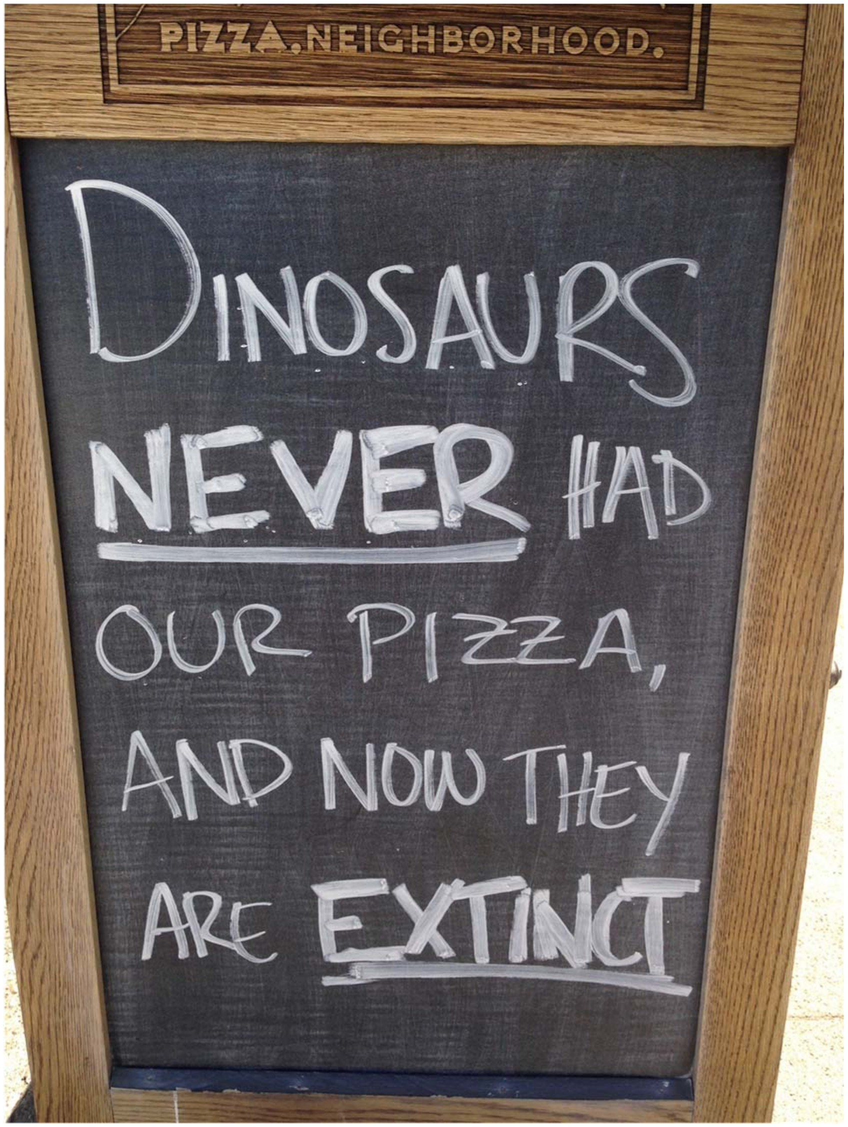
UNIVERSITY of WASHINGTON

Data Science UW
Methods for Data Analysis

Logistic Regression and Time Series
Lecture 7
Steve Elston

A dubious hypothesis?



Topics

- > Review
- > Other regularization methods
- > Overview of logistic regression
- > Time series

Review

- Bootstrapping regression models
- Linear Algebra overview
- Decomposition methods
- SVD
  - SVD as linear regression
  - Variable reduction
  - Storing data

More on Regularization

Regularization is widely applied in machine learning

> SVD regularization is a bit awkward.
> Are there other approaches?
> Constrain the coefficients to be close to zero
  – Provides stable, but biased, solutions

Ridge Regression

- Ridge regression is a way to limit the amount of independent variables in the regression.
- Our regular least squares criterion minimizes the least squares of the error plus a regularization term that is a product of a constant and the sum of squared coefficients :

$$\min \sum (y - y_i)^2 + \alpha \sum \beta^2$$

- Essentially this is preventing the partial slope terms from getting too large.

