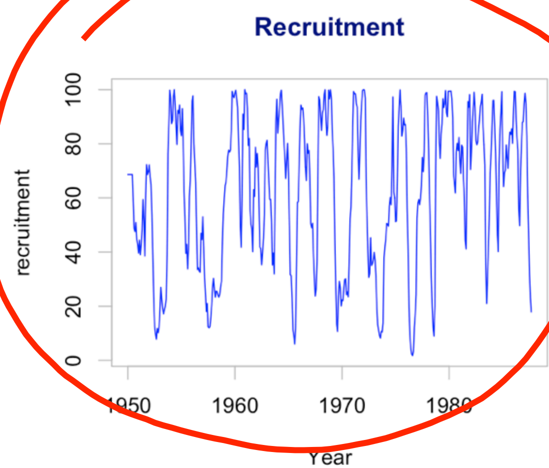
$$\widehat{R}_t = 65.79 - 44.28_{(2.78)} S_{t-6}$$

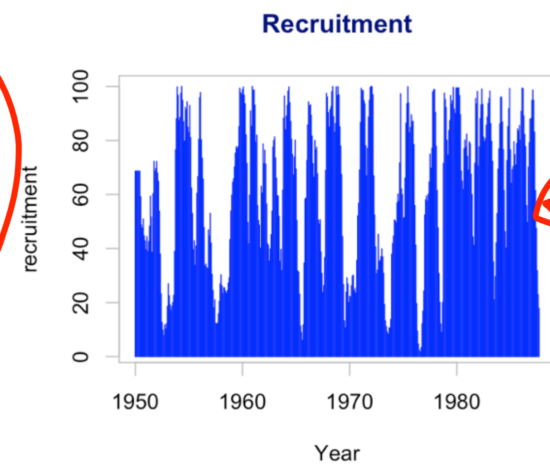
with $\widehat{\sigma}_w = 22.5$ on 445 degrees of freedom.

Visualize the Series









Estimate a Model With Lag SOI

```
> str(rec)
 Time-Series [1:453] from 1950 to 1988: 68.6 68.6 68.6 68.6 68.6 ...
> str(soi)
 Time-Series [1:453] from 1950 to 1988: 0.377 0.246 0.311 0.104 -0.016
```

```
# time index alignment
fish = ts.intersect(rec, soiL6=lag(soi,-6))
summary(fit2 <- lm(rec ~ soiL6, data=fish,na.action = NULL))
```

$$\widehat{R}_t = 65.79 - 44.28_{(2.78)} S_{t-6}$$

```
Call:
lm(formula = rec ~ soiL6, data = fish, na.action = NULL)

Residuals:
    Min      1Q  Median      3Q     Max
-65.187 -18.234   0.354  16.580  55.790

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   65.790      1.088   60.47   <2e-16 ***
soiL6        -44.283      2.781  -15.92   <2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 22.5 on 445 degrees of freedom
Multiple R-squared:  0.3629,	Adjusted R-squared:  0.3615
F-statistic: 253.5 on 1 and 445 DF,  p-value: < 2.2e-16
```

Time Series Smoothing Techniques: Introduction and Mathematical Formulation

Introduction to Smoothing Techniques

- Smoothing techniques ("smoothers") are often used to uncover trend and cyclical components of a series.

- The general concept of a smoothing technique that it is formed using a **weighted average of pass values** of a series.

- We will discuss some popular smoothing techniques:
    1. Moving averages
    2. Polynomial and periodic regression smoothers
    3. Spline smoothers
    4. Kernel smoothers

- We will
    - Define the mathematical form of each of these smoothers.
    - Illustrate each of these techniques and their empirical patterns using two examples. The dataset used in one of the examples can be downloaded directly from the Federal Reserve's website.

1. Symmetric Moving Average Smoother

A symmetric moving average smoother takes the following formulation:

$$m_t = \sum_{j=-k}^{k} a_j x_{t-j}$$

where $a_j = a_{j-1} \geq 0$ and the sum of the weights equal to one: $\sum_{j=-k}^{k} a_j = 1$

1. Symmetric Moving Average Smoother

Setting $k = 2$ essentially gives a monthly series, if the underlying series is a weekly series, and can help bring out the seasonality pattern, if exists:

$$m_t = \frac{1}{5}\sum_{j=-2}^{2} x_{t-j}$$
$$= \frac{1}{5}\left(x_{t-2} + x_{t-1} + x_t + x_{t+1} + x_{t+2}\right)$$

where $a_j = \frac{1}{5} \forall a_j$

1. Symmetric Moving Average Smoother

Setting $k = 26$ essentially gives an annual series, if the underlying series is a weekly series, and can help identify the long-term trend underlying the series:

$$\begin{aligned} m_t &= \frac{1}{53} \sum_{j=-26}^{26} x_{t-j} \\ &= \frac{1}{55} \left(x_{t-26} + x_{t-25} + \cdots + x_t + \cdots + x_{t+25} + x_{t+26} \right) \end{aligned}$$

where $a_j = \frac{1}{53} \forall a_j$

2. Regression and Periodic Smoothers

Another class of time series smoothing technique has the following general setup:

$$x_t = f_t + z_t$$

where f_t is some smooth function of time and z_t is a stationary process. One choice of f_t is a polynomial:

$$\boxed{f_t = \sum_{i=0}^{p} \beta_i t^i}$$

For periodic data, **periodic function** is used:

$$f_t = \sum_{i=0}^{p} \alpha_i cos(2\pi\omega_i t)\beta_i sin(2\pi\omega_i t)$$

where $cos(2\pi\omega_0 t) = sin(2\pi\omega_0 t) = 1$, and $\omega_1 \ldots \omega_p$ are distinct, specified frequencies.

The polynomial and periodic polynomial functions can be combined as one smoother in a classical linear regression.

3. Spline Smoother

Smoothing splines Extending the polynomial regression as a smoothing technique is to use spline function.

Consider dividing the modeling time horizon into k mutual exclusive and exhaustive intervals:

$$[t_0 = 1, t_1], [t_1 + 1, t_2], \ldots, [t_{k-1} + 1, t_k = n]$$

where $t_0, t_1, \ldots, t_k$ are called knots.

The generalization of the polynomial regression comes from the fact that one fits a regression of the form

$$f_t = \beta_0 + \beta_1 t + \cdots + \beta_p t^p$$

in each of the time intervals defined above. When $p = 3$, it is called *cubic splines*.

3. Spline Smoother

Smoothing splines technique modifies the spline method by incorporating the penalized smoothness component in the objective function such that the minimization problem accounts for the trade-off between the model fit and the degree of smoothness. The objective function is written as

$$\sum_{t=1}^{n} [x_t - f_t]^2 + \lambda \int \left(f_t''\right)^2 dt$$

where f_t is a cubic spline with a knot at each t and λ is the smoothing parameter

4. Kernel Smoother

Kernel Smoothing is a symmetric moving average smoother with a probability density weight function.

$$\hat{f}_t = \sum_{i=1}^{n} w_i(t) x_i$$

where

$$w_i(t) = \frac{K\left(\frac{t-i}{b}\right)}{\sum_{j=1}^{n} K\left(\frac{t-i}{b}\right)}$$

Some example kernel functions are…

Time-Series Smoothing Techniques: Examples Using a Real-World Time Series

Initial Unemployment Insurance Claim

- This example make uses of the initial unemployment insurance claim (or initial claims) data collected by the Bureau of Labor Statistics (BLS) of the U.S. Department of Labor.

- This is one of the most watched economic measures by policy makers, business leaders, economists, professionals in many fields, and consumers. It tracks the number of people who have **filed jobless claims for the first time during the specified period** with the appropriate government labor office. This number represents an inflow of people receiving unemployment benefits.

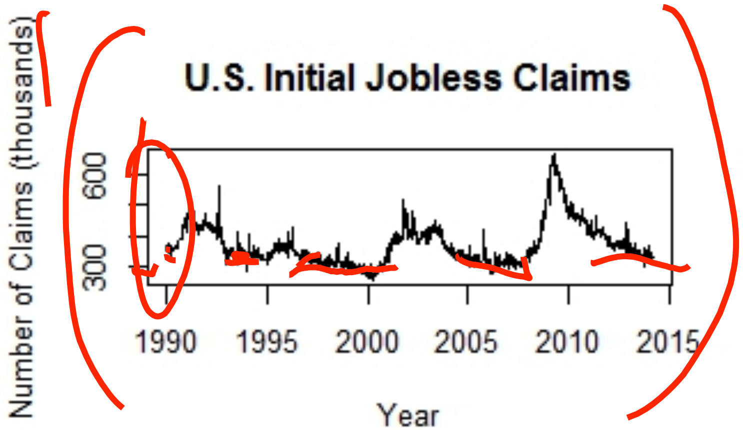
- **Definition of initial UI claims:** "The Department of Labor's Unemployment Insurance (UI) programs provide unemployment benefits to eligible workers who become unemployed through no fault of their own, and meet certain other eligibility requirements." (http://www.dol.gov/dol/topic/unemployment-insurance/)

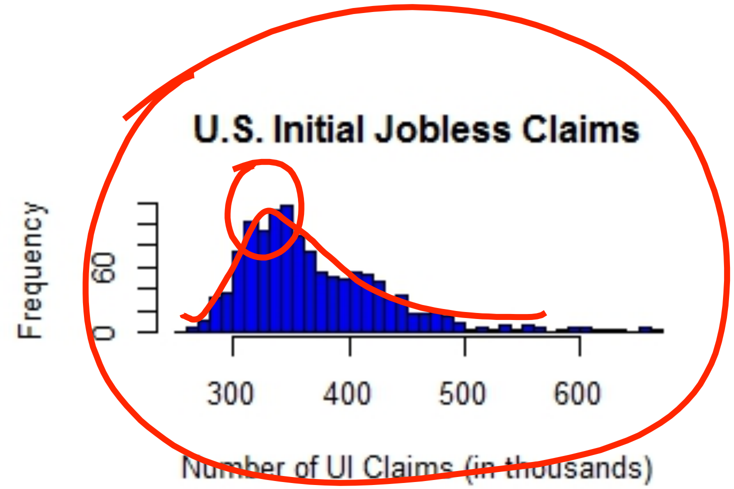
- The **UI weekly claims data** are used in current economic analysis of unemployment trends in the nation and in each state. Initial claims measure emerging unemployment, and continued weeks claimed measure the number of persons claiming unemployment benefits.

