Time Series Analysis I

SYS 4021/6021

Laura Barnes and Julianne Quinn

Organization of lecture

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

1. Review of Multiple Linear Regression and Generalized Linear Models
2. Introduction to Time Series
3. Common Elements of Time Series
4. Build Regression Models of Time Series

Models Discussed in Class So Far

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
    - Trends
    - Seasons / Cycles
    - Auto-correlation
- Summary

Multiple Linear Regression:

$$Y = \beta_0 + \beta_1 X_1 + \ldots + \beta_k X_k + \varepsilon$$

Generalized Linear Models:

$$g(E[Y]) = \beta_0 + \beta_1 X_1 + \ldots + \beta_k X_k$$

where $g(E[Y]) = \log\left(\frac{P(Y=1)}{1-P(Y=1)}\right)$ for logistic regression

So far, we have only used the above models for cross-sectional data. What if we have a time series or longitudinal data?

Time Series Overview

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

A time series is a sequence of data that have been observed in successive order at different points in time.

It is commonly assumed time series data are spaced equally in time.

Examples:

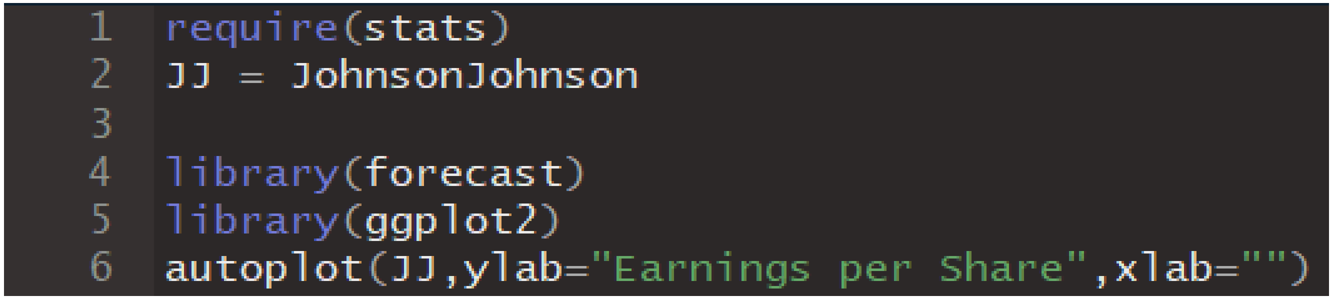
- Company earnings
- Global temperature
- Air pollution
- Stock prices

Time Series Example

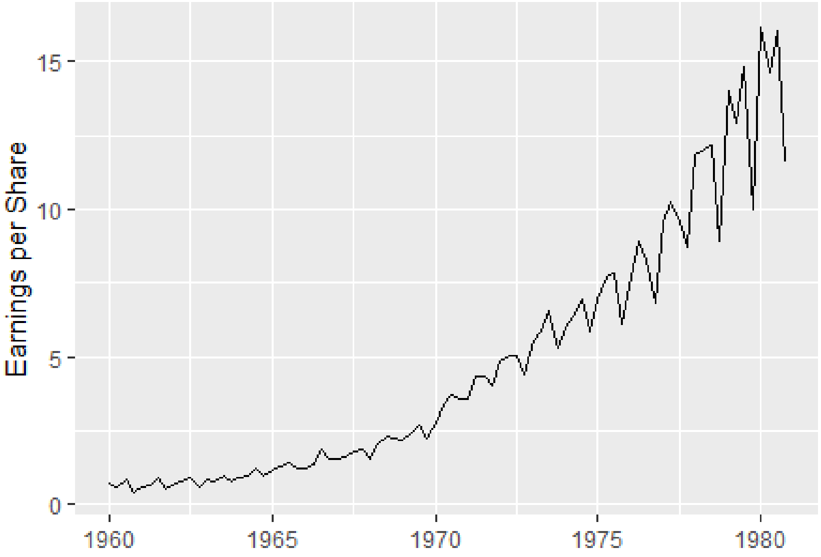
Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Johnson & Johnson quarterly earnings per share, 1960-I to 1980-IV:



```r
require(stats)
JJ = JohnsonJohnson

library(forecast)
library(ggplot2)
autoplot(JJ,ylab="Earnings per Share",xlab="")
```

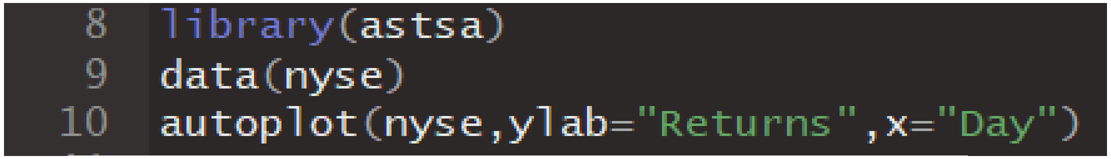


Time Series Example

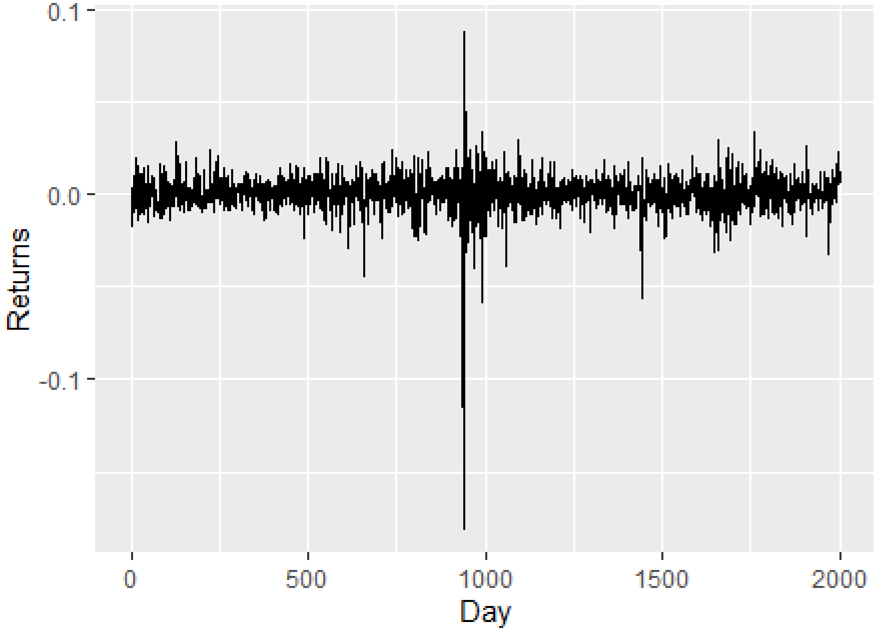
Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Returns of the NYSE. Daily value-weighted market returns of 2000 trading days, Feb 2, 1984 – Dec 31, 1991:



```r
library(astsa)
data(nyse)
autoplot(nyse,ylab="Returns",x="Day")
```

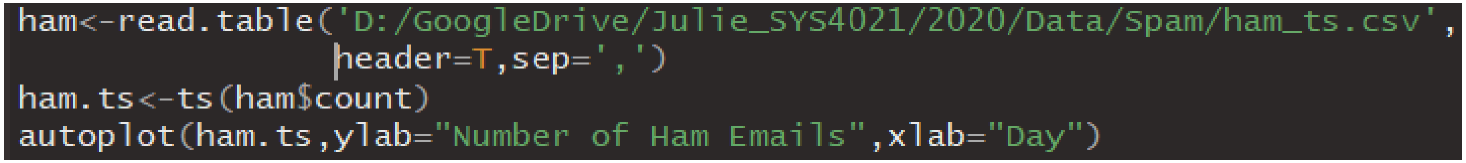


Time Series Example

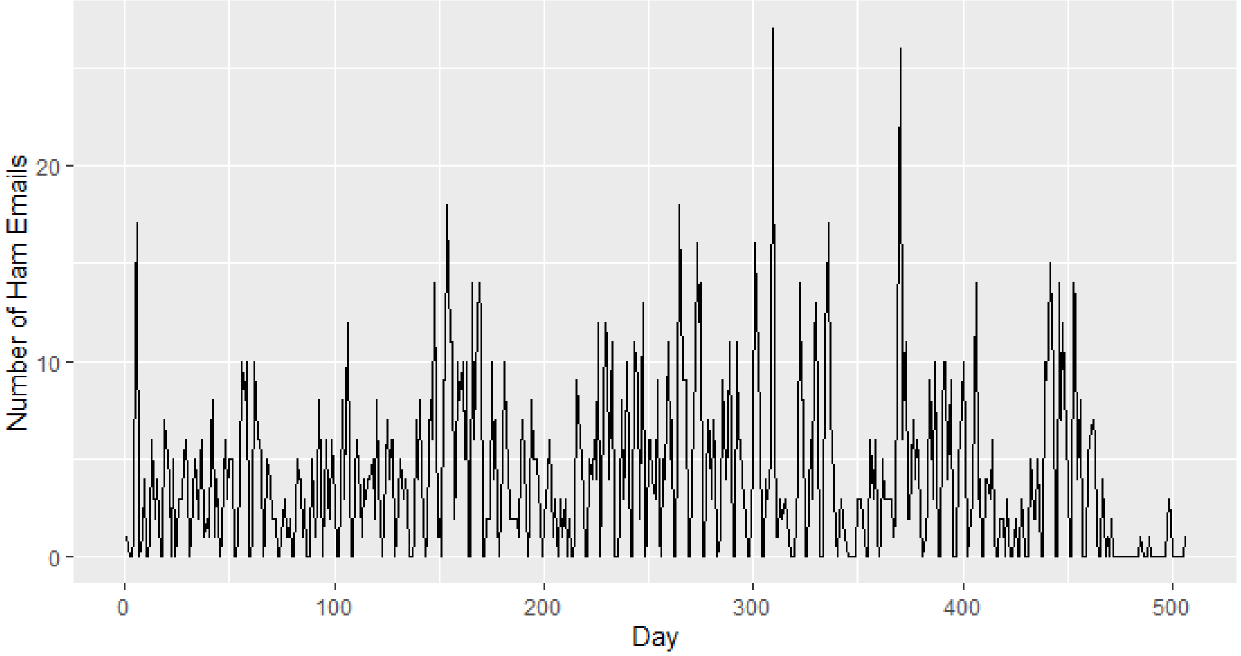
Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Ham emails received per day, January 2000 – June 2001:



```
ham<-read.table('D:/GoogleDrive/Julie_SYS4021/2020/Data/Spam/ham_ts.csv',
                header=T,sep=',')
ham.ts<-ts(ham$count)
autoplot(ham.ts,ylab="Number of Ham Emails",xlab="Day")
```



Time Series Analysis

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

If we want to predict the value of some variable Y at time t, Y_t, can we use MLR or GLM?

These methods both assume adjacent observations are independent and identically distributed (iid).

This is often violated with time series data. If we don't account for that, we might get biased estimates or underestimate variability.

In time series analysis we'll need to account for this "auto-correlation" (correlation of a variable with itself).

Time Series Analysis

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

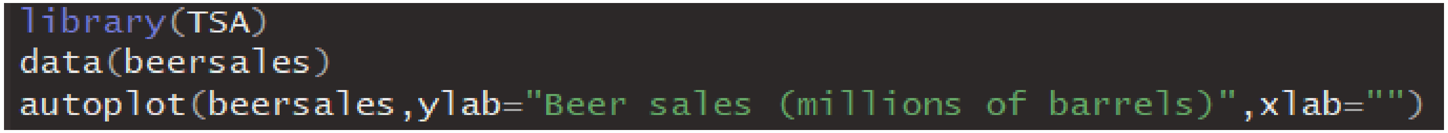
Autocorrelation is one common element of time series data we’ll need to account for in applying MLR or GLM to make forecasts.

What other common elements might time series include? Let’s consider an example.

Time Series Components

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

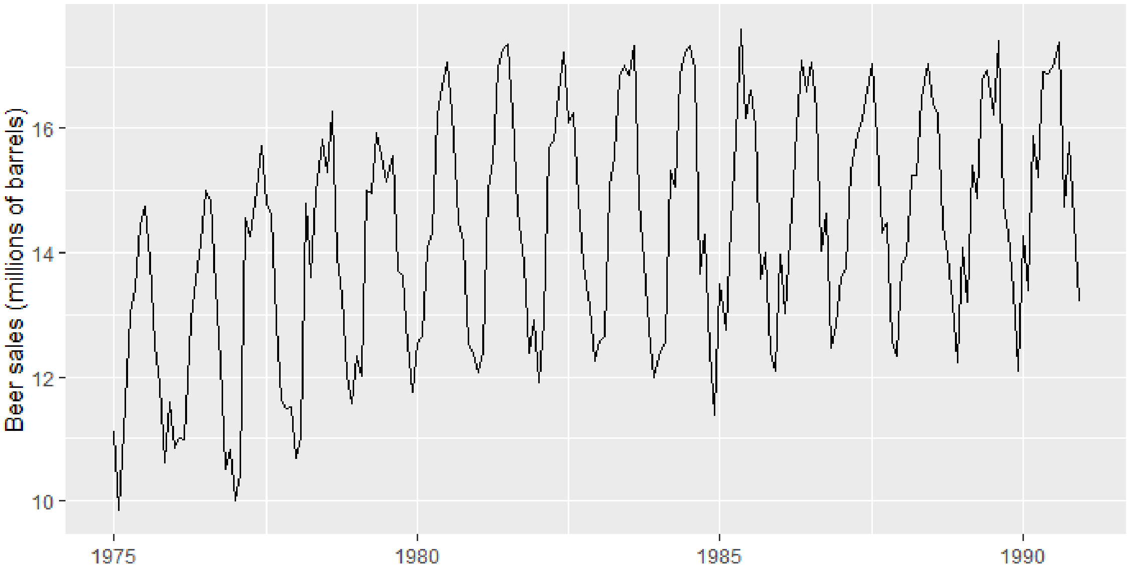


```r
library(TSA)
data(beersales)
autoplot(beersales,ylab="Beer sales (millions of barrels)",xlab="")
```

Trend: long term upward/downward movement

How could we model this?

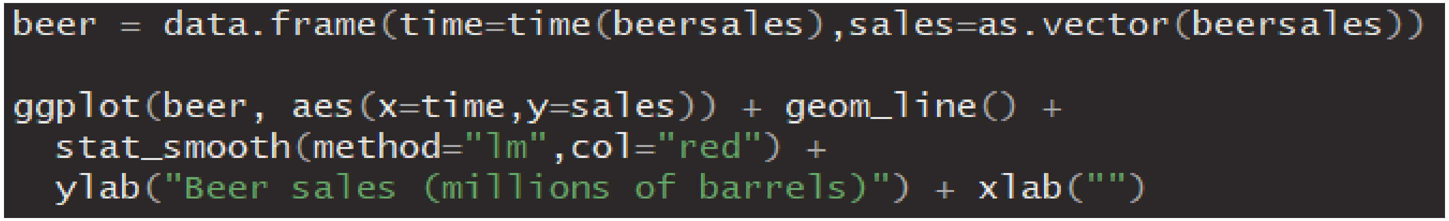
$$Y_t = \beta_0 + \beta_1 t + \varepsilon_t$$



Trend

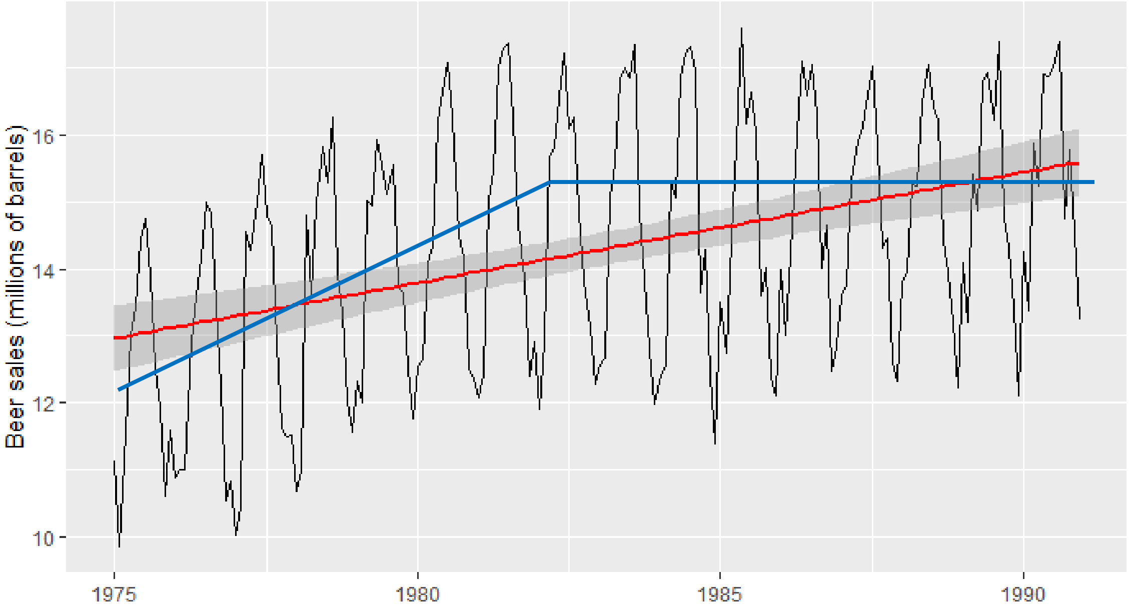
Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary



```
beer = data.frame(time=time(beersales),sales=as.vector(beersales))

ggplot(beer, aes(x=time,y=sales)) + geom_line() +
  stat_smooth(method="lm",col="red") +
  ylab("Beer sales (millions of barrels)") + xlab("")
```

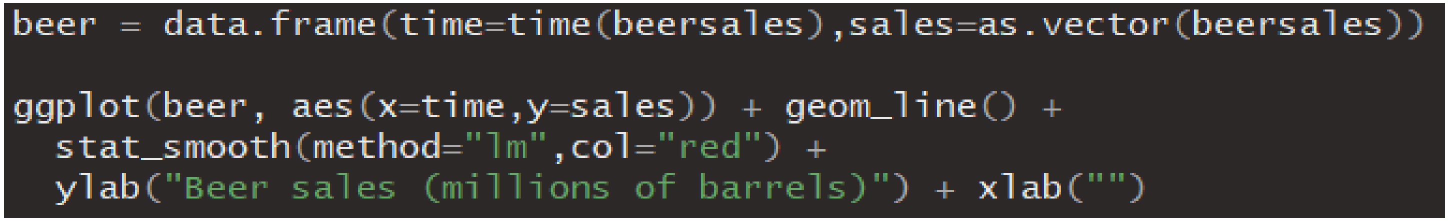
Might there be a better way to model this trend?



Trend

Agenda

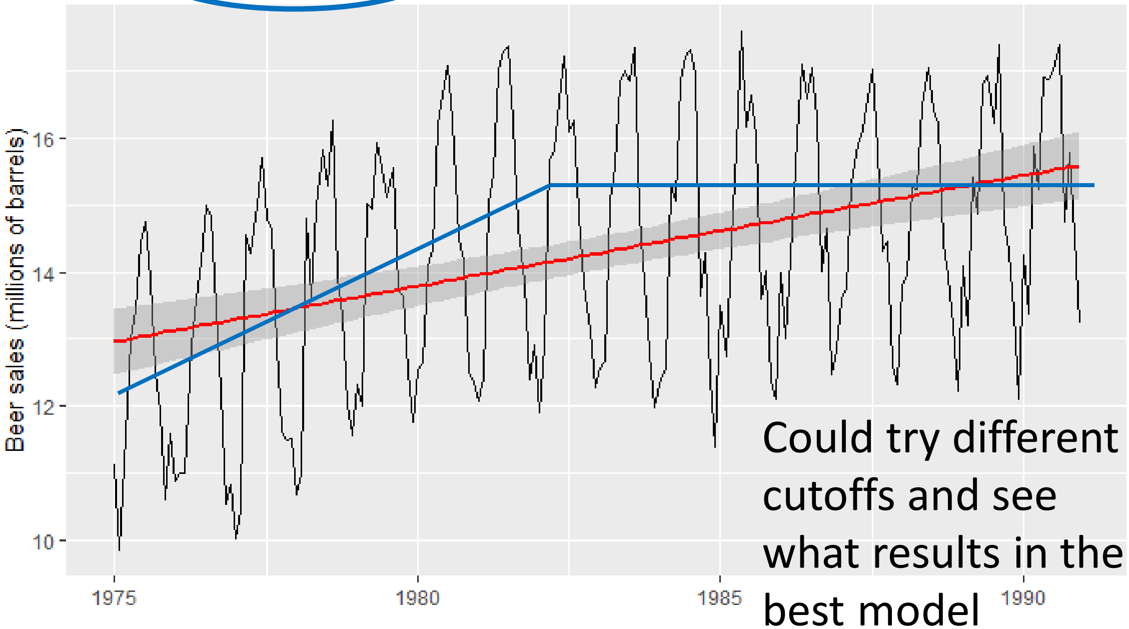
- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary



```
beer = data.frame(time=time(beersales),sales=as.vector(beersales))

ggplot(beer, aes(x=time,y=sales)) + geom_line() +
  stat_smooth(method="lm",col="red") +
  ylab("Beer sales (millions of barrels)") + xlab("")
```

$$Y_t = \beta_0 + \beta_1 t + \beta_2 X_1 + \beta_3 X_1 t + \varepsilon_t$$

where $X_1 = 1$ if $t > 1982$, 0 otherwise

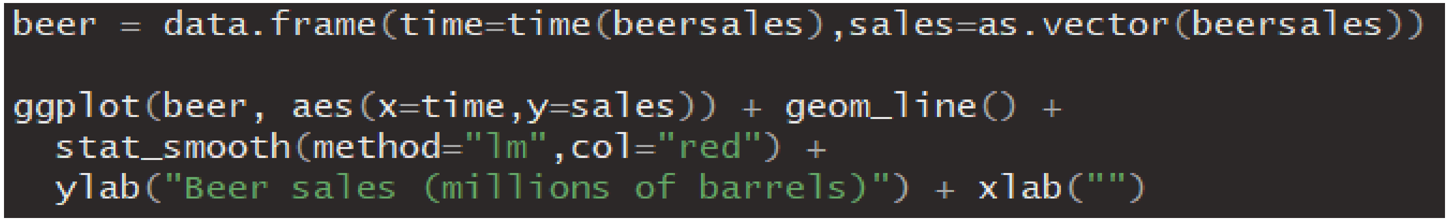


Could try different cutoffs and see what results in the best model

Trend

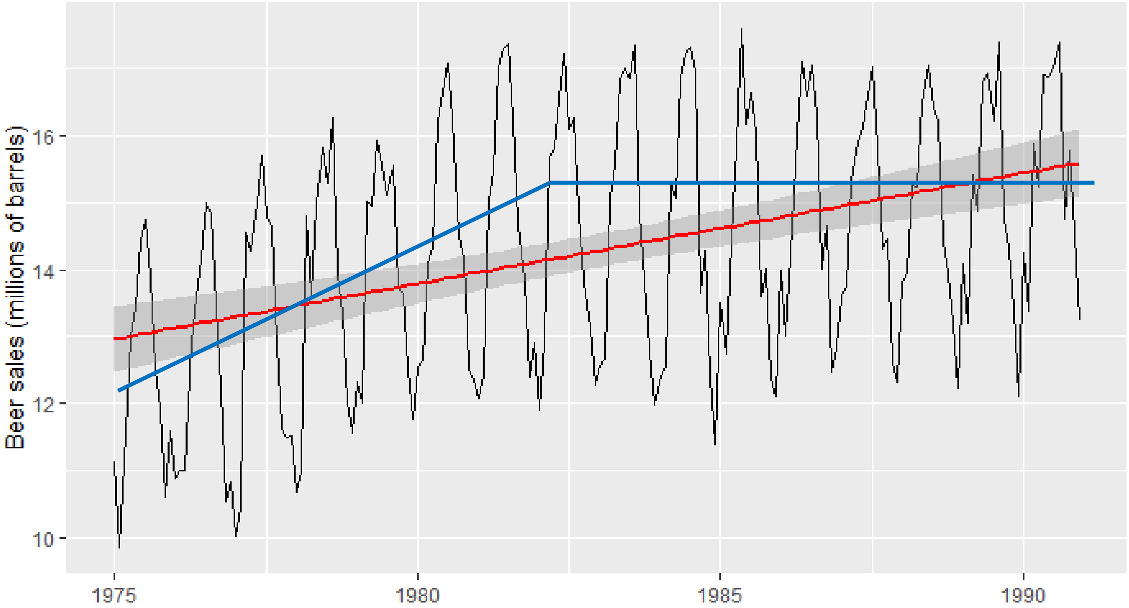
Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary



```
beer = data.frame(time=time(beersales),sales=as.vector(beersales))

ggplot(beer, aes(x=time,y=sales)) + geom_line() +
  stat_smooth(method="lm",col="red") +
  ylab("Beer sales (millions of barrels)") + xlab("")
```

What else do we need to model in this dataset?



