Time Series Analysis

Vivek Kantamani

TIME SERIES ANALYSIS

INTRODUCTION

Motivation:
Recall that in linear regression, we assume that the ε error terms exhibit an iid structure.

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$$

In numerous fields of study related to statistics, data is often recorded over time (or over space), and thus exhibits a temporal (spatial) structure more complex than "independent and identically distributed," because cases are dependent on the past.

Time series analysis extends the traditional linear regression setting to include important aspects of temporal dependence using time series models - discrete-time stochastic process with equal time intervals.

Objectives of Time Series Analysis:
1. Drawing inferences from time series $\{Y_t : t = 1, \ldots, n\}$.

2. Forecasting $\hat{Y}_{n+d}$ using a prediction interval (for some specified level of confidence): $\hat{Y}_{n+d} \pm ME$.

   Forecasting d steps ahead is extrapolation.
   However, by applying suitable transformations to the dataset, the data may become stationary - stationary data will allow for more accurate and tractable forecasting.

Methods - Study the Error Structure of Time Series Data:
1. Seasonal Adjustment

   Recognize and remove any seasonal component (seasonality) so it is not confused with the long-term trend.

2. Trend Adjustment

   Recognize and remove the trend.
   Note that in linear regression, the trend is the linear function $m_t = \hat{y} = \beta_0 + \beta_1 t$, and that we often subtracted the estimated trend to study the residuals and ensure that the regression model satisfied our assumptions.

3. Study the error structure (temporal residuals, noise) of time series data to examine temporal dependence structure, e.g. by constructing a time series model on the errors (e.g. ARMA).

4. Forecast using the constructed time series model.

5. Reintroduce the seasonality and trend into the model.

Classical Additive Time Series Decomposition:

$$Z_t = Y_t - m_t - s_t$$

Z_t indicates the noise, or residual, of the time series (the de-trended de-seasonalized time series).
Y_t indicates the original time series.
m_t indicates the trend of the time series.
s_t indicates the seasonality of the time series.

Using estimated values, we have

$$Z_t \approx Y_t - \hat{m}_t - \hat{s}_t$$

We will evaluate if Z_t (noise, residuals) exhibit a temporal dependence structure, or if they constitute white noise, $Z_t \sim WN$. If the residuals exhibit white noise, there is no dependence structure to exploit, so predictions will be based on seasonality

and trend.
If residuals exhibit a temporal dependence structure, this structure can be modeled using a time series model, which allows us to construct a more accurate forecast.

White Noise (WN):
$Z_1, \ldots, Z_n \sim WN$ ($White\ Noise$) iff
1. $Cov(Z_i, Z_j) = 0$ for $i \neq j$
2. $E[Z_i] = 0$
3. $Var(Z_i) = \sigma^2$

Data is commonly centered when evaluating for white noise, i.e. if $E[Z_i] = \mu \neq 0$.

Note that $Z_t \sim WN(0, \sigma^2)$ is not a distributional assumption - it is a statement regarding the correlation structure of Z_i, for a specified mean and variance.
In particular, white noise is not iid - the definition of white noise indicates that the residuals Z_i are uncorrelated - not that they are independent.
For uncorrelated residuals to be independent, additional assumptions must be placed on the residuals - e.g., the residuals must be normally distributed.

Thus $\{Z_t\}$ is iid (or independent) $\implies \{Z_t\} \sim WN$.
However, $\{Z_t\} \sim WN$ does not imply $\{Z_t\}$ is iid (unless $\{Z_t\}$ is also normal).

AR(1) Process - First-Order Autoregressive Model:
Consider the AR(1) process, the first-order autoregressive model.

$$Y_t = \phi Y_{t-1} + Z_t, \quad t \in \mathbb{Z}, \quad Z_t \sim WN(0, \sigma^2)$$

ϕ is a statistical parameter relating Y_t to the previous case Y_{t-1}.

The AR(1) process is a stochastic process where each case is governed by the previous case plus noise, throughout the integers.

Our goal is to find a solution to this equation that characterizes Y_t as a process that satisfies the AR(1) model.

The solution is given by the following linear process

$$Y_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}, \quad |\phi| < 1$$

The restriction $|\phi| < 1$ guarantees convergence (of the underlying geometric series).

We can verify this solution by substitution.

$$\phi Y_{t-1} + Z_t = \phi \sum_{j=0}^{\infty} \phi^j Z_{(t-1)-j} + Z_t = \sum_{j=0}^{\infty} \phi^{j+1} Z_{t-(j+1)} + Z_t$$

For $k = j + 1$,

$$= \sum_{k=1}^{\infty} \phi^k Z_{t-k} + \phi^0 Z_{t-0} = \sum_{k=0}^{\infty} \phi^k Z_{t-k} = Y_t$$

Can we estimate ϕ?
The model resembles regression through the origin, allowing us to identify a least squares (or maximum likelihood estimator, depending on the assumptions imposed on Z_i).

$$Y_t \sim \beta_1 x_i + \varepsilon_i$$

In other words, we can regress Y_t on Y_{t-1} using ordinary least squares.

For instance, the least squares estimator of ϕ is given by

$$\arg\min_{\phi} \sum_{i=1}^{n} (Y_t - \phi Y_{t-1})^2$$

Note that the least squares estimator can be solved for, regardless of whether $|\phi| < 1$ or not.
Note however, that for $|\phi| \geq 1$, the causal structure of our model will change.

In R, we can perform regression through the origin, regressing on the previous case as follows.

```
lm(Y[2:n] ~ Y[1:(n-1)] - 1)
```

We will later discuss
1. How to derive such solutions analytically.

2. Why we require $|\phi| < 1$.

   The infinite sum expression for Y_t will only converge if the geometric series involving ϕ^j converges - otherwise the variance of Y_t would grow explosively and fail to converge.
   Recall the convergence of a geometric series.

$$\sum_{k=0}^{\infty} r^k = \frac{1}{1-r}, \quad |r| < 1$$

3. How an (infinite) linear process solution helps practitioners.

   A linear process represents an infinite linear combination of white noise.
   A solution in the form of a linear process will provide a constraint on the range of values for the parameters - leading to stability of the theoretical solution.
   In particular, expressing a time series model as a linear process of the past is a very important property with many consequences for the model.

Examples:

Example: Violation of the iid Assumption

Data is often recorded over time (or over space), and thus exhibits a temporal (spatial) structure more complex than "independent and identically distributed," because cases are dependent on the past.

Consider the following:

a) The closing price of a financial instrument P_t heavily depends on the previous observation P_{t-1}, where t could be any time index (month, day, minute, etc.).

b) Windspeed W_t recorded on minute by minute basis exhibits spatial and temporal dependence, because windspeet at minute t, W_t, depends on the previous case W_{t-1}.

c) In classical experimental design, respondents are often recorded over time (repeated measures design).

d) In the field of machine learning, speech recognition falls within the scope of time series analysis.

Example: White Noise

Depending on the selected variance, there are infinite ways to compose white noise $Z_t \sim WN(0, \sigma^2)$.
Simulated iid noise resembles white noise, because iid noise is a stronger assumption than WN.

White noise structure can be composed from non-iid random variables.

Assume $Z_t \overset{iid}{\sim} N(0,1)$.
Define X_t as follows.

$$X_t = \begin{cases} Z_t & if\ t\ is\ even \\ \frac{(Z_{t-1}^2 - 1)}{\sqrt{2}} & if\ t\ is\ odd \end{cases}$$

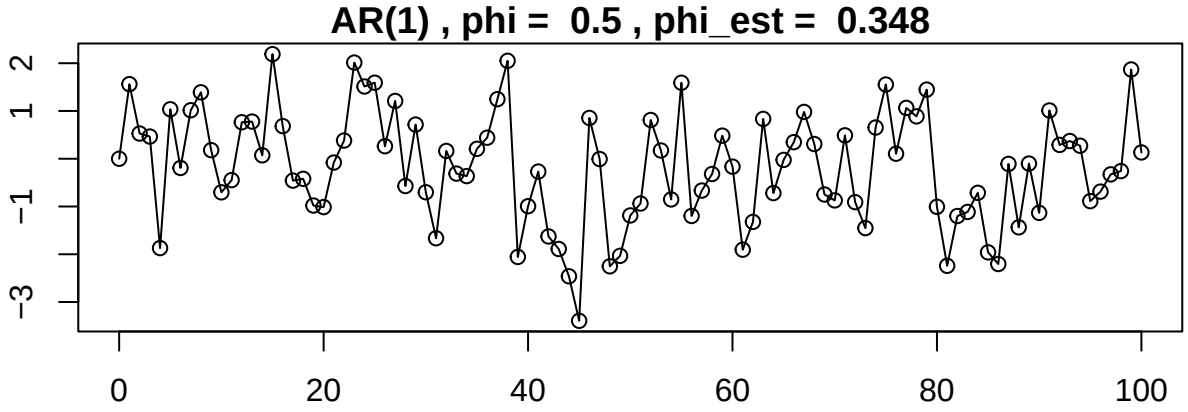
We can show that X_t is WN, although it is not iid.

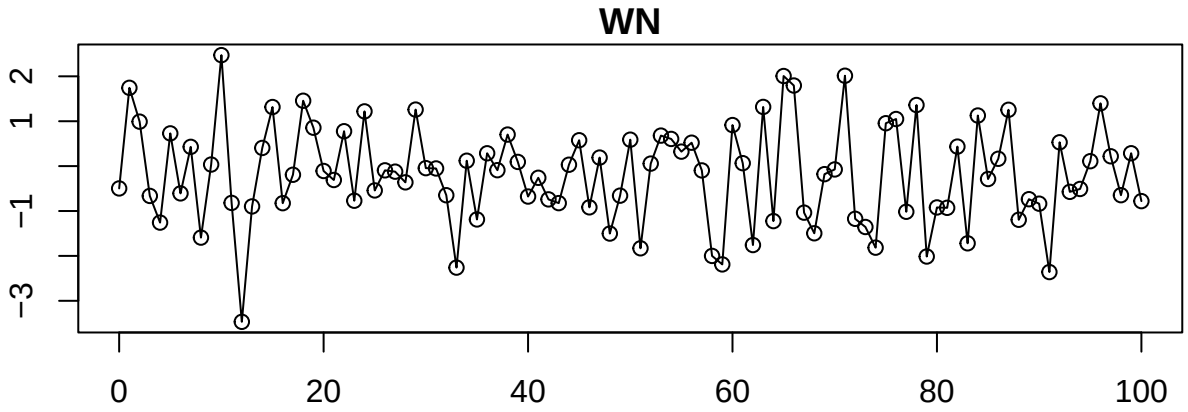
$$E[X_t] = 0$$

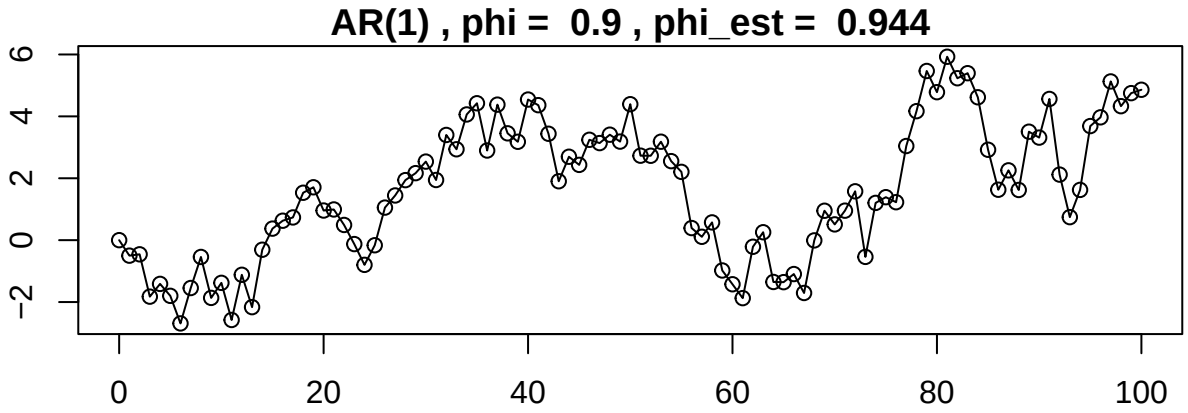
$$\gamma(h) = \begin{cases} \sigma^2 & if\ h = 0 \\ 0 & if\ h \neq 0 \end{cases}$$

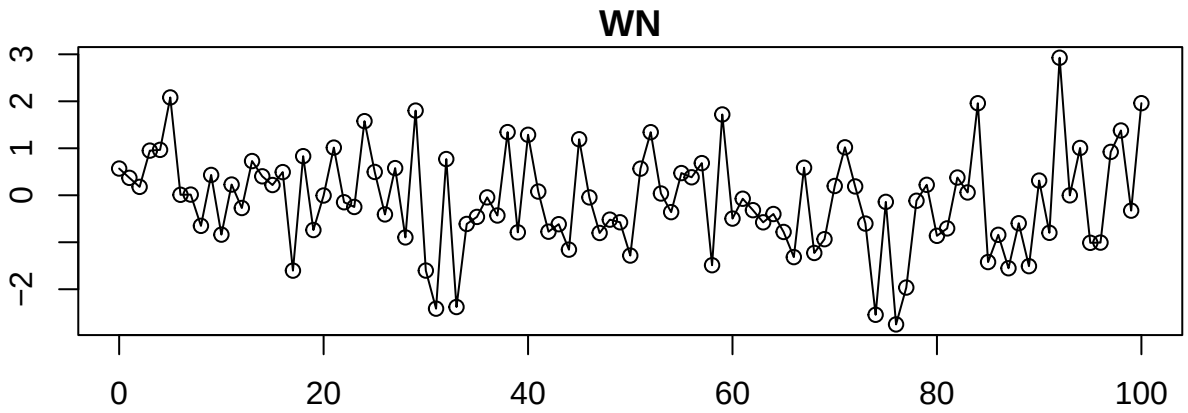
$$\rho(h) = \begin{cases} 1 & if\ h = 0 \\ 0 & if\ h \neq 0 \end{cases}$$