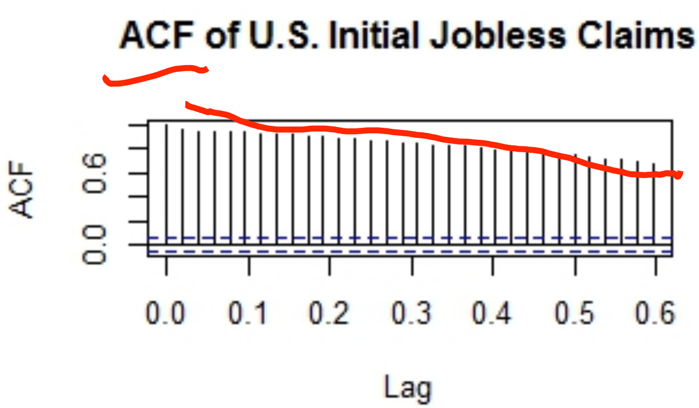
- After importing the data to R, we first look at the basic structure of the data, such as number of variables, number of observations, time intervals, and total time period it is covered.
- This is an important practice and can be used to check if the dataset loaded to R is the same as the original data. It is particular important if the dataset is large.

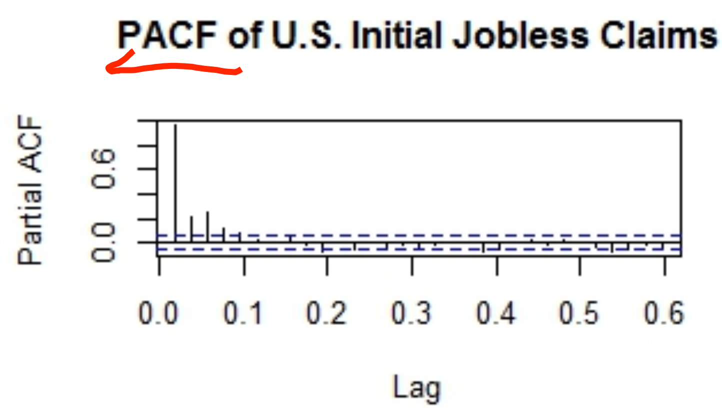
```
'data.frame':	1300 obs. of  3 variables:
 $ Date    : chr  "5-Jan-90" "12-Jan-90" "19-Jan-90" "26-Jan-90" ...
 $ INJCJC  : int  355 369 375 345 368 367 348 350 351 349 ...
 $ INJCJC4: num  362 366 364 361 364 ...
> dim(data1)
[1] 1300    3
> head(data1)
       Date INJCJC  INJCJC4
1  5-Jan-90    355  362.25
2 12-Jan-90    369  365.75
3 19-Jan-90    375  364.25
4 26-Jan-90    345  361.00
5  2-Feb-90    368  364.25
6  9-Feb-90    367  363.75
```

- Next we will look at the time plot, density plot, and scatter plot matrix of the current vs. the lag values, ACF, and PACF.
- Since we have already introduced the initial UI data series earlier in the lecture, I will not spend time on these graphs.

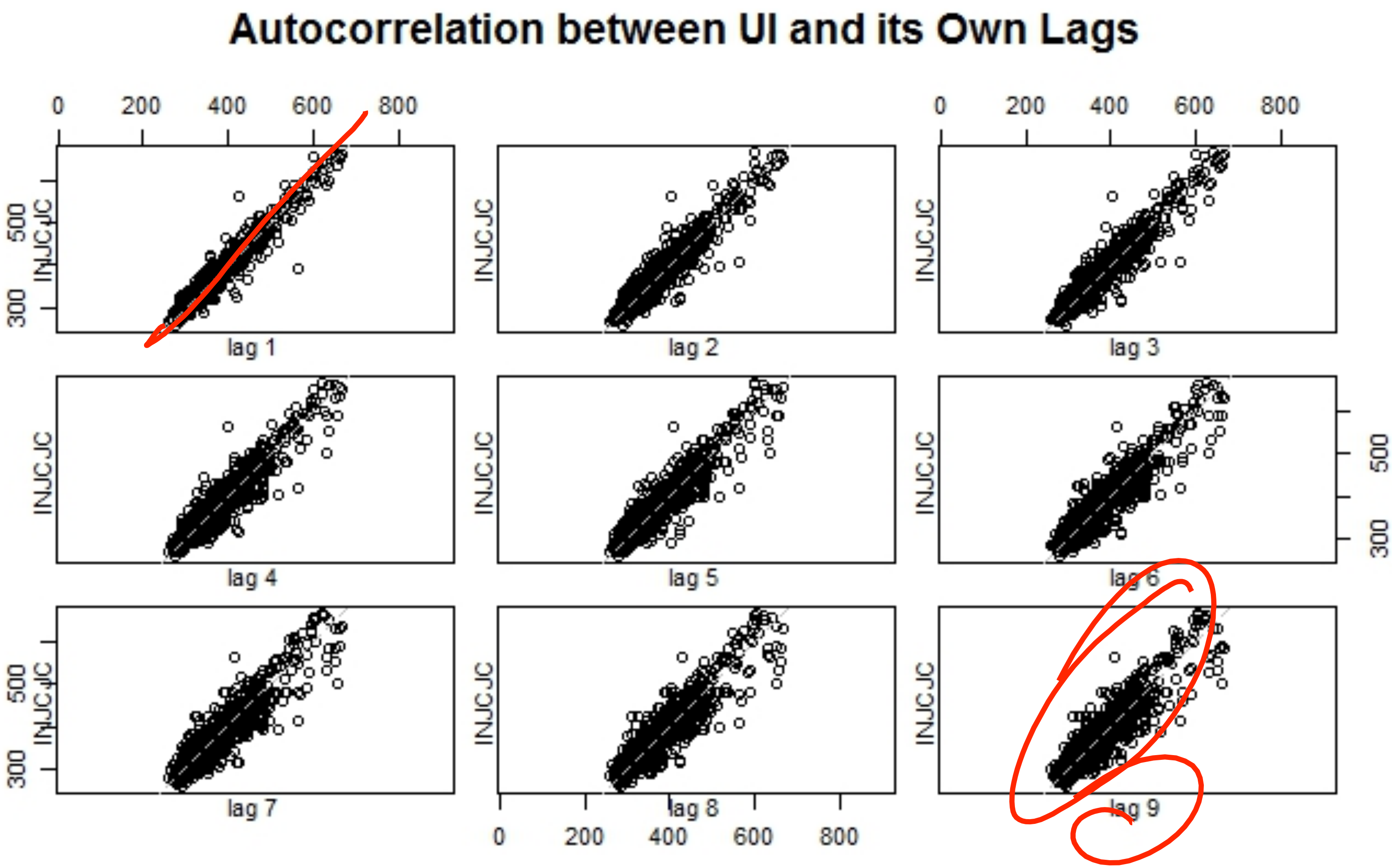








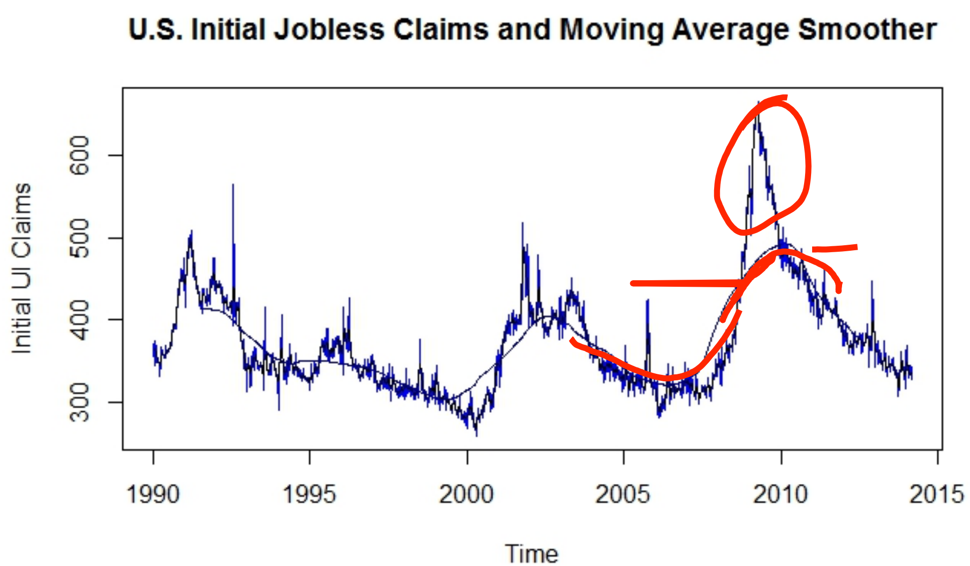
- Next we will look at the time plot, density plot, and scatter plot matrix of the current vs. the lag values, ACF, and PACF.