Cycles and seasons

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Cycles and seasons are recurring long-term up and down movements

Cycles can have a duration of any length of time. They are measured from peak to peak, or trough to trough

Seasons are periodic patterns that complete themselves within a specified time period and are then repeated in the same amount of time.

Example cycles vs. seasons

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Cycles

Business cycle: Recurrent periods of prosperity (expansion) alternating with recession (contraction). Expansion ends at the peak and contraction ends at the trough.

Climate cycles: e.g. El Nino-Southern Oscillations

Seasons

Weather measurements (e.g. daily temperature repeats over a year)

Sales in a department store (repeats over a year)

Tides (high tides repeat twice per day; spring tides repeat twice per month)

Modeling seasonality

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

We model trends by including time as a predictor. How can we model seasonality?

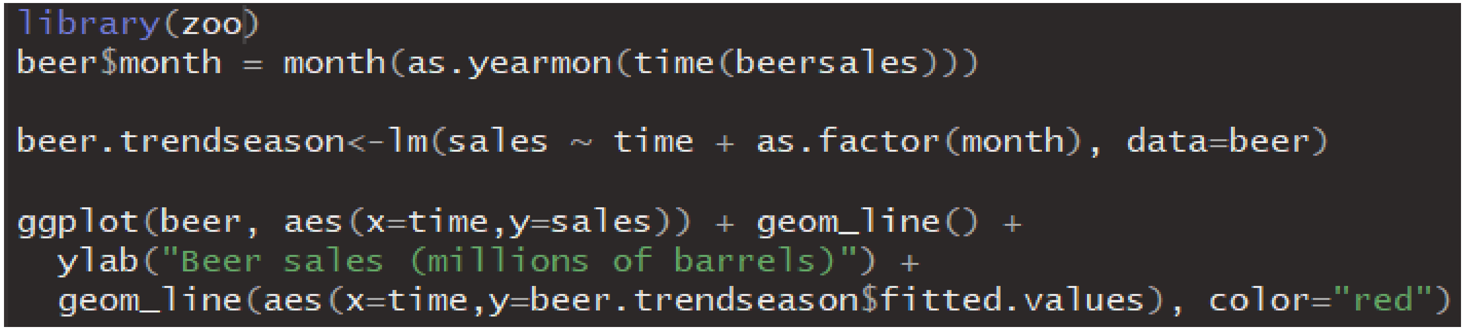
There are a couple ways. One is through dummy variables.

For L seasons, we can use L-1 dummy variables to adjust the mean in each season relative to the base case season.

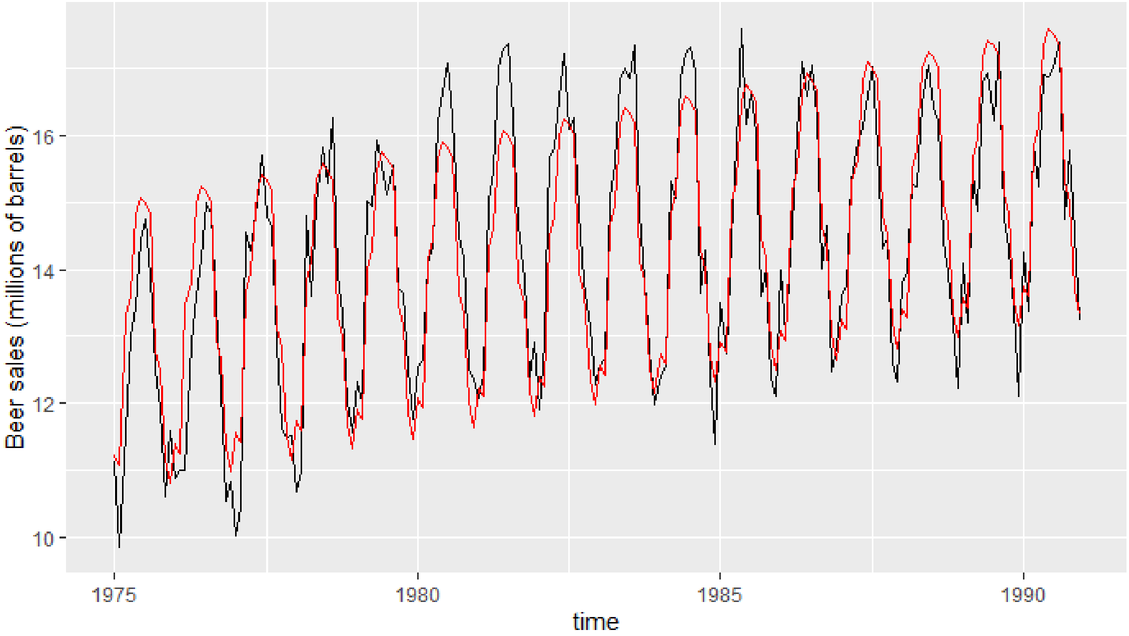
Seasonality

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary



```
library(zoo)
beer$month = month(as.yearmon(time(beersales)))

beer.trendseason<-lm(sales ~ time + as.factor(month), data=beer)

ggplot(beer, aes(x=time,y=sales)) + geom_line() +
  ylab("Beer sales (millions of barrels)") +
  geom_line(aes(x=time,y=beer.trendseason$fitted.values), color="red")
```



Modeling seasonality

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

For monthly data exhibiting an annual season, we need 11 dummy variables.

What if we had daily data? Is it reasonable to use 364 dummy variables? We would need significantly > 364 observations to do that.

For smoothly varying seasons that require more dummy variables than we have enough data to estimate, we can instead use trigonometric functions, e.g. $\sin(2\pi t/T)$ and $\cos(2\pi t/T)$ where $T = \#$ of time steps in a season.

Modeling seasonality

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Dummy variables allow for easier interpretation and testing.

Trigonometric models have better parsimony. Using trigonometric terms usually requires less parameters, but requires smoothly varying seasons.

Autocorrelation

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

While trends and cycles/seasons are common in time series data, modeling them still does not account for the autocorrelation in the dataset that violates the assumptions of MLR and GLMs.

To account for autocorrelation, we can regress the response variable on past values of itself. This is called an autoregressive model:

$$Y_t = \beta_0 + \beta_1 Y_{t-1} + \ \dots + \beta_k Y_{t-k} + \varepsilon_t$$

We might also use past values of other predictors, X:

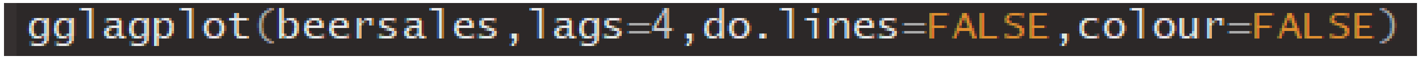
$$Y_t = \beta_0 + \beta_1 Y_{t-1} + \ \dots + \beta_k Y_{t-k} + \beta_{k+1} X_{t-1} + \ \dots + \beta_{2k} X_{t-k} + \varepsilon_t$$

This significantly increases the complexity of our model selection problem.

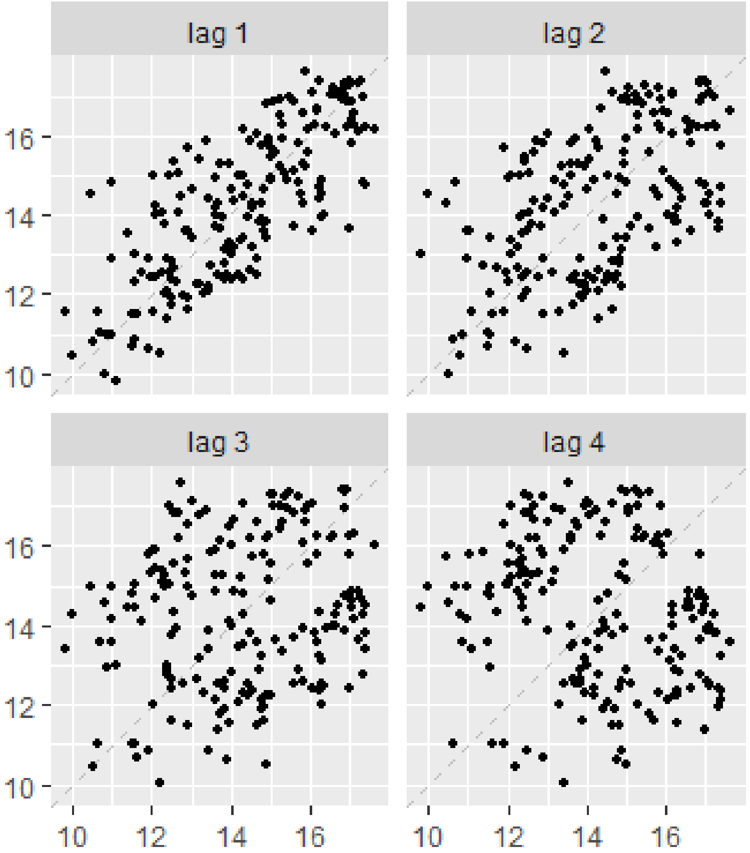
Lag Plots of Beer Sales

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary



```
gglagplot(beersales,lags=4,do.lines=FALSE,colour=FALSE)
```



Autocorrelation

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
    - Trends
    - Seasons / Cycles
    - Auto-correlation
- Summary

Rather than making pairwise scatterplots of Y_t with past observations of itself at different lags k (i.e. k time steps ago: $Y_{t\text{-}k}$), we will often plot these correlations on one plot.

This is called an Autocorrelation Function (ACF). It shows Corr(Y_t, $Y_{t\text{-}k}$) at different lags k.

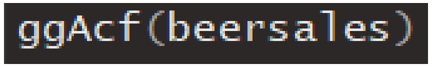
The standard error for the autocorrelation is $\sim \frac{1}{\sqrt{n}}$ where n is the number of observations in the time series.

Any observation within $\pm\frac{2}{\sqrt{n}}$ of 0 is not statistically significant.

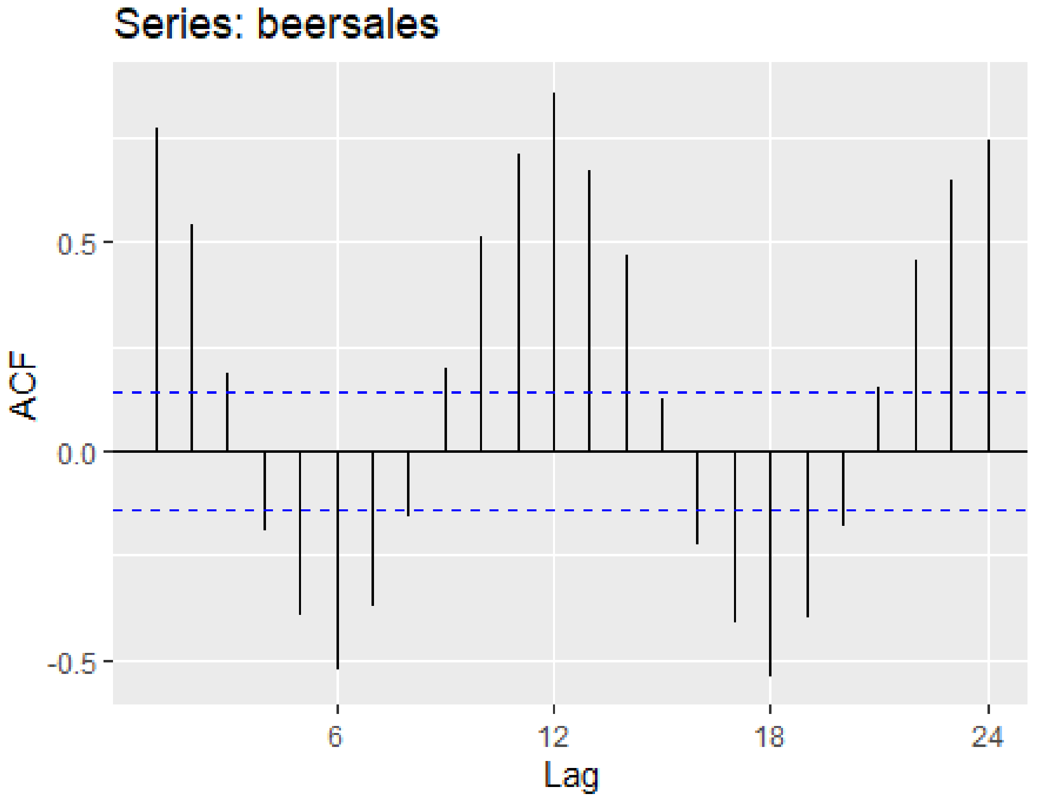
Beer Sales ACF

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary



```
ggAcf(beersales)
```



Regression for Time Series Analysis

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Often we won’t build the autoregressive model directly to the observations, but to the residuals of our trend + seasonality model. This will account for the fact that the residuals are not iid.

For example, to model a variable Y that exhibits a trend and seasonality over L seasons, use:

$$Y_t = \beta_0 + \beta_1 t + \sum_{i=1}^{L-1} \beta_{i+1} X_i + \varepsilon_t = E[Y_t] + \varepsilon_t$$

$$\varepsilon_t = \sum_{j=1}^{k} \varphi_j \varepsilon_{t-j} + w_t$$

where $X_i = 1$ if season *i*, 0 otherwise and $w_t \sim \mathcal{N}(0, \sigma^2)$ and iid.

Summary

Agenda

- Review of MLR and GLM assumptions
- Intro to Time Series Analysis
- Common Elements of Time Series
  - Trends
  - Seasons / Cycles
  - Auto-correlation
- Summary

Time series can be modeled with MLR and GLMs, but we have to account for autocorrelation to meet the iid assumptions of these methods.

There are many other common elements of time series that may need to be modeled:

- Trends: use t as a predictor
- Seasonality/cycles: use dummy variables or trigonometric functions
- Auto-correlation: consider past observations/past residuals as predictors

Time Series Analysis 2

SYS 4021/ 6021

Laura Barnes and Julianne Quinn

Organization of lecture

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

1. Review of Time Series Concept
2. Trigonometric Models of Seasonality
3. Autoregressive (AR) Models
4. Moving Average (MA) Models
5. Autoregressive Moving Average (ARMA) Models

Time Series Overview

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

A time series is a sequence of data that have been observed in successive order at discrete points in time.

It is commonly assumed time series data are spaced equally in time.

Time series often exhibit:

- Trends
- Cycles (repeated up and down movements)
- Seasons (cycles with constant period)
- Autocorrelation

Simple Time Series Models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

We can capture trends, seasonality and autocorrelation through a series of models.

For example, to model a variable Y that exhibits a trend and seasonality over L seasons, use:

$$Y_t = \beta_0 + \beta_1 t + \sum_{i=1}^{L-1} \beta_{i+1} X_i + \varepsilon_t = E[Y_t] + \varepsilon_t$$

$$\varepsilon_t = \sum_{j=1}^{k} \varphi_j \varepsilon_{t-j} + w_t$$

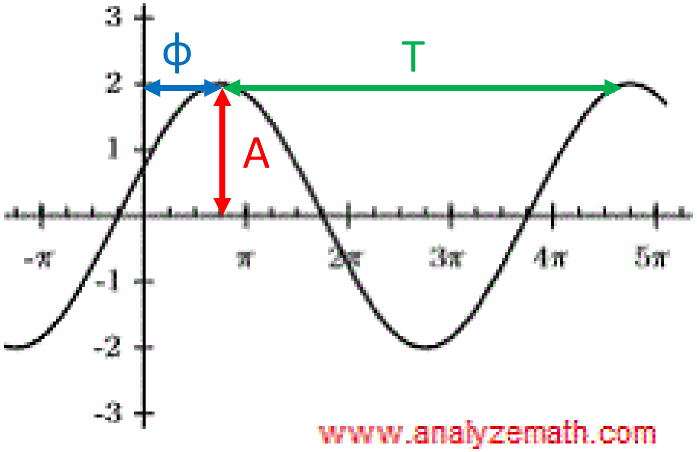
where $X_i = 1$ if season *i*, 0 otherwise and $w_t \sim \mathcal{N}(0, \sigma^2)$ and iid.

Modeling seasonality

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
    - Spectral Analysis
- Autoregressive (AR) Models
    - Stationarity
    - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Instead of using L-1 dummy variables to capture seasons, we noted this can also be captured with trigonometric functions.



$$A\cos\left(\frac{2\pi t}{T} - \phi\right)$$

Sometimes we will know the period (T) intuitively. But what about the phase shift (ϕ) and amplitude (A)? And what if there are multiple seasons in the dataset?

Modeling seasonality

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Using trigonometric identities, we can rewrite the cosine function in a way that allows us to estimate the amplitude and phase like in a regression equation.

Recall $\cos(\alpha \pm \beta) = \cos(\alpha)\cos(\beta) \mp \sin(\alpha)\sin(\beta)$. So

$$A\cos\left(\frac{2\pi t}{T} - \phi\right) = A\cos\left(\frac{2\pi t}{T}\right)\cos(\phi) + A\sin\left(\frac{2\pi t}{T}\right)\sin(\phi)$$

$$A\cos\left(\frac{2\pi t}{T} - \phi\right) = \beta_1\cos\left(\frac{2\pi t}{T}\right) + \beta_2\sin\left(\frac{2\pi t}{T}\right)$$

where $\beta_1 = A\cos(\phi)$ and $\beta_2 = A\sin(\phi)$

Therefore we can use $X_1 = \cos\left(\frac{2\pi t}{T}\right)$ and $X_2 = \sin\left(\frac{2\pi t}{T}\right)$ as predictors in a regression model, and estimate β_1 and β_2 instead of A and ϕ. We just need to know the period T.

Period hunting

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Sometimes the period T is intuitive, e.g. monthly temperatures should have a period of 12, corresponding to 1 year.

If there are multiple seasons, it may not be obvious. They may add up to create cycles of non-constant period (e.g. El Nino Southern Oscillations).

To find these periods we will consider spectral analysis.

Spectral Analysis

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

There are two branches of time series analysis:

- Time domain: The present value is modeled with the past values (time series analysis)
- Frequency domain: The present value is modeled with periodic sines and cosines (spectral analysis)

Spectral analysis acknowledges that any time series can be represented as a sum of sinusoids with different amplitudes, A, oscillating at different frequencies, f.

The frequency is the inverse of the period, i.e. T = 1/ f.

Periodogram

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

The periodogram is used in spectral analysis to discover the periodic components of a time series

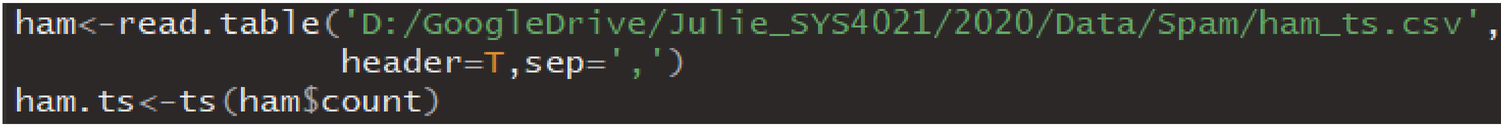
It plots the squared amplitude (often smoothed) of the data with sinusoids oscillating at a frequency $f_j = j/n$ where n is the number of time steps in the time series and $j \epsilon (1, \dots, n/2)$.

A peak in the periodogram at some frequency indicates a strong (high amplitude) sinusoid at that frequency. This suggests there is seasonality at the corresponding period.