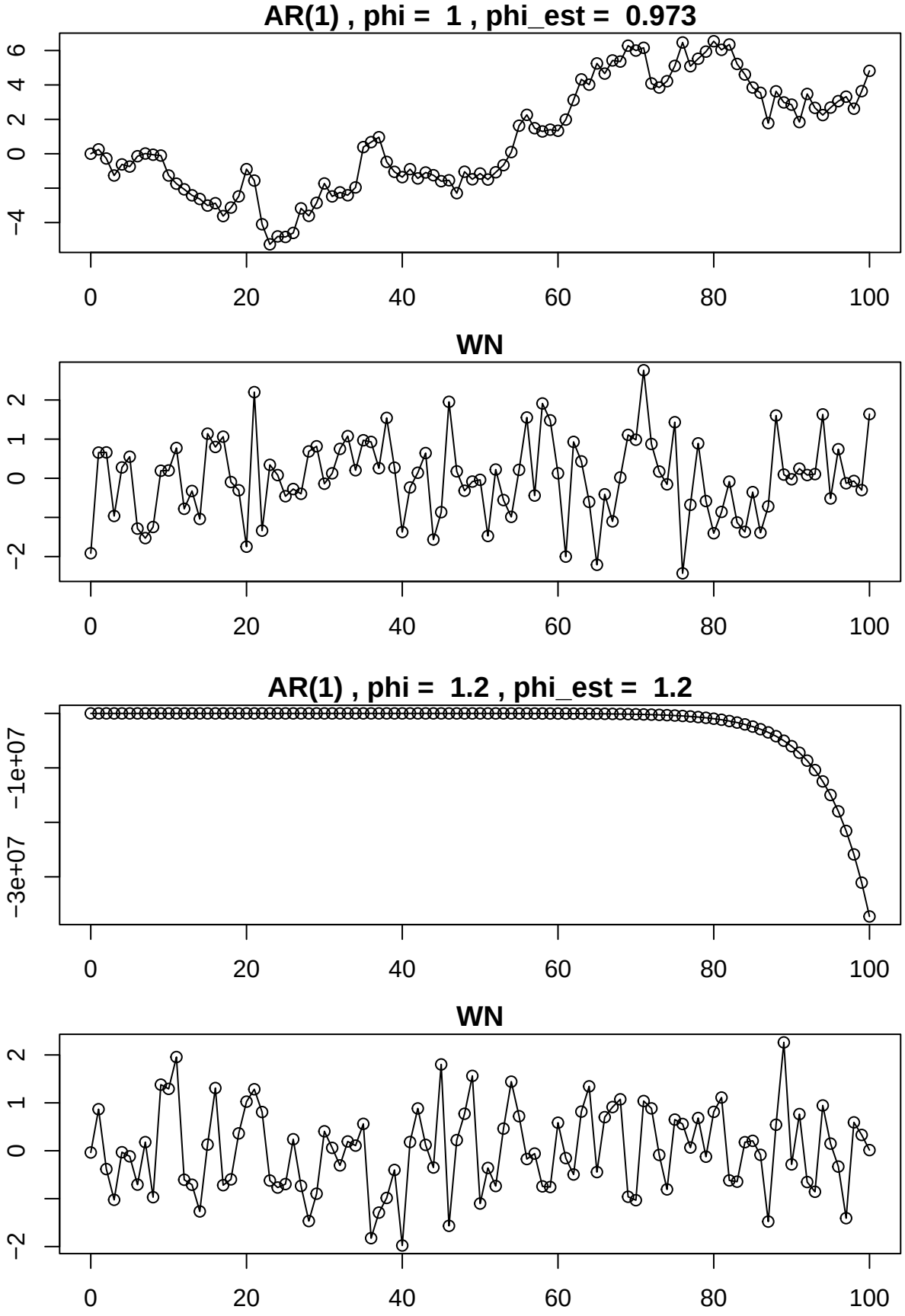
Example: AR(1) Simulation


AR(1) , phi = 0.5 , phi_est = 0.348
−3
−1
1
2
0
20
40
60
80
100


WN
−3
−1
1
2
0
20
40
60
80
100


AR(1) , phi = 0.9 , phi_est = 0.944
−2
0
2
4
6
0
20
40
60
80
100


WN
−2
0
1
2
3
0
20
40
60
80
100



Compared to WN, the AR(1) process has exhibits temporal dependence on the previous case.
Increasing ϕ (for $|\phi| < 1$) increases the temporal dependence.
$|\phi| = 1$ results in a (nonstationary) random walk - note that the random walk is the classic nonstationary example after which AR(1) processes are explosively nonstationary.
$\phi > 1$ results in a nonstationary (explosive) time series.

If de-trended and de-seasonalized data resembles white noise, a regression model would be sufficient.

To forecast the $n+1$ case, we would use the estimated value of ϕ and the nth case.
To forecast the $n+2$ case, we would use the estimated value of ϕ in the form ϕ^2 and the nth case - the further out we forecast, we become less confident in our predictions.

STOCHASTIC PROCESSES

Stochastic Process:
Consider a random experiment.

A random variable (RV) X is a function that maps from the set Ω of all possible outcomes of the random experiment to (real-valued) numbers.

$$X(\omega) : \Omega \to \mathbb{R}$$

A stochastic process (SP) is a collection of random variables that are all defined on the same probability space (random experiment) - i.e. defined on some sample space Ω and indexed by a set T.

$$(X_t : t \in T) := (X(t, \omega), t \in T, \omega \in \Omega)$$

Therefore a stochastic process is a function of two variables: T and Ω.
The index set T often represents time, but can be any ordered collection.
The state space S is the range of all possible values the random variables X_t can assume (it characterizes Ω).

Note that a specified outcome ω in the sample space Ω of a stochastic process refers to the entire collection of random variables (which take values in the state space S) represented by repetition of the same random experiment indexed by the (time) set T.

Types of Stochastic Processes:
Stochastic processes can be classified according to the characteristics of their index set T and state space S.

1. Discrete Stochastic Process (T discrete)

   Index set T is a finite or countably infinite set.

   $$T = \mathbb{Z} = \{\dots, -2, -1, 0, 1, 2, \dots\}$$
   $$(X_t : t \in T) = \dots, X_{-2}, X_{-1}, X_0, X_1, X_2, \dots$$

   a. Discrete State Space (S)
   e.g. # of deadly car accidents in NY on day $t = 1, 2, \dots$

   b. Continuous State Space (S)
   e.g. Time Series - average daily temperature in NY on day $t = 1, 2, \dots$

2. Continuous Stochastic Process (T continuous)

   Index set T is defined on some uncountably infinite set (interval).

   $$(X_t : t \geq 0)$$

   a. Discrete State Space (S)
   e.g. Poisson Process - # of customers at a store by time t between 8am and 9pm

   b. Continuous State Space (S)
   e.g. Brownian Motion - the euro/dollar exchange rate at time t between 7am and 20pm GMT

Visualizing Stochastic Processes:
For a fixed $t \in T$, a stochastic process is a random variable (with a probability distribution).

$$\omega \mapsto X_t(\omega), \quad \omega \in \Omega$$

For a fixed random outcome $\omega \in \Omega$, a stochastic process is a function of time.

$$t \mapsto X_t(\omega), \quad t \in T$$

This is known as a sample path, trajectory, or realization of $(X_t : t \in T)$.

In time series, we will consider this latter characterization, in which a stochastic process can be viewed as a function that maps from outcomes ω in the sample space Ω to all possible functions defined on T.

$$\{X_t : t \in T\} : \omega \in \Omega \to X_t(\omega), \quad t \in T$$

Characterizing a Discrete Stochastic Process:
Consider a discrete stochastic process.

$$\{X_t : t \in T\} : \omega \in \Omega \to X_t(\omega), \quad t \in T$$

A finite dimensional distribution is the joint probability distribution of a random vector - i.e., for an observed stochastic process at the time points $\{t_1, \ldots, t_n\}$, the probability distribution of the random vector $\{X_{t_1} = i_{t_1}, \ldots, X_{t_n} = i_{t_n}\}$, $P(X_{t_1} = i_{t_n}, \ldots, X_{t_n} = i_{t_n})$.
The finite dimensional distributions (for various $\{t_1, \ldots, t_n\}$ for any $n \in \mathbb{N}$) characterize the probability law of the stochastic process for countable state spaces.

However, it is often difficult to identify the finite dimensional distribution (joint probability distribution) in practice.
Instead, we will characterize a stochastic process by its moment definitions and measures of its dependence structure.

Characterizing a Stochastic Process - Moment Definitions and Measures of Dependence Structure:

1. Mean Function - First Moment Definition

   The mean function of a stochastic process is defined as

   $$E[X_t] = \mu_X(t) = \mu_t, \quad t \in T$$

   provided that the first moment is finite, $E[|X_t|] < \infty$.

   For a fixed $t \in T$, a stochastic process is a random variable (with a probability distribution) - the mean function is the trajectory of the averages of the distributions for all t.

   Note that we often assume that $E[X_t^2] < \infty$, i.e. that X_t has a finite second moment.
   Recall that the LLN requires a finite first moment, and the CLT requires a finite first and second moment.

2. Autocovariance Function (ACVF) - Second Moment Definition

   The autocovariance function (ACVF) of a stochastic process with finite second moments is defined as

   $$\gamma_X(t, s) = \gamma(t, s) = Cov(X_t, X_s) = E[(X_t - \mu_t)(X_s - \mu_s)], \quad \forall\ s, t \in T$$

Properties:

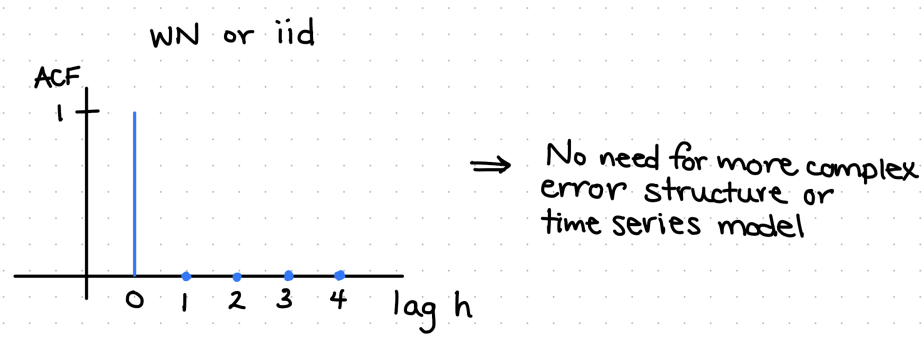
1. $\gamma(t, t) = Var(X_t)$
2. ACVF is symmetric

   $$\gamma_X(t, s) = \gamma_X(s, t)$$

3. Autocorrelation Function (ACF) - Second Moment Definition

   The autocorrelation function (ACF) of a stochastic process with non-zero and finite second moments (i.e. variance) is defined as

   $$\rho_X(t, s) = \rho(t, s) = Corr(X_t, X_s) = \frac{\gamma(t, s)}{\sqrt{\gamma(t, t)\gamma(s, s)}}$$



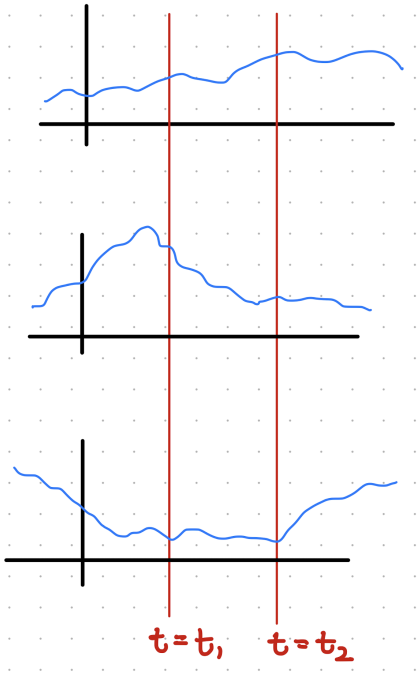
Examples:

Example: Visualizing the Trajectory of a Stochastic Process

Consider visualizing the time series trajectory of stocks.

Consider visualizing stocks where stocks are the outcomes $\omega \in \Omega$ being considered.
For $S_1 \in \Omega$, $S_2 \in \Omega$, and $S_3 \in \Omega$, we consider the following trajectories.

$$X_t = X(t, S_1), \quad X_t = X(t, S_2), \quad X_t = X(t, S_3)$$



Example: Characterizing a Stochastic Process

Consider a sequence of random variables $\{Z_1, Z_2, \dots\}$, for $Z_i \overset{iid}{\sim} N(0, \sigma^2)$, $t \in \mathbb{Z}^+$.