Symmetric Moving Average Smoother

- Recall that when applying a smoothing technique, we need to pay attention to the smoother parameter. Some smoothing techniques require an explicit specification of the smoothing parameter.
- In the case of moving average smoother, the "implicit" smoothing parameter is the number of values used in calculating each of the moving average value.

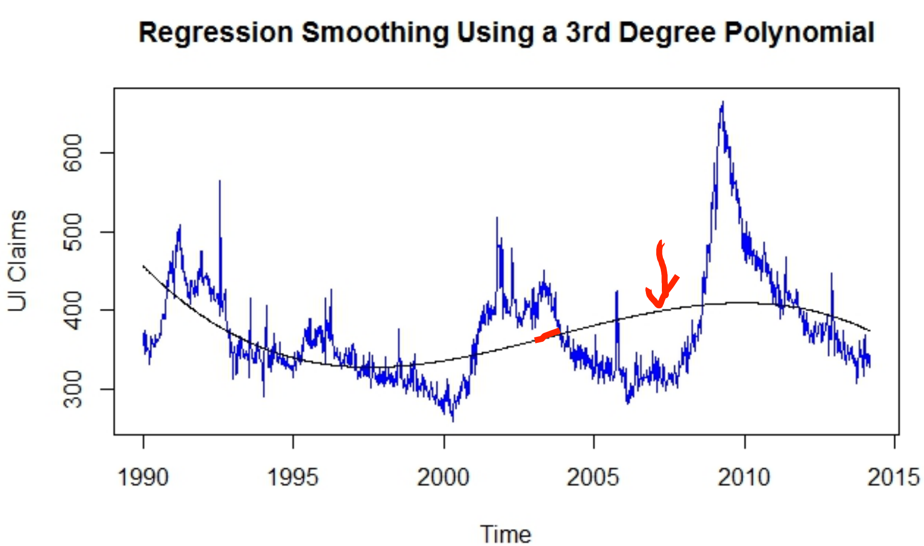
```
ma5 = filter(INJCJC, sides=2, rep(1,5)/5)
ma157 = filter(INJCJC, sides=2, rep(1,157)/157)
```



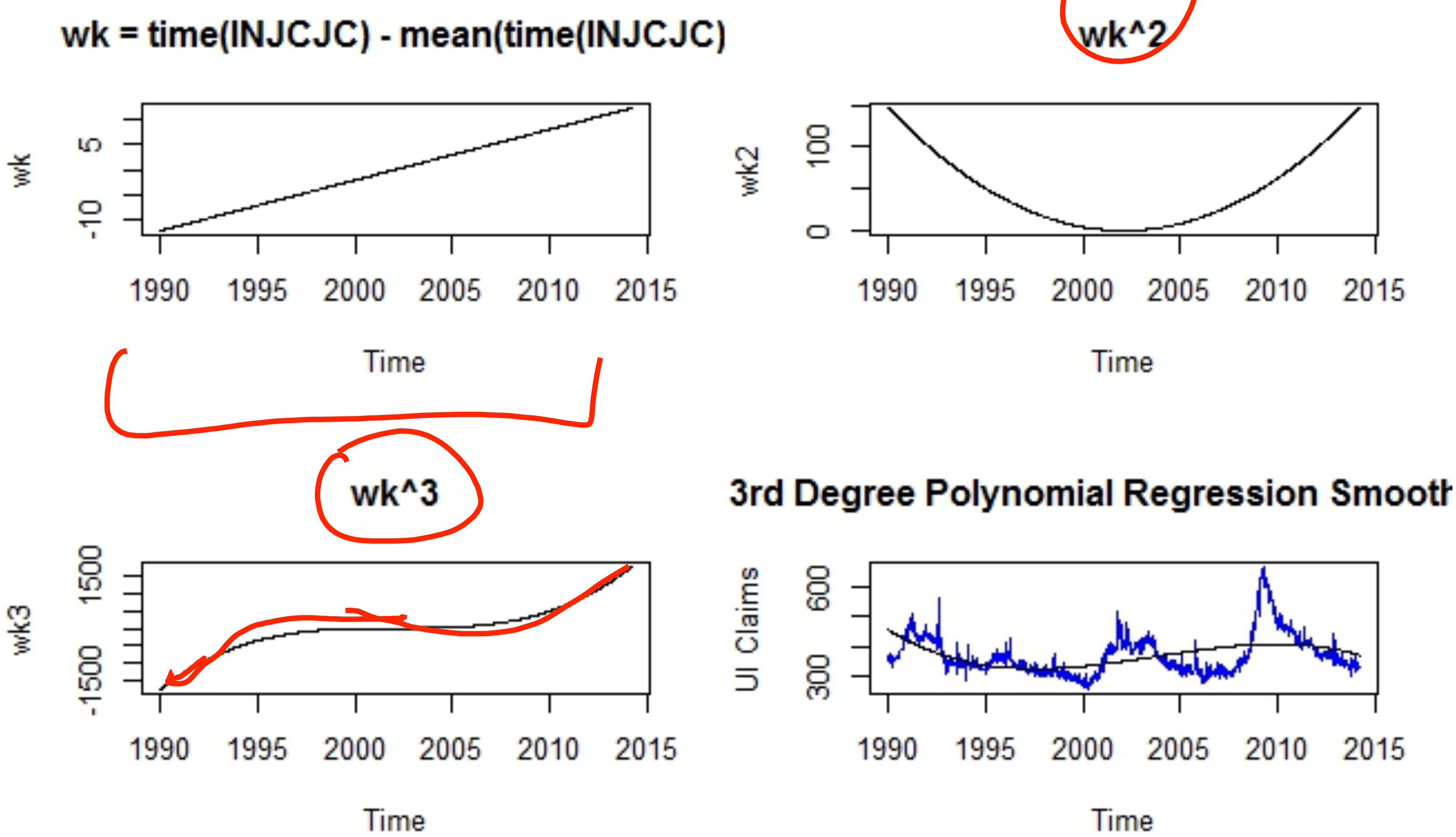
Regression Smoother

- A regression smoothing using a third-degree polynomial regression.
- It could capture the general shape of the series at the first 10 years of the historical observations, but it does not capture well the trend dynamics in the last 15 years of the historical observation period.

```
wk = time(INJCJC) - mean(time(INJCJC))
wk2 = wk^2
wk3 = wk^3
reg1 = lm(INJCJC~wk + wk2 + wk3, na.action=NULL)
plot(INJCJC, type="l", col="blue",
     main="Regression Smoothing Using a 3rd Degree Polynomial",
     ylab="UI Claims")
lines(fitted(reg1), col="black")
```



- A regression smoothing using a third-degree polynomial regression.
- It could capture the general shape of the series at the first 10 years of the historical observations.



Spline Smoothers

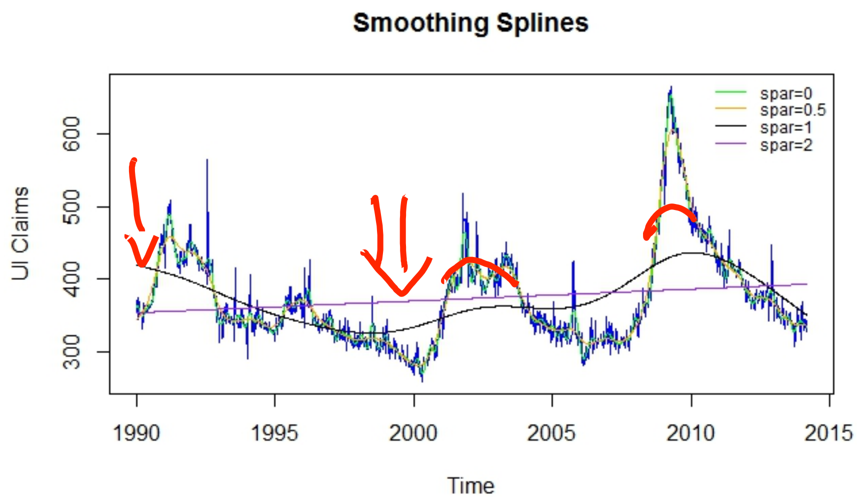
- Analogous to the role bandwidth plays in a kernel smoother, lambda is the smoothing parameter in the spline smoothers.
- We estimate four spline smoothers using different smoother parameter values.

```
plot(INJCJC, type="l", col="blue",
     main="Smoothing Splines",
     ylab="UI Claims")
lines(smooth.spline(time(INJCJC), INJCJC, spar=0)  , col="green")
lines(smooth.spline(time(INJCJC), INJCJC, spar=0.5), col="orange")
lines(smooth.spline(time(INJCJC), INJCJC, spar=1)  , col="black")
lines(smooth.spline(time(INJCJC), INJCJC, spar=2)  , col="purple")
```