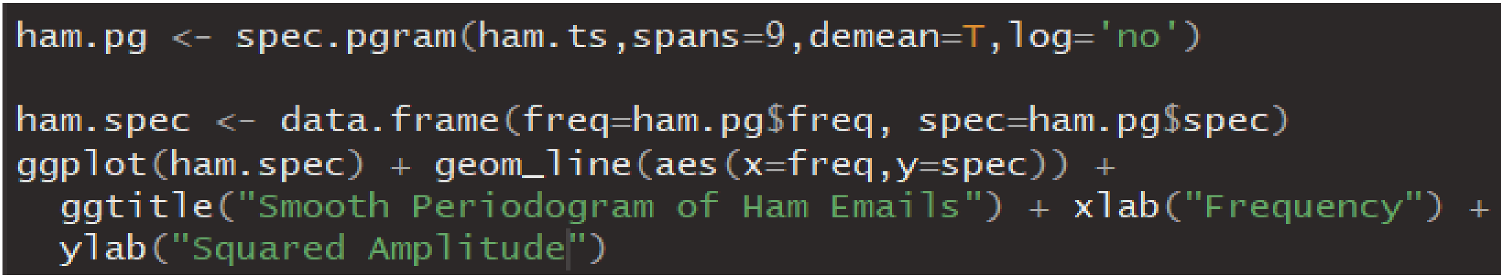
Example Periodogram

Agenda

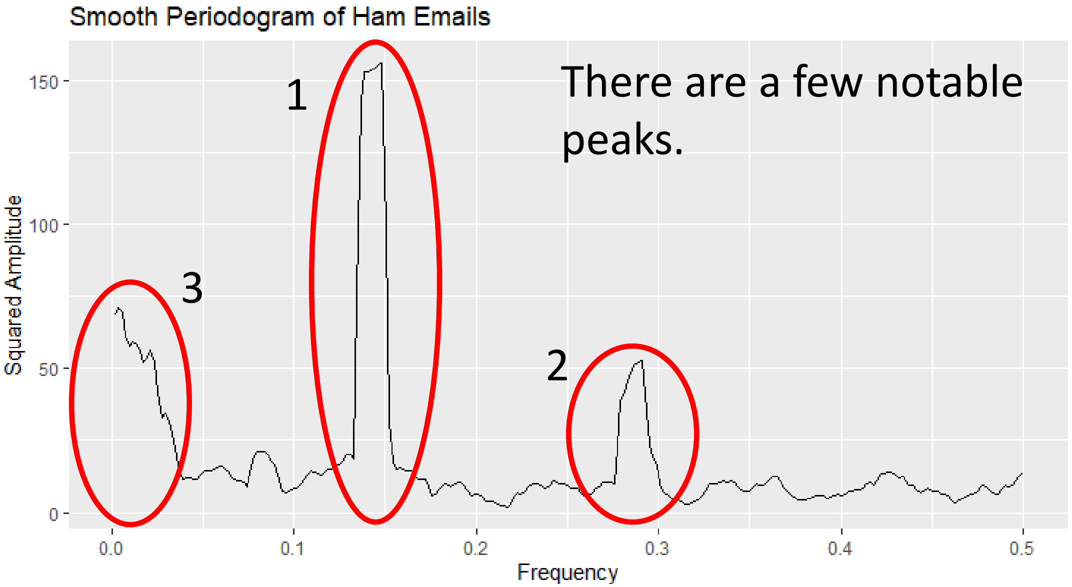
- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
ham<-read.table('D:/GoogleDrive/Julie_SYS4021/2020/Data/Spam/ham_ts.csv',
                header=T,sep=',')
ham.ts<-ts(ham$count)
```



```
ham.pg <- spec.pgram(ham.ts,spans=9,demean=T,log='no')

ham.spec <- data.frame(freq=ham.pg$freq, spec=ham.pg$spec)
ggplot(ham.spec) + geom_line(aes(x=freq,y=spec)) +
  ggtitle("Smooth Periodogram of Ham Emails") + xlab("Frequency") +
  ylab("Squared Amplitude")
```

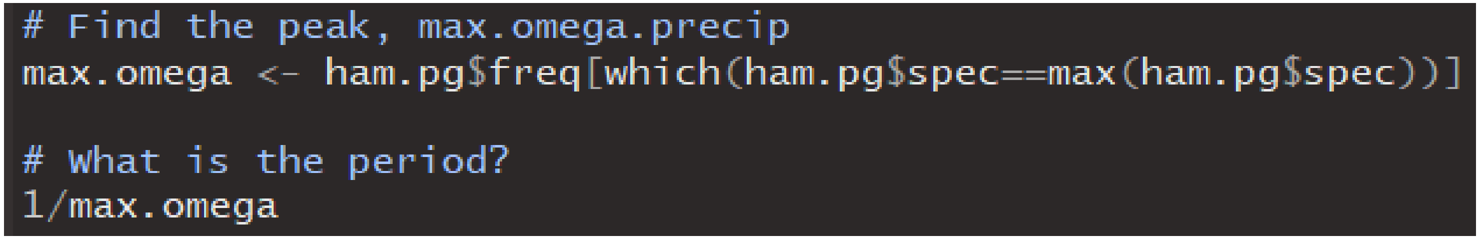


There are a few notable peaks.

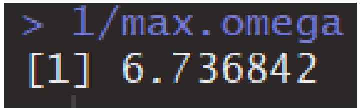
Example Periodogram

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

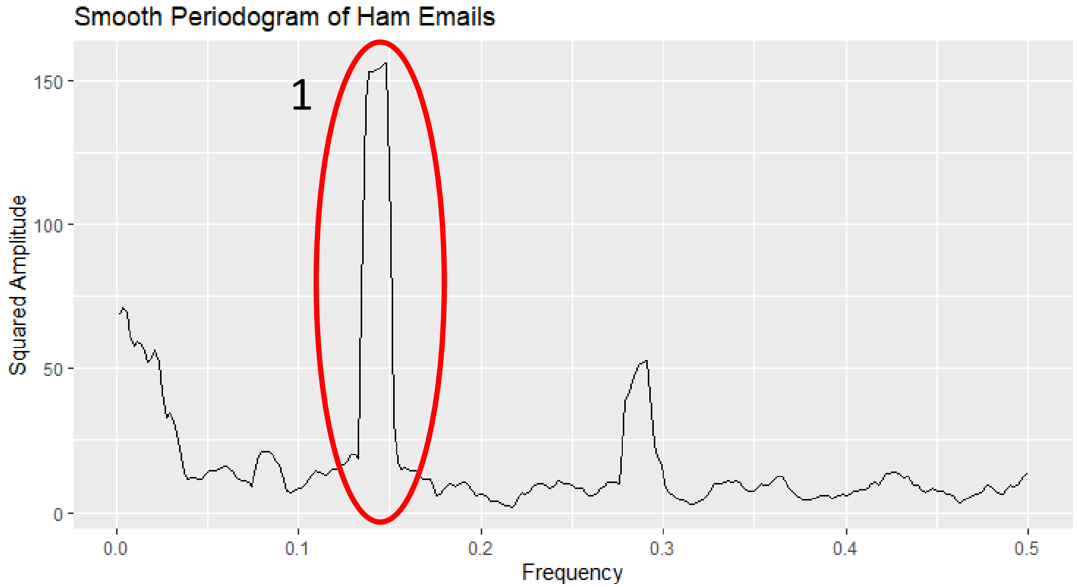


```
# Find the peak, max.omega.precip
max.omega <- ham.pg$freq[which(ham.pg$spec==max(ham.pg$spec))]

# What is the period?
1/max.omega
```



```
> 1/max.omega
[1] 6.736842
```

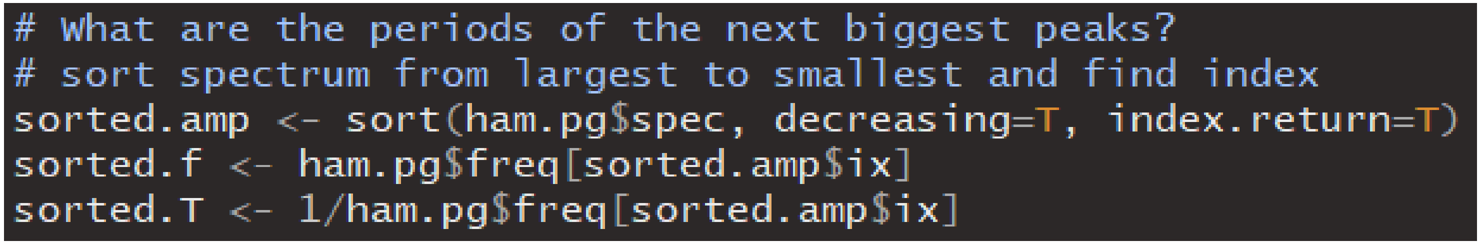
This corresponds to ~ 7 days, the expected weekly season.



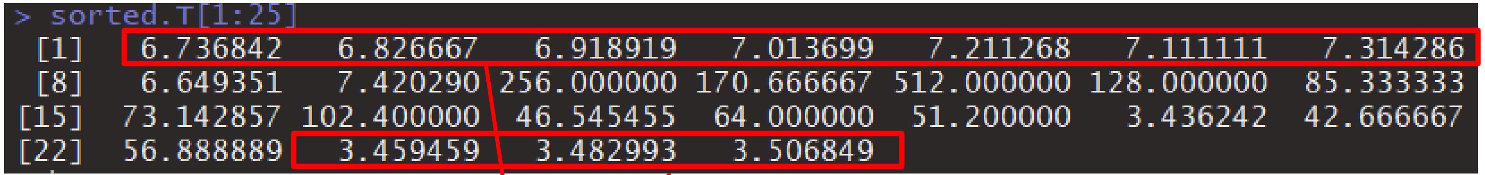
Example Periodogram

Agenda

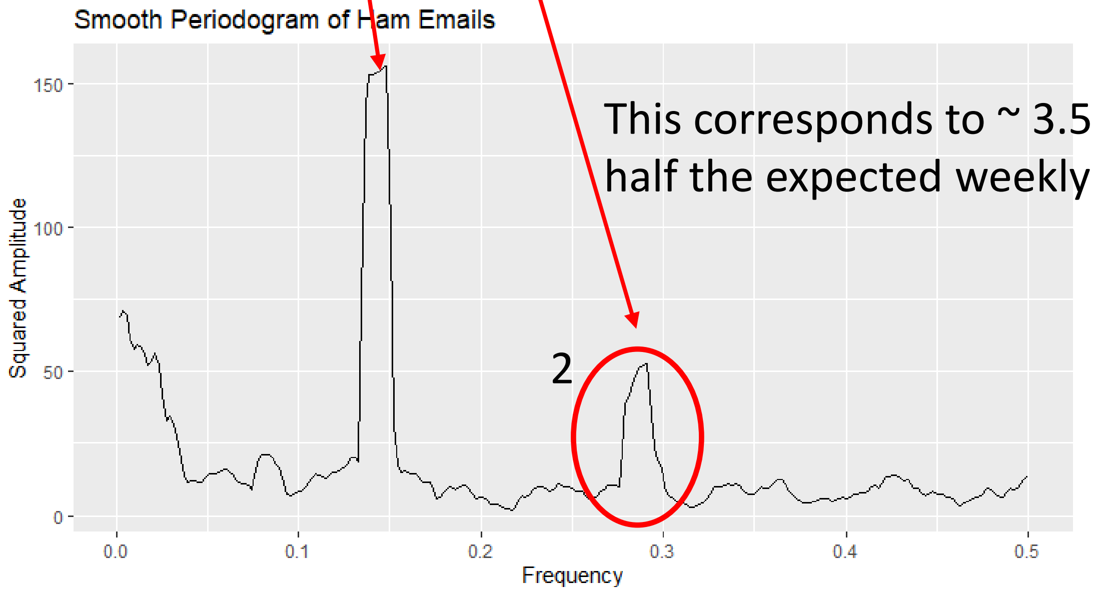
- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
# What are the periods of the next biggest peaks?
# sort spectrum from largest to smallest and find index
sorted.amp <- sort(ham.pg$spec, decreasing=T, index.return=T)
sorted.f <- ham.pg$freq[sorted.amp$ix]
sorted.T <- 1/ham.pg$freq[sorted.amp$ix]
```



```
> sorted.T[1:25]
 [1]   6.736842   6.826667   6.918919   7.013699   7.211268   7.111111   7.314286
 [8]   6.649351   7.420290 256.000000 170.666667 512.000000 128.000000  85.333333
[15]  73.142857 102.400000  46.545455  64.000000  51.200000   3.436242  42.666667
[22]  56.888889   3.459459   3.482993   3.506849
```



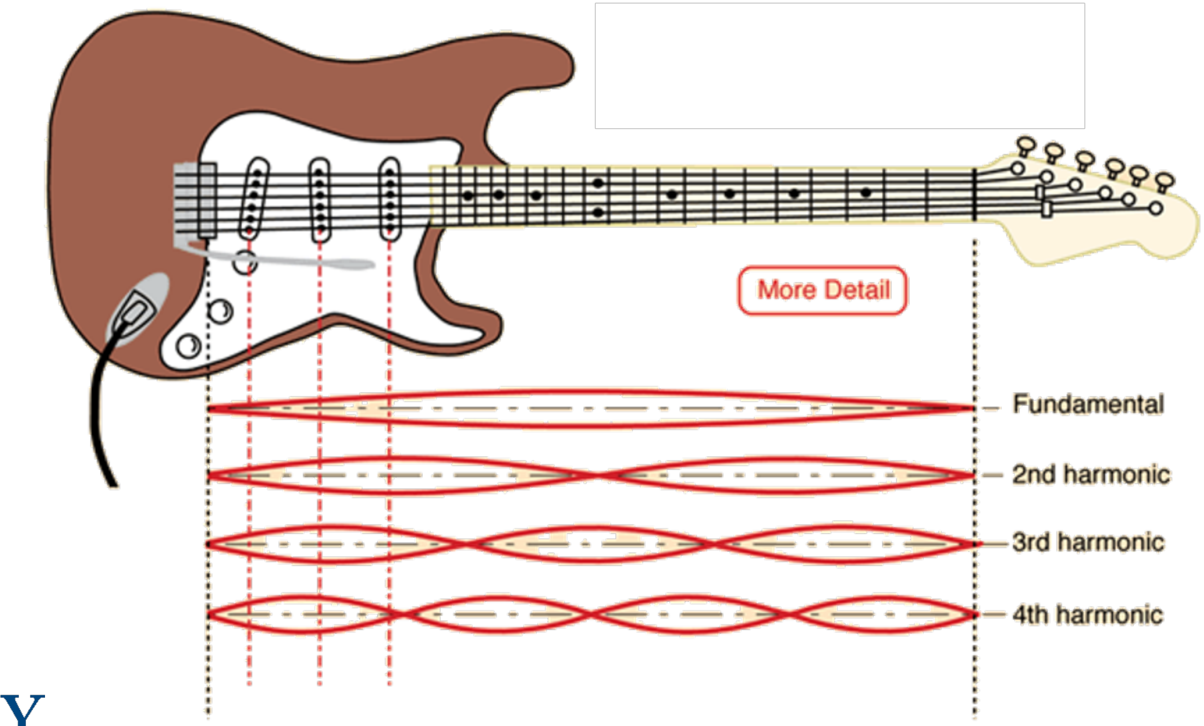
This corresponds to ~ 3.5 days, half the expected weekly season.

Harmonics

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

We will often get modulated peaks at harmonics like this. Their weighted sum (different amplitudes and phases) allows for asymmetric seasons (e.g. rise faster than fall).



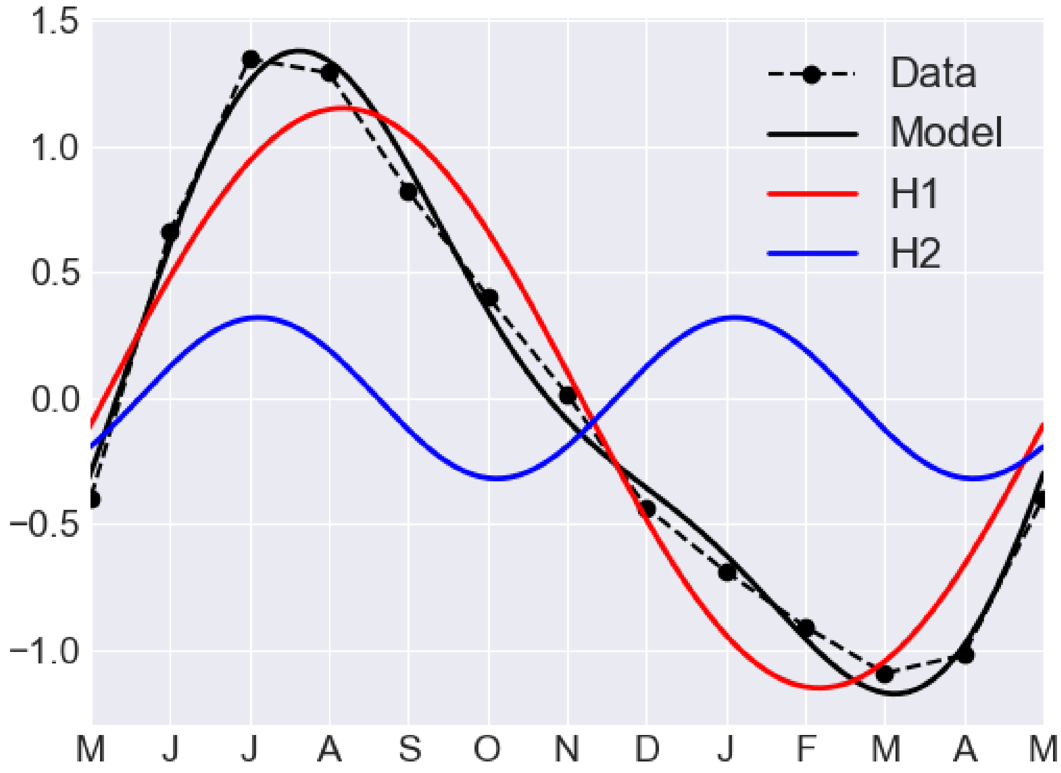
Example of harmonics in hydrologic data

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Q = Flow

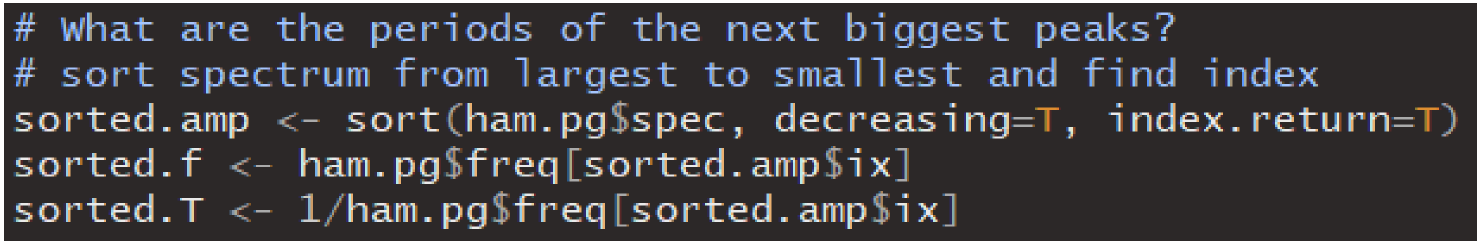
$$Z = \frac{\log(Q) - \mu_{\log(Q)}}{\sigma_{\log(Q)}}$$



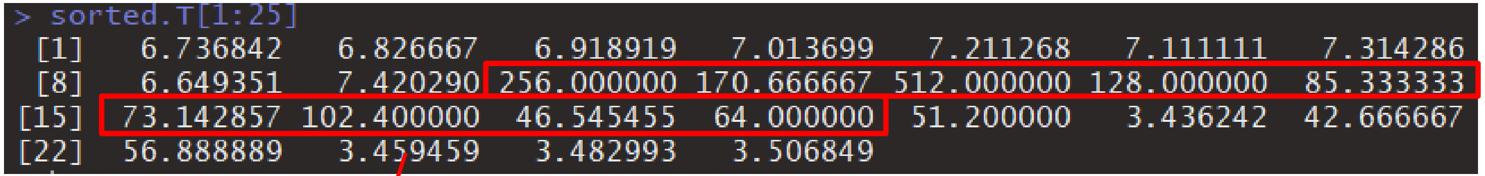
Example Periodogram

Agenda

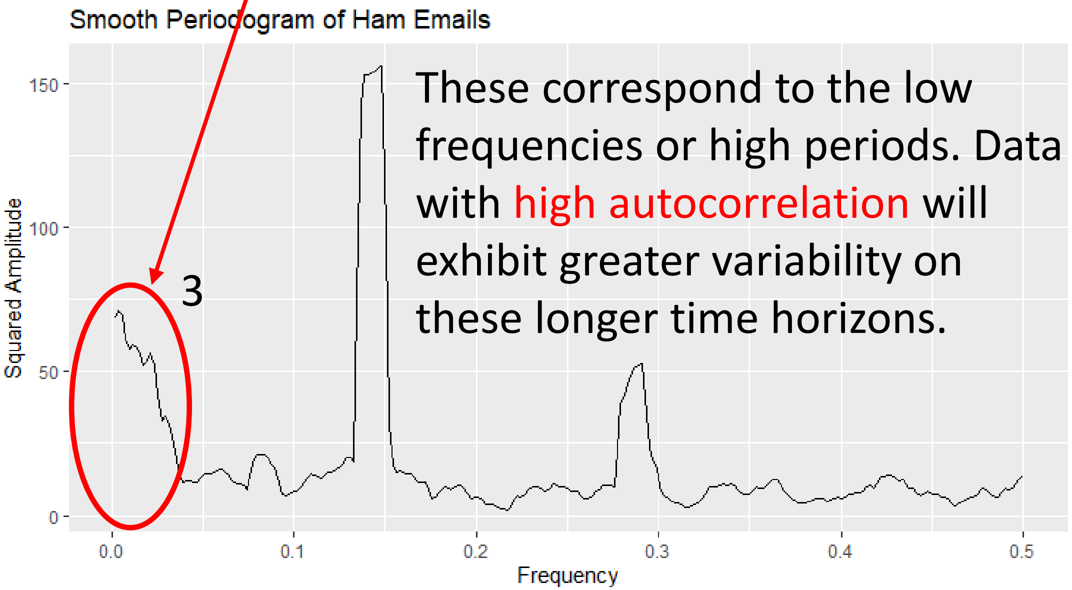
- Recap of Time Series Concept
- Modeling Seasonality
    - Spectral Analysis
- Autoregressive (AR) Models
    - Stationarity
    - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
# What are the periods of the next biggest peaks?
# sort spectrum from largest to smallest and find index
sorted.amp <- sort(ham.pg$spec, decreasing=T, index.return=T)
sorted.f <- ham.pg$freq[sorted.amp$ix]
sorted.T <- 1/ham.pg$freq[sorted.amp$ix]
```



```
> sorted.T[1:25]
 [1]   6.736842   6.826667   6.918919   7.013699   7.211268   7.111111   7.314286
 [8]   6.649351   7.420290 256.000000 170.666667 512.000000 128.000000  85.333333
[15]  73.142857 102.400000  46.545455  64.000000  51.200000   3.436242  42.666667
[22]  56.888889   3.459459   3.482993   3.506849
```

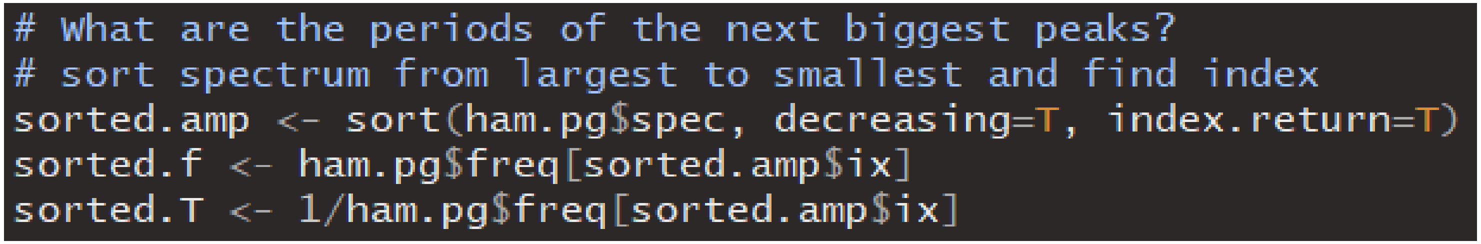


These correspond to the low frequencies or high periods. Data with high autocorrelation will exhibit greater variability on these longer time horizons.

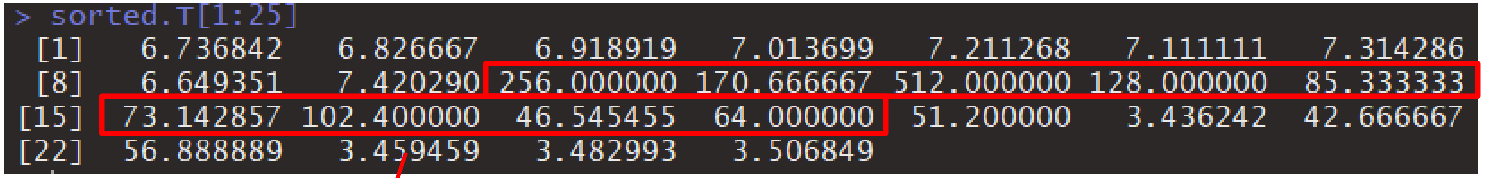
Example Periodogram

Agenda

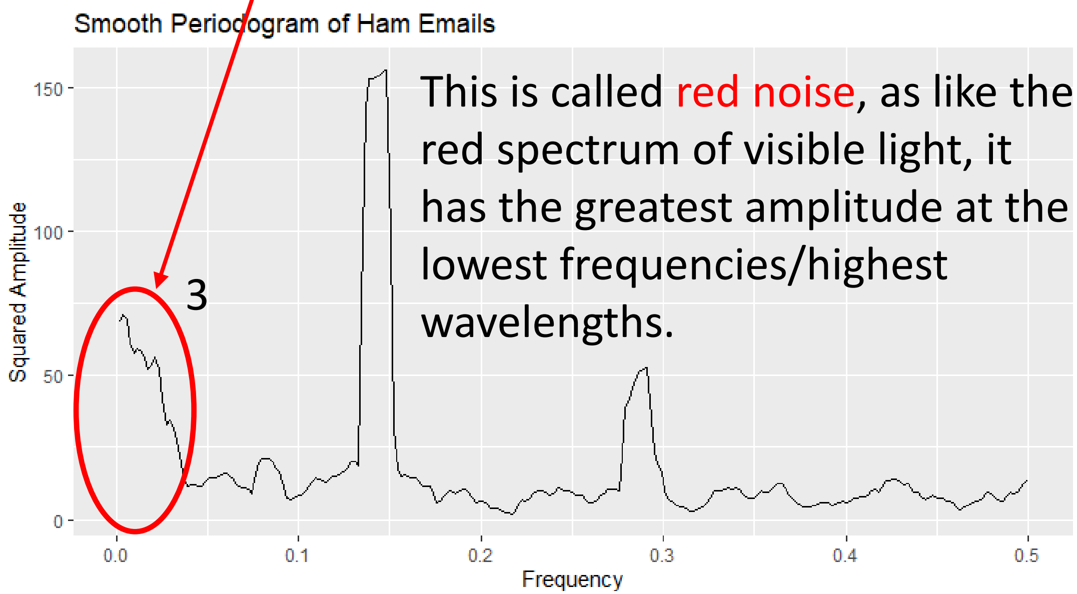
- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
# What are the periods of the next biggest peaks?
# sort spectrum from largest to smallest and find index
sorted.amp <- sort(ham.pg$spec, decreasing=T, index.return=T)
sorted.f <- ham.pg$freq[sorted.amp$ix]
sorted.T <- 1/ham.pg$freq[sorted.amp$ix]
```



```
> sorted.T[1:25]
 [1]   6.736842   6.826667   6.918919   7.013699   7.211268   7.111111   7.314286
 [8]   6.649351   7.420290 256.000000 170.666667 512.000000 128.000000  85.333333
[15]  73.142857 102.400000  46.545455  64.000000  51.200000   3.436242  42.666667
[22]  56.888889   3.459459   3.482993   3.506849
```

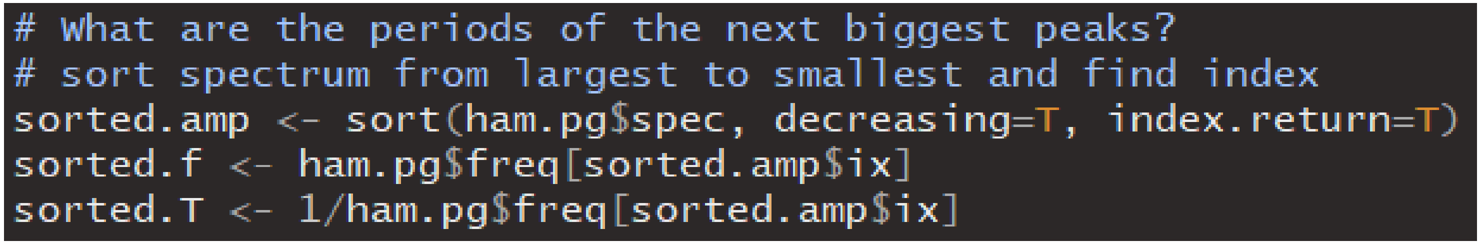


This is called red noise, as like the red spectrum of visible light, it has the greatest amplitude at the lowest frequencies/highest wavelengths.