Find the mean function, autocovariance function (ACVF), and autocorrelation function (ACF) of this stochastic process.

$$X_t = Z_t, \quad Z_i \overset{iid}{\sim} N(0, \sigma^2), \quad t \in \mathbb{Z}^+$$

$$\mu_t = E[X_t] = 0$$

$$\gamma(t,s) = Cov(X_t, X_s) = E[(X_t - \mu_t)(X_s - \mu_s)] = E[X_t X_s] = E[Z_t Z_s] = Cov(Z_t, Z_s)$$

$$= \begin{cases} = \gamma(t,t) = \sigma^2 & if\ t = s \\ \overset{iid}{=} 0 & if\ t \neq s \end{cases}$$

$$\rho(t,s) = \begin{cases} 1 & if\ t = s \\ 0 & if\ t \neq s \end{cases}$$

TIME SERIES MODELS

Time Series Model:
A time series model is a discrete (evenly spaced) stochastic process $\{X_t : t \in T\}$ with a continuous state space.
We will often define the index set as $T = \mathbb{Z}$ or $T \subset \mathbb{Z}$.

In particular, a time series model for the observed data $\{x_t : t \in T\}$ is a specification of the finite dimensional (joint) distributions (or simply the mean and autocovariance functions) of a sequence of random variables $\{X_t : t \in T\}$, of which $\{x_t : t \in T\}$ is postulated to be a realization.

Characterizing a Time Series Model - Moment Definitions and Measures of Dependence Structure:
We can characterize a time series model by its finite dimensional (joint) probability distribution.

$$P(X_1 \leq x_1, X_2 \leq x_2, \ldots, X_n \leq x_n)$$

for $-\infty < x_1, \ldots, x_n < \infty$, and $n = 1, 2, \ldots$

For instance, if we assume that a time series model is Gaussian, the joint probability distribution is given by the multivariate Gaussian (which characterizes the first and second order moments).
In particular, if $\{X_t\}$ is multivarate normal, then the second order moments provide a complete characterization of $\{X_t\}$.

However, the joint probability distribution is rarely known or used in practice.
Instead, we typically focus on the first and second order moments (e.g., mean function, ACVF, ACF).

General Approach to Time Series Modeling:
In time series analysis, we are interested in studying the error structure of our models.
By eliminating trend and seasonality, we seek to exploit the remaining error structure using time series models to forecast future cases.
If error structure is iid or resembles white noise (WN), regression methods are appropriate.

1. EDA to check for
   a. Trend
   b. Seasonality
   c. Sharp Changes - violations of stationarity (transform data, e.g. difference operator, or use a piecewise mean function)
   d. Outlying Cases - use a robust estimator (e.g. absolute loss instead of squared loss)
2. Remove trend and seasonal components to get "stationary residuals."

   If the residuals resemble WN or IID structure, there is no way to exploit the structure with further models.
   Thus, a WN / IID structure is not optimal for forecasting - there is no error structure to exploit, it is completely random.
3. Choose a model to fit the error structure / residuals (AR, MA, ARMA, ARIMA, GARCH).

   Note that there are ways to fit trend and time series model parameters in a single step.
4. Estimate the model (MOM, MLE, etc.).
   a. Check diagnostics.
5. Forecast residuals and "invert" de-trending and de-seasonalizing transformations (i.e. to forecast the raw time series)

Removing Trend:

The time series X_t can be represented as the sum of the trend m_t and the de-trended time series Y_t.

$$X_t = m_t + Y_t$$

We would like to study the de-trended time series Y_t.

$$Y_t \approx X_t - m_t$$

1. Ordinary Least Squares.

   Assuming the trend is given by $m_t = \beta_0 + \beta_1 t$, we can estimate the trend using OLS (i.e. squared loss).

$$\arg\min_{\beta} \sum_{t=1}^{n} (X_t - \hat{m}_t)^2$$

   The de-trended time series $\hat{Y}_t$ is therefore given by the residuals of the line of best fit.

Note that ordinary least squares (OLS) can be used for more complex regression models - e.g. polynomial regression, harmonic regression, etc.

2. Simple Moving Average (Rolling Average) Smoother

We can estimate the trend using a simple moving average (SMA, rolling average) model.

$$w_t = \frac{1}{2q+1} \sum_{j=-q}^{q} X_{i-j}$$

q represents the region over which we would like to select cases.
The model takes $2q+1$ observations around time point i to find the average of X_i.

Recall $X_t = m_t + Y_t$. Then

$$X_t \approx w_t = \frac{1}{2q+1} \sum_{j=-q}^{q} m_{t-j} + \frac{1}{2q+1} \sum_{j=-q}^{q} Y_{t-j}$$

Therefore, assuming m_t is approximately linear over $[t-q, t+q]$,

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

Removing Seasonality:
The time series X_t can be represented as the sum of the seasonality of the time series s_t and the de-seasonalized time series Y_t.

$$X_t = s_t + Y_t$$

We would like to study the de-seasonalized time series Y_t.

$$Y_t \approx X_t - s_t$$

1. Harmonic Regression (Parametric Sigmoidal Model)

There are many methods for estimating seasonality, including using seasonal averages.
We can fit a harmonic regression model (a parametric linear combination of sigmoidal functions) to estimate seasonality (and trend).

$$s_t = \alpha_0 + \sum_{j=1}^{k} \alpha_j \cos(\lambda_j t) + \beta_j \sin(\lambda_j t)$$

Where s_t is parametrized by α_i, $i = 0, \ldots, k$, and β_j, $j = 0, \ldots, k$.
$\lambda_1, \ldots, \lambda_k$ are fixed frequencies (integer multiples of $\frac{2\pi}{period}$).
Note that both the *cosine* and *sine* functions have the same frequency.

The α and β terms will be estimated using least squares (OLS, squared loss) harmonic regression.

$$\arg\min_{\alpha} (X_t - \hat{s}_t)^2$$

Where α is the parameter vector that includes the α and β terms.
The de-seasonalized time series $\hat{Y}_t$ is therefore given by the residuals of the line of best fit.

Stationary Models:
Loosely speaking, $\{X_t : t \in \mathbb{Z}\}$ is stationary if $X_t \overset{d}{\approx} X_{t+h}$ for $h > 0$ (if X_t and X_{t+h} are similar in distribution).
In other words, $X_t \approx$ *"shifted RV"*.

We will restrict our attention to first and second order models (the mean and the covariance functions).

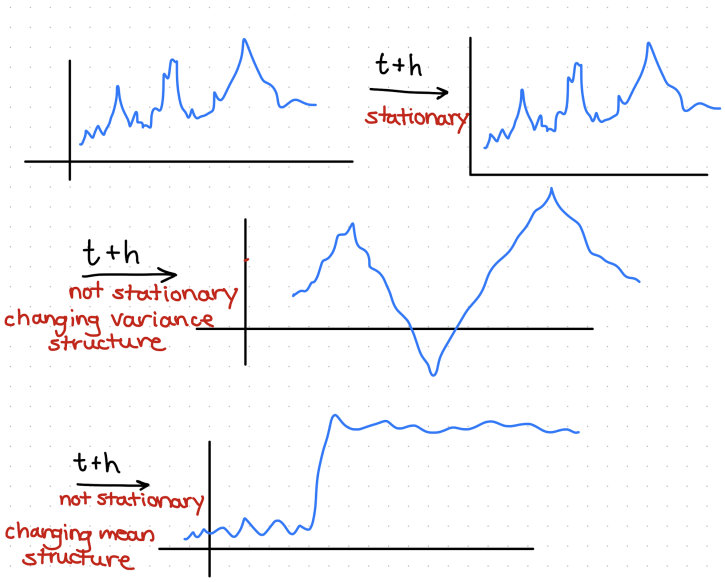
$$E[X_t] = \mu_t, \quad \gamma(t,s) = Cov(X_t, X_s)$$

Note that in both a theoretical and applied perspective, we check the first and second order properties of a time series model to evaluate (weak) stationarity.

Weak Stationarity:

$\{X_t\}$ is weakly stationary if
1. $E[X_t] = \mu_X(t)$ is independent of t (does not depend on t)
2. $\gamma_X(t+h, t) = \gamma(h)$ is independent of t for each h (for a fixed value of h, it should not depend on t).

Whenever we say '$\{X_t\}$ is stationary,' we mean weakly stationary, unless otherwise indicated.



Strict Stationarity:

$\{X_t\}$ is strictly stationary if

$$\mathbf{X} = (X_1, \ldots, X_n) \stackrel{d}{=} (X_{1+h}, \ldots, X_{n+h}), \quad \forall\, h \in \mathbb{Z},\ n \geq 1$$

Note that equality in distribution of the finite dimensional distributions implies equality of the marginal distributions.

We would like to work with strict stationarity for its mathematical properties (e.g. iid sequence is strictly stationary).
However many properties are easier to handle for weak stationarity.
Thus, whenever we say '$\{X_t\}$ is stationary,' we mean weakly stationary, unless otherwise indicated.

Properties:

1. Strict stationarity implies weak stationarity.
Weak stationarity does not imply strict stationarity.

ACVF and ACF of Stationary Time Series:

Let $\{X_t\}$ be a stationary time series.
The ACVF of $\{X_t\}$ at lag h is

$$\gamma_X(h) = Cov(X_{t+h}, X_t)$$

The ACF of $\{X_t\}$ at lag h is

$$\rho_X(h) = Corr(X_{t+h}, X_t) = \frac{\gamma_X(h)}{\gamma_X(0)}$$

Properties:

1. For stationary time series, $\gamma_X(h) \equiv \gamma_X(t+h, t)$, because stationary time series do not depend on t.
2. $\gamma(0) = Var(X_t) \geq 0$
3. $|\gamma(h)| \leq \gamma(0), \quad \forall\, h$

Proof:

Holds by the Cauchy-Schwartz inequality.

4. $\gamma(\cdot)$ is an even function, $\implies \gamma(h) = \gamma(-h), \quad \forall\ h$

Proof:

The covariance function is symmetric.

$$\gamma(h) = Cov(X_{t+h}, X_t) = Cov(X_t, X_{t+h}) = \gamma(-h)$$

Sample Autocovariance and Autocorrelation Function:
The goal is to estimate ρ_X from observed time series data.

Let $x_1, \ldots, x_n$ be observations of a time series.

The sample mean is $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$.

The sample autocovariance function is

$$\hat{\gamma}(h) = \frac{1}{n}\sum_{t=1}^{n-|h|}(x_{t+|h|} - \bar{x})(x_t - \bar{x})$$

$|h|$ is used to produce a meaningful sum whether a positive or negative lag is used.
For $h = 0$ ($\hat{\gamma}(0)$), we obtain the sample variance formula, except we divide by n instead of by the $n - h$ degrees of freedom - for large samples this difference is negligible.
For instance, recall that MLE estimators in regression use $\frac{1}{n}$.

The sample autocorrelation function is

$$\hat{\rho}(h) = \frac{\hat{\gamma}(h)}{\hat{\gamma}(0)}$$

In R, the sample autocorrelation function is given by `acf()`.

Approximate Sampling Distribution for the Sample Autocorrelation Function - Is X_t IID / WN?:
We would like to establish a sampling distribution for the sample autocorrelation function for the case when X_t is IID noise (or WN) to evaluate if the observed data contradicts this framework (e.g., in a testing procedure).

Assuming iid noise, the correlation structure should approximate a normal distribution

$$\hat{\rho}(h) \overset{d}{\approx} N\left(0, \frac{1}{n}\right)$$

For large n, $h > 0$, $\sigma^2 = 1$.
Note that the process is scaled so that the WN variance is 1.

Assuming iid noise, for 95% confidence, we have the confidence bands

$$\hat{\rho}(h) \pm \frac{1.96}{\sqrt{n}}$$

These are the confidence bands that appear in the `acf()` plot.

Differencing and Difference Operator, Backward Shift (Backshift) Operator:
The lag-1 difference operator ∇_d is defined as

$$\nabla_d X_t = X_t - X_{t-1} = (1 - B^d)X_t$$

Where the backward shift operator B is defined as

$$BX_t = X_{t-1}$$

In other words, the backward shift operator moves the case back by 1-lag.

Properties:

1. Powers of ∇ and B follow intuitively.

$$\nabla^2 X_t = \nabla(\nabla X_t) = \nabla(X_t - X_{t-1})$$
$$= (X_t - X_{t-1}) - (X_{t-1} - X_{t-2}) = X_t - 2X_{t-1} - X_{t-2}$$

$$\nabla^2 X_t = \nabla(\nabla X_t) = (1-B)(1-B)X_t = (1-B)^2 X_t$$
$$= (1 - 2B + B^2)X_t = X_t - 2BX_t + B^2 X_t = X_t - 2BX_t + B(BX_t)$$
$$= X_t - 2X_{t-1} + X_{t-2}$$

2. It is not possible to "divide" by an operator - an inverse operator must be defined instead.

Uses:

1. Differencing can be used to eliminate trend (and seasonality) in time series data.
Recall that we de-trend and de-seasonalize (or difference) a time series dataset to exploit the remaining error structure.