$$\sum_{t=1}^{n}[x_t - f_t]^2 + \lambda \int \left(f_t''\right)^2 dt$$

f_t is a cubic spline with a knot at each t

λ is the smoothing parameter

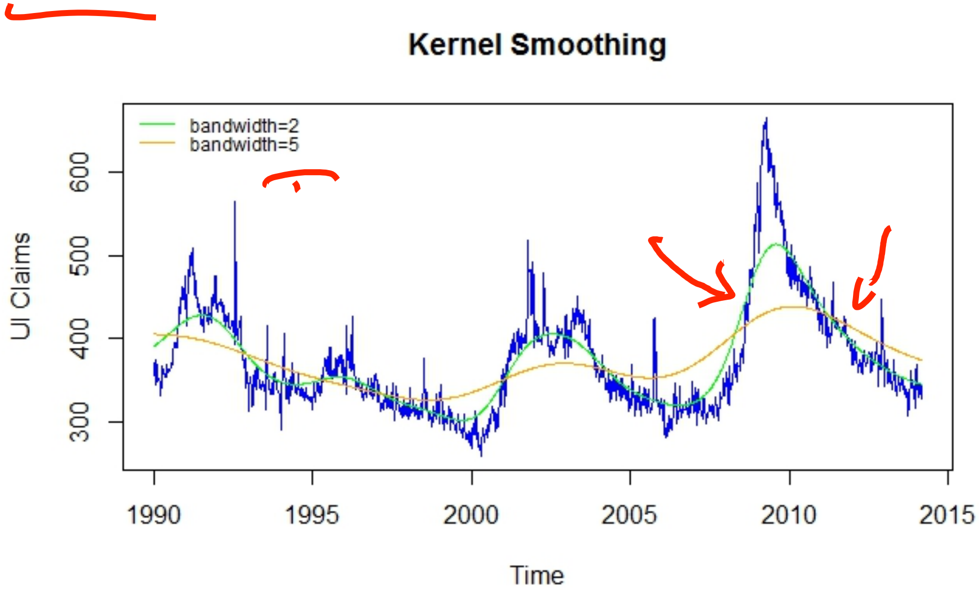


Kernel Smoothers

- For kernel smoother, the choice of bandwidth is critical, as it has a great influence on the smoothness of the kernel estimates.
- The choice of the kernel function, however, is less important.

$$\hat{f}_t = \sum_{i=1}^{n} w_i(t) x_i \qquad w_i(t) = \frac{K\left(\frac{t-i}{b}\right)}{\sum_{j=1}^{n} K\left(\frac{t-i}{b}\right)}$$

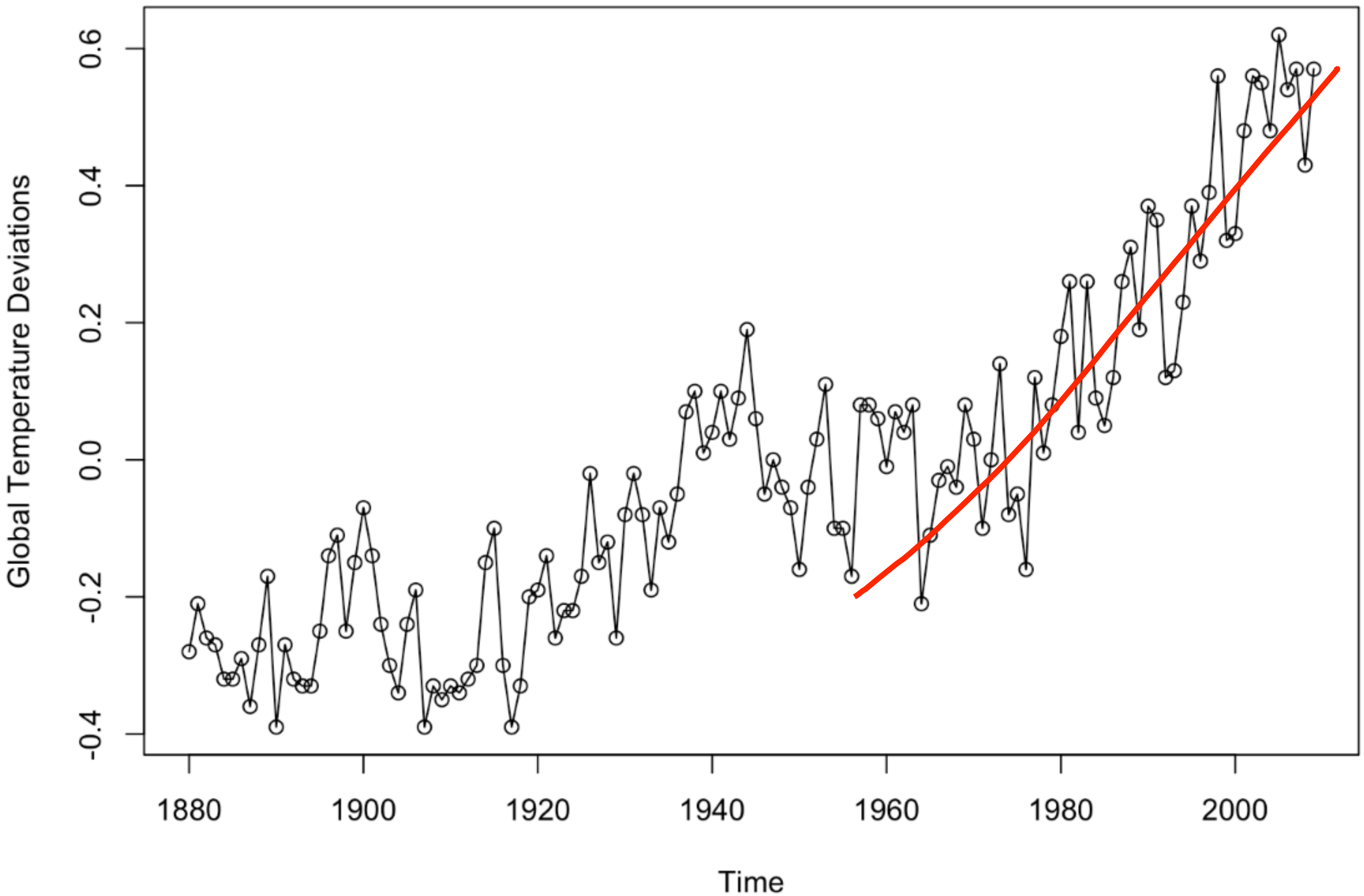
```
lines(ksmooth(time(INJCJC), INJCJC, "normal", bandwidth=2))
lines(ksmooth(time(INJCJC), INJCJC, "normal", bandwidth=5))
```



Time Trend Elimination

Global Temperature Deviations

```
plot(gtemp, type="o", ylab="Global Temperature Deviations")
```



Global Temperature in the Last 130 Years

- estimate a linear trend for a series
- detrend the series
- substract the detrend series from the observed series
- model the residuals using a stationary model

```
library(astsa)
# Look at the structure of the data
str(gtemp)
```

```
##  Time-Series [1:130] from 1880 to 2009: -0.28 -0.21 -0.26 -0.27 -0.32 -0.32 -0.29 -0.36 -0.27 -0.17 ...
```

Keeping Track of the Time Index of a Series Is Important in Time Series Analysis

```
#obtain the time index associated with the time series
time(gtemp)
```

```
## Time Series:
## Start = 1880
## End = 2009
## Frequency = 1
##   [1] 1880 1881 1882 1883 1884 1885 1886 1887 1888 1889 1890 1891 1892 1893
##  [15] 1894 1895 1896 1897 1898 1899 1900 1901 1902 1903 1904 1905 1906 1907
##  [29] 1908 1909 1910 1911 1912 1913 1914 1915 1916 1917 1918 1919 1920 1921
##  [43] 1922 1923 1924 1925 1926 1927 1928 1929 1930 1931 1932 1933 1934 1935
##  [57] 1936 1937 1938 1939 1940 1941 1942 1943 1944 1945 1946 1947 1948 1949
##  [71] 1950 1951 1952 1953 1954 1955 1956 1957 1958 1959 1960 1961 1962 1963
##  [85] 1964 1965 1966 1967 1968 1969 1970 1971 1972 1973 1974 1975 1976 1977
##  [99] 1978 1979 1980 1981 1982 1983 1984 1985 1986 1987 1988 1989 1990 1991
## [113] 1992 1993 1994 1995 1996 1997 1998 1999 2000 2001 2002 2003 2004 2005
## [127] 2006 2007 2008 2009
```