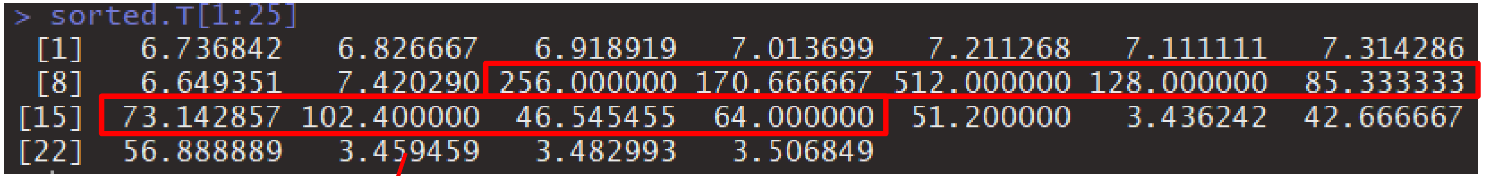
Example Periodogram

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

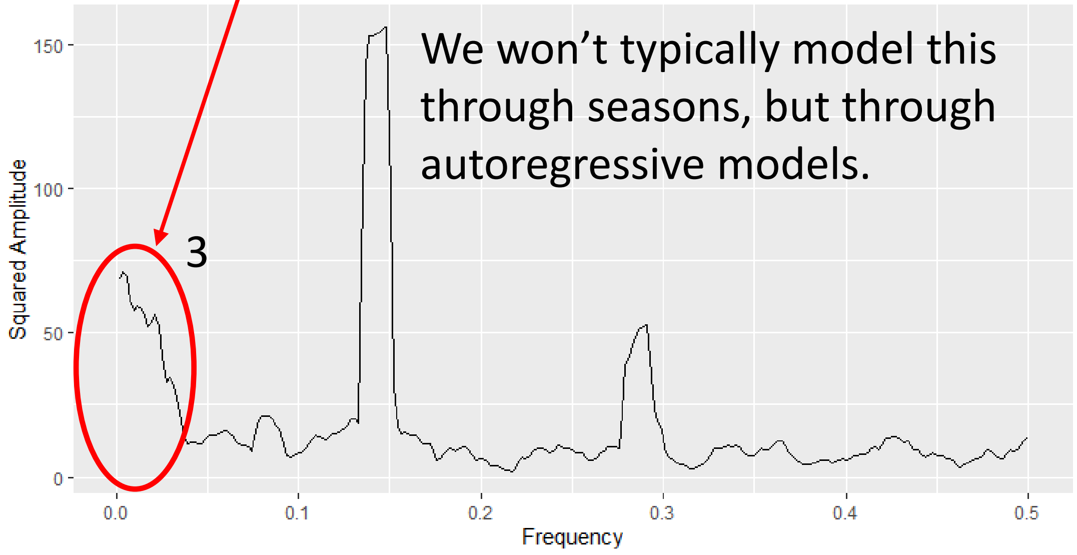


```
# What are the periods of the next biggest peaks?
# sort spectrum from largest to smallest and find index
sorted.amp <- sort(ham.pg$spec, decreasing=T, index.return=T)
sorted.f <- ham.pg$freq[sorted.amp$ix]
sorted.T <- 1/ham.pg$freq[sorted.amp$ix]
```



```
> sorted.T[1:25]
 [1]   6.736842   6.826667   6.918919   7.013699   7.211268   7.111111   7.314286
 [8]   6.649351   7.420290 256.000000 170.666667 512.000000 128.000000  85.333333
[15]  73.142857 102.400000  46.545455  64.000000  51.200000   3.436242  42.666667
[22]  56.888889   3.459459   3.482993   3.506849
```

Smooth Periodogram of Ham Emails

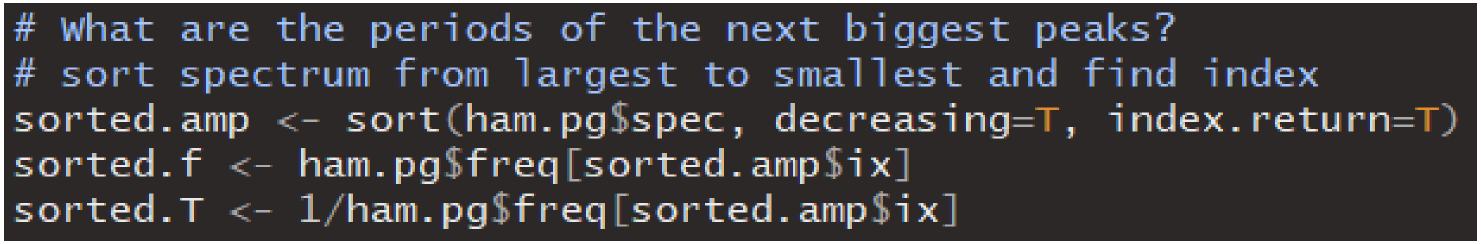


We won't typically model this through seasons, but through autoregressive models.

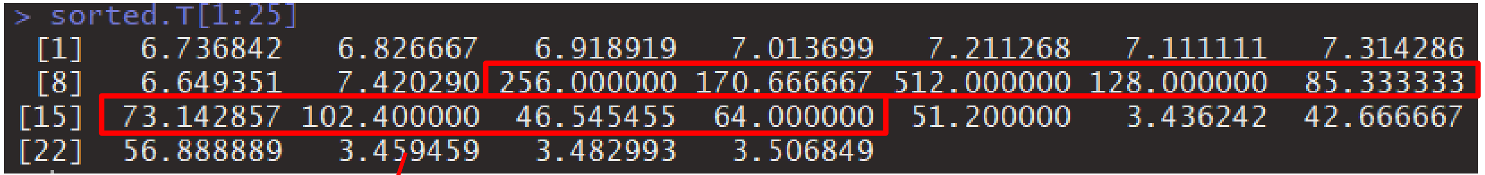
Example Periodogram

Agenda

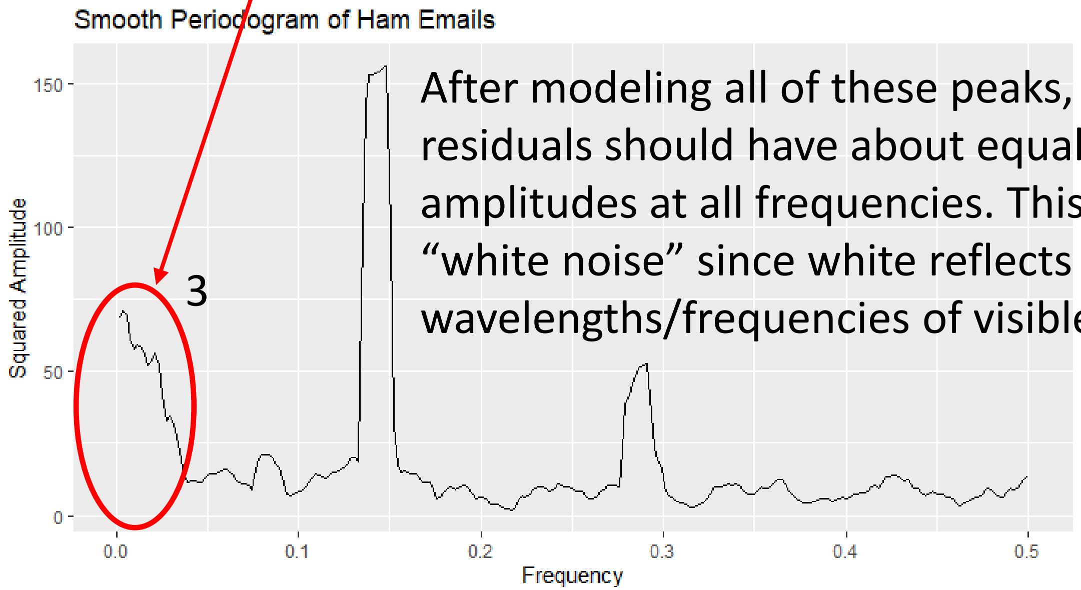
- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```r
# What are the periods of the next biggest peaks?
# sort spectrum from largest to smallest and find index
sorted.amp <- sort(ham.pg$spec, decreasing=T, index.return=T)
sorted.f <- ham.pg$freq[sorted.amp$ix]
sorted.T <- 1/ham.pg$freq[sorted.amp$ix]
```



```
> sorted.T[1:25]
 [1]   6.736842   6.826667   6.918919   7.013699   7.211268   7.111111   7.314286
 [8]   6.649351   7.420290 256.000000 170.666667 512.000000 128.000000  85.333333
[15]  73.142857 102.400000  46.545455  64.000000  51.200000   3.436242  42.666667
[22]  56.888889   3.459459   3.482993   3.506849
```



3

After modeling all of these peaks, the residuals should have about equal amplitudes at all frequencies. This is called “white noise” since white reflects all wavelengths/frequencies of visible light.

Autoregressive Models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

After modeling trends and seasonality, we will often be left with residuals that are autocorrelated. These can be modeled with autoregressive models.

An autoregressive model of order *p*, abbreviated AR(*p*), is of the form:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ \dots + \phi_p x_{t-p} + w_t$$

where

- x_t is stationary (has a constant distribution over time)
- $\phi_1, \dots, \phi_p$ are unknown regression parameters to be estimated
- w_t is assumed iid $\mathcal{N}(0, \sigma^2)$
- The mean μ of x_t is assumed to be zero. If not, it is replaced by $x_t - \mu$

Stationarity

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

We define a weakly stationary time series as one with:

- Constant mean
- Autocorrelation is a function of the lag

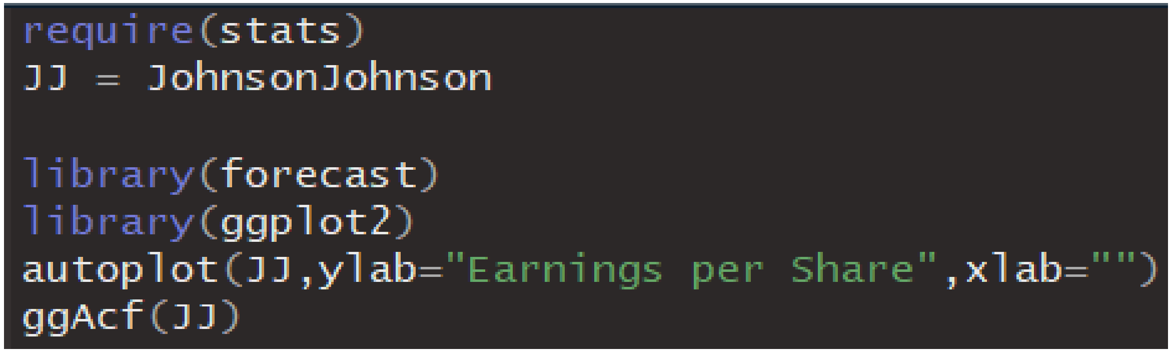
We can detect nonstationary time series by looking at the time series plot or the autocorrelation function.

- A time series with a trend is nonstationary
- An ACF with slow or linear decay is nonstationary

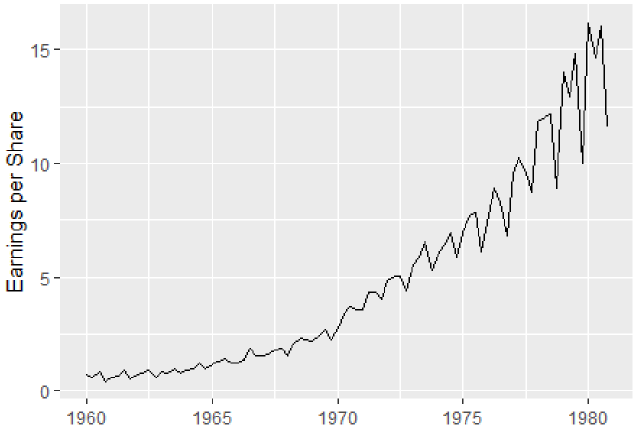
Example Non-Stationary Time Series

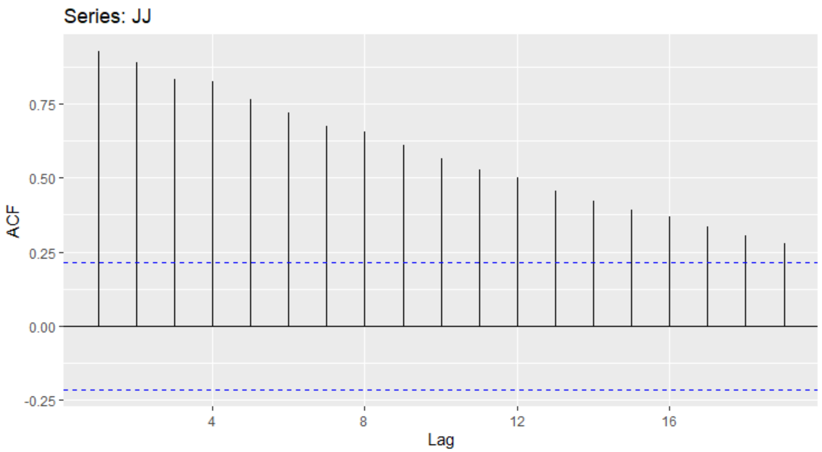
Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
require(stats)
JJ = JohnsonJohnson

library(forecast)
library(ggplot2)
autoplot(JJ,ylab="Earnings per Share",xlab="")
ggAcf(JJ)
```