Suppose we have a linear trend $m_t(t) = \beta_0 + \beta_1 t$.

$$\nabla m_t(t) = \nabla(\beta_0 + \beta_1 t) = (\beta_0 + \beta_1 t) - (\beta_0 + \beta_1(t-1))$$
$$= \beta_0 + \beta_1 t - \beta_0 - \beta_1 t + \beta_1 = \beta_1$$

Examples:

Example: Time Series Model - Binary Process

Consider the stochastic process $\{X_t : t = 1, 2, \dots\}$ defined as follows for $0 \leq p \leq 1$,

$$P(X_t = 1) = p$$
$$P(X_t = -1) = 1 - p$$

Example: Time Series Model - Random Walk Model (Discrete Noise Structure)

a) Consider the stochastic process $\{S_t : t = 0, 1, \dots\}$, for

$$S_t = X_1 + X_2 + \cdots + X_t, \quad t = 1, 2, \dots$$

for $\{X_t : t = 1, 2, \dots\} \sim Binary\ Process$,

$$P(X_t = 1) = p, \quad 0 \leq p \leq 1$$
$$P(X_t = -1) = 1 - p$$

Note that the binary process will have an ACF similar to WN, because it resembles an iid process (i.e. there is no relationship between X_t's).
In general, when you consider a white noise component for a random walk, the random walk is not stationary (e.g. the binary process resembles WN as an iid process).
However, we note that the difference operator $S_t - S_{t-1} = X_t$, yields a stationary time series (sometimes, as in this case).

Example: HW 1 #1 - Characterizing a Random Walk Model

Let $\{X_t : t \in \mathbb{N}\}$ be a stochastic process (or time series) defined by

$$X_t = \sum_{i=1}^{t} Z_i, \quad t \in \mathbb{N}$$

where $\{Z_t : t \in \mathbb{Z}^+\} \sim WN(0, \sigma^2)$.

a) Derive the mean function $\mu_X(t)$ of $\{X_t : t \in \mathbb{N}\}$.

$$\mu_X(t) = E[X_t] = E\left[\sum_{i=1}^{t} Z_i\right] \overset{linear}{=} \sum_{i=1}^{t} E[Z_i] = (t)(0) = 0$$

b) Derive the covariance function $\gamma_X(s,t)$ of $\{X_t : t \in \mathbb{N}\}$.

$$\gamma_X(s,t) = Cov(X_s, X_t) = E[(X_s - \mu_X(s))(X_t - \mu_X(t))] = E[X_s X_t] = E\left[\sum_{i=1}^{s} Z_i \sum_{j=1}^{t} Z_j\right]$$

For $s < t$,

$$E[(Z_1 + \cdots + Z_s)(Z_1 + \cdots + Z_t)] = E\left[\sum_{i=1}^{s} Z_i^2 + k\sum\sum_{j<k} Z_j Z_k\right] \overset{linearity}{=} \sum_{i=1}^{s} E[Z_i^2] + k\sum\sum_{j\in[1,s]<k\in[1,t]} E[Z_i Z_j]$$

Note that $Var(Z_i) = E[Z_i^2] - (E[Z_i])^2 = E[Z_i^2]$, for $E[Z_i] = 0$.
Note that $Cov(Z_i, Z_j) = E[Z_i Z_j] - E[Z_i]E[Z_j] = E[Z_i Z_j] = 0$, because the WN structure is uncorrelated and has mean 0.

$$= sVar(Z_i) = s\sigma^2, \quad for\ s < t$$

$$\therefore \gamma_X(s,t) = \min(s,t)\sigma^2$$

Note that $\gamma(t,t) = Var(X_t) = (\sigma^2)\min(t,t) = t\sigma^2$.
In the random walk, because the variance increases with t, the further out that we go, the less certain we are of where we are located.
In fact, the best forecast is the previous case.

c) Derive the correlation function $\rho_X(t,s)$ of $\{X_t : t \in \mathbb{N}\}$\$.

$$\rho_X(s,t) = \frac{\gamma_X(s,t)}{\sqrt{\gamma_X(s,s)\gamma_X(t,t)}} = \frac{\min(s,t)\sigma^2}{\sigma^2\sqrt{s}\sqrt{t}} = \frac{s\sigma^2}{\sigma^2\sqrt{s}\sqrt{t}} = \frac{\sqrt{s}}{\sqrt{t}}, \quad for \ \ s < t$$

The random walk is not stationary because $\gamma_X(t+h,t) = t\sigma^2$ depends on t.
Hypothesis testing procedures that test for stationarity often test against the random walk as the alternative hypothesis.

Example: HW 1 #2 - Characterizing an AR(1) Model, Causal for $|\phi| < 1$, Random Walk for $|\phi| = 1$

Let $\{X_t : t \in \mathbb{Z}\}$ be a time series defined by the First-Order Autoregressive Model AR(1).

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0,\sigma^2), \quad t \in \mathbb{N}, \quad |\phi| < 1$$

Note that the assumption that $|\phi| < 1$ ensures stationarity and causality - that observations (X's) are composed of past WN terms.
Thus Z_t is uncorrelated with (past X's) X_s for each $s < t$.

Note also that because X_t is observable, it is easier to estimate ϕ than it is to estimate θ in the MA(1) case.

a) Derive $\mu_X(t)$.

Note that this is a zero-mean process.
We will take expectation of both side of the equation.

$$E[X_t] = \phi E[X_{t-1}] + E[Z_t] = \phi E[X_{t-1}]$$

Given that $|\phi| < 1$, because this is a recursive process $\forall\ mathbbZ$, the only way for this equality to hold is if $E[X_t] = 0$.

$$E[X_t] = \phi E[X_{t-1}]$$
$$E[X_{t-1}] = \phi E[X_{t-2}]$$
$$\vdots$$

$$\therefore |\phi| < 1 \implies E[X_t] = 0$$

b) Derive the variance of $\{X_t\}$.

The AR(1) process is a zero-mean process for $|\phi| < 1$.

$$\mu_X(t) = E[X_t] = 0$$

$$Var(X_t) = \gamma_X(t,t) = Cov(X_t, X_t) = E[(X_t - \mu_X(t))^2]$$
$$= E[X_t^2] = E[(\phi X_{t-1} + Z_t)(\phi X_{t-1} + Z_t)] = E[\phi^2 X_{t-1}^2 + 2\phi Z_t X_{t-1} + Z_t^2]$$

Note that Z_t and X_t are zero-mean processes.
Note that Z_t and past X_{t-h} terms are uncorrelated.

$$Cov(Z_t, X_{t-1}) = E[Z_t X_{t-1}] - E[Z_t]E[X_{t-1}] = E[Z_t X_{t-1}] = 0$$

Note that

$$Var(Z_t) = E[Z_t^2] - (E[Z_t])^2 = E[Z_t^2] = \sigma^2$$

$$= \phi^2 E[X_{t-1}^2] + 2\phi E[Z_t X_{t-1}] + E[Z_t^2]$$
$$= \phi^2 E[X_{t-1}^2] + \sigma^2$$

$$\therefore E[X_t^2] = \phi^2 E[X_{t-1}^2] + \sigma^2$$

We note that this is a recursive relationship.
Expressing the equation in terms of the covariance function, both terms are lag 0.

$$\gamma_X(0) = \phi^2 \gamma_X(0) + \sigma^2$$
$$(1 - \phi^2)\gamma_X(0) = \sigma^2$$
$$\gamma_X(0) = Var(X_t) = \frac{\sigma^2}{1 - \phi^2}$$

c) Derive the ACVF $\gamma_X(h)$ and ACF $\rho_X(h)$.

We multiply both sides of the equation by X_{t-h}.

$$(X_t = \phi X_{t-1} + Z_t) \cdot X_{t-h}$$
$$X_t X_{t-h} = \phi X_{t-1} X_{t-h} + Z_t X_{t-h}$$

Taking the expectation, we have

$$E[X_t X_{t-h}] = \phi E[X_{t-1} X_{t-h}] + E[Z_t X_{t-h}]$$

Note that covariance is symmetric for stationary time series, so $\gamma(h) = \gamma(-h)$, which implies that $E[X_t X_{t-h}] = \gamma_X(h)$.

Note that the lag between $t-1$ and $t-h$ is $(t-1) - (t-h) = h - 1$.
Because they have mean 0, $Cov(X_{t-1}, X_{t-h}) = E[X_{t-1} X_{t-h}] = \gamma_X(h-1)$.

Note that Z_t is uncorrelated with past X terms (and both Z_t, X_s have mean 0), implying that $Cov(Z_t, X_{t-h}) = E[Z_t X_{t-h}] = 0$.

$$\therefore \gamma_X(h) = \phi\ \gamma_X(h-1)$$

Note that this is a recursive relationship for $|\phi| < 1$.

$$\gamma_X(h) = \phi\ \gamma_X(h-1) = \phi^2 \gamma_X(h-2) = \phi^3 \gamma_X(h-3) = \cdots = \phi^h \gamma_X(0)$$

$$\therefore \rho_X(h) = \frac{\gamma_X(h)}{\gamma_X(0)} = \phi^{|h|}, \quad h \in \mathbb{Z}$$

The absolute value of h is used for simplification purposes, and this is justified because the ACF is symmetric for stationary processes.

Both the mean function and ACVF/ACF do not depend on t, implying that the AR(1) process is stationary.

d) Show that if $\phi = 1$, the AR(1) process is the random walk model (assume $X_0 = 0$).

For $\phi = 1$, the AR(1) process becomes a random walk.

$$Y_t = Z_t + \phi Y_{t-1} = Z_t + Y_{t-1} = Z_t + (Y_{t-2} + Z_{t-1}) = Z_t + Z_{t-1} + (Y_{t-3} + Z_{t-2}) = \cdots$$

$$X_0 = 0$$
$$X_1 = \phi X_0 + Z_1 = (1)(0) + Z_1 = Z_1$$
$$X_2 = \phi X_1 + Z_2 = (1)(Z_1) + Z_2 = Z_1 + Z_2$$
$$X_3 = \phi X_2 + Z_3 = (1)(Z_1 + Z_2) + Z_3 = Z_1 + Z_2 + Z_3$$
$$\vdots$$
$$\therefore X_n = \sum_{i=1}^{n} Z_i, \quad Z_i \sim WN(0, \sigma^2), \quad t \in \mathbb{N}_0$$

In other words, for $\phi = 1$, the AR(1) model sums white noise terms, which is a random walk model.

Example: HW 1 #3 - Characterizing a MA(1) Model

Let $\{X_t : t \in \mathbb{Z}\}$ be a time series defined by the First-Order Moving Average Model MA(1).

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad \theta \in \mathbb{R}$$

a) Derive the mean function $\mu_X(t)$ of $\{X_t : t \in \mathbb{Z}\}$.

$$\mu_X(t) = E[X_t] = E[Z_t + \theta Z_{t-1}] = E[Z_t] + \theta E[Z_{t-1}] = 0$$

b) Derive the covariance function $\gamma_X(t+h, t)$ of $\{X_t : t \in \mathbb{Z}\}$.

$$\gamma_X(t+h, t) = Cov(X_{t+h}, X_t) = E[X_{t+h}X_t] = E[(Z_{t+h} + \theta Z_{t+h-1})(Z_t + \theta Z_{t-1})]$$
$$E[Z_{t+h}Z_t + \theta Z_{t+h}Z_{t-1} + \theta Z_{t+h-1}Z_t + \theta^2 Z_{t+h-1}Z_{t-1}]$$
$$= E[Z_{t+h}Z_t] + \theta E[Z_{t+h}Z_{t-1}] + \theta E[Z_{t+h-1}Z_t] + \theta^2 E[Z_{t+h-1}Z_{t-1}]$$

For $h = 0$,

$$E[Z_t^2] + \theta E[Z_t Z_{t-1}] + \theta E[Z_{t-1}Z_t] + \theta^2 E[Z_{t-1}^2]$$

Note that $E[Z_t^2] = Var(Z_t) = \sigma^2$ and $E[Z_{t-1}^2] = Var(Z_{t-1}) = \sigma^2$.
Note that $E[Z_t Z_{t-1}] = Cov(Z_t, Z_{t-1}) = 0$ and $E[Z_{t-1}Z_t] = Cov(Z_{t-1}, Z_t) = 0$.

$$= (1 + \theta^2)\sigma^2$$

For $h = \pm 1$ (the MA(1) process is stationary),

$$E[Z_{t+1}Z_t] + \theta E[Z_{t+1}Z_{t-1}] + \theta E[Z_t^2] + \theta^2 E[Z_t Z_{t-1}]$$

Z_t is a zero-mean process, so the covariance terms are 0, and $E[Z_t^2] = Var(Z_t) = \sigma^2$.

$$= \sigma^2 \theta$$

For $h = \pm 2$ (the MA(1) process is stationary),

$$E[Z_{t+2}Z_t] + \theta E[Z_{t+2}Z_{t-1}] + \theta E[Z_{t+1}Z_t] + \theta^2 [Z_{t+1}Z_{t-1}]$$

Z_t is a zero-mean process so the covariance terms are 0.

$$= 0$$

For $|h| > 1$, we note that the terms $\{(t+h, t), (t+h, t-1), (t+h-1, t), (t+h-1, t-1)\}$ will never match, implying that the expectations of products (covariances) will all go to 0.

$$\gamma(t+h,t) = \gamma(h) = \begin{cases} \sigma^2(1+\theta^2) & if\ h=0 \\ \sigma^2\theta & if\ |h|=1 \\ 0 & if\ |h|>1 \end{cases}$$

c) Is $\{X_t : t \in \mathbb{Z}\}$ stationary and why?
Does the stationarity of X_t depend on θ?