Lasso Regression

- Lasso regression is another way to limit the amount of independent variables in the regression.
- Our regular least squares criterion minimizes the least squares of the error:

$$\min \sum (y - y_i)^2$$

- Lasso regression minimizes the same with the addition of a ‘regularization’ term:

$$\min \sum \left(y - y_j\right)^2 \qquad \text{Such that} \qquad \sum |\beta_i| < \lambda$$

- Here, y is the predicted for j points. There are i terms with beta coefficients. Lambda is a fixed value that limits the betas.
- Sometimes called **Elasticnet**

Regularization Summary

Regularization is necessary

- Most real-world machine learning problems are under-determined or over-paramterized
- All solutions involve adding bias and using an approximation
  - Bias variance trade-off in training models
- Some common approaches
  - Feature selection
  - SVD/PCA
  - Ridge and Lasso methods - elasticnet

Logistic Regression

- The purpose of logistic regression is to use linear regression to predict a limited dependent variable.
- Usually our dependent variable has 2 outcomes (1 or 0) or occurrence.
- Examples:
  - Bank gives a yes (1) or no (0) outcome to loan applications.
  - Success/Failures of clinical trials.
  - Mortality outcomes.
  - Marketing outcomes (will a user click on an add).
- Logistic predictions will result in a probability of success.

Why focus on logistic regression?

Logistic regression widely used

- An early classifier method - Joseph Berkson 1944
- Logistic function has analogs in most classifers
- Computationally efficient

Metrics for Classification

Confusion matrix

	Predicted Positive	Predicted Negative
Actual Positive	**TP**	**FN**
Actual Negative	**FP**	**TN**

Metrics for Classification

- Accuracy = TP + TN / (TP + TN + FP + FN)
- Precision or positive predictive value = TP/(TP + FP)
- Recall = TP/(TP + FN)
- + Many others!

Logistic Regression

- Logistic regression is also called the 'logit' model:
- Original model:

$$y_i = \beta_0 + \beta_1 x_1 + \varepsilon_0$$

- Logit model:

$$\ln\left[\frac{p_i}{1-p_i}\right] = \beta_0 + \beta_1 x_1 + \varepsilon_0$$

Log-odds-ratio

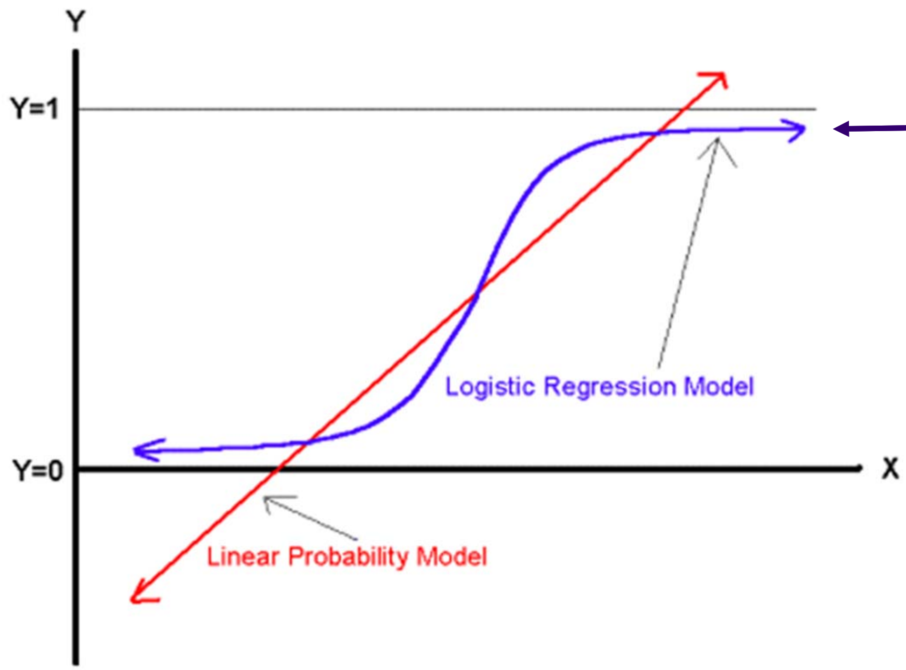
- So estimated probabilities follow: (solving for p)

$$p_i = \frac{1}{1+\exp(-(\beta_0 + \beta_1 x_1))}$$

Logistic Regression

$$p_i = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1))}$$

- > As $(\beta_0 + \beta_1 x_1)$ gets really big, p approaches 1.
- > As $(\beta_0 + \beta_1 x_1)$ gets really small, p approaches 0.



$$\frac{1}{1 + \exp(-x)}$$

Logistic Regression

- Differences between linear and logistic regression.
- Predictions
  - Linear regression outcomes are unbounded.
  - Logistic regression outcomes are bounded between 0 and 1.

$$p_i = \frac{1}{1 + \exp(-(\beta_0 + \beta_1 x_1))}$$

- Error distribution
  - Linear regression errors are normally distributed.
  - Logistic regression errors are Bernoulli distributed.
- R demo

Logistic Regression Summary

> Logistic function is analog for most classifiers
> Computationally efficient
> Accuracy depends on separation of classes
> Error trade-off by changing decision probability

Time Series Analysis

Time series data are everywhere!