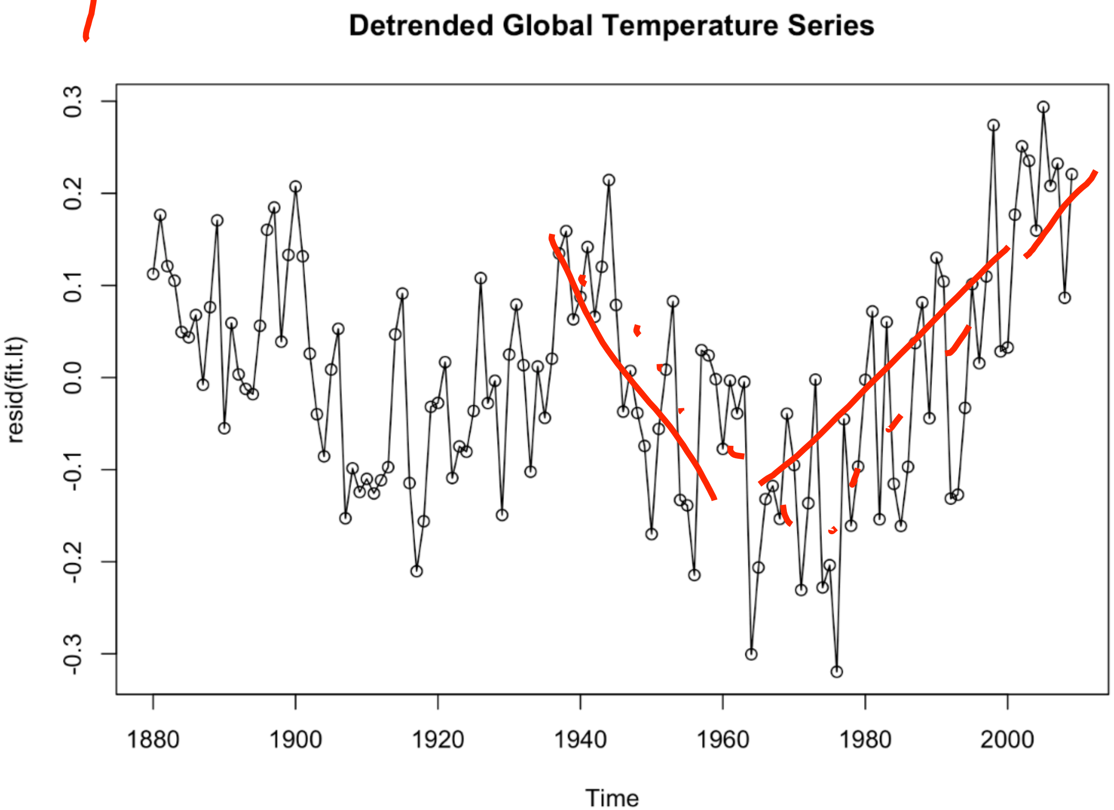
Step 1: Estimate a Linear Trend

```
#Estimate a linear time trend
fit.lt = lm(gtemp ~ time(gtemp), na.action = NULL)
summary(fit.lt)
```

```
##
## Call:
## lm(formula = gtemp ~ time(gtemp), na.action = NULL)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.31946 -0.09722  0.00084  0.08245  0.29383
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -1.120e+01  5.689e-01  -19.69   <2e-16 ***
## time(gtemp)  5.749e-03  2.925e-04   19.65   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1251 on 128 degrees of freedom
## Multiple R-squared:  0.7511, Adjusted R-squared:  0.7492
## F-statistic: 386.3 on 1 and 128 DF,  p-value: < 2.2e-16
```

Step 2: Detrend the Series

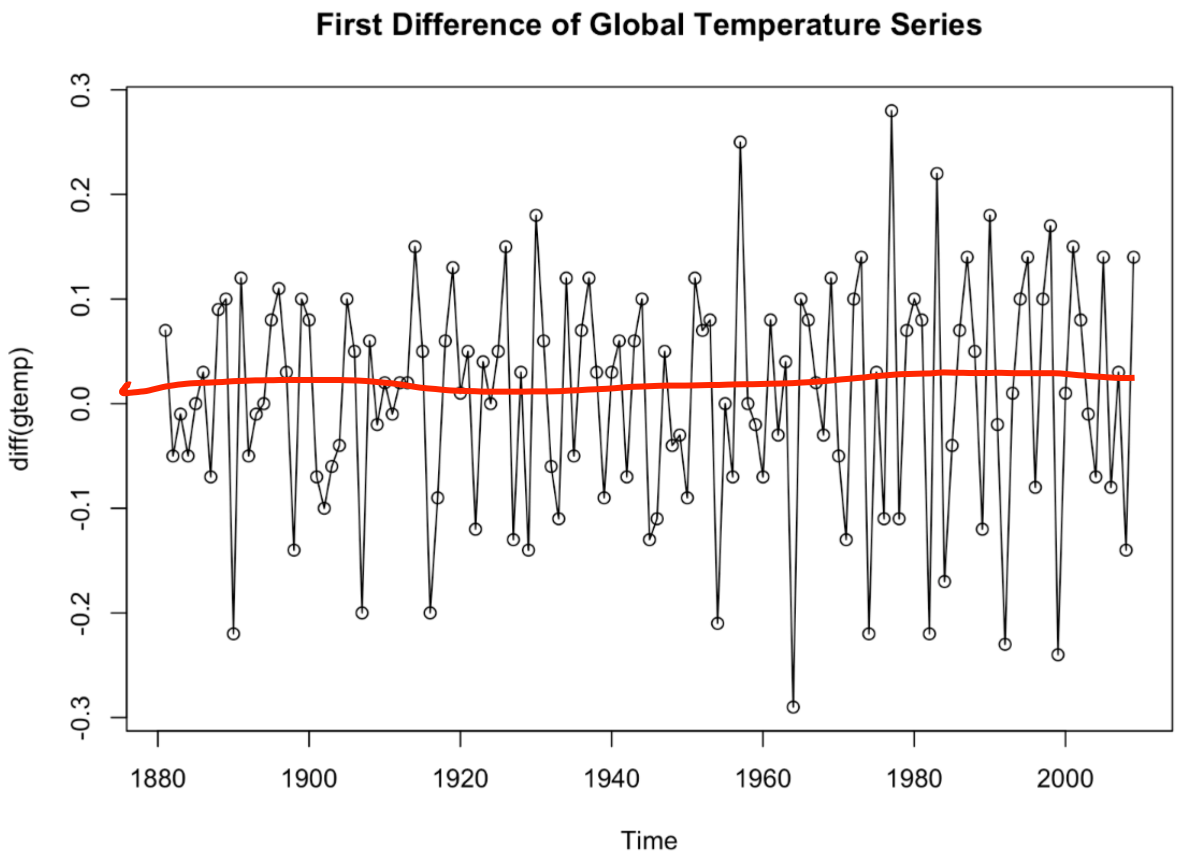
```
plot(resid(fit.lt), type="o", main="Detrended Global Temperature Series")
```



Step 3:
Model the detrended series using a stationary model, if the interest is on modeling the stationary series and when the assumption of stationarity is established.

Another Approach… First Differencing

```
plot(diff(gtemp), type="o", main="First Difference of Global Temperature Series")
```



Autoregressive Models Part 1a
Mathematical Formulation and Properties

Autoregressive Models

Autoregressive models are found on the idea that the current value of a series x_t can be explained as a function of p past values $x_{t-1}, x_{t-2}, \cdots, x_{t-p}$, where p is called the order of the autoregressive model and determines the number of steps into the past needed to *forecast* the current value.

A **stationary autoregressive model** of *order* p, $AR(p)$ takes the form

$$x_t = \alpha + \phi_1 x_{t-1} + \phi_2 x_{t-2} + \cdots + \phi_p x_{t-p} + \omega_t$$

where

x_t is a *stationary* series

$\phi_1, \phi_2, \cdots, \phi_p \; (\phi_p \neq 0)$ are constants (i.e. unknown parameters to be estimated)

ω_t is a *Gaussian white noise series with mean zero and variance* σ_w^2, and

$\alpha = \mu/(1 - \phi_1 - \phi_2 - \cdots - \phi_p)$

Autoregressive Models

To thoroughly study the properties of the autoregressive model, we first consider the first-order model $AR(1)$:

$$x_t = \phi x_{t-1} + \omega_t$$

$$\begin{aligned} x_t &= \phi x_{t-1} + \omega_t \\ &= \phi\left(\phi x_{t-1}\right) + \omega_t \\ &= \phi^2 x_{t-2} + \left(\omega_t + \phi \omega_{t-1}\right) \\ &\vdots \\ &= \phi^k x_{t-p} + \sum_{j=0}^{k-1} \phi^j \omega_{t-j} \end{aligned}$$

Autoregressive Models

Continue to iterate backward, the $AR(1)$ model can be written as a **linear process:**

$$x_t = \sum_{j=0}^{\infty} \phi^j w_{t-j}$$

provided that

1. $|\phi| < 1$
2. x_t is stationary

In other words, the linear process defined above exists in the mean square sense:

$$\lim_{k \Rightarrow \infty} E\left(x_t - \sum_{j=0}^{k-1} \phi^j w_{t-j}\right) = \lim_{k \Rightarrow \infty} \phi^{2k} E\left(x_{t-k}^2\right) = 0$$