$\{X_t : t \in \mathbb{Z}\}$ is stationary because $\mu_X = 0$ is constant in t, and $\gamma_X(t+h,t)$ is constant in t for each h. Stationarity of $\{X_t : t \in \mathbb{Z}\}$ does not depend on θ, because no matter the value of θ, t would not be introduced in into μ_X or $\gamma_X(t+h,t)$.
.

d) Derive the correlation function $\rho_X(h)$ of $\{X_t : t \in \mathbb{Z}\}$.

$$\rho_X(t+h,t) = \rho_X(h) = \frac{\gamma_X(h)}{\gamma_X(0)} = \begin{cases} \frac{\sigma^2(1+\theta^2)}{\sigma^2(1+\theta^2)} = 1 & if\ h=0 \\ \frac{\sigma^2\theta}{\sigma^2(1+\theta^2)} = \frac{\theta}{1+\theta^2} & if\ |h|=1 \\ \frac{0}{\sigma^2(1+\theta^2)} = 0 & if\ |h|>1 \end{cases}$$

Example: Stationarity of a First Order Moving Average Model, MA(1)

A first order moving average model is not to be confused with a simple moving average model.

Recall that in an AR model, X_t depends on a previous X case, X_{t-1}.
In a MA(1) model, the noise Z_t depends on the previous Z case, Z_{t-1}.

$$X_t = Z_t + \theta Z_{t-1}, \quad t \in \mathbb{Z},\ \theta \in \mathbb{R}$$

Where $Z_t \sim WN(0, \sigma^2)$.

Note that $\theta \in \mathbb{R}$ lacks a constraint to guarantee stationarity (unlike in the AR(1) model).

$$\mu_X(t) = E[X_t] = E[Z_t + \theta Z_{t-1}] = E[Z_t] + \theta E[Z_{t-1}] = 0$$

We need to find the variance to find the ACF.
For lag $h = 0$,

$$Var(X_t) = E\left[X_t^2\right] = E\left[(Z_t + \theta Z_{t-1})(Z_t + \theta Z_{t-1})\right] = E[Z_t^2 + \theta Z_t Z_{t-1} + \theta Z_{t-1} Z_t + \theta^2 Z_{t-1}]$$
$$= E[Z_t^2] + \theta E[Z_t Z_{t-1}] + \theta E[Z_{t-1} Z_t] + \theta^2 E[Z_{t-1}^2]$$

Note that for WN, $Cov(Z_i, Z_j) = 0 \implies E[Z_i Z_j] = 0$.

$$= \sigma^2 + 0 + 0 + \theta^2\sigma^2 = \sigma^2(1+\theta^2)$$

We will derive the remainder in the homework.

$$\gamma_X(t+h,t) = \gamma_X(h) = \begin{cases} \sigma^2(1+\theta^2) & if\ h=0 \\ \sigma^2\theta & if\ h=\pm 1 \\ 0 & if\ |h|>1 \end{cases}$$

Note that for lag $h = 0$, $\gamma_X(t, t+h) = \gamma_X(t,t)$, which is the variance.

$$\rho_X(t+h,t) = \rho_X(h) = \begin{cases} \frac{\sigma^2(1+\theta^2)}{\sigma^2(1+\theta^2)} = 1 & if\ h=0 \\ \frac{\sigma^2\theta}{\sigma^2(1+\theta^2)} & if\ h=\pm 1 \\ \frac{0}{\sigma^2(1+\theta^2)} = 0 & if\ |h|>1 \end{cases}$$

Note that the MA(1) model is stationary and has lag-1 dependence.
The MA(2) model is stationary and has lag-2 dependence, etc.
Note that $\theta \in \mathbb{R}$ lacks a constraint to guarantee stationarity (unlike in the AR(1) model).

Note that the dependence structure is based on unobservable cases Z_t, which are analogous to error.
Thus we require significant statistical machinery to estimate θ, and forecast X_t (unlike in the AR(1) model).

Example: HW 1 #6 - Characterizing a Product of IID Noise

Let $\{X_t : t \in \mathbb{Z}\}$ be a time series defined by

$$X_t = Z_t Z_{t-1}, \quad Z_t \sim IID(0, \sigma^2)$$

a) Derive the mean function $\mu_X(t)$ of $\{X_t : t \in \mathbb{Z}\}$.

$$\mu_X(t) = E[X_t] = E[Z_t Z_{t-1}] = Cov(Z_t, Z_{t-1}) = 0$$

b) Derive the covariance function $\gamma_X(t+h, t)$ of $\{X_t : t \in \mathbb{Z}\}$.

$$\gamma_X(t+h, t) = Cov(X_{t+h}, X_t) = E[X_{t+h} X_t] = E[Z_{t+h} Z_{t+h-1} Z_t Z_{t-1}]$$

For $h = -2$,

$$E[Z_{t-2} Z_{t-3} Z_t Z_{t-1}] \overset{iid}{=} E[Z_{t-2}] E[Z_{t-3}] E[Z_t] E[Z_{t-1}] = 0$$

For $h = -1$,

$$E[Z_{t-1} Z_{t-2} Z_t Z_{t-1}] \overset{iid}{=} E[Z_{t-1}^2] E[Z_{t-2}] E[Z_t] = \sigma^2(0)(0) = 0$$

For $h = 0$,

$$E[Z_t Z_{t-1} Z_t Z_{t-1}] = E[Z_t^2] E[Z_{t-1}^2] = (\sigma^2)(\sigma^2) = \sigma^4$$

For $h = 1$,

$$E[Z_{t+1} Z_t Z_t Z_{t-1}] = E[Z_t^2] E[Z_{t+1}] E[Z_{t-1}] = (\sigma^2)(0)(0) = 0$$

For $h = 2$,

$$E[Z_{t+2} Z_{t+1} Z_t Z_{t-1}] = E[Z_{t+2}] E[Z_{t+1}] E[Z_t] E[Z_{t-1}] = 0$$

Note that for $|h| > 2$, all expectations will be 0.

$$\gamma_X(t+h, t) = \gamma_X(h) = \begin{cases} \sigma^4 & if\ h = 0 \\ 0 & otherwise \end{cases}$$

c) Is $\{X_t : t \in \mathbb{Z}\}$ a stationary time series?

$\{X_t : t \in \mathbb{Z}\}$ is stationary because μ_X and $\gamma_X(t+h, t)$ are constant in h.

Example: Differencing - Random Walk

Let $X_t = \sum_{i=1}^{t} Z_i$, $Z_i \sim WN(0, \sigma^2)$.

Recall $\gamma_X(t+h, t) = t\sigma^2$, so the random walk process is not stationary.

Let $Y_t = \nabla X_t$.

$$Y_t = \nabla X_t = X_t - X_{t-1} = \sum_{i=1}^{t} Z_i - \sum_{i=1}^{t-1} Z_i = Z_t$$

Applying the difference operator on X_t returns only the WN error structure.

$$\gamma_Y(h) = \begin{cases} \sigma^2 & if\ h = 0 \\ 0 & if\ h \neq 0 \end{cases}$$

Thus Y_t is a stationary series, but we cannot exploit a WN error structure.

Suppose we would like to forecast a random walk, X_{t+1}.
Because we cannot exploit the WN structure, our forecast is the previous case.

$$\hat{X}_{t+1} = X_t$$

Example: De-Trended Time Series - US Population

Suppose that the US Population is given by the time series $X_t = m_t + Y_t$.

$$Y_t \approx X_t - m_t$$

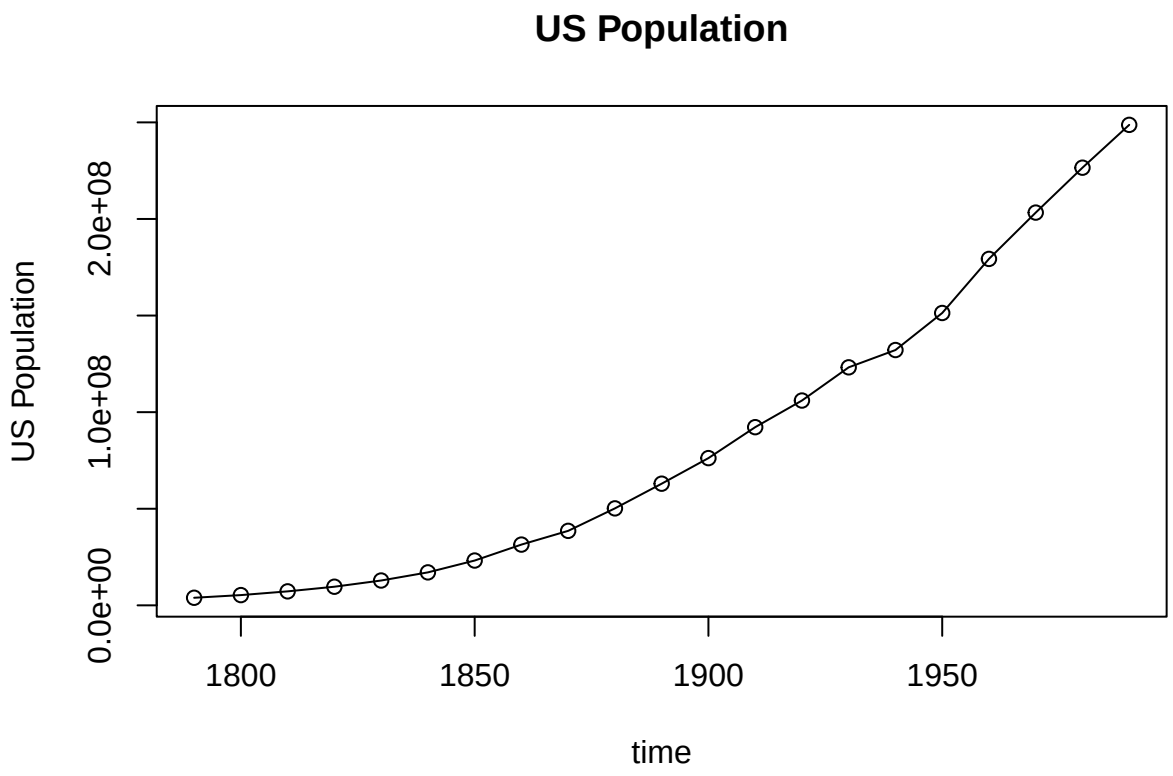
Assume the trend is given by $m_t = \beta_0 + \beta_1 t + \beta_2 t^2$.
Therefore, we can estimate the trend using OLS.

$$\arg\min_{\beta} \sum_{t=1}^{n} (X_t - \hat{m}_t)^2$$

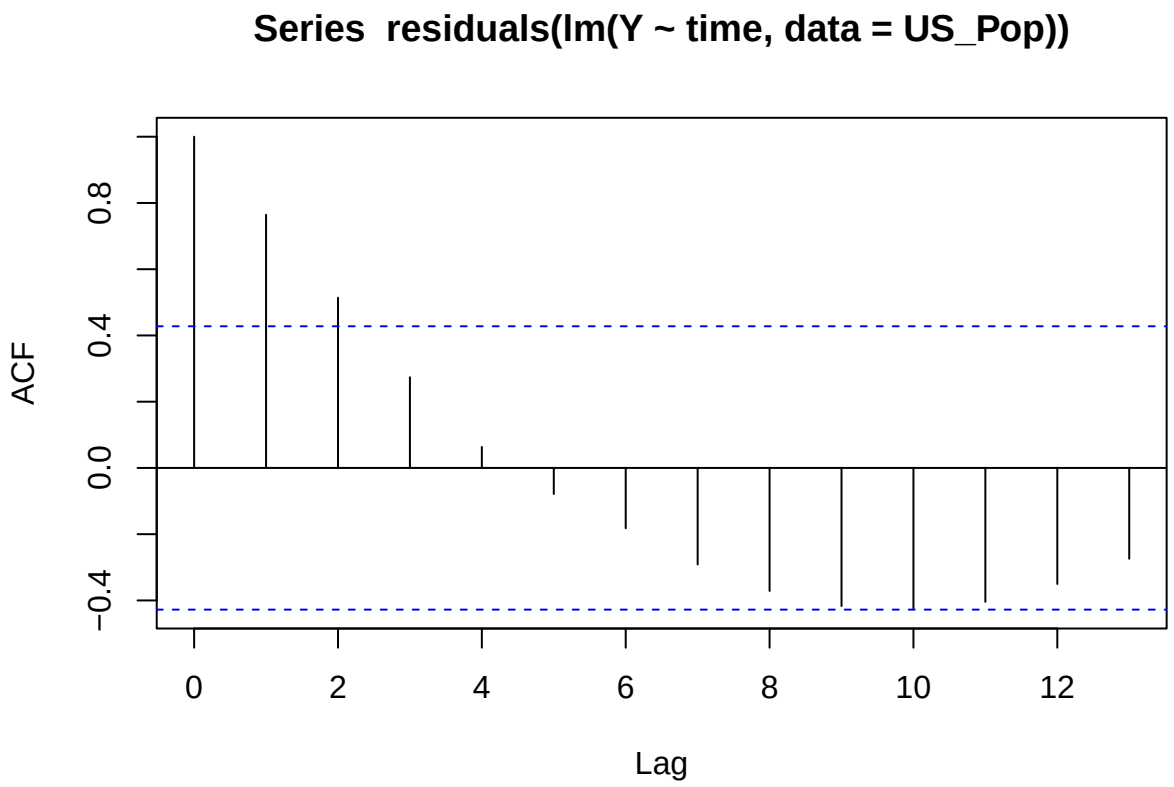
The de-trended time series $\hat{Y}_t$ is therefore given by the residuals of the line of best fit.