- Demand forecasting – Electricity production, Internet bandwidth, Traffic management
- Medicine – Time dependent treatment effect, EKG, EEG
- Engineering and Science – Signal analysis, Analysis of physical measurements
- Capital markets and economics – Seasonal unemployment, Price/return series, Risk analysis
- And many others!

Time Series Modeling

- Representations
  - Continuous Functions: Solutions to ODEs, PDEs…
- Random processes
  - Random variables that depend on the previous observation.
- Sum of Periodic functions
  - Daily Trend + Weekly Trend + Seasonal Trend + …
- Time Series Analysis Objectives
  - Estimate True values in the presence of noise or trend
  - Forecast future values

Time Series Modeling

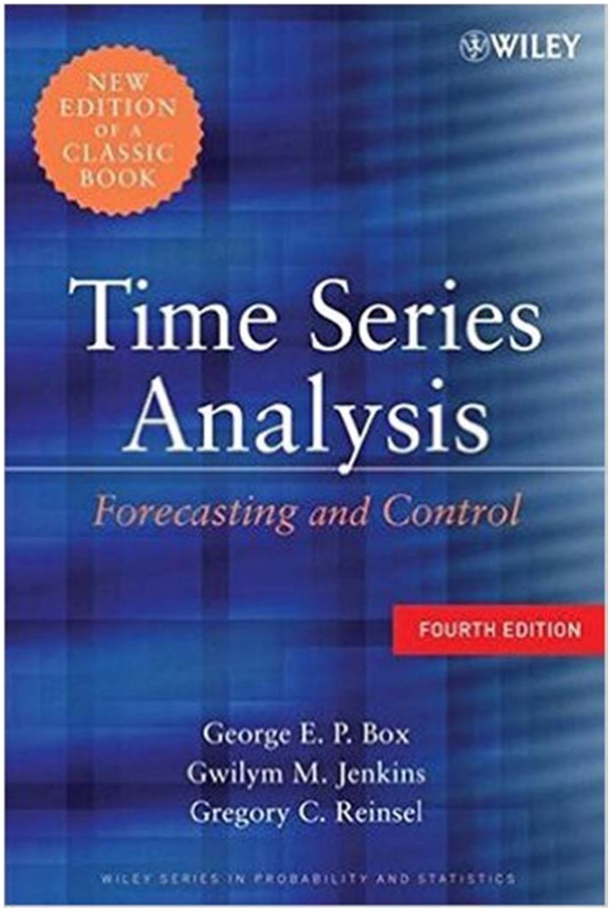
- Time series measurements are represented by observations over time:

$$Y = (Y_1, Y_2, Y_3, \ldots, Y_T)$$

- Stochastic Process is a process that evolves over time.
- Regular statistical analysis is concerned with estimations of repeated samples.
- With time series, we usually cannot measure repeatedly and have to observe over time how something changes.
  - E.g.: Mortality Rate, Temperature, …
- A stochastic process is ‘stationary’ if there is **no trend** in the data and **constant variance**.
  - This is a nice assumption because it implies the correlation of a process is fixed over time. I.e. any two points should have the same relationship.

Box-Jenkins Models for Time Series

Classic book: Time Series Analysis, Forecasting and Control, First Ed, Wiley, 1970