Autoregressive Models Part 1b Mathematical Formulation and Properties

Autoregressive Models

The $AR(1)$ model

$$x_t = \phi x_{t-1} + \omega_t$$

is **stationary** with mean

$$E\left(x_t\right) = \sum_{j=0}^{\infty} \phi^j E\left(w_{t-j}\right) = 0$$

0

due to the assumption that w_t is a *white noise series with mean zero and a constant variance.*

The variance and autocorrelation functions are

$$\rho_k = \phi^k (k \geq 0 and \left|\phi\right| < 1)$$

$$\gamma_k = \frac{\phi^k \omega_w^2}{(1 - \phi^2)}$$

Autoregressive Models: Second-Order Properties

Second-order properties of an AR(1) model

AR(1) process is given by

$$x_t = \alpha x_{t-1} + w_t$$

where $\{w_t\}$ is a white noise series with mean zero and variance σ^2. It can be shown that the second-order properties follow as

$$\left.\begin{aligned} \mu_x &= 0 \\ \gamma_k &= \alpha^k \sigma^2 / (1 - \alpha^2) \end{aligned}\right\}$$

$Bx_t = x_{t-1}$

Using B, a stable AR(1) process ($|\alpha| < 1$) can be written as

$$\left.\begin{aligned} (1 - \alpha \mathrm{B}) x_t &= w_t \\ \Rightarrow x_t &= (1 - \alpha \mathrm{B})^{-1} w_t \\ &= w_t + \alpha w_{t-1} + \alpha^2 w_{t-2} + \ldots = \textstyle\sum_{i=0}^{\infty} \alpha^i w_{t-i} \end{aligned}\right\}$$

Autoregressive Models: Second-Order Properties

Hence, the mean is given by

$$E(x_t) = E\left(\sum_{i=0}^{\infty} \alpha^i w_{t-i}\right) = \sum_{i=0}^{\infty} \alpha^i E(w_{t-i}) = 0$$

and the autocovariance follows as

$$\begin{aligned}\gamma_k = \mathrm{Cov}(x_t, x_{t+k}) &= \mathrm{Cov}\left(\sum_{i=0}^{\infty} \alpha^i w_{t-i}, \sum_{j=0}^{\infty} \alpha^j w_{t+k-j}\right) \\ &= \sum_{j=k+i} \alpha^i \alpha^j \mathrm{Cov}(w_{t-i}, w_{t+k-j}) \\ &= \alpha^k \sigma^2 \sum_{i=0}^{\infty} \alpha^{2i} = \alpha^k \sigma^2 / (1 - \alpha^2)\end{aligned}$$

Autoregressive Models: General Case

- As we will see when studying MA models, we need to specify the invertability condition because this condition ensures the existence of the autoregressive representation of a MA model. As for stationarity condition, MA model is always stationary regardless of its parameter values.

- In contract to MA models, an AR model is always invertible, but restrictions on parameters need to be imposed in order for the process to be stationary.

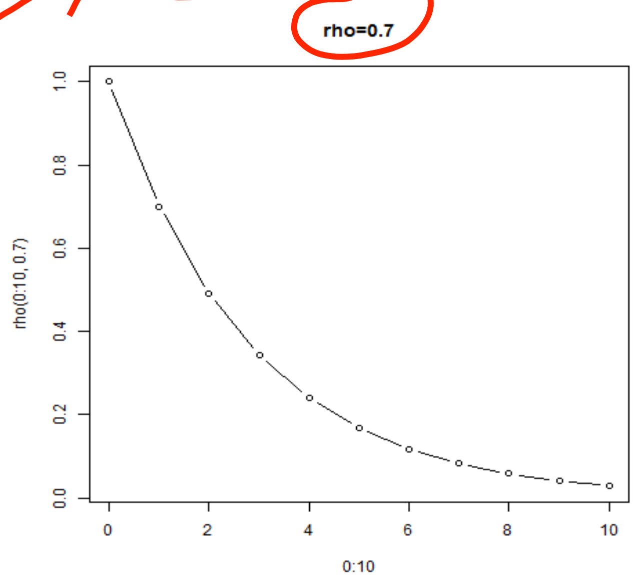
Autoregressive Models Part 2 Simulation and Empirical Properties of AR(1) Models

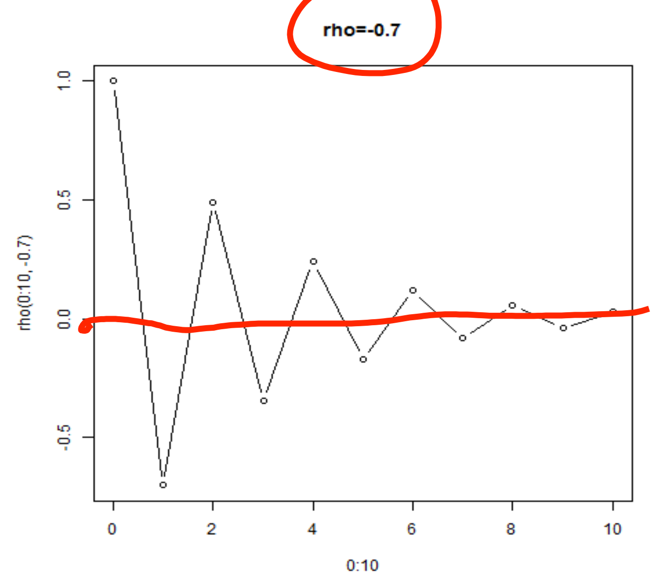
AR(1) Model: Theoretical Autocorrelation Function

The autocorrelation function of an AR(1) process, and in fact, AR(p) process in general, decay to zero gradually, either monotonically or in a oscillated fashion, although the general process allows for richer dynamics.

The damped oscillation comes from the switching signs at each successively longer time displacement.

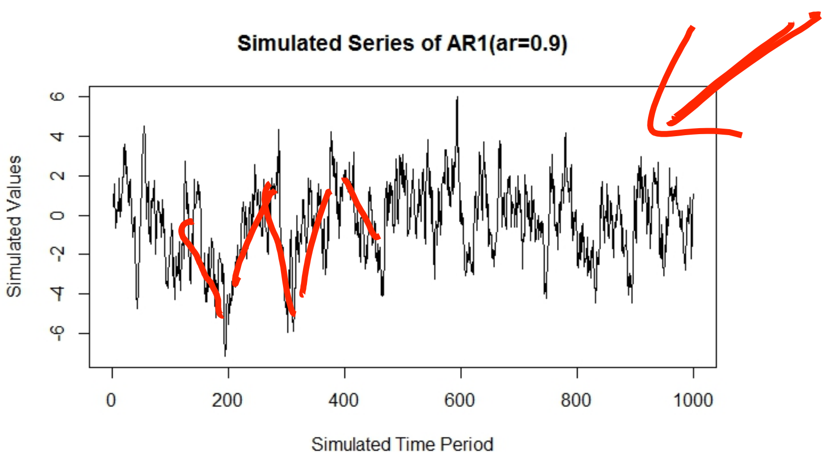
$$\rho_k = \phi^k \quad \text{(provided that } k \geq 0 \text{ and } |\phi| < 1)$$

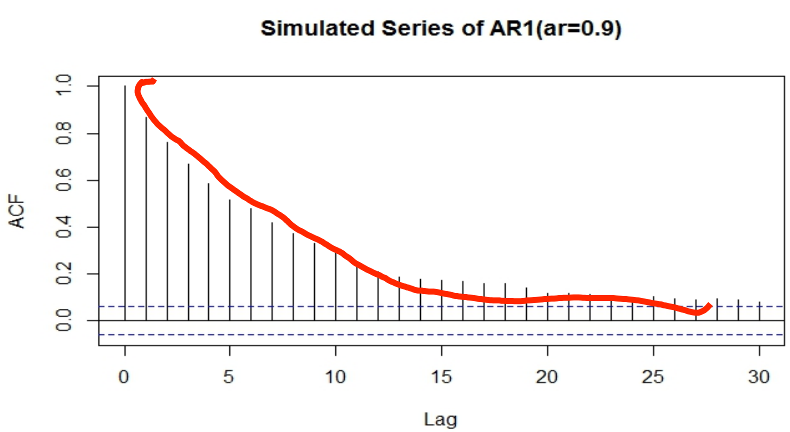




The ACF of AR(1) Model: Case 1

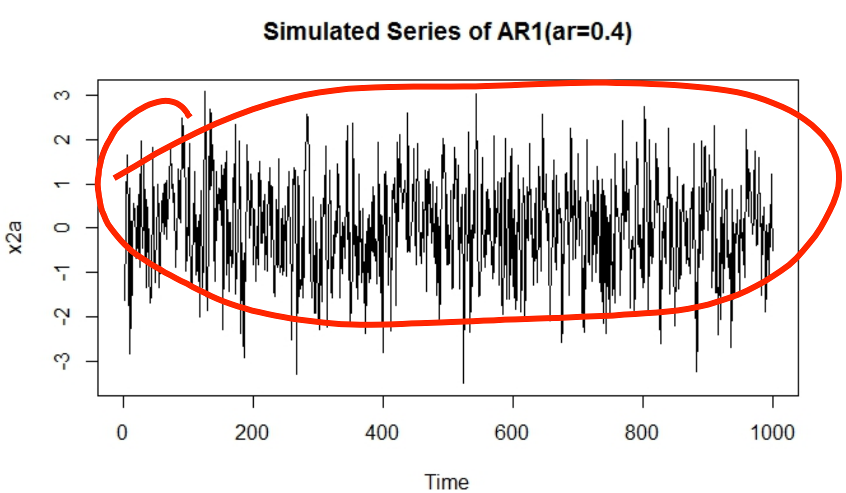
- The time plot of the simulated **AR(1) model with parameter = 0.9** appears to be persistent, meaning that there are time periods in which the series "tends" to go up or tends to go down. This is not surprising because with the AR parameter being 0.9, the simulated series has high autocorrelation.
- The correlogram decays gradually to zero, and the ACF is still statistically significant after 25 lags. In general, the autocorrelations converge to 0 in the limit as the time displacement approaches infinity. **The implication is that they do not "cut off" at zero abruptly.**
- Remember that we simulated 1,000 observations, so it is a pretty long time series and can compress the confidence interval.