Does $\hat{Y}_t \approx X_t - \hat{m}_t$ resemble white noise (WN)?
If $\hat{Y}_t$ resembles WN, there is no need for a time series model - the regression model is sufficient.
Yes, $\hat{Y}_t$ resembles WN or iid noise (based on the ACF plot) - although there is some periodicity/seasonality.

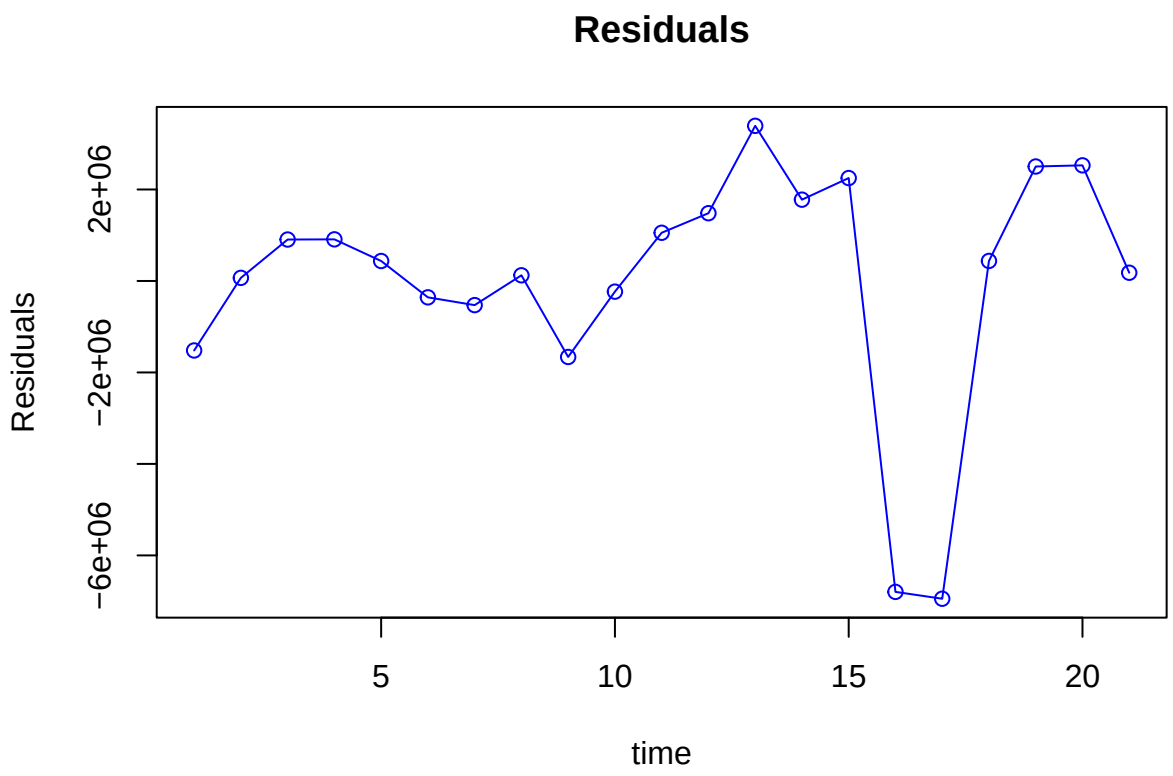
```
US_Pop = read.csv("US_Population.csv")

# Plot raw time series X_t
# The data clearly has an exponential trend
plot(US_Pop$time,
     US_Pop$Y,
     type = "o",
     main = "US Population",
     xlab = "time",
     ylab = "US Population")
```



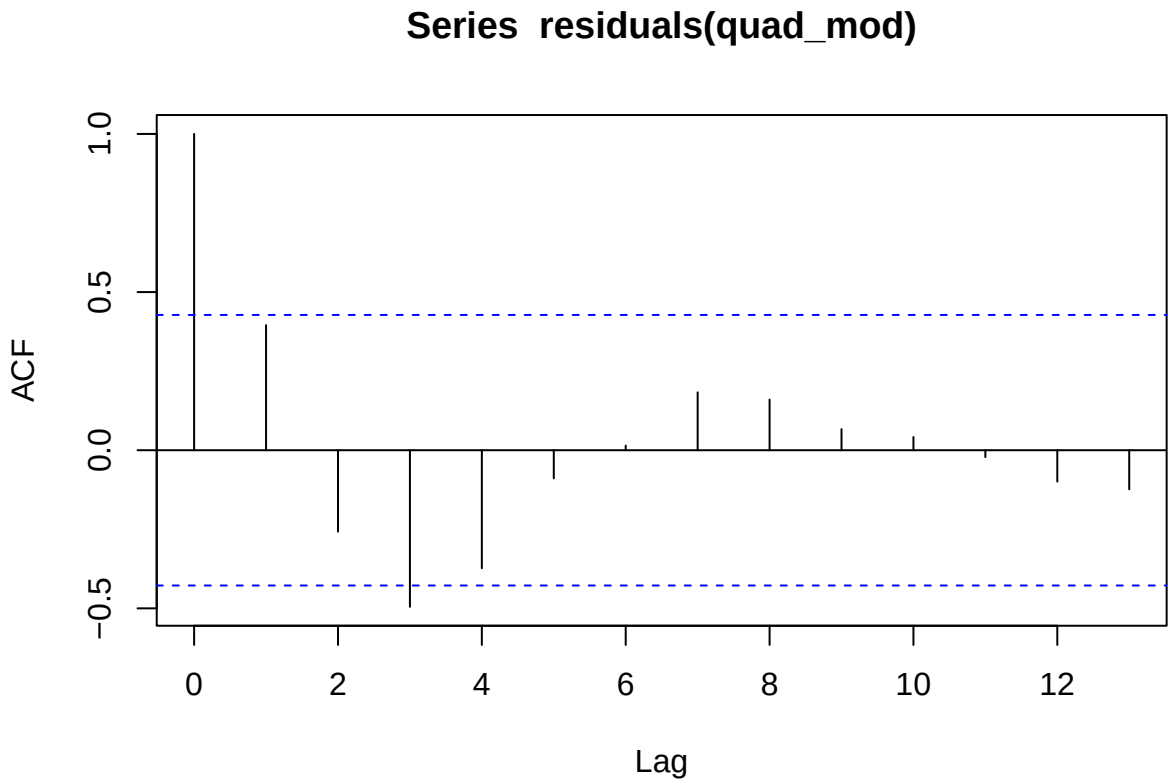
```r
# ACF of raw time series X_t
# There is periodic structure and many lags cross the confidence bands
acf(residuals(lm(Y~time,
                 data = US_Pop)))
```




```r
# For convenience, let us de-trend using a quadratic regression model
# hat(m)_t
quad_mod = lm(Y~poly(time,degree = 2,raw = T),data=US_Pop)

# Y_t = X_t - hat(m)_t
plot(residuals(quad_mod),type="o",col="blue",
     main = "Residuals", xlab = "time", ylab = "Residuals")
```




```
# Plot ACF for residuals (de-trended time series Y_t)
# ACF looks like iid noise after de-trending (no TS model needed)
acf(residuals(quad_mod))
```



Example: De-Trended Time Series - Lake Huron 1875-1972

Suppose that the level of Lake Huron is given by the time series $X_t = m_t + Y_t$.

$$Y_t \approx X_t - m_t$$

Assume the trend is given by $m_t = \beta_0 + \beta_1 t$.
Therefore, we can estimate the trend using ordinary least squares (OLS, squared loss).

$$\arg\min_{\beta} \sum_{t=1}^{n} (X_t - \hat{m}_t)^2$$

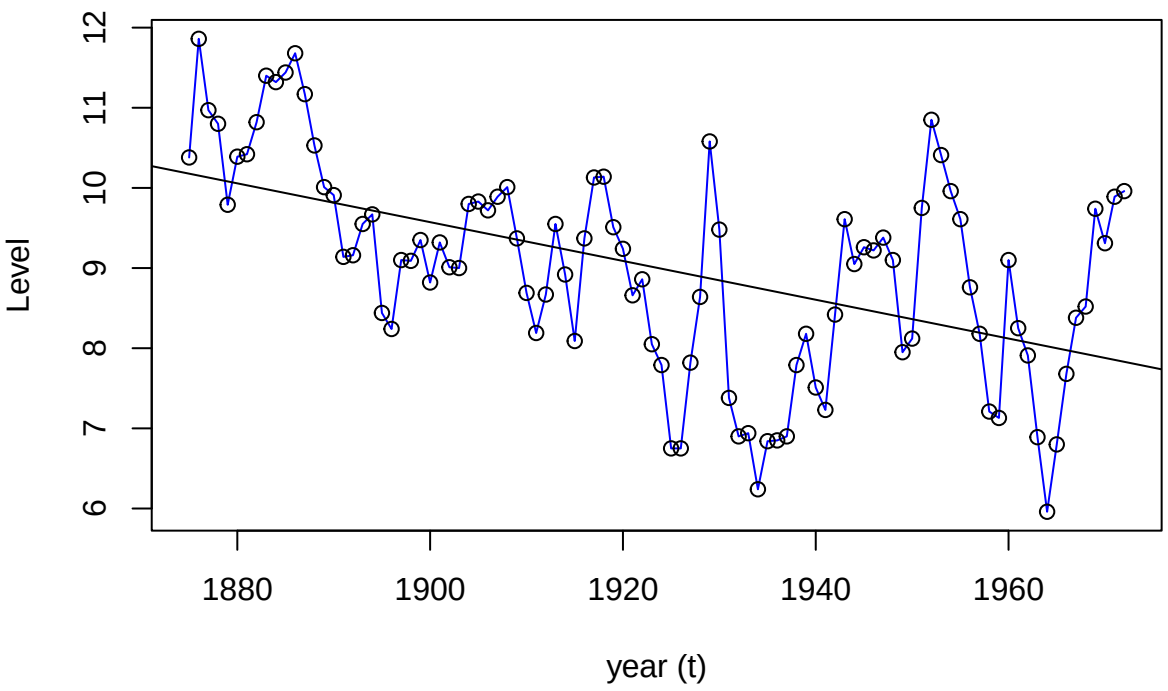
The de-trended time series $\hat{Y}_t$ is therefore given by the residuals of the line of best fit.

Does $\hat{Y}_t \approx X_t - \hat{m}_t$ resemble white noise (WN)?
If $\hat{Y}_t$ resembles WN, there is no need for a time series model - the regression model is sufficient.

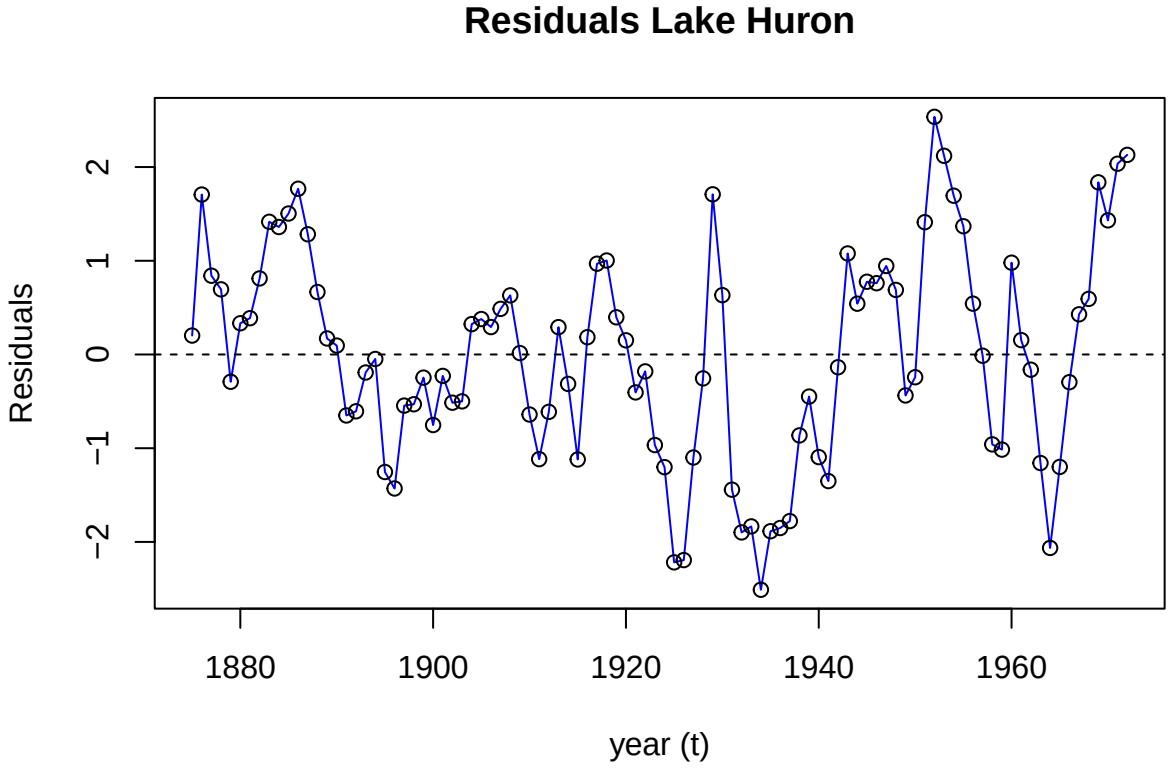
No, the de-trended Lake Huron data still demonstrates temporal behavior.
Applying an AR(1) model to the de-trended data yields WN.
To forecast this, we would forecast the AR(1) model and then add the trend.