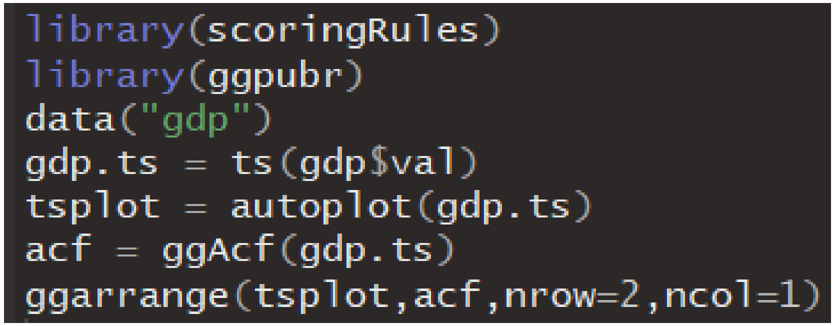




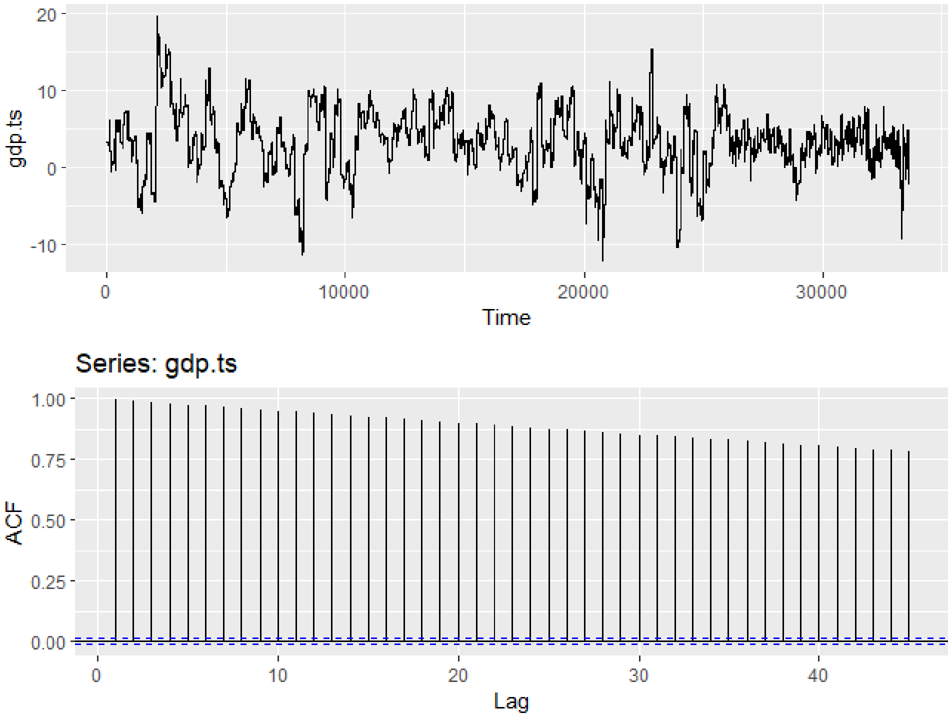
Example Non-Stationary Time Series

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
library(scoringRules)
library(ggpubr)
data("gdp")
gdp.ts = ts(gdp$val)
tsplot = autoplot(gdp.ts)
acf = ggAcf(gdp.ts)
ggarrange(tsplot,acf,nrow=2,ncol=1)
```



Stationary Time Series

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Before modeling the correlation structure in the series, we make the time series stationary

Two popular approaches are to 1) regress on time and model the residuals, or 2) take differences and model those.

First difference: $z_t = x_t - x_{t-1}$ where t = 2, 3, …, n

If first differences don't work, consider the second differences: $a_t = z_t - z_{t-1}$ where t = 3, …, n. This could capture a quadratic trend.

You can take higher order differences, but that is rare.

Autoregressive Models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

After modeling trends and seasonality, we will often be left with residuals that are autocorrelated. These can be modeled with autoregressive models.

An autoregressive model of order *p*, abbreviated AR(p), is of the form:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ \ldots + \phi_p x_{t-p} + w_t$$

where

- x_t is stationary (has a constant distribution over time)
- $\phi_1, \ldots, \phi_p$ are unknown regression parameters to be estimated
- w_t is assumed iid $\mathcal{N}(0, \sigma^2)$
- The mean μ of x_t is assumed to be zero. If not, it is replaced by $x_t - \mu$

The Order of Autoregressive Models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

We can use OLS regression to estimate $\phi_1, \dots, \phi_p$, but how do we decide on the order p?

One could build multiple models and select the model with the lowest AIC/ BIC or highest adjusted R^2.

You can also estimate the model order from plots of the ACF and what is called the partial ACF or PACF.

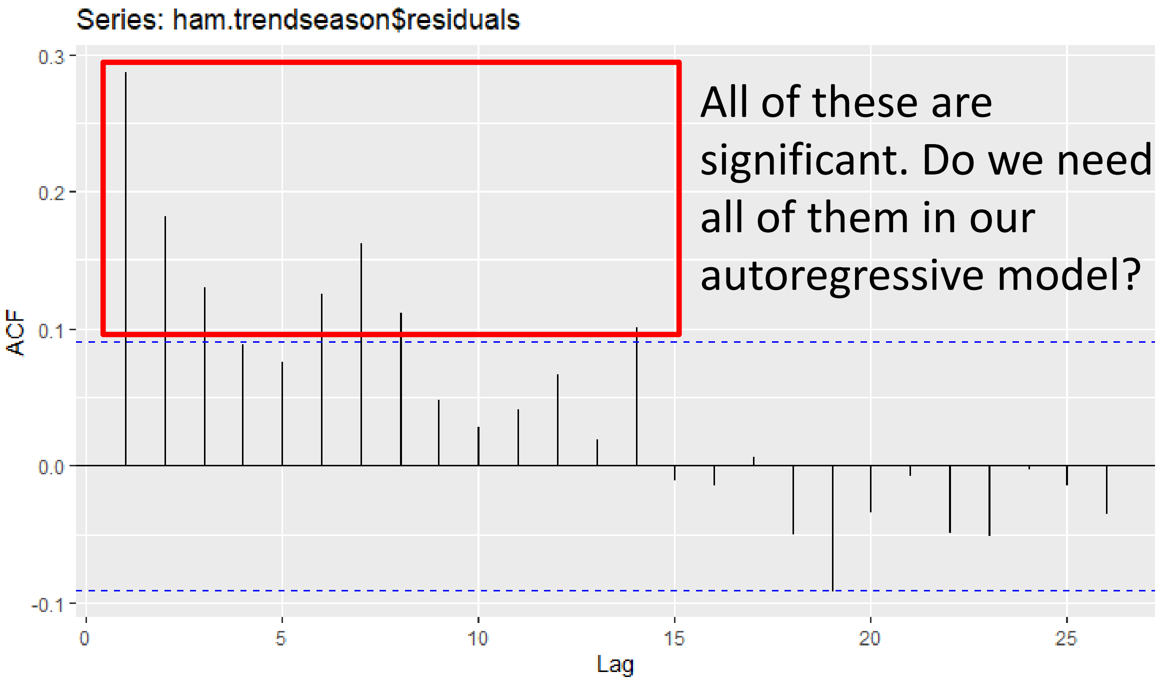
Autocorrelation

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Recall the ACF shows Corr(Y_t, Y_{t-k}) at different lags k.

Any correlation outside $\pm \frac{2}{\sqrt{n}}$ of 0 is statistically significant.



All of these are significant. Do we need all of them in our autoregressive model?

Partial Autocorrelation

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

If Corr(Y_t, Y_{t-1}) is high, so too will Corr(Y_{t-1}, Y_{t-2}) be high.

Therefore Corr(Y_t, Y_{t-2}) will likely be high as well.

If we put but Y_{t-2} and Y_{t-1} in our model, we might have problems with multicollinearity.

We only want to add Y_{t-2} in addition to Y_{t-1} if it adds predictive value while controlling for Y_{t-1}.

The partial autocorrelation assesses this value.

Partial Autocorrelation Function

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Partial autocorrelation is the correlation between values of the time series and itself at a specified lag *k* while controlling for correlations at *t*<*k*.

It is estimated by fitting AR(*p*) regression models progressively, incrementing *p* by one. The last coefficient in each regression (ϕ_p for x_{t-p}) is the partial autocorrelation at lag *p*.

For example, the partial autocorrelation at lag 1 is ϕ_1 in:

$$x_t = \phi_1 x_{t-1} + w_t$$

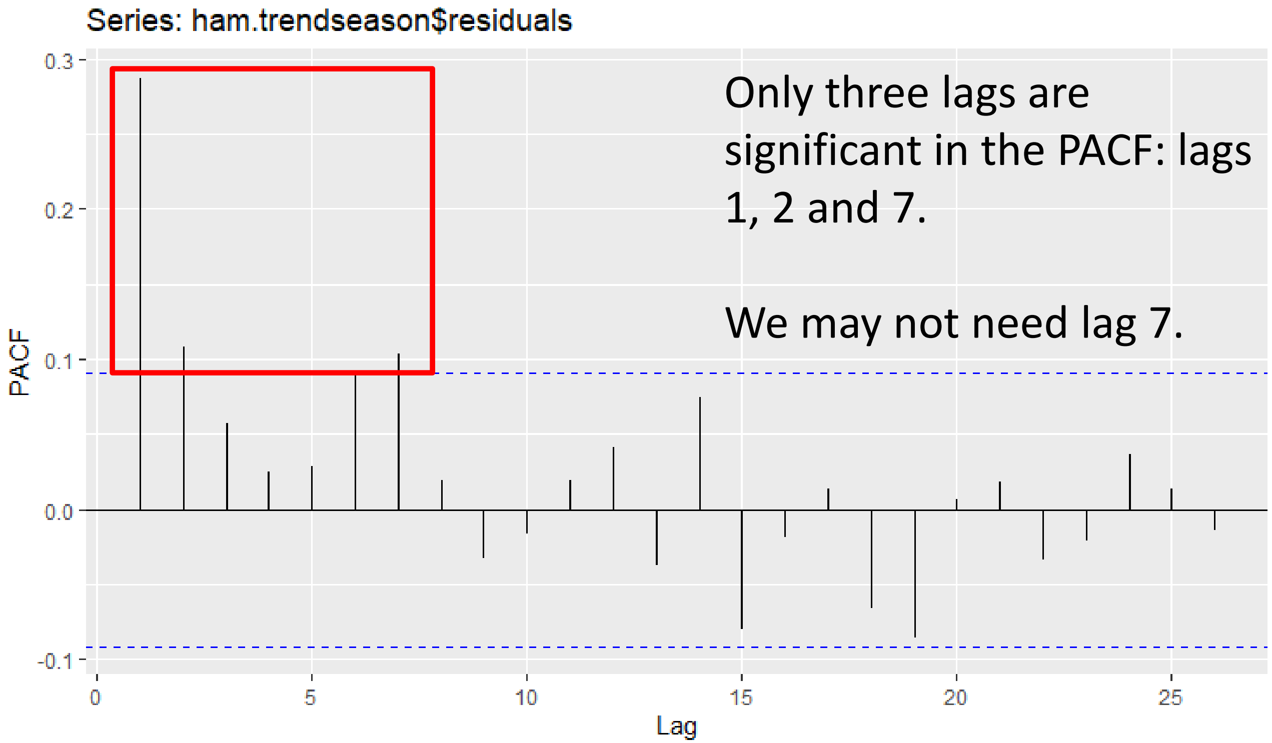
The partial autocorrelation at lag 2 is ϕ_2 in:

$$x_t = \phi_1 x_{t-1} + \phi_2\, x_{t-2} + w_t \text{ etc.}$$

Partial Autocorrelation Function

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



Only three lags are significant in the PACF: lags 1, 2 and 7.

We may not need lag 7.

Choosing the order of AR models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
    - Spectral Analysis
- Autoregressive (AR) Models
    - Stationarity
    - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

For AR(p) models the PACF cuts-off, or becomes insignificant, after p lags.

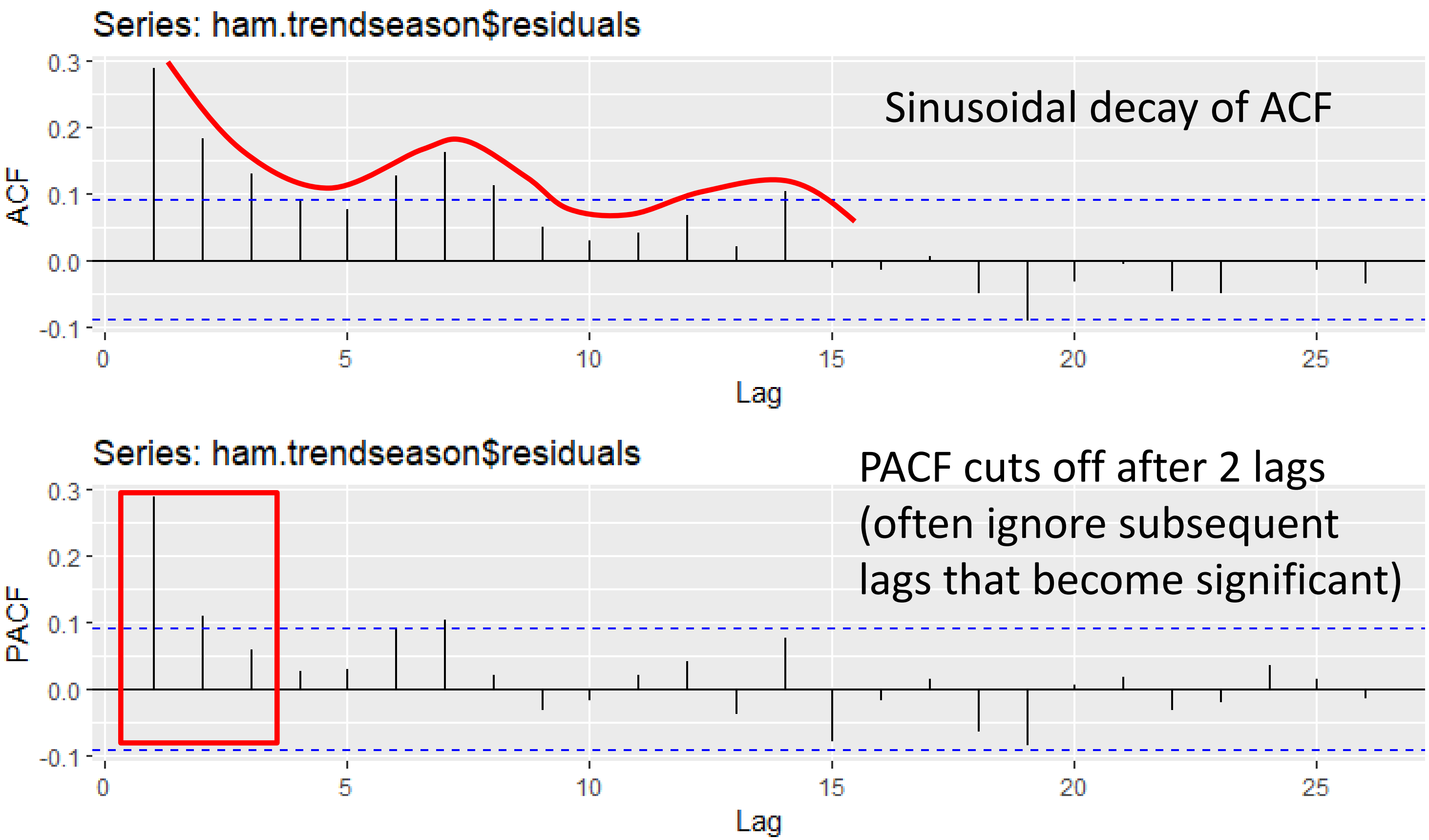
For AR(p) models the ACF shows sinusoidal or exponential decay (tails off).

For nonstationary processes the ACF shows linear decay.

What order would you choose for these residuals?

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
    - Spectral Analysis
- Autoregressive (AR) Models
    - Stationarity
    - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



Sinusoidal decay of ACF

PACF cuts off after 2 lags (often ignore subsequent lags that become significant)

Try AR(2) model. But what if the ACF and PACF don't follow this pattern?

Moving Average Models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Another way to model persistence in residuals is with moving average models.

A moving average model of order *q*, abbreviated as MA(*q*), is defined as:

$$x_t = \theta_1 w_{t-1} + \theta_2 w_{t-2} + \cdots + \theta_q w_{t-q} + w_t$$

where

- $\theta_1, \ldots, \theta_q$ are unknown regression parameters to be estimated
- w_t is assumed iid $\mathcal{N}(0, \sigma^2)$
- w_{t-k} are past residuals of the moving average model at lag *k*.

ACF and PACF of MA Models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Because MA models must estimate coefficients on past residuals, which depend on the model fit, they have to be estimated iteratively through maximum likelihood estimation.

MA models are not as easily interpreted as AR models, but they are useful for modeling many real-world time series.

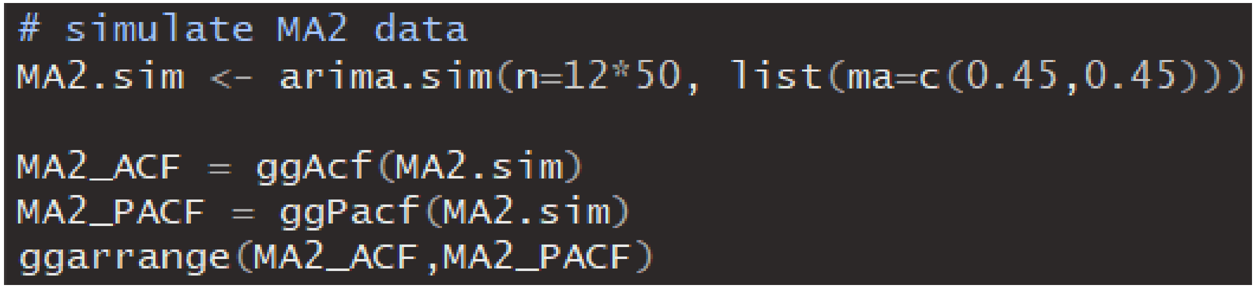
The order q of MA models can be determined according to the opposite pattern of AR models:

- The ACF cuts off or becomes insignificant after q lags.
- The PACF shows sinusoidal or exponential decay (tails off)

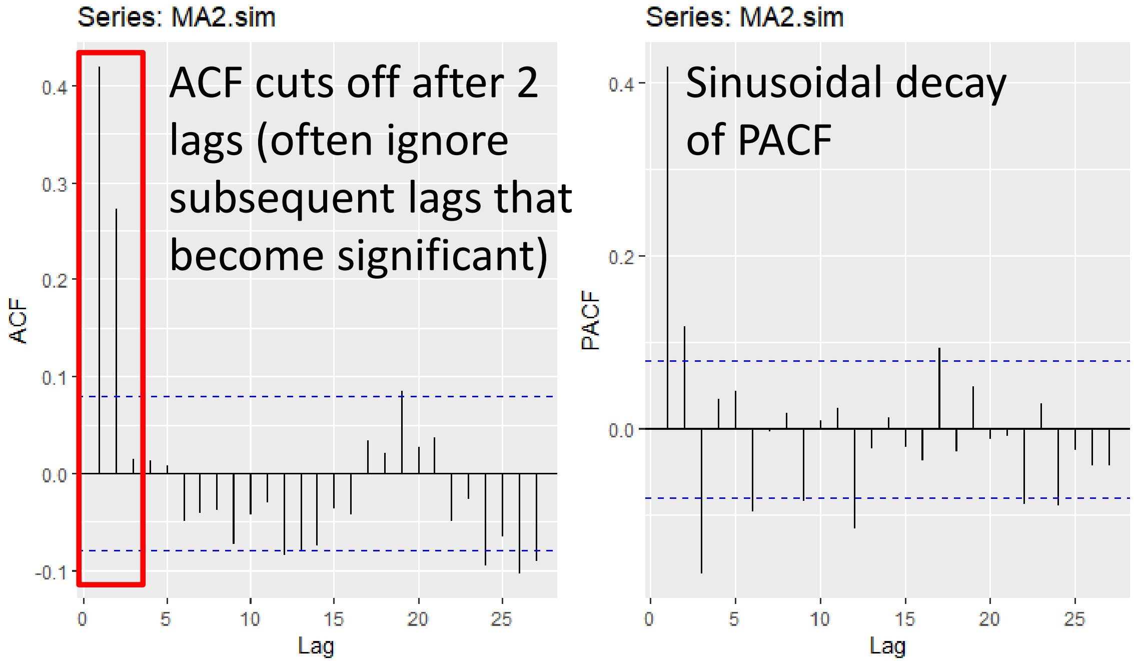
Example Moving Average Time Series

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
# simulate MA2 data
MA2.sim <- arima.sim(n=12*50, list(ma=c(0.45,0.45)))

MA2_ACF = ggAcf(MA2.sim)
MA2_PACF = ggPacf(MA2.sim)
ggarrange(MA2_ACF,MA2_PACF)
```



ACF cuts off after 2 lags (often ignore subsequent lags that become significant)

Sinusoidal decay of PACF

Try MA(2) model. But what if the ACF and PACF don't follow this pattern either?

Autoregressive Moving Average Models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

Some time series contain autoregressive and moving average components

An autoregressive moving average model, abbreviated as ARMA(*p*,*q*) has *p* AR terms and *q* MA terms:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ \ldots + \phi_p x_{t-p} + \theta_1 w_{t-1} + \theta_2 w_{t-2} + \cdots + \theta_q w_{t-q} + w_t$$

where

- x_t is stationary with 0 mean
- $\phi_p \neq 0$ and $\theta_q \neq 0$ (but ϕ_i for $i<p$ and θ_j $j<q$ might be 0)
- w_t is assumed iid $\mathcal{N}(0, \sigma^2)$
- w_{t-k} are past residuals of the ARMA model at lag *k*.

Estimating ARMA models

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

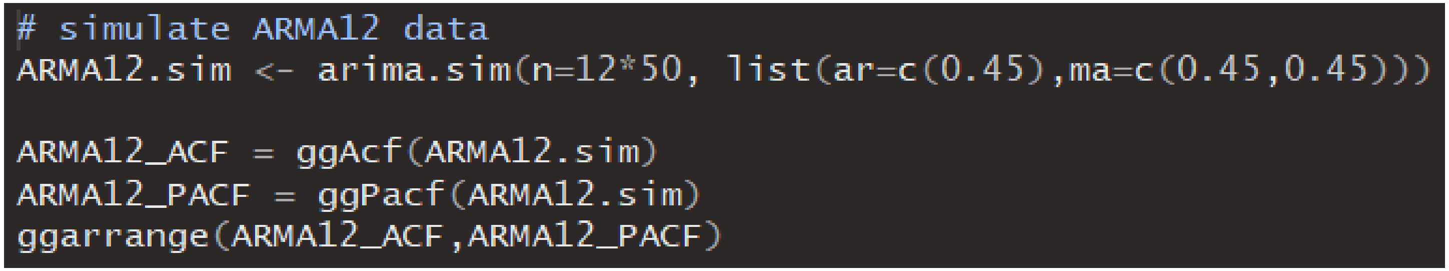
Like MA models, the coefficients of ARMA models must be estimated through maximum likelihood estimation.

Since both AR and MA components are embedded in the time series, both the ACF and PACF tail off in a sinusoidal pattern or exponentially decay.

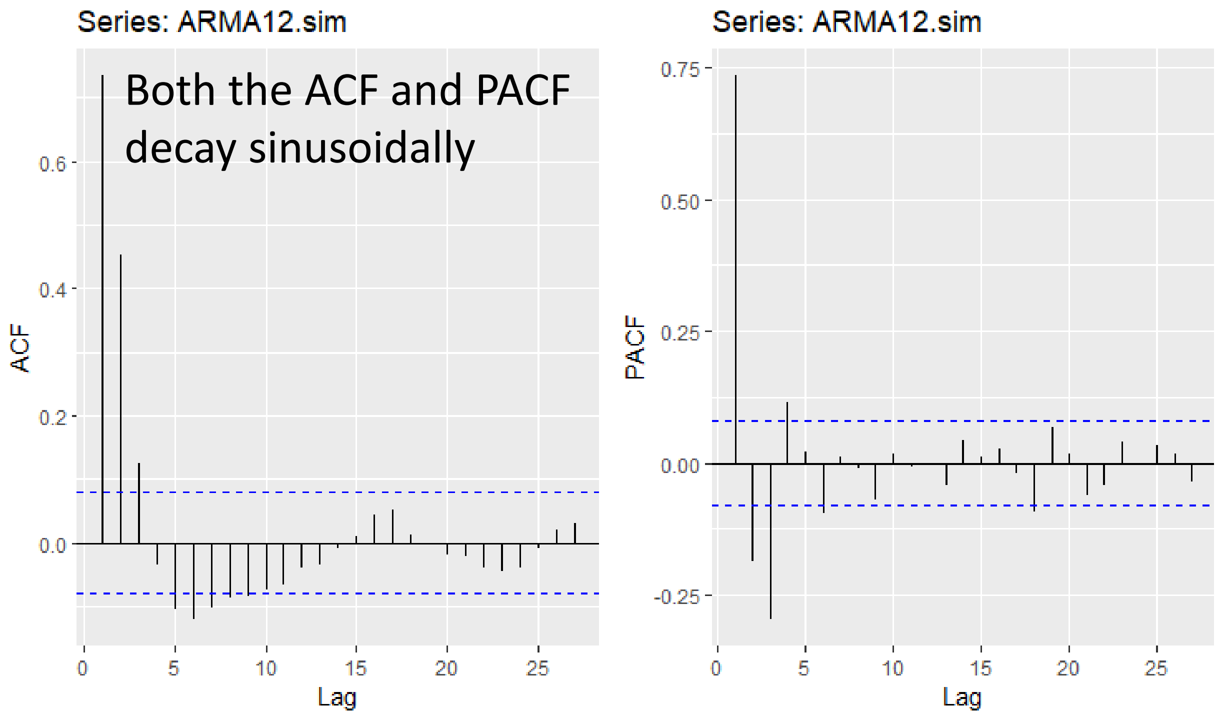
Example ARMA Time Series

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary



```
# simulate ARMA12 data
ARMA12.sim <- arima.sim(n=12*50, list(ar=c(0.45),ma=c(0.45,0.45)))

ARMA12_ACF = ggAcf(ARMA12.sim)
ARMA12_PACF = ggPacf(ARMA12.sim)
ggarrange(ARMA12_ACF,ARMA12_PACF)
```



Both the ACF and PACF decay sinusoidally

Summary

Agenda

- Recap of Time Series Concept
- Modeling Seasonality
  - Spectral Analysis
- Autoregressive (AR) Models
  - Stationarity
  - ACF and PACF
- Moving Average (MA) Models
- Autoregressive Moving Average (ARMA) Models
- Summary

To see if there are seasonal components in a time series, and to estimate their periods, one can use the periodogram.

Peaks in the periodogram at low frequencies can be a sign of autocorrelation, not seasonality.

Autocorrelation can be modeled through autoregressive (AR), moving average (MA), and autoregressive moving average (ARMA) models.

The ACF and PACF can help identify the orders of these models. When unclear, metrics like AIC can be used.

Time Series Analysis 3

SYS 4021/ 6021

Laura Barnes and Julianne Quinn

Organization of lecture

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

1. Review of Time Series and Autoregressive (AR) Models
2. Autoregressive Moving Average (ARMA) Models
3. Autoregressive Integrated Moving Average (ARIMA) Models
4. Time Series Model Diagnostics
5. Forecasting and Simulating with Time Series Models

Review of Time Series Models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
    - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Trait	Visualize with	Model with
Trends	- Monotonically increasing or decreasing plot of time series itself - Slow, linearly decaying ACF	- Linear model with time as a predictor if significant (control for seasonality when testing) - Differencing time series
Seasons	- Repeated peaks in ACF at period T - Peaks in periodogram at frequency f (better approach than ACF if there are multiple seasons)	- Dummy variables if small number of seasons - Linear model with sin(2πt/T) and cos(2πt/T) as predictors at all relevant periods T if smoothly varying seasons
Cycles	- Plot of time series	- Possibly sum of multiple seasons from periodogram
Auto-correlation	- ACF and PACF (of residuals of trend + seasonality model)	- ARMA models

Simple Time Series Models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

We can capture trends, seasonality and autocorrelation through a series of models.

For example, to model a variable Y that exhibits a trend and seasonality over L seasons, use:

$$Y_t = \beta_0 + \beta_1 t + \sum_{i=1}^{L-1} \beta_{i+1} X_i + \varepsilon_t = E[Y_t] + \varepsilon_t$$

$$\varepsilon_t = \sum_{j=1}^{k} \varphi_j \varepsilon_{t-j} + w_t$$

where $X_i = 1$ if season *i*, 0 otherwise and $w_t \sim \mathcal{N}(0, \sigma^2)$ and iid.

Review of Autoregressive (AR) Models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

After modeling trends and seasonality, we will often be left with residuals that are autocorrelated. These can be modeled with autoregressive models.

An autoregressive model of order *p*, abbreviated AR(*p*), is of the form:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ \ldots + \phi_p x_{t-p} + w_t$$

where

- x_t is stationary (has a constant distribution over time)
- $\phi_1, \ldots, \phi_p$ are unknown regression parameters to be estimated
- w_t is assumed iid $\mathcal{N}(0, \sigma^2)$
- The mean μ of x_t is assumed to be zero. If not, it is replaced by $x_t - \mu$

Choosing the order of AR models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

For AR(p) models the PACF cuts-off, or becomes insignificant, after p lags.