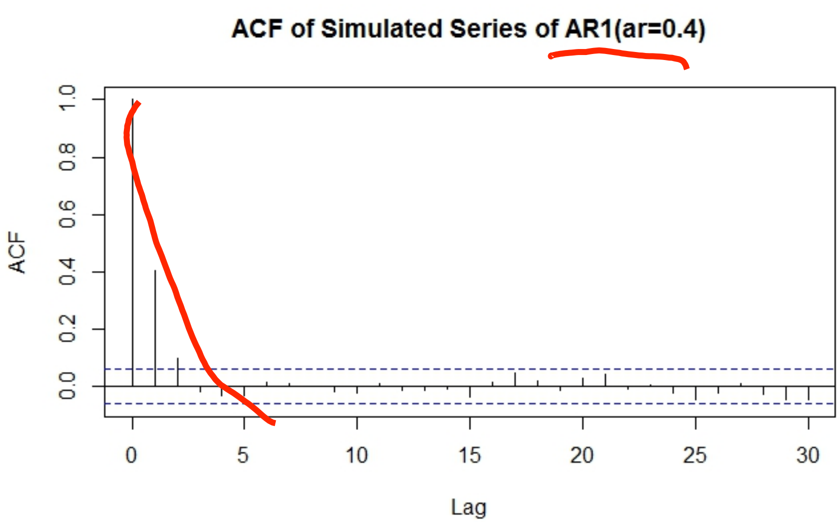




AR(1) Model: Case 2

- The time plot of the simulated AR(1) model with parameter = 0.4 looks very different from the previous AR(1) model. The persistence is much less apparent.
- The correlogram also decays to zero but in a much fast fashion. In fact, the ACF is no longer statistically significant after three lags.
- So, the correlogram of the ACF of AR models depends on the AR parameter.
- As we will see, this is one of the biggest different the correlogram ke a comment that this is in a sharp contrast with MA processes.





PACF of AR(1) Model

- Partial autocorrelation function for the AR(1) process, on the other hand, cuts off abruptly after the first time displacement:

$$\boxed{p(\tau) = \begin{cases} \varphi, & \tau = 1 \\ 0, & \tau > 1 \end{cases}}$$

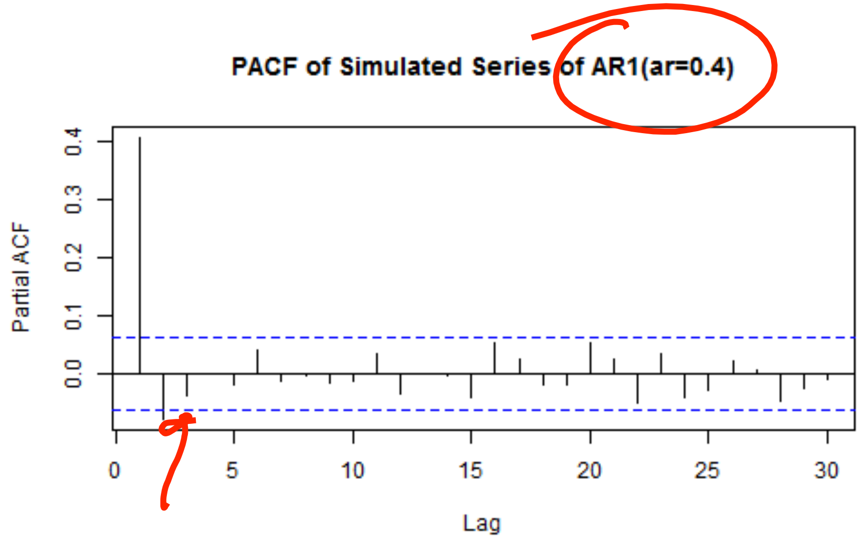
- For AR(1) model, this is obvious, because the first partial autocorrelation is just the autoregressive coefficient, and the coefficients on all other longer lags, by construction, are zero.

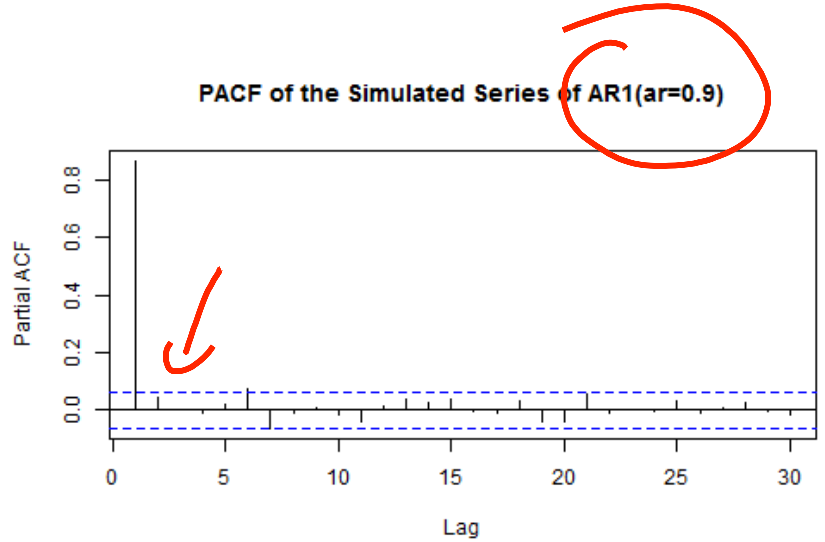
- This is an important property of AR models because it helps identify the order of an autoregressive models.

- In fact, using the autocorrelation function alone is hard to identify the order of an autoregressive models.

PACF of AR(1) Model

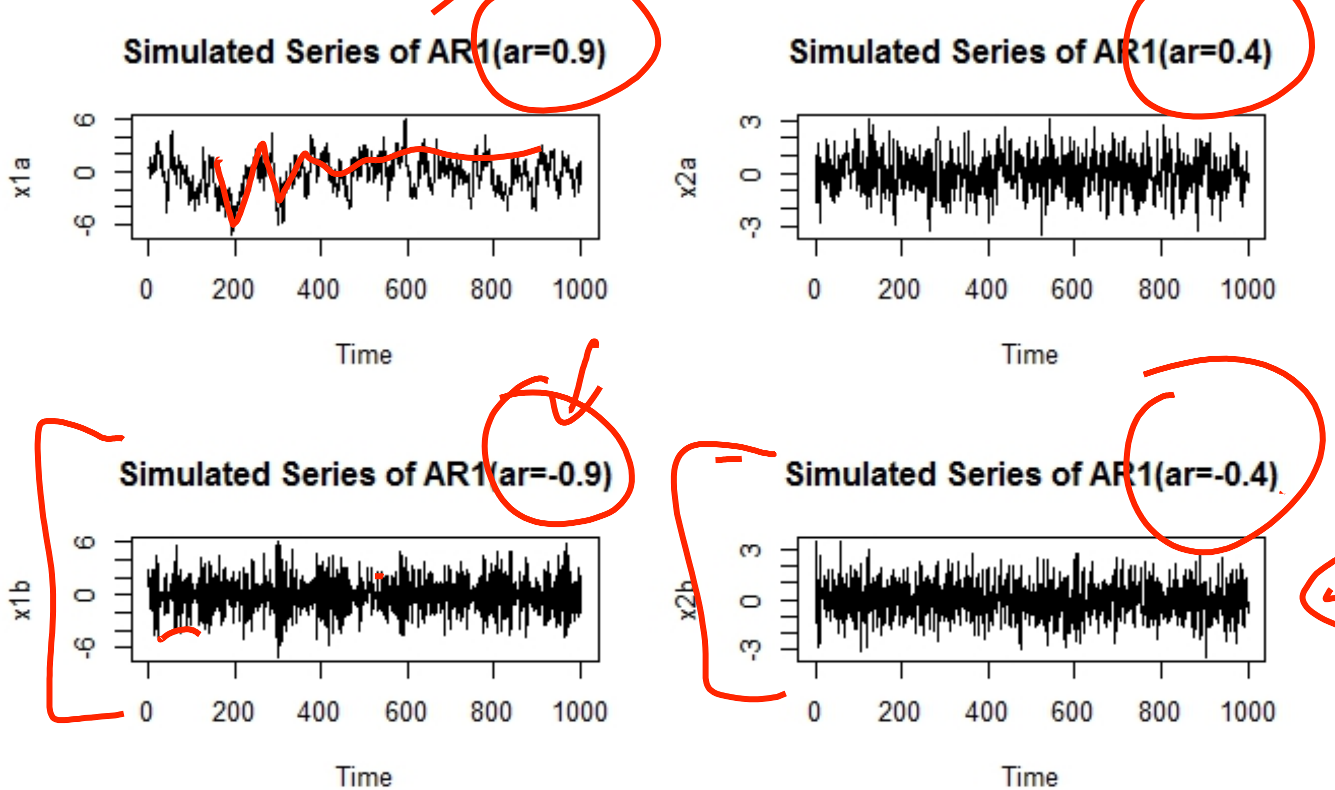
- Both of the partial autocorrelation function graphs indicate that the partial autocorrelations cut off to zero after lag 1.
- So, we know that if the underlying data-generating process is a class of autoregressive models, then these PACF graphs indicate that the process should be of order 1.
- When analyzing real-world data, it is never that clear cut. As such, we need to use combine various graphical techniques and statistical tests to identify the order of an AR model.
- In fact, we may be able to narrow down only to a few models and use other statistics, such as AIC and BIC, to choose a model.