Note that the AR(2) model yields even more improvement.

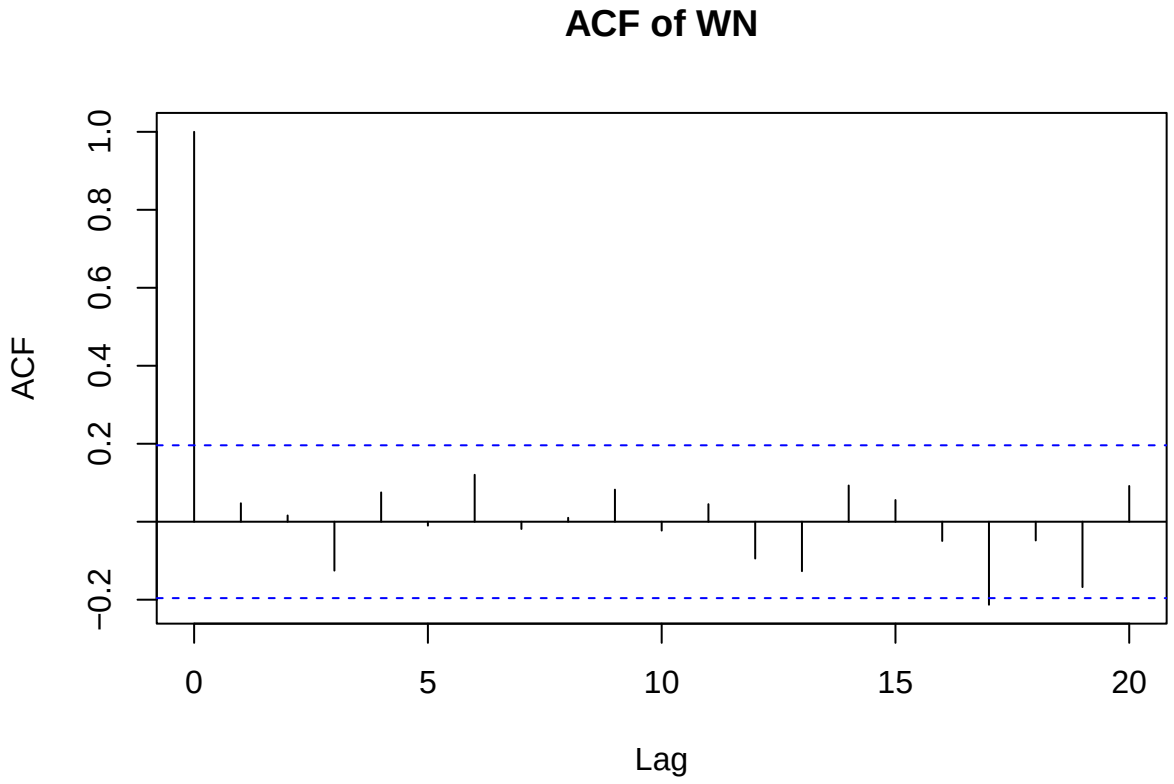
```
Lake_Huron = read.csv("Lake_Huron.csv")

# Plot raw time series X_t
plot(Lake_Huron$time, Lake_Huron$Y,
     type="l", col="blue",
     main="Level of Lake Huron",
     xlab="year (t)",
     ylab="Level")

points(Lake_Huron$time,Lake_Huron$Y,
       pch=1)

# fit linear trend hat(m)_t
lm_lake = lm(Y~time,data=Lake_Huron)
abline(lm_lake)
```



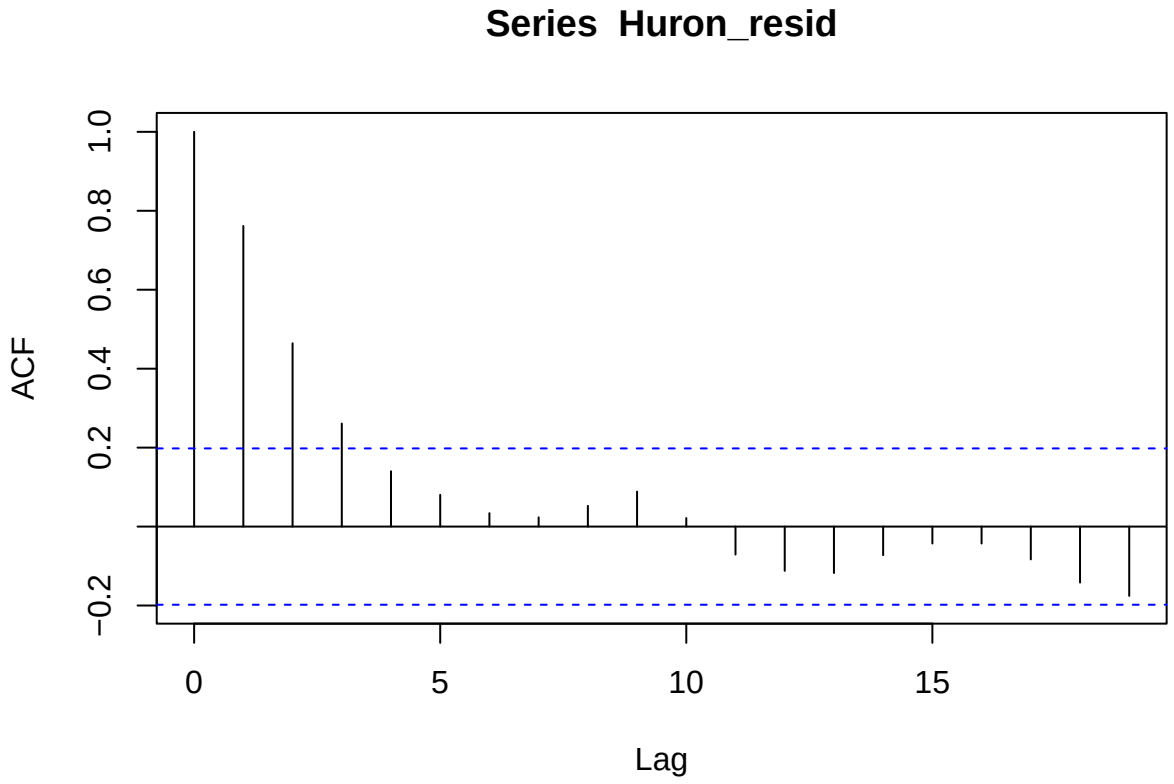

```
# define residuals - de-trended time series Y_t = X_t = hat(m)_t
Huron_resid = residuals(lm_lake)

# Plot residuals (de-trended time series Y_t) - trend is made the horizontal axis
plot(Lake_Huron$time,Huron_resid,type="l",col="blue",
     main="Residuals Lake Huron",xlab="year (t)",ylab="Residuals")
points(Lake_Huron$time,Huron_resid,pch=1)
abline(h=0,lty=2)
```



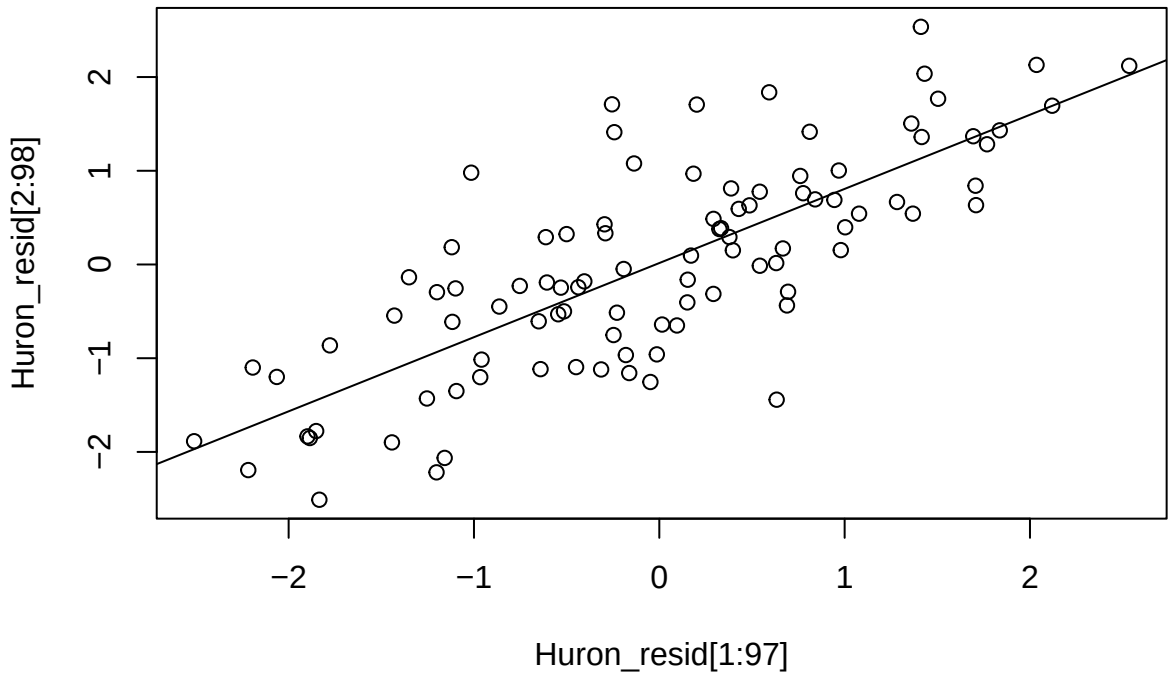

```
# ACF for iid (stronger than WN assumption)

# Lag 0 correlation is 1, because this is how the data is correlated with itself.
# The correlation for different lags is within the confidence bands.

# For 5% significance, approximately no more than one case out of 20 should fall outside confidence bands
# This is why 20 lags are displayed

acf(rnorm(100), main="ACF of WN")
```