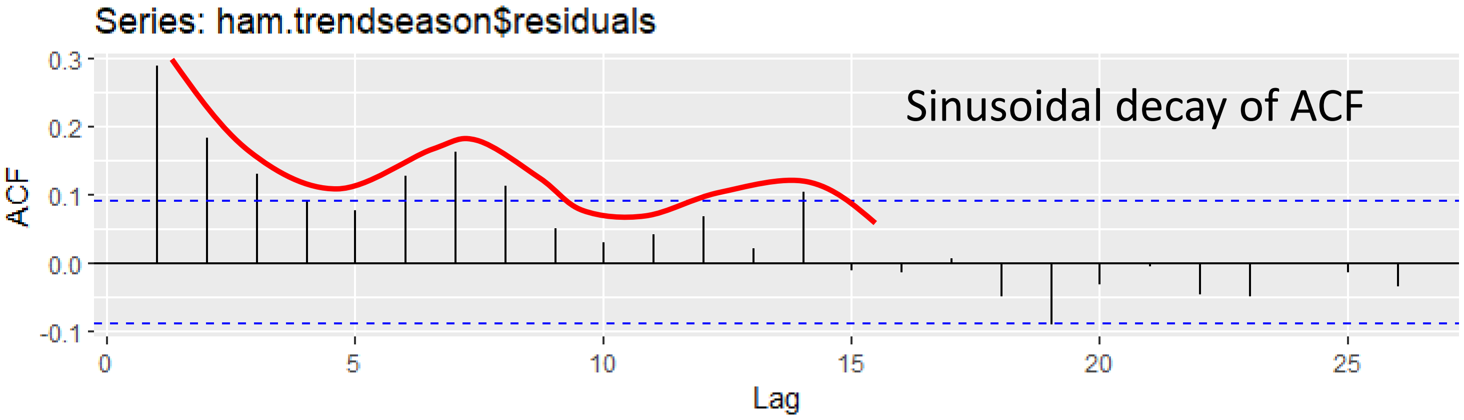
For AR(p) models the ACF shows sinusoidal or exponential decay (tails off).

For nonstationary processes the ACF shows linear decay.

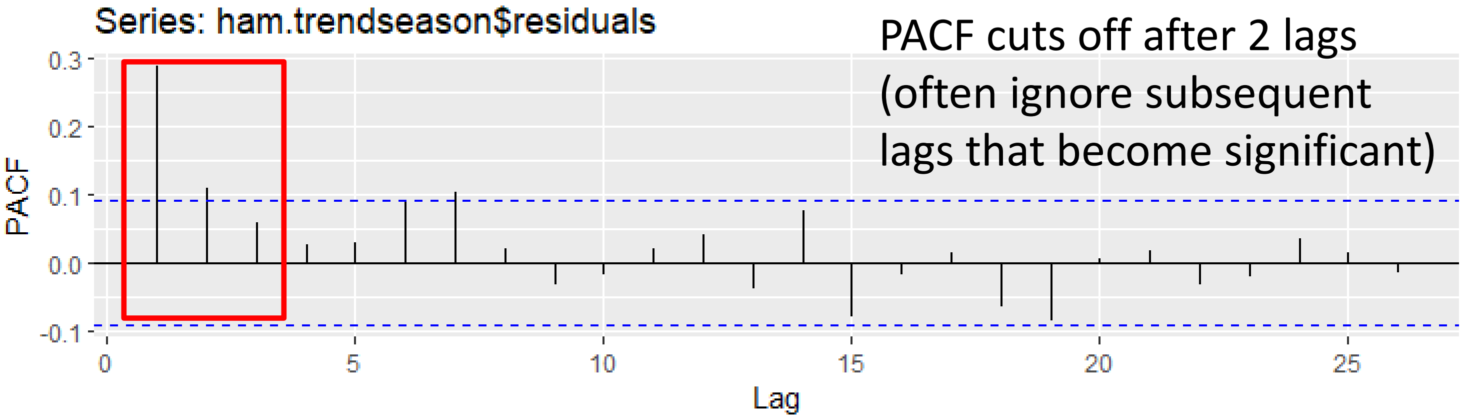
What order would you choose for these residuals?

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



Sinusoidal decay of ACF



PACF cuts off after 2 lags (often ignore subsequent lags that become significant)

Try AR(2) model. But what if the ACF and PACF don't follow this pattern?

Moving Average Models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Another way to model persistence in residuals is with moving average models.

A moving average model of order q, abbreviated as MA(q), is defined as:

$$x_t = \theta_1 w_{t-1} + \theta_2 w_{t-2} + \cdots + \theta_q w_{t-q} + w_t$$

where

- $\theta_1, \ldots, \theta_q$ are unknown regression parameters to be estimated
- w_t is assumed iid $\mathcal{N}(0, \sigma^2)$
- w_{t-k} are past residuals of the moving average model at lag k.

ACF and PACF of MA Models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Because MA models must estimate coefficients on past residuals, which depend on the model fit, they have to be estimated iteratively through maximum likelihood estimation.

MA models are not as easily interpreted as AR models, but they are useful for modeling many real-world time series.

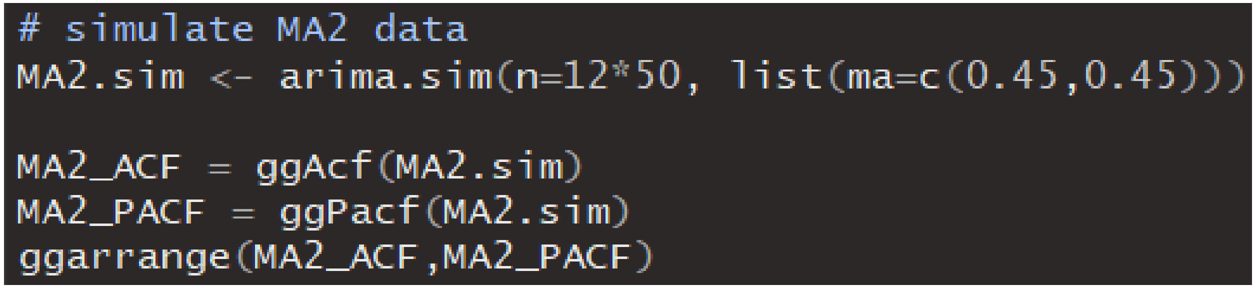
The order *q* of MA models can be determined according to the opposite pattern of AR models:

- The ACF cuts off or becomes insignificant after *q* lags.
- The PACF shows sinusoidal or exponential decay (tails off)

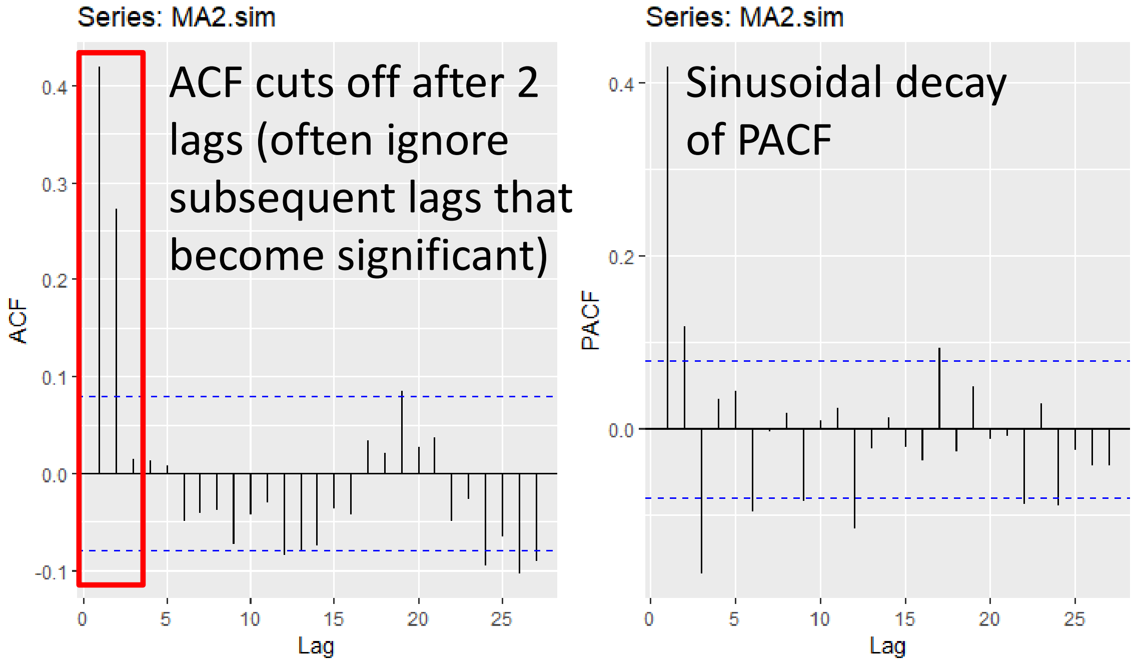
Example Moving Average Time Series

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
# simulate MA2 data
MA2.sim <- arima.sim(n=12*50, list(ma=c(0.45,0.45)))

MA2_ACF = ggAcf(MA2.sim)
MA2_PACF = ggPacf(MA2.sim)
ggarrange(MA2_ACF,MA2_PACF)
```



ACF cuts off after 2 lags (often ignore subsequent lags that become significant)

Sinusoidal decay of PACF

Try MA(2) model. But what if the ACF and PACF don't follow this pattern either?

Autoregressive Moving Average Models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Some time series contain autoregressive and moving average components

An autoregressive moving average model, abbreviated as ARMA(p,q) has p AR terms and q MA terms:

$$x_t = \phi_1 x_{t-1} + \phi_2 x_{t-2} + \ldots + \phi_p x_{t-p} + \theta_1 w_{t-1} + \theta_2 w_{t-2} + \cdots + \theta_q w_{t-q} + w_t$$

where

- x_t is stationary with 0 mean
- $\phi_p \neq 0$ and $\theta_q \neq 0$ (but ϕ_i for $i<p$ and θ_j $j<q$ might be 0)
- w_t is assumed iid $\mathcal{N}(0, \sigma^2)$
- w_{t-k} are past residuals of the ARMA model at lag k.

Estimating ARMA models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

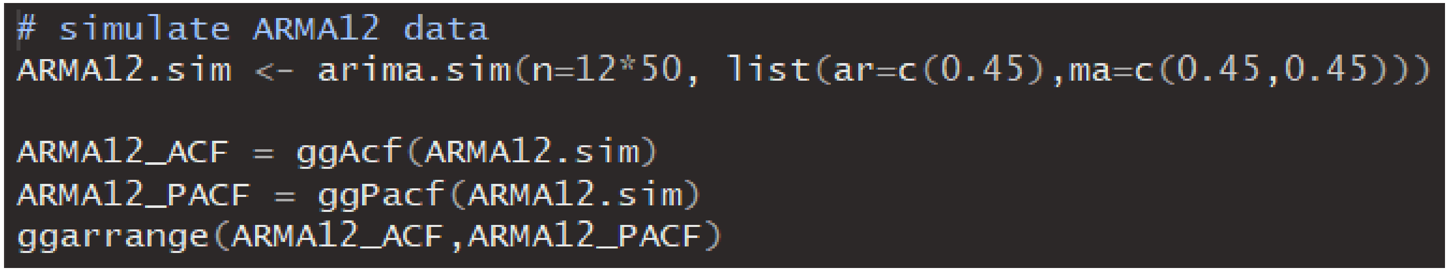
Like MA models, the coefficients of ARMA models must be estimated through maximum likelihood estimation.

Since both AR and MA components are embedded in the time series, both the ACF and PACF tail off in a sinusoidal pattern or exponentially decay.

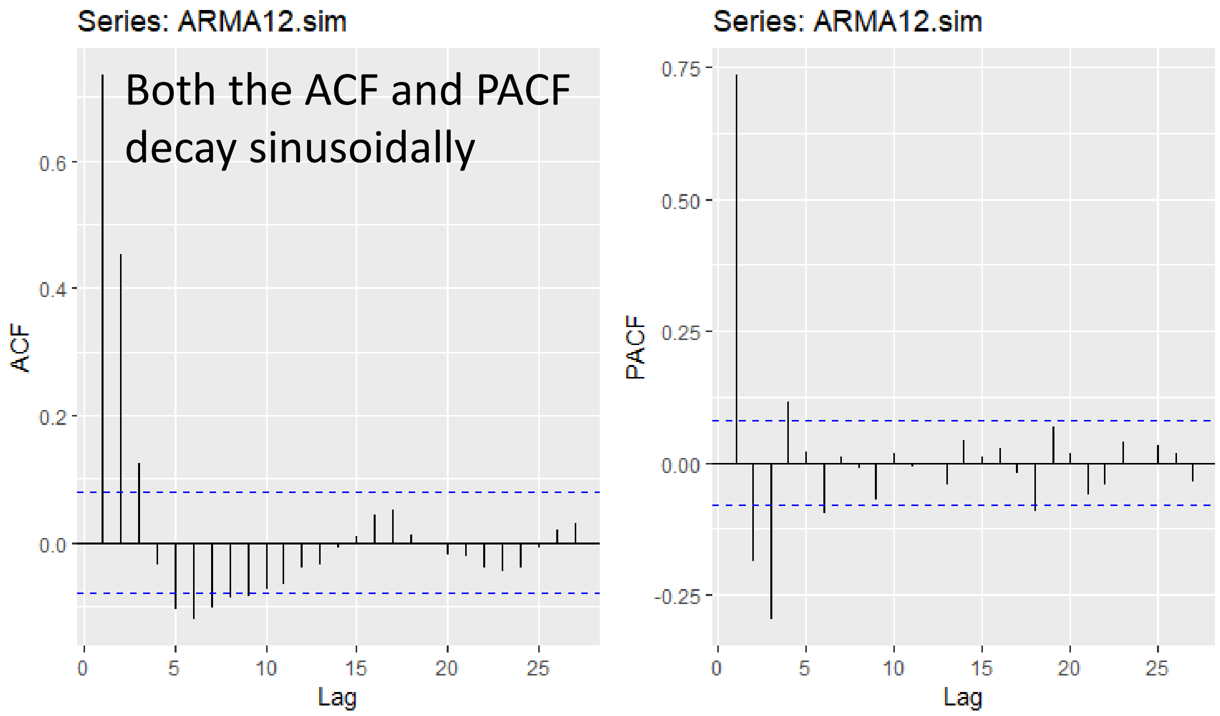
Example ARMA Time Series

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
    - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
# simulate ARMA12 data
ARMA12.sim <- arima.sim(n=12*50, list(ar=c(0.45),ma=c(0.45,0.45)))

ARMA12_ACF = ggAcf(ARMA12.sim)
ARMA12_PACF = ggPacf(ARMA12.sim)
ggarrange(ARMA12_ACF,ARMA12_PACF)
```



Both the ACF and PACF decay sinusoidally

Recap: Behavior of ACF and PACF

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

We can often identify appropriate time series models from the autocorrelation and partial autocorrelation functions

	Nonstationary	AR(*p*)	MA(*q*)	ARMA(*p*,*q*)
ACF	Slow, linear decay	Sinusoidal or exponential decay	Cuts off after *q* lags	Sinusoidal or exponential decay
PACF		Cuts off after *p* lags	Sinusoidal or exponential decay	Sinusoidal or exponential decay

Backshift Operator

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

To simplify the notation of ARMA models, we often use the backshift operator, *B*:

$$Bx_t = x_{t-1}$$

We can extend it to powers:

$$B^2 x_t = B(Bx_t) = Bx_{t-1} = x_{t-2}$$

We can use similar notation for differencing with ∇:

$$\nabla x_t = x_t - x_{t-1}$$

Writing this definition using the backshift operator we see

$$\nabla x_t = x_t - Bx_t = (1 - B)x_t$$

Differences of order *d* can be written generally as:

$$\nabla^d x_t = (1 - B)^d x_t$$

Autoregressive Integrated Moving Average Models

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

ARMA models can only model x_t if it is stationary.

If it is not stationary we can take differences.

This can be done in one step with autoregressive integrated moving average (ARIMA) models.

A stochastic process, x_t, is said to be ARIMA(p,d,q) if

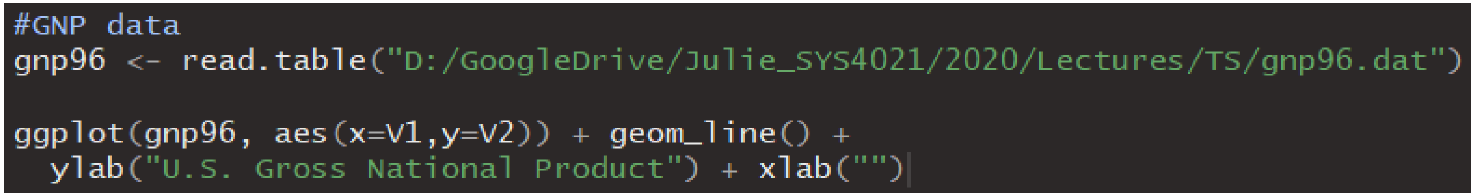
$$\nabla^d x_t = (1 - B)^d x_t$$

is ARMA(p,q).

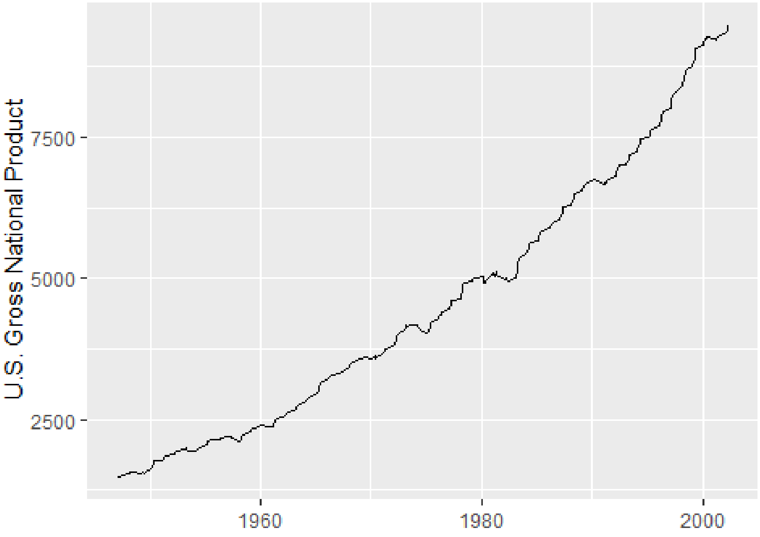
Example ARIMA time series: GNP

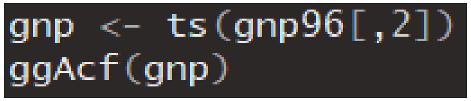
Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

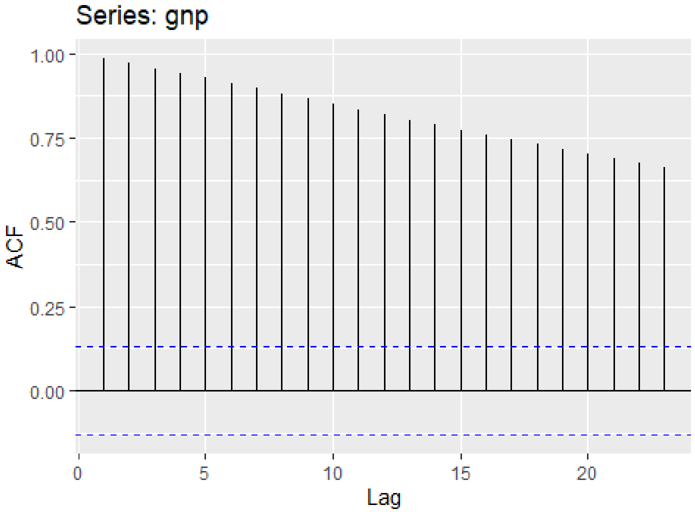


```
#GNP data
gnp96 <- read.table("D:/GoogleDrive/Julie_SYS4021/2020/Lectures/TS/gnp96.dat")

ggplot(gnp96, aes(x=V1,y=V2)) + geom_line() +
  ylab("U.S. Gross National Product") + xlab("")
```





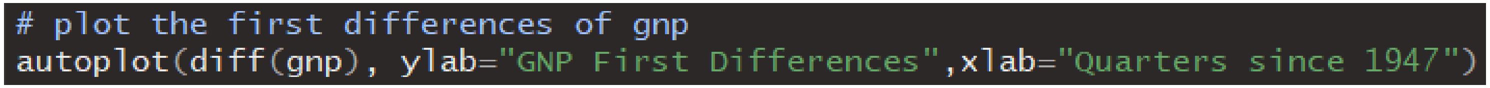
```
gnp <- ts(gnp96[,2])
ggAcf(gnp)
```



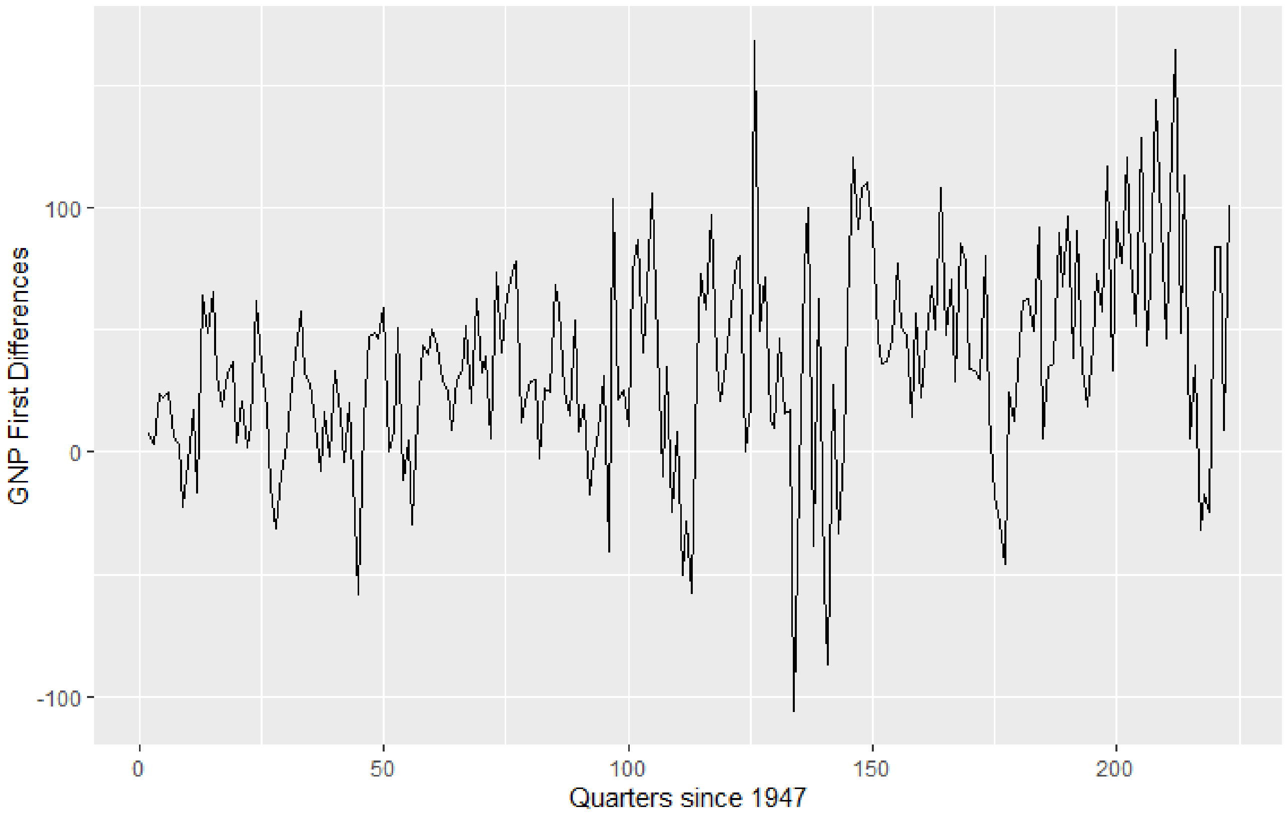
Example ARIMA time series: GNP

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



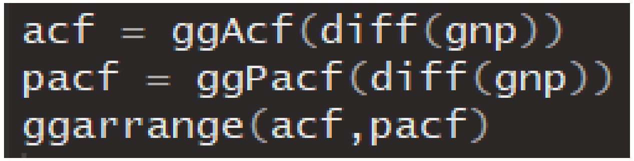
```
# plot the first differences of gnp
autoplot(diff(gnp), ylab="GNP First Differences",xlab="Quarters since 1947")
```