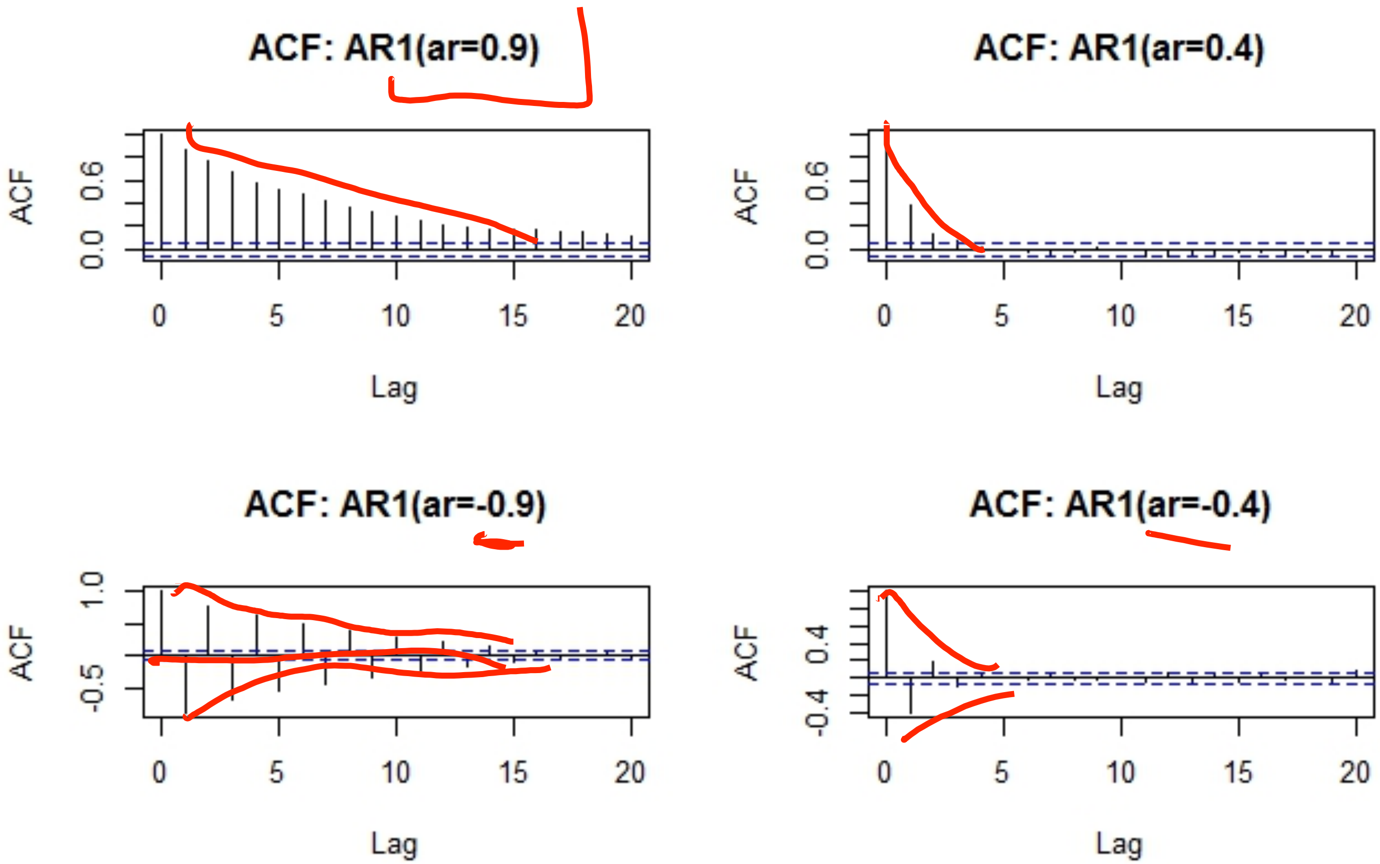
PACF of AR(1) Model

- Let's look at two other AR(1) models that have AR parameters being negative.
- Notice the difference among these time plots:
  - For the models with positive autocorrelation, the time series plots exhibit much more persistence for the series with high correlation.
  - For the models with negative autocorrelation, however, the time series plots look quick similar.



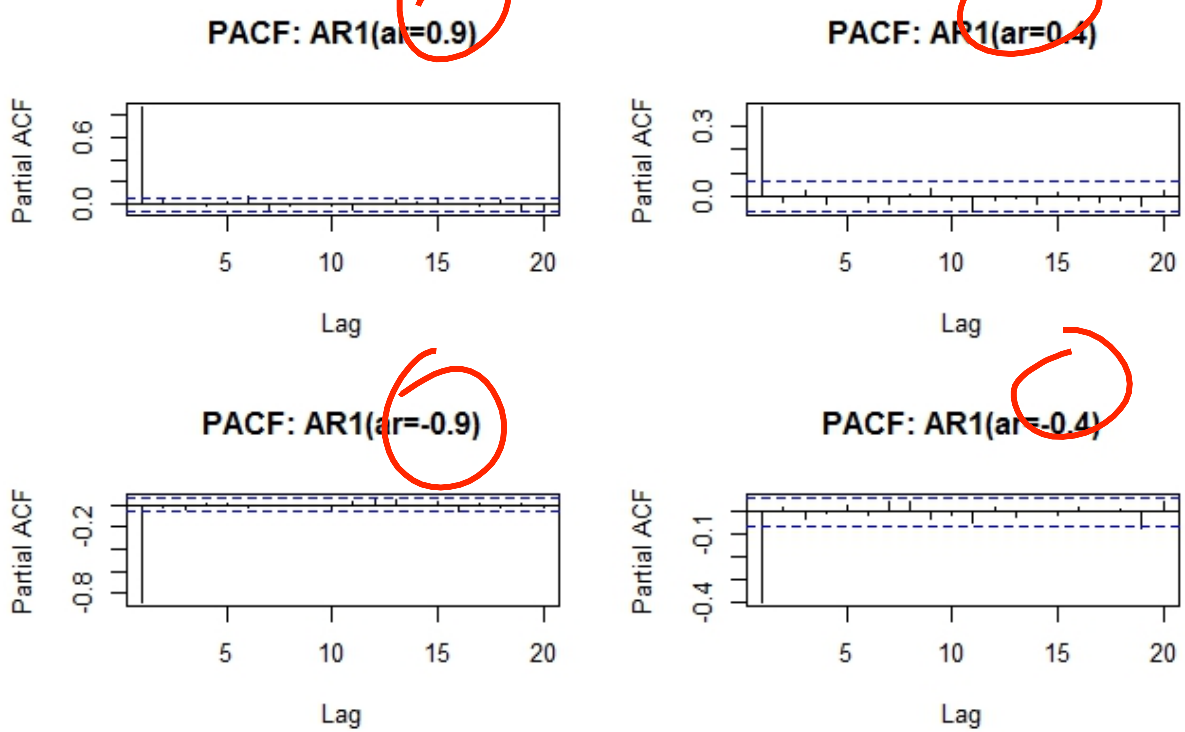
PACF of AR(1) Model

- To distinguish the two, we can rely on autocorrelation function because
  - ➢ The autocorrelations of AR(1) models with positive AR parameter decay to zero gradually and monotonically, and
  - ➢ The autocorrelation of AR(1) models with negative AR parameters decay to zero in an oscillated fashion.



PACF of AR(1) Model

- The PACFs of AR(1) models with and without positive AR parameters behave very different.
- Therefore, PACF can be used to help learning about the sign of the parameter of an AR(1) model. Remember that it is possible because we assume that the underlying data-generating process is an AR(1) model.
- In general, given a data series, we will have to plot the time series plot, ACF, and PACF as a starting point to choose candidate models for the data.



Berkeley SCHOOL OF INFORMATION