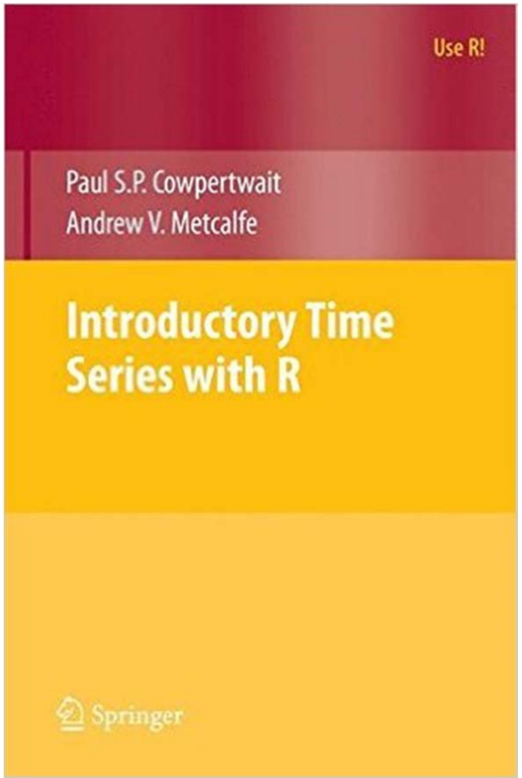
Time series in R

Multiple time series classes available

- > We will only use base time series class; ts
- > Time series object contains one of more ordered values
- > Time attributes
  - – Represents units of time
- > Can perform arithmetic on time attributes

Time Series Resources

- R forecast package https://cran.r-project.org/web/packages/forecast/index.html
- Cowpertwait and Metcalfe, Introductory Time Series with R, Springer, 2009



Basic Time Series Statistics

- Variance

$$\sigma^2 = E[(y_t - \mu)^2]$$

- Autocovariance at lag k

$$\gamma_k = E[(y_t - \mu)(y_{t+k} - \mu)]$$

- Autocorrelation at lag k

$$\rho_k = \gamma_k / \sigma^2$$

- Partial autocoorelation at lag *k* is the correlation that results from removing the effect of any correlations due to the terms at smaller lags
- Autocorrelogram plots ρ_k vs. k

White noise

> White noise is random and independent

$$y_t = w_t = N(\upsilon,\sigma)$$

No time dependency in series, so:

$$p_0 = 1$$

$$p_k = 0 \quad k \mathrel{!}= 0$$

$$\gamma_k = Cov(x_t, x_{t+k}) = 0$$

> White noise is stationary; no change in variance with time

Random walk

- Random walk is sum of white noise

$$y_t = y_{t-1} + w_t$$

Or,

$$w_t = y_t - y_{t-1}$$ are the innovations

And,

$$\gamma_k = \mathrm{Cov}(x_t, x_{t+k}) = t\sigma^2$$ is **unbounded!**

- Random walk is not stationary; variance dependent on time
- R Demo

Time Series with trend

Trend is a systematic change in the value of y_t

- A time series with trend is not stationary
- Mean depends on time
- Variance depends on time

Difference Series

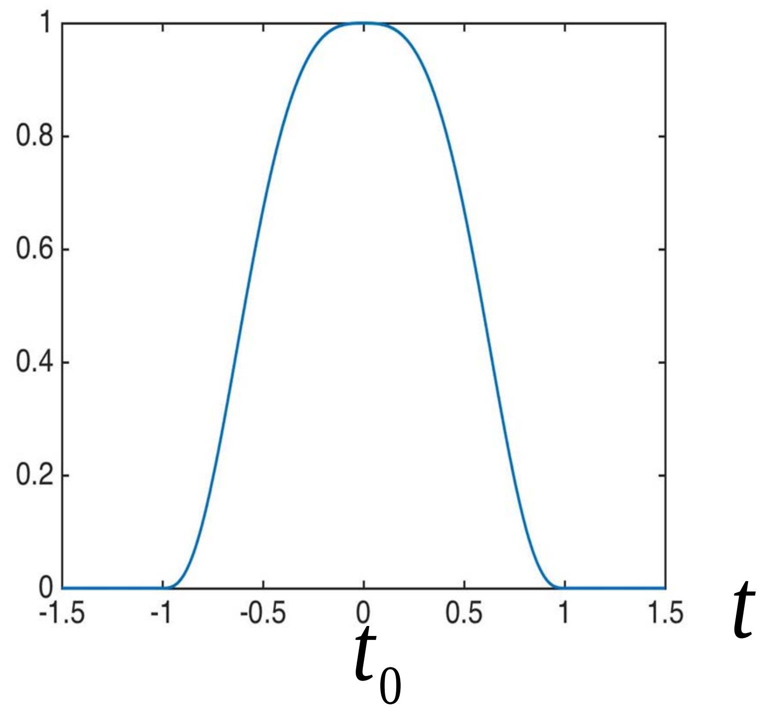
- Taking lag p difference of time series can remove trend
- Result is order p difference series
- Can transform non-stationary series to stationary

loess Nonlinear Regression

loess = local scatterplot smoother

- loess uses a smooth kernel function to find weighted average within window
- Adjusting the window of span gives difference results

$$\left(1-\text{distance}(t,t_0)^3\right)^3$$



Seasonal Time Series

- Many time series have seasonal components
- Seasonal component is periodic
- Examples; Unemployment rate, Number of people travel by air, Agricultural Production, Ice cream consumption…