Example ARIMA time series: GNP

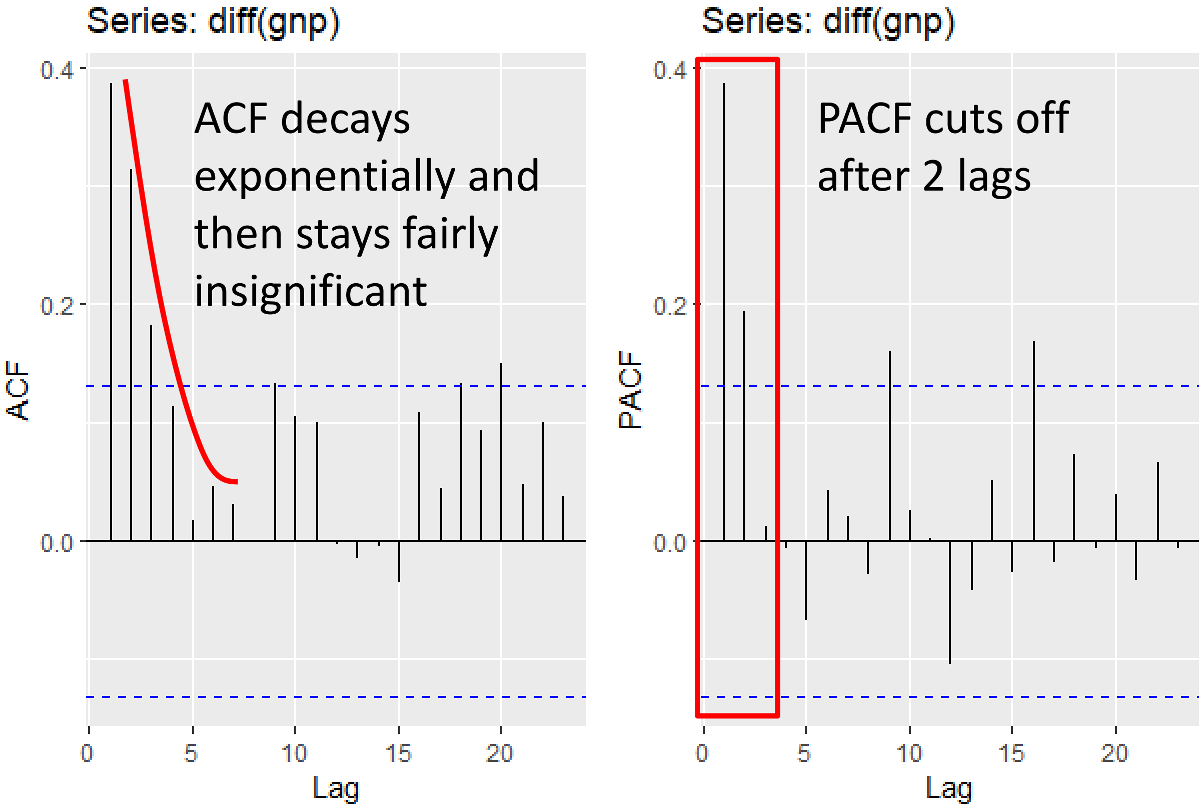
Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
acf = ggAcf(diff(gnp))
pacf = ggPacf(diff(gnp))
ggarrange(acf,pacf)
```

AR(2) model of first differences seems appropriate, so ARIMA(2,1,0) of GNP



ACF decays exponentially and then stays fairly insignificant

PACF cuts off after 2 lags

Diagnostics in Time Series Analysis

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

We now know several models we can capture autocorrelation in a time series, or in the residuals of a trend + seasonality model of a time series.

But how do we know if these models are adequate?

We want a model that is parsimonious as possible: we need enough parameters to model correlated observations, but we don’t want to overfit the data and increase the variance of our forecasts.

As with MLR, we will assess model performance with both metrics like AIC, BIC, Adjusted R^2 and diagnostics.

Diagnostics in Time Series Analysis

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Like MLR, we want our residuals w_t to be normal with 0 mean and constant variance.

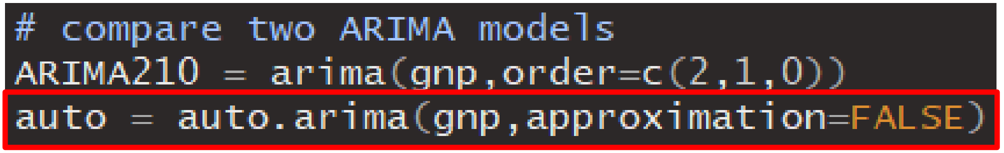
More importantly, we want them to be independent.

We can still use the residuals vs. fitted to check for lack of fit and heteroscedasticity, and the qqplot to check for normality.

GNP ARIMA model diagnostics

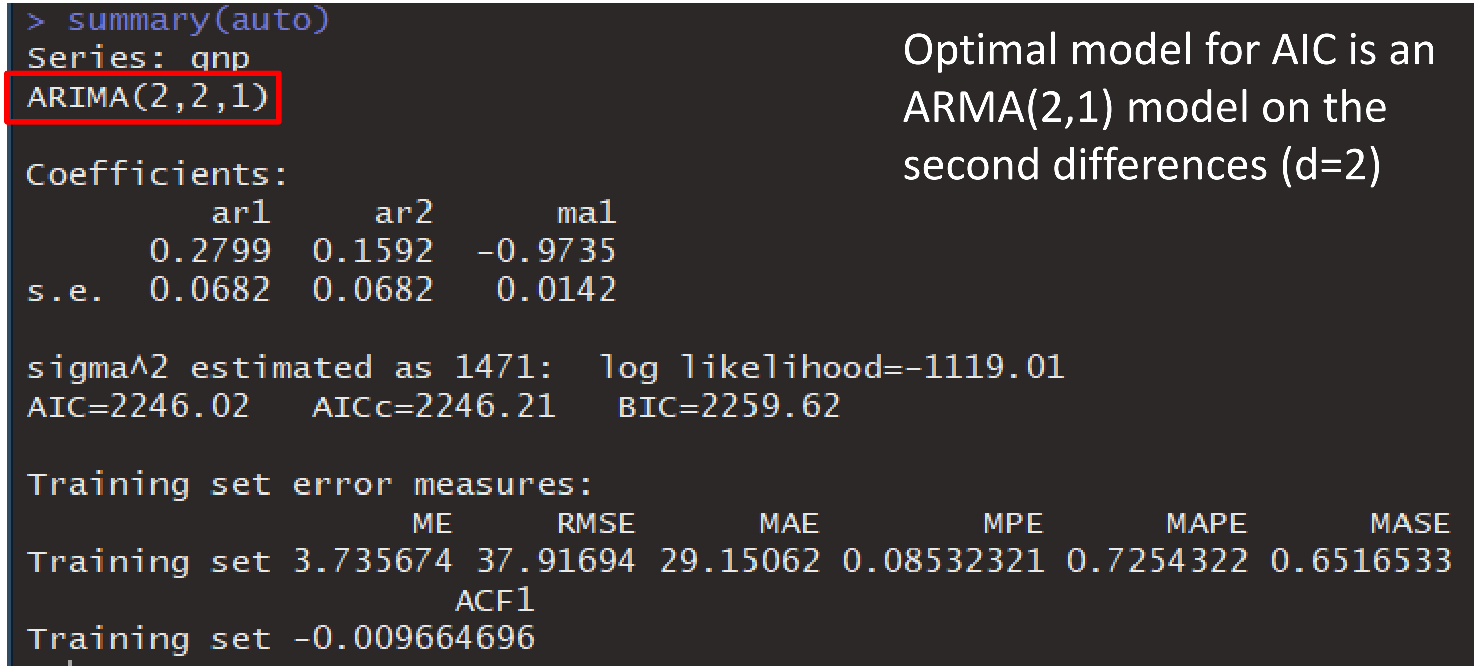
Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
# compare two ARIMA models
ARIMA210 = arima(gnp,order=c(2,1,0))
auto = auto.arima(gnp,approximation=FALSE)
```

Finds ARIMA model that minimizes the AIC



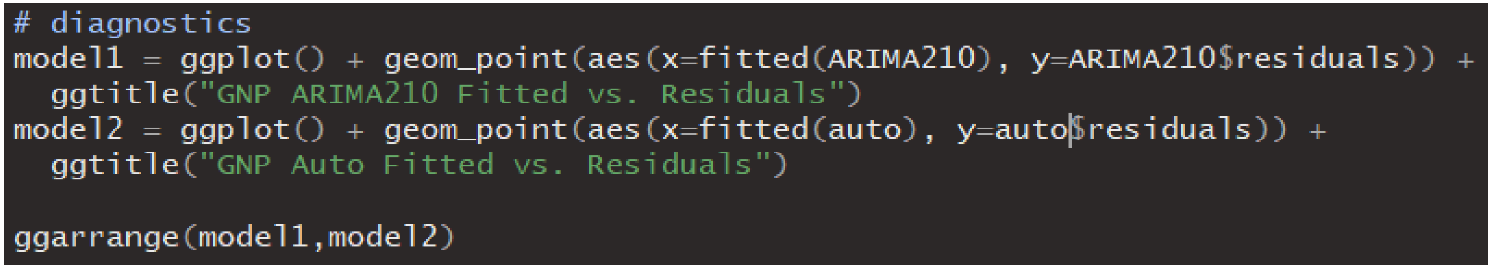
```
> summary(auto)
Series: gnp
ARIMA(2,2,1)

Coefficients:
         ar1     ar2      ma1
      0.2799  0.1592  -0.9735
s.e.  0.0682  0.0682   0.0142

sigma^2 estimated as 1471:  log likelihood=-1119.01
AIC=2246.02   AICc=2246.21   BIC=2259.62

Training set error measures:
                   ME     RMSE      MAE        MPE      MAPE      MASE
Training set 3.735674 37.91694 29.15062 0.08532321 0.7254322 0.6516533
                     ACF1
Training set -0.009664696
```

Optimal model for AIC is an ARMA(2,1) model on the second differences (d=2)

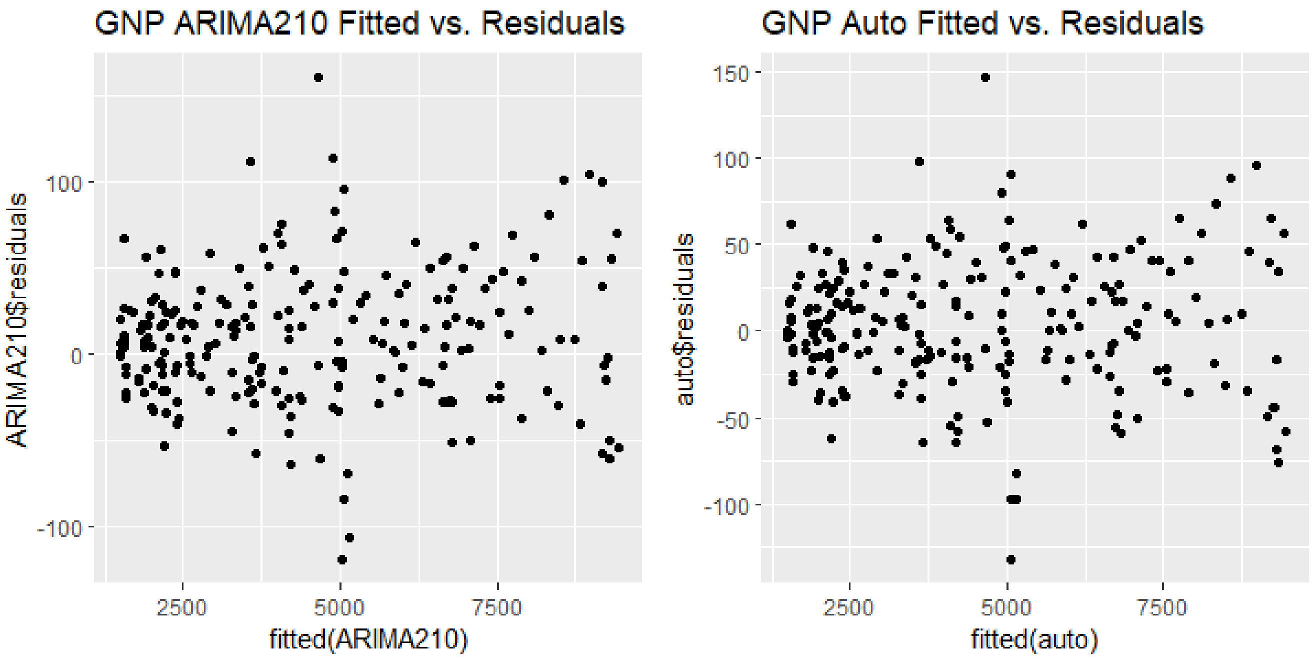
GNP ARIMA model diagnostics

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
# diagnostics
model1 = ggplot() + geom_point(aes(x=fitted(ARIMA210), y=ARIMA210$residuals)) +
  ggtitle("GNP ARIMA210 Fitted vs. Residuals")
model2 = ggplot() + geom_point(aes(x=fitted(auto), y=auto$residuals)) +
  ggtitle("GNP Auto Fitted vs. Residuals")

ggarrange(model1,model2)
```

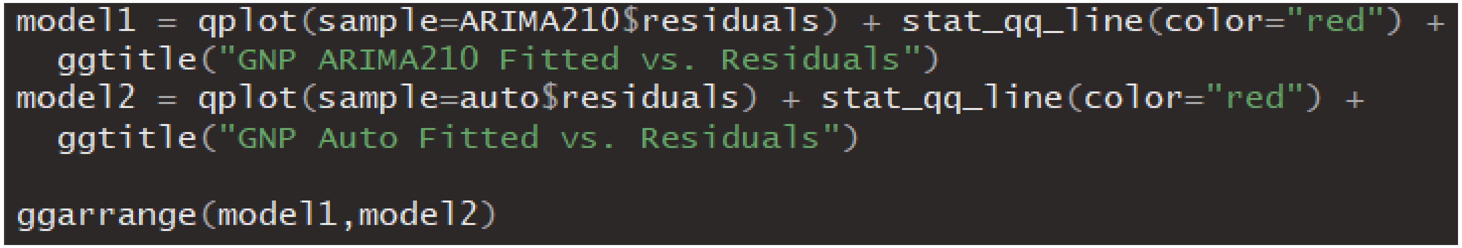


Residuals vs. fitted look similar for both models

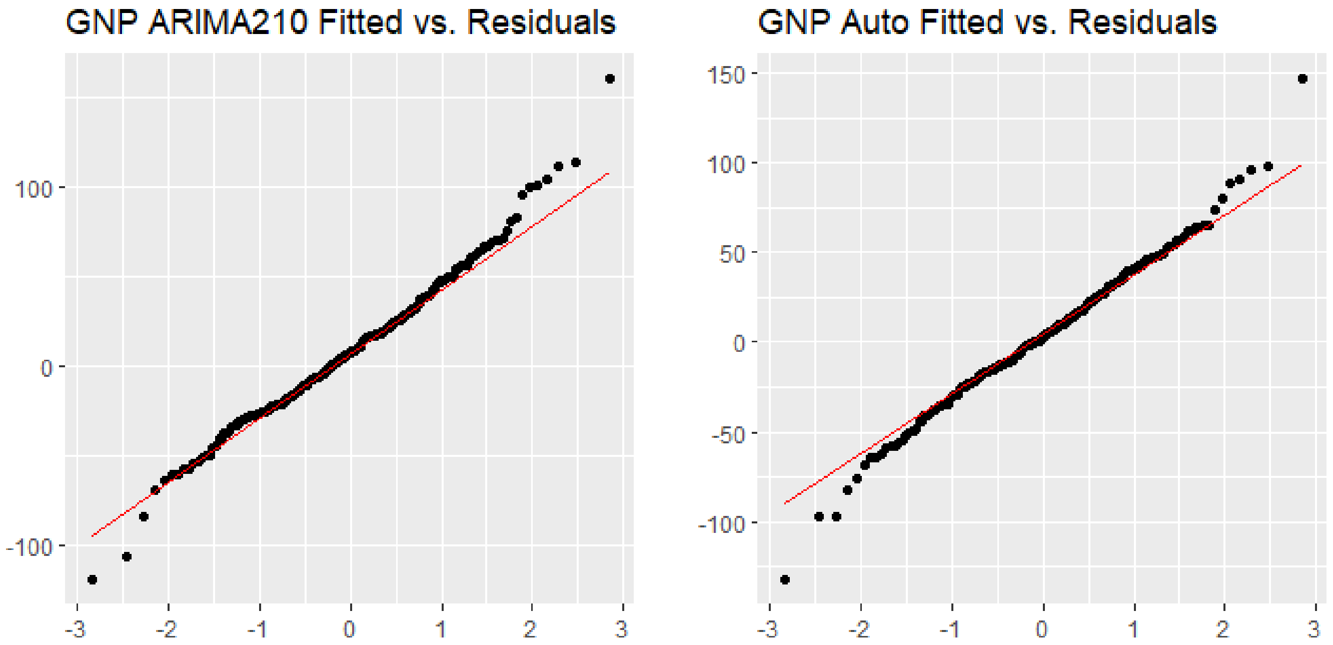
GNP ARIMA model diagnostics

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
model1 = qplot(sample=ARIMA210$residuals) + stat_qq_line(color="red") +
  ggtitle("GNP ARIMA210 Fitted vs. Residuals")
model2 = qplot(sample=auto$residuals) + stat_qq_line(color="red") +
  ggtitle("GNP Auto Fitted vs. Residuals")

ggarrange(model1,model2)
```



Residuals are about equally Gaussian for both models

Diagnostics in Time Series Analysis

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Like MLR, we want our residuals w_t to be normal with 0 mean and constant variance.

More importantly, we want them to be independent.

We can still use the residuals vs. fitted to check for lack of fit and heteroscedasticity, and the qqplot to check for normality.

We can use the ACF and PACF to check for independence, as well as the Ljung-Box test.

Ljung Box Test

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

The Ljung-Box test determines whether the first *H* sample autocorrelations of the residuals, considered together, are significant.

H0: Autocorrelation is not significant
(model is adequate up to lag H)

Ha: Autocorrelation is significant
(model is not adequate up to lag H)

Ljung-Box Test

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

The test statistic of the Ljung-Box test is the Q-statistic:

$$Q = n'(n'+2)\sum_{h=1}^{H}\frac{\hat{\rho}_w^2(h)}{n'-h}$$

where

n' = *n* – *d*, *n* = # of observations in original time series, *d* = degree of differencing used to transform original time series into stationary time series

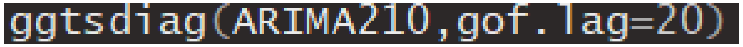
$\hat{\rho}_w^2(h)$ is the squared sample autocorrelation of the residuals, *w*, at lag *h*

Under H0, $Q \sim \chi^2_{H-p-q}$ as $n \rightarrow \infty$.

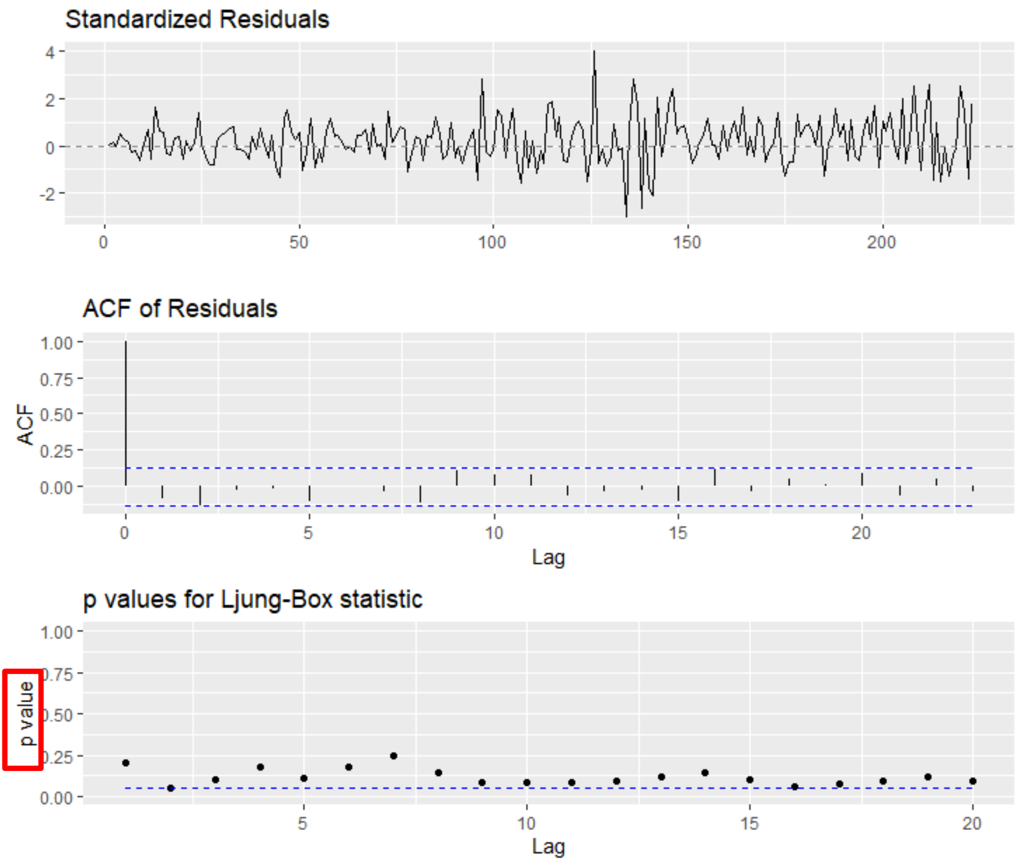
GNP ARIMA model diagnostics

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
ggtsdiag(ARIMA210,gof.lag=20)
```



If the p-value is <0.05, i.e. below the blue line, we reject H0 that the correlations are not significant.

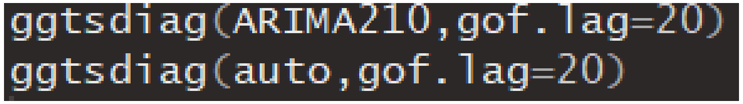
That is, we reject that the model is adequate at that lag. So the higher the dots, the better.

This model becomes inadequate after only 1 lag.

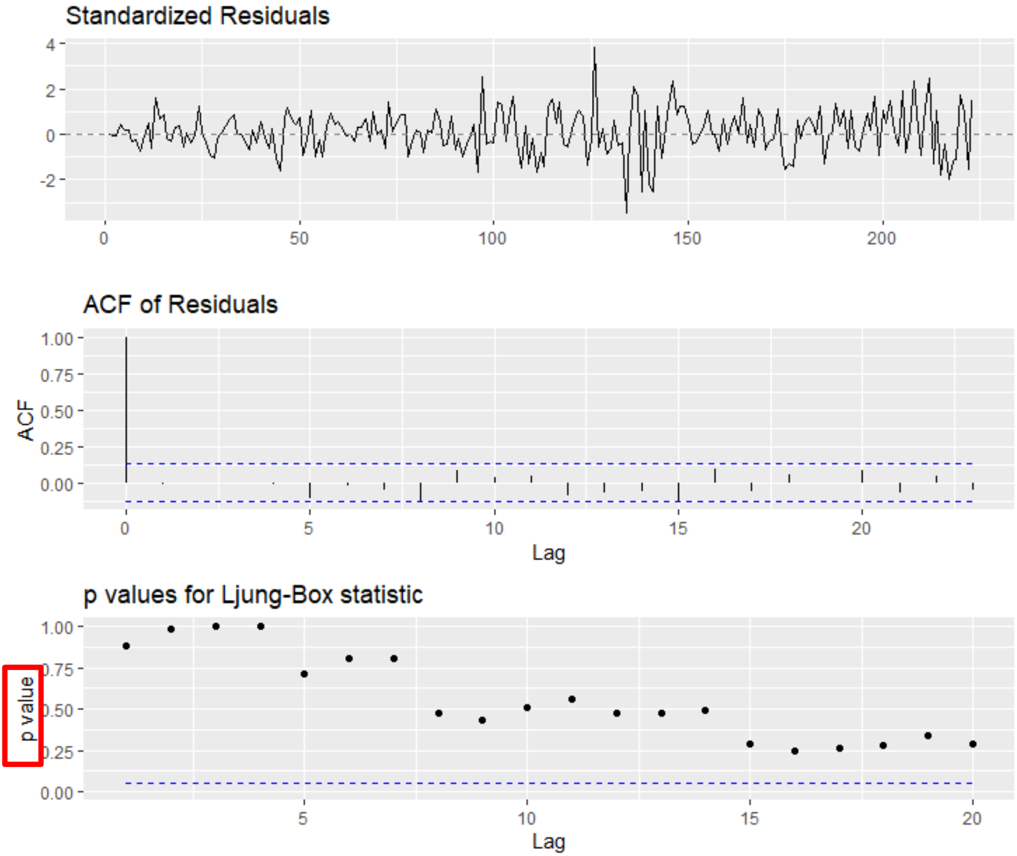
GNP ARIMA model diagnostics

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
ggtsdiag(ARIMA210,gof.lag=20)
ggtsdiag(auto,gof.lag=20)
```



The auto.arima model, which was ARIMA(2,2,1), is good for at least 20 lags.