Decomposition of Time Series

Decompose time series into components, trend, seasonal, and remainder

- Model can be additive or multiplicative
  - Log transform for multiplicative
- Several possible models for trend
  - Simple windowed Moving Average
  - Nonlinear regression model; lowess
- Seasonal component
  - Can use simple linear model
  - More sophisticated moving window models
  - Multiplicative model if seasonal component increases with trend

Autoregressive Model (AR)

- If a series is stationary (no trend) and auto-correlated, it should be able to be predicted as some weighted sum of previous values.
- Every new observed point relies on what the previous p-points were:

$$y_t = c + \sum_{i=1}^{p} (\varphi_i y_{t-i}) + \varepsilon_t$$

- Coefficients ϕ_i determine the time dependency
- The above is shown as AR(p), this means it has "order p"
- R Demo

Correlogram of AR(p) Process

Measure of time dependence

- (Auto) Correlate time series with lagged version of itself

  $\rho_k = \phi^k$

  $\rho_0 = 1$ always
- Stationary process autocorrelation has small order p.
- Number of non-zero partial autocorrelations is the order of the AR process

AR Models

> ARIMA(1,0,0) = 1st order auto regressive

$$y_t = c + \varphi_1 y_{t-1} + \varepsilon_t$$

> ARIMA(2,0,0) = 2nd order auto regressive

$$y_t = c + \varphi_1 y_{t-1} + \varphi_2 y_{t-2} + \varepsilon_t$$

*Note we assume that the sum of all the coefficients in the model < 1, otherwise the series is not stationary.

Moving Average Model

MA processes averages the noise or error terms

> An MA process of order p is represented as, MA(q):

$$y_t = c + \sum_{i=1}^{q} (\theta_i \varepsilon_{t-i}) + \varepsilon_t$$

Where ε_t is the error,

If ε_t is white noise we have MA(0) process

> Number of non-zero autocorrelations is the order of the MA process

MA Models

> ARIMA(0,0,1) = 1st order Moving Average

$$y_t = c + \theta_1 \varepsilon_{t-1} + \varepsilon_t$$

> ARIMA(0,0,2) = 2nd order Moving Average

$$y_t = c + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \varepsilon_t$$

Auto Regressive Moving Average (ARMA)

- > Auto-Regressive Moving Average (ARMA)
- > ARMA is denoted by two variables (p,q)
  - – p = Auto regression order
  - – q = Order of moving average

$$y_t = c + \sum_{i=1}^{p}(\varphi_i y_{t-i}) + \sum_{i=1}^{q}(\theta_i \varepsilon_{t-i}) + \varepsilon_t$$

AR (p,q) = AR(p) filter + MA(q) filter + error terms

- > R-demo

ARIMA

- Auto-Regressive Integrated Moving Average (ARIMA)
- ARIMA(p,d,q) model has three parameters:
- p = Order of autoregressive process
- d = Degree of difference operator for the ‘integrated’ part, this is how the model takes into account the differences needed for finding trend.
- q = Order of Moving Average Process

- *Note ARIMA(0,0,0)=> $y_t = \varepsilon_t$(random noise)

Integrated Models (Random Walks)

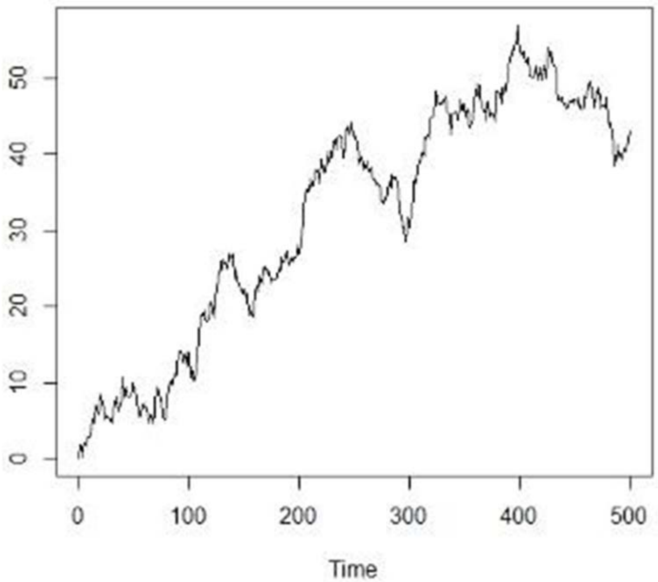
> ARIMA(0,1,0) = Random Walk Model

$$y_t = y_{t-1} + \varepsilon_t \quad \text{OR} \quad \Delta y_t = \varepsilon_t$$

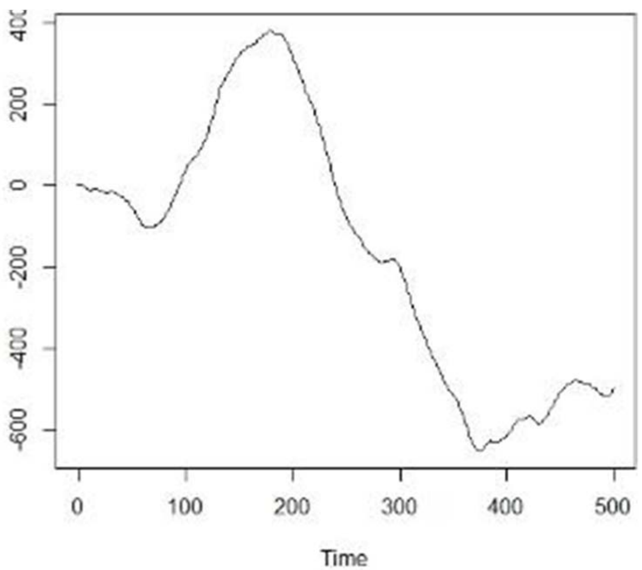
> ARIMA(0,2,0) = 2nd order random walk

$$y_t = (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) + \varepsilon_t$$

1st order



2nd order



ARIMA + Seasonal

- Add in seasonal (cyclic) factors
- If Arima models have three factors, pdq, then seasonal Arima models have 6: the same pdq, and seasonal PDQ

$$Arima(p,d,q)X(P,D,Q)$$

- p = Autoregressive order (non – seasonal)
- d = Integrative part (non – seasonal)
- q = Moving Average order (non – seasonal)
- Seasonal (cyclic) parameters (lagged by a time difference)
- P = Autoregressive order (seasonal)
- D = Integrative part (seasonal)
- Q = Moving Average order (non – seasonal)

Forecasting

Forecasting is the whole point!

> Use time series model to predict the next values

e.g. $y_{t+1}, y_{t+2}, \ldots, y_{t+n}$

> R forecasting package makes life simple.

> R Demo

Time Series Using Linear Regression Models

- Approximate time series using linear models if we are careful.
- Insert factors into linear model that account for time.
  - E.g. Number of days/weeks/months/years since start.
- If neighboring points are related add in our auto-regressive terms:
  - Add a ‘time before’ and/or ‘2 times before’ values, etc…
- Add in the integrated terms:
  - Add in a difference of two prior observations, etc…
- Moving average:
  - Term is the average of the past X observations.

Time Series Summary

- Time series data are everywhere!
- A stochastic process is ‘stationary’ if there is **no trend** and **constant variance**.
- Time series values can only be sampled once
- Time series have serial dependency
- Decompose time series into trend, seasonal and remainder (noise) components
- ARIMA process:
  - AR(p) – AR process of order p, for dependency in values
  - I(d) – dth order difference operator, removes trend
  - MA(q) – MA process of order q, dependency in noise

Time Series Summary

- ARIMA(p,q,q,P,D,Q) process to model seasonal compoent
- White noise is an ARIMA(0,0,0) process
- Random walk is not stationary, but difference series is
- Use R forecast package to make life easy
- With variable volatility use ARCH or GARCH models – Beyond the scope of course

Assignment 7

- Perform time series analysis on the data for one of Milk Production, or Ice Cream Production (your choice), in the CADairyProduction.csv file to answer the following questions
  - Is this time series stationary?
  - Is there a significant seasonal component?
  - For the remainder from the decomposition of the time series what is the order of the ARMA(p,q) process that best fits.
  - Forecast production for 12 months and examine numeric values **and** plot the confidence intervals. Are the confidence intervals reasonably small compared to the forecast means.
- You should submit:
  - An R-script written in a professional style and with clear comments
  - A report summarizing your conclusions and providing charts and tables to support those conclusions