Therefore this model outperforms the ARIMA(2,1,0) model in both AIC and diagnostics.

Forecasting

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Another important criterion for performance is how well the model does out of sample, specifically in forecasting future values of the time series.

The standard criterion is to find which model has a lower MSE in forecasting future observations: $E[(x_{n+m} - \hat{x}_{n+m})^2]$

where

- n is the number of observations the model was built to
- m is the number of forecast observations
- $\hat{x}_{n+m} = E[x_{n+m} | x_n, x_{n-1}, \ldots]$.

If the time series is AR(p), then for $n \geq p$, the one-step-ahead prediction is $\hat{x}_{n+1} = \phi_1 x_n + \phi_2 x_{n-1} + \ldots + \phi_p x_{n-p+1}$.

For ARMA models it is more complicated, because it also depends on the error you made in modeling/ forecasting previous time steps, w_{t-k}.

Forecasting

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

We will often be interested in probabilistic forecasts.

We can report a 100(1-α)% prediction interval around our point forecast as $(\hat{x}_{n+m} \pm t_{n-n_p}^{1-\alpha/2}\sigma_{n+m})$ where σ_{n+m} is the standard error of the forecast and n_p is the number of parameters in our time series model.

The variance σ_{n+m}^2 of our forecast error gets bigger and then levels off with increasing forecast lead time, m.

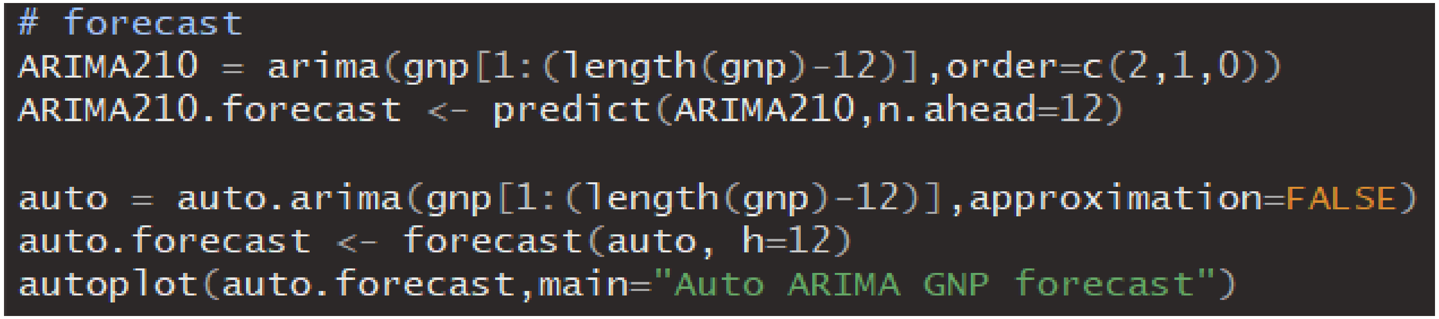
ARMA forecasts quickly settle to the mean with a constant prediction error as the forecast lead time grows.

Example Forecast with GNP data

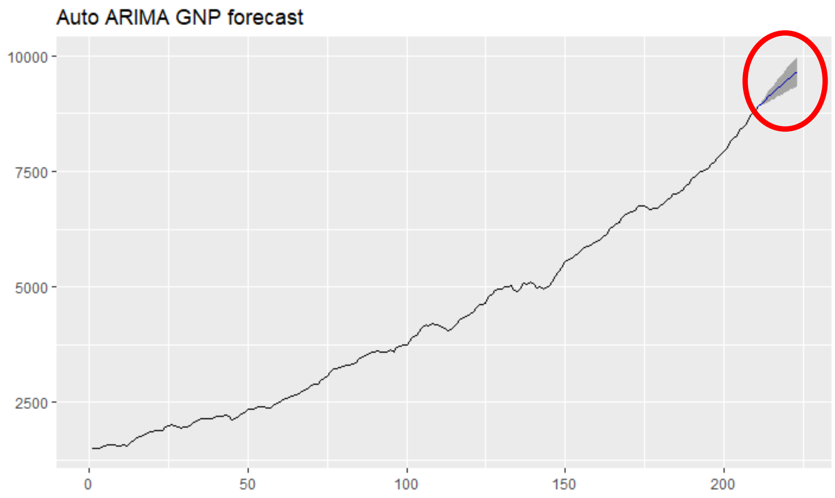
Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Build a forecast to all but the last 3 years of quarterly GNP and then forecast the remainder.



```
# forecast
ARIMA210 = arima(gnp[1:(length(gnp)-12)],order=c(2,1,0))
ARIMA210.forecast <- predict(ARIMA210,n.ahead=12)

auto = auto.arima(gnp[1:(length(gnp)-12)],approximation=FALSE)
auto.forecast <- forecast(auto, h=12)
autoplot(auto.forecast,main="Auto ARIMA GNP forecast")
```



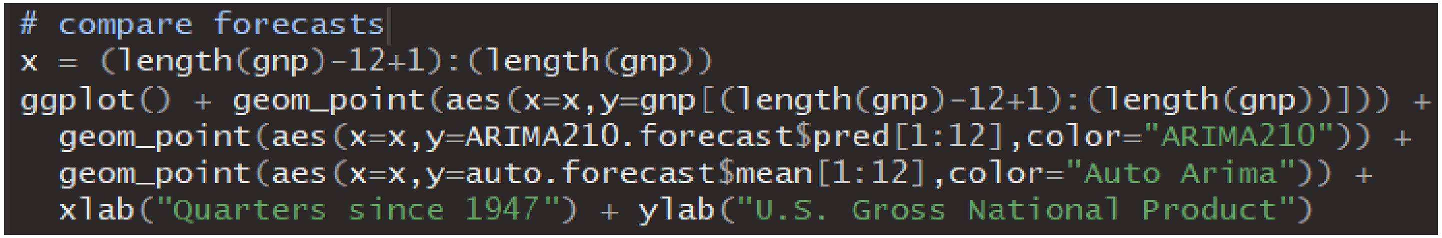
Forecast and 95% confidence interval

Example Forecast with GNP data

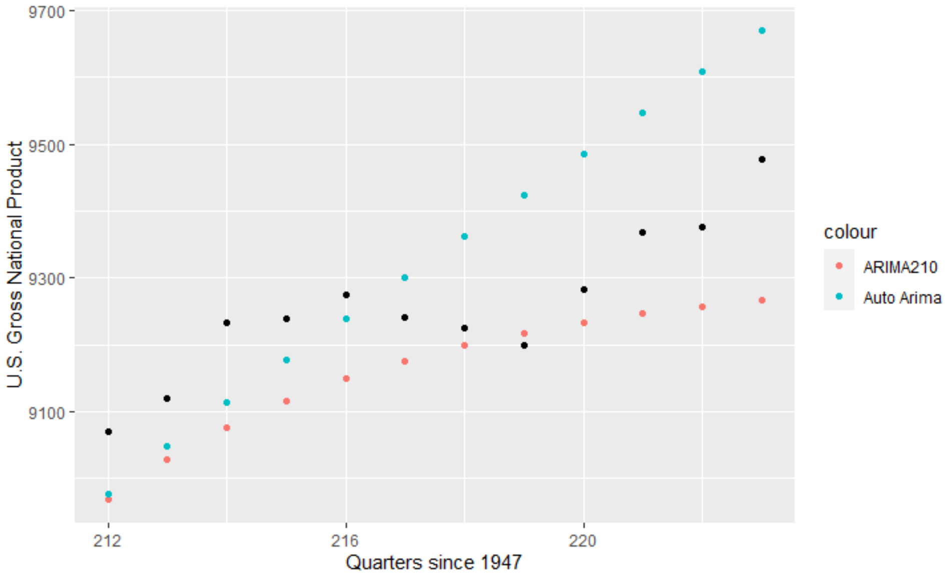
Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Build a forecast to all but the last 3 years of quarterly GNP and then forecast the remainder.



```
# compare forecasts
x = (length(gnp)-12+1):(length(gnp))
ggplot() + geom_point(aes(x=x,y=gnp[(length(gnp)-12+1):(length(gnp))])) +
  geom_point(aes(x=x,y=ARIMA210.forecast$pred[1:12],color="ARIMA210")) +
  geom_point(aes(x=x,y=auto.forecast$mean[1:12],color="Auto Arima")) +
  xlab("Quarters since 1947") + ylab("U.S. Gross National Product")
```

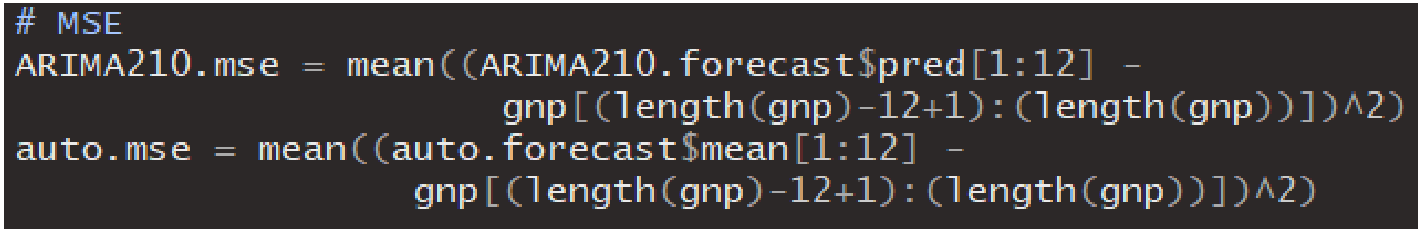


Example Forecast with GNP data

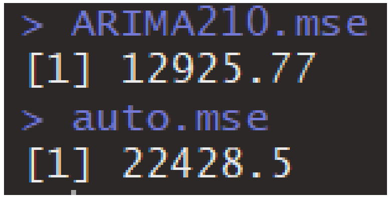
Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

Build a forecast to all but the last 3 years of quarterly GNP and then forecast the remainder.



```
# MSE
ARIMA210.mse = mean((ARIMA210.forecast$pred[1:12] -
                      gnp[(length(gnp)-12+1):(length(gnp))])^2)
auto.mse = mean((auto.forecast$mean[1:12] -
                  gnp[(length(gnp)-12+1):(length(gnp))])^2)
```



```
> ARIMA210.mse
[1] 12925.77
> auto.mse
[1] 22428.5
```

While the auto.arima model (ARIMA(2,2,1)) had better diagnostics and a lower AIC, the ARIMA(2,1,0) model had a lower forecast MSE

Time Series Simulation

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

After deciding on a model, you can not only use it for forecasting but for simulation.

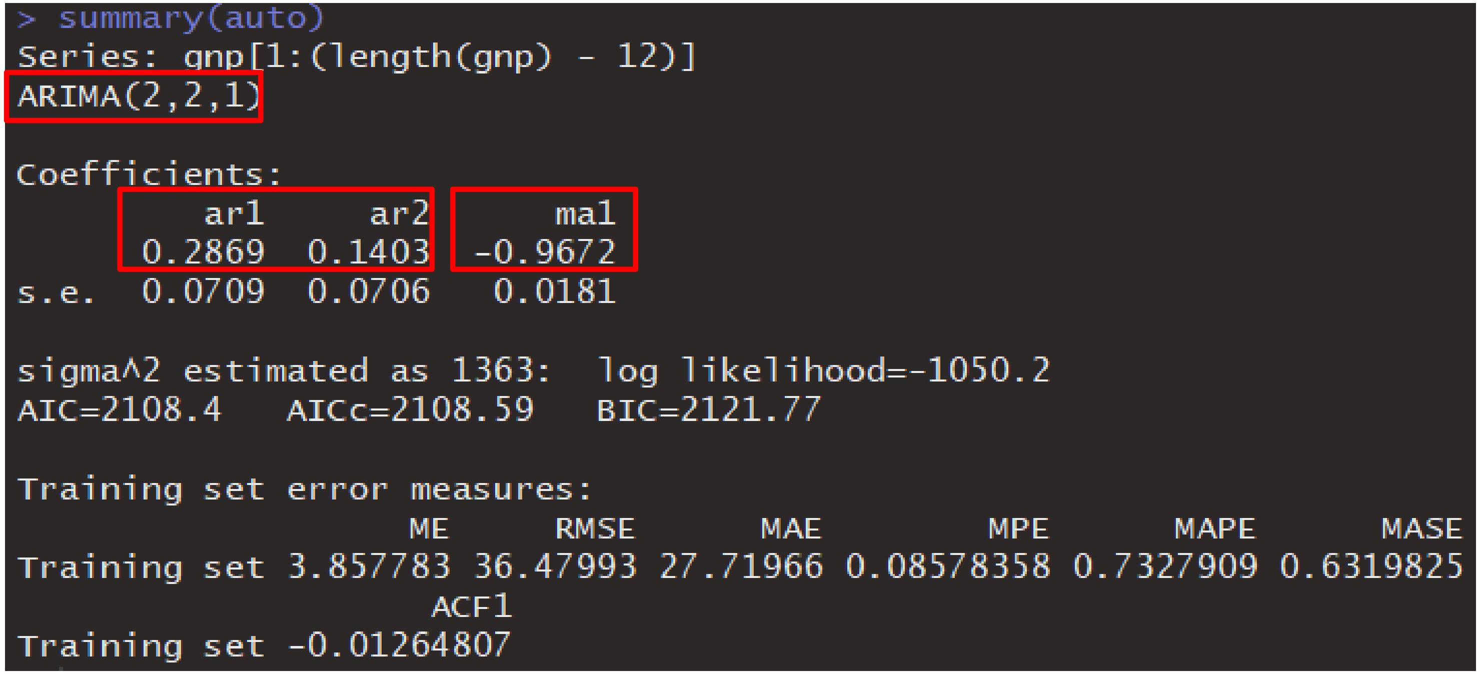
This is useful for risk analysis, e.g. if you want to estimate the probability of seeing a value above some threshold. Simulated data can also be used as input to a decision-making model to capture events outside of what has occurred historically.

Simulations simply generate random normal Gaussian noise for the w_t term of the regression model and propagate it through to estimate Y_t.

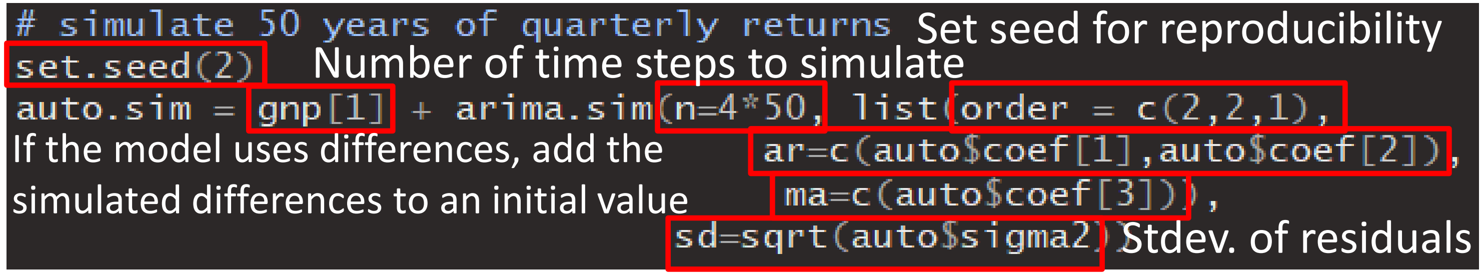
Example Simulation

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
> summary(auto)
Series: gnp[1:(length(gnp) - 12)]
ARIMA(2,2,1)

Coefficients:
         ar1     ar2      ma1
      0.2869  0.1403  -0.9672
s.e.  0.0709  0.0706   0.0181

sigma^2 estimated as 1363:  log likelihood=-1050.2
AIC=2108.4   AICc=2108.59   BIC=2121.77

Training set error measures:
                   ME     RMSE      MAE        MPE      MAPE      MASE
Training set 3.857783 36.47993 27.71966 0.08578358 0.7327909 0.6319825
                    ACF1
Training set -0.01264807
```



```
# simulate 50 years of quarterly returns
set.seed(2)
auto.sim = gnp[1] + arima.sim(n=4*50, list(order = c(2,2,1),
                              ar=c(auto$coef[1],auto$coef[2]),
                               ma=c(auto$coef[3])),
                           sd=sqrt(auto$sigma2))
```

Set seed for reproducibility

Number of time steps to simulate

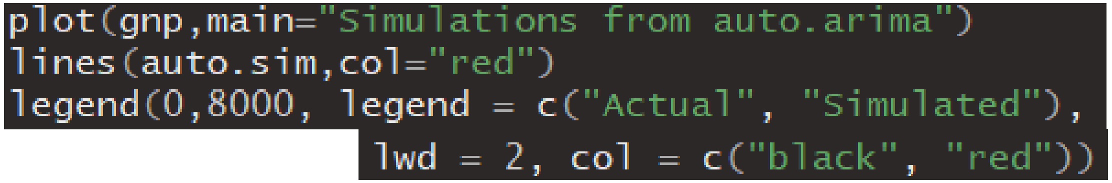
If the model uses differences, add the simulated differences to an initial value

Stdev. of residuals

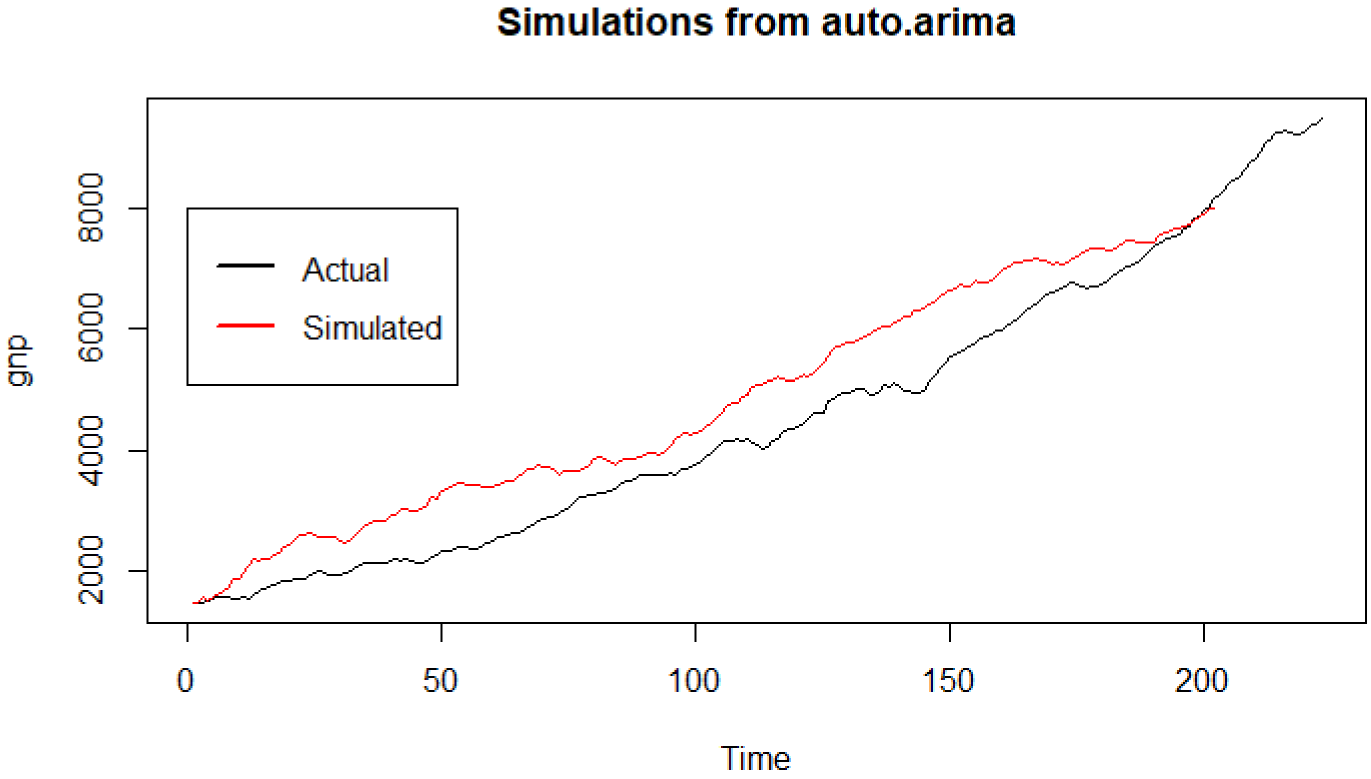
Example Simulation

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process



```
plot(gnp,main="Simulations from auto.arima")
lines(auto.sim,col="red")
legend(0,8000, legend = c("Actual", "Simulated"),
                  lwd = 2, col = c("black", "red"))
```



Time Series Simulation

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

When generating synthetic data, you should check that it reproduces the statistics of the raw data, such as:

- Mean
- Variance
- Seasonality (compare periodogram)
- Trend (compare coefficient on time in regression)
- Autocorrelation (compare ACF and PACF)

Summary: Box Jenkins Process for Time Series Analysis

Agenda

- Review of Time Series and ARMA models
- Backshift Operator
- ARIMA models
- Time Series Diagnostics
  - Ljung-Box Test
- Forecasting
- Simulation
- Summary: Box-Jenkins Process

1. Tentative Identification: Historical time series data are used to identify possible ARIMA models.

2. Estimation: Estimate the parameters of the tentatively identified models.

3. Diagnostic checking: Check the adequacy of the tentatively identified models.

4. Forecasting & Simulation: Once a final model is selected, it is used to forecast/ simulate future time series values.