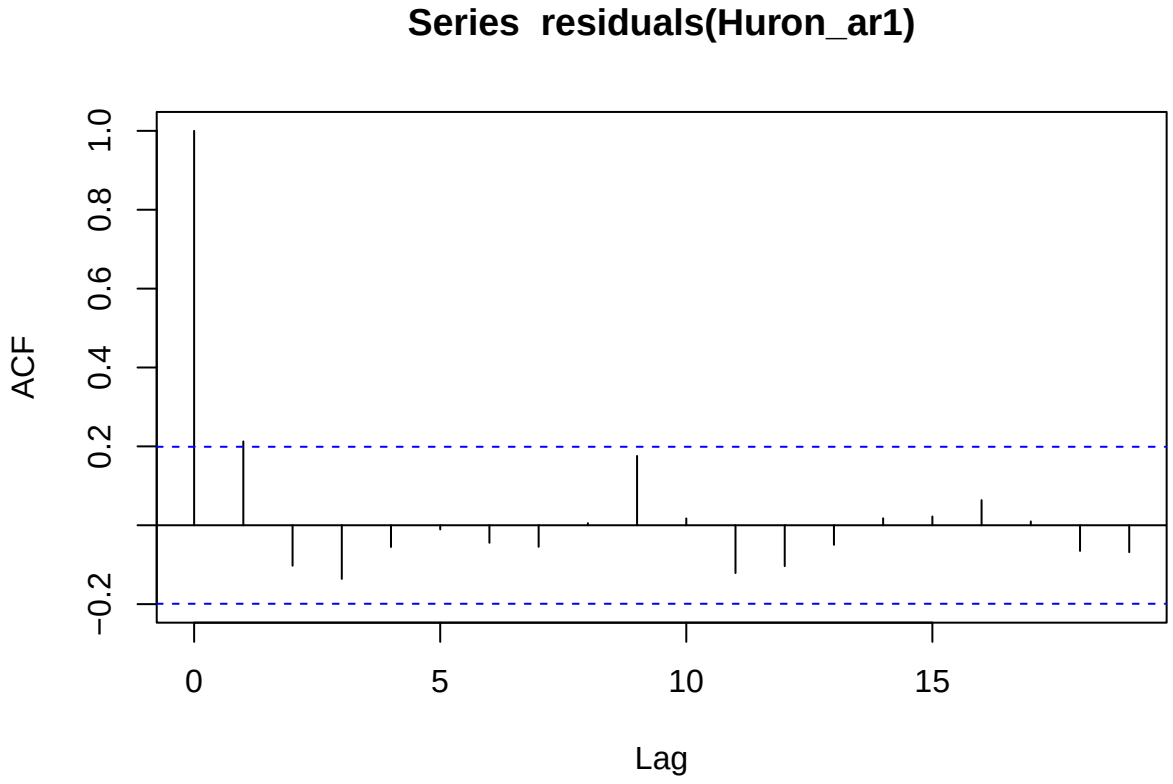

```
# ACF for residuals (de-trended time series Y_t)
# ACF for de-trended data clearly exhibits temporal behavior
# While it falls within the confidence bands for increasing lag (and could be modeled with an ARMA model),
# there is clear periodicity (that is apparent in the original dataset as well)
acf(Huron_resid)
```




```
# Since we lack the tools to fit a more sophisticated model, we will fit an AR(1) model
# Find the length of the dataset (n=98) to structure AR(1) model
n = length(Lake_Huron$Y)

# We fit an estimated AR(1) to the de-trended data using regression - lm(Y[2:n] ~ Y[1:(n-1)])
# Y_t= phi Y_{t-1} + Z_t
plot(Huron_resid[1:97],Huron_resid[2:98])
Huron_ar1 = lm(Huron_resid[2:98]~Huron_resid[1:97])
abline(Huron_ar1)
```




```
# Z_t = Y_t - phi Y_{t-1}
# Residuals of AR(1) should resemble WN - and it does
# ACF of residuals of AR(1)
acf(residuals(Huron_ar1))
```



```
# Try AR(2)  (multiple regression) on residuals - improves acf of de-trended data
# Y_t = phi_1 Y_{t-1} + phi_2 Y_{t-2} + Z_t
Huron_ar2 = lm(Huron_resid[3:98]~Huron_resid[2:97]+Huron_resid[1:96])

# Z_t = Y_t - phi_1 Y_{t-1} - phi_2 Y_{t-2}
acf(residuals(Huron_ar2))
```


```
# However note that the first coefficient is showing stationarity for non-causal behavior,
# or non-stationarityy for causal behavior (phi1_est = 1.002137 <1? >1? approximately 1?)
# Perhaps a random walk is dominating this time series (because 1 is a reasonable value for phi1_est)
# Note that the variance increases over time - this is another indication of a random walk
# While an AR(2) structure produced iid erros, we may have unreliable estimates
Huron_ar2
```

```
##
## Call:
## lm(formula = Huron_resid[3:98] ~ Huron_resid[2:97] + Huron_resid[1:96])
##
## Coefficients:
##       (Intercept)  Huron_resid[2:97]  Huron_resid[1:96]
##         -0.007852           1.002137          -0.283798
```

```
confint(Huron_ar2)
```

```
##                         2.5 %      97.5 %
## (Intercept)       -0.1451124  0.12940879
## Huron_resid[2:97]  0.8090870  1.19518727
## Huron_resid[1:96] -0.4804011 -0.08719571
```

Example: De-Seasonalized Time Series - Accidental Deaths

The accidental deaths dataset has a clear periodic structure.

We will assume a parametric (linear combination of sigmoidal functions) structure on the seasonality.
The two models that will be considered are as follows.

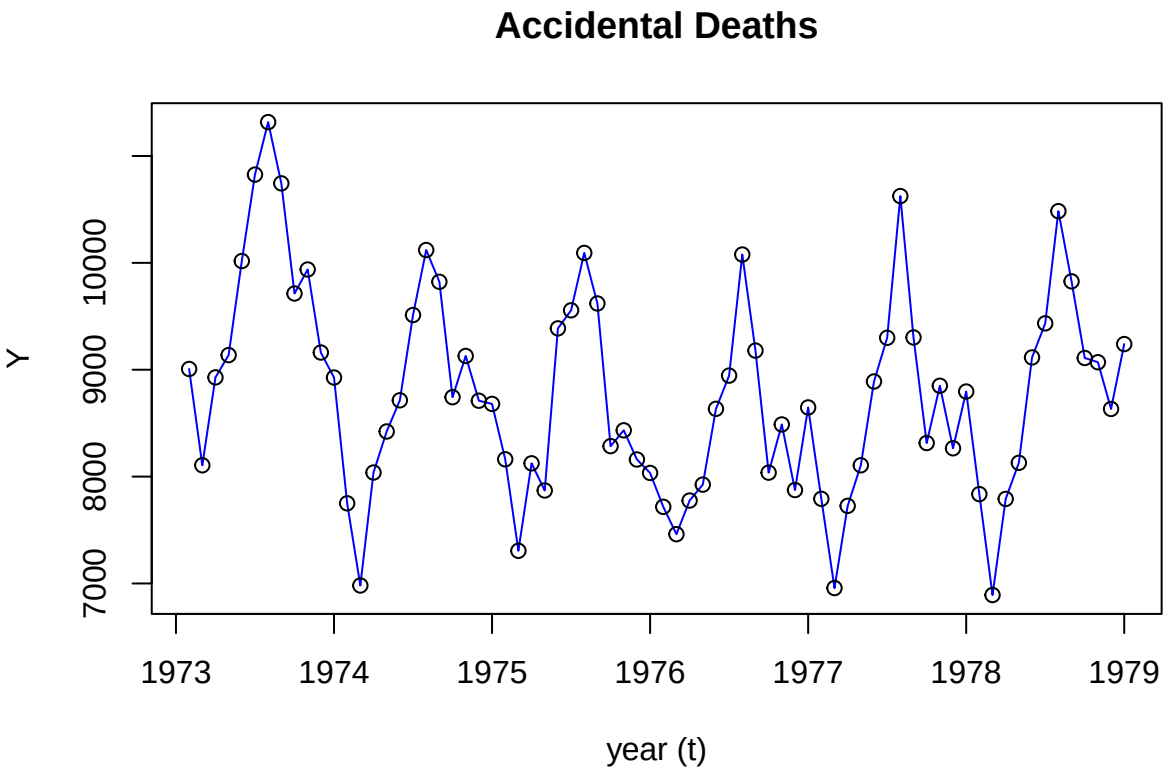
$$s_t = \alpha_0 + \alpha_1 \cos(6t) + \beta_1 \sin(6t)$$
$$s_t = \alpha_0 + \alpha_1 \cos(6t) + \beta_1 \sin(6t) + \alpha_2 \cos(12t) + \beta_2 \sin(12t)$$

The α and β terms will be estimated using least squares (OLS, squared loss) harmonic regression.

$$\arg\min_{\alpha}(X_t - \hat{s}_t)^2$$

Where α is the parameter vector that includes the α and β terms.
The de-seasonalized time series $\hat{Y}_t$ is therefore given by the residuals of the line of best fit.

```
AD = read.csv("Accidental_Deaths.csv")

# Plot raw time series X_t
# There is a clear periodic structure
plot(AD$time, AD$Y,
     type="l", col="blue",
     main="Accidental Deaths", ylab="Y", xlab="year (t)")

points(AD$time,AD$Y)
```




```
# 72 observations
length(AD$Y)
```

```
## [1] 72
```

```r
# To guess frequency - try to match the length of a single period
# Note that 2*pi/6 yields 1.04 which is approximately the length of a single period
72/12
```

```
## [1] 6
```

```r
72/6
```

```
## [1] 12
```

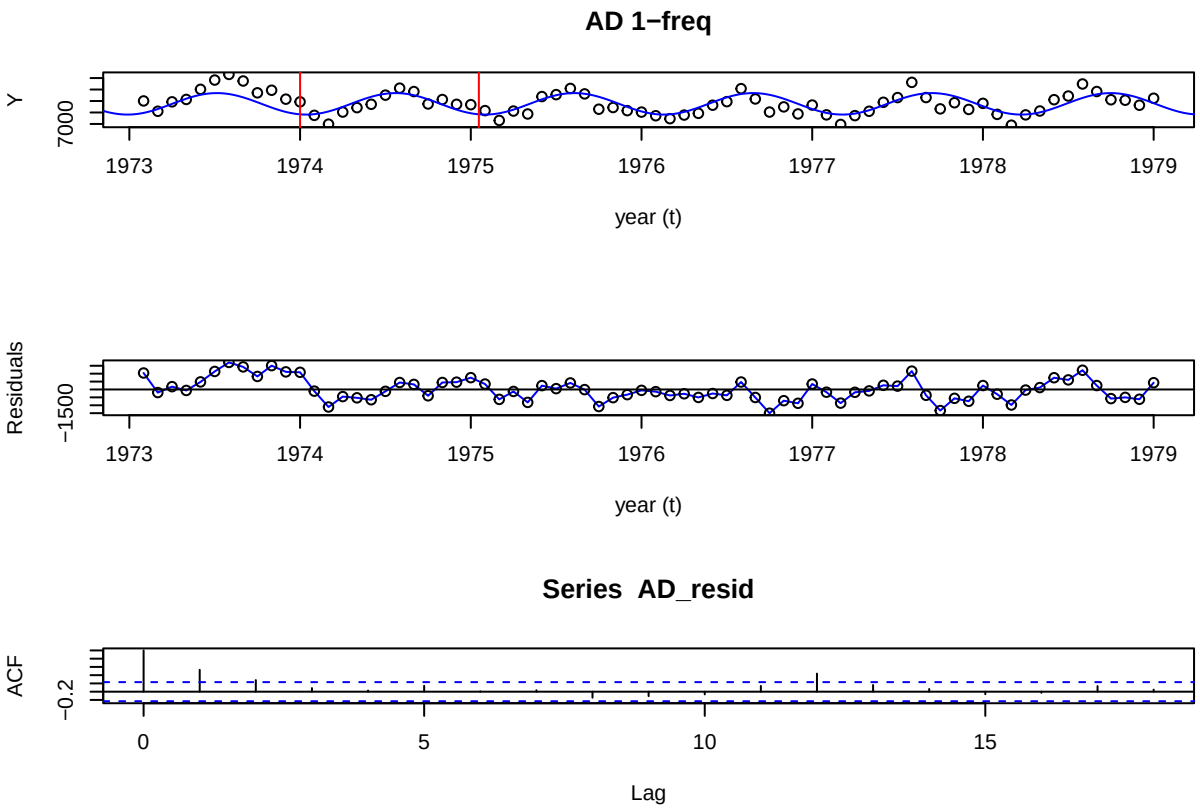
```r
2*pi/12
```

```
## [1] 0.5235988
```

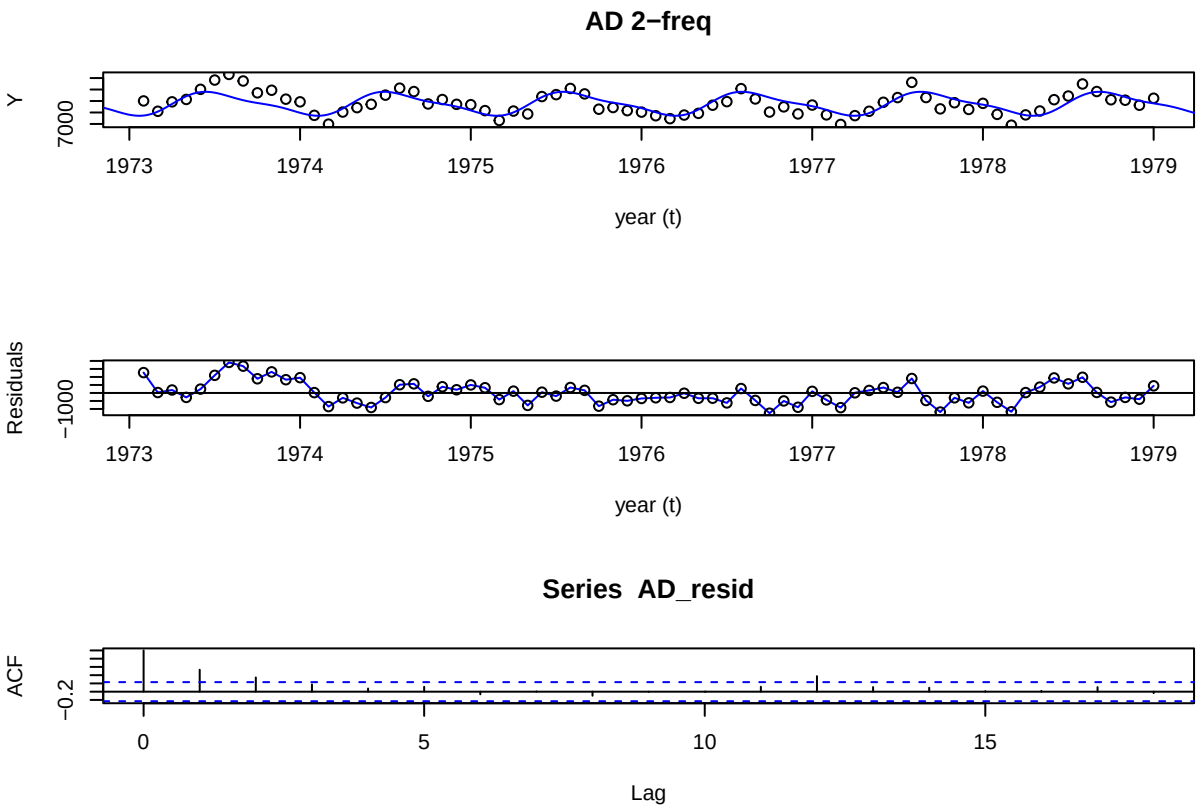
```r
2*pi/6
```

```
## [1] 1.047198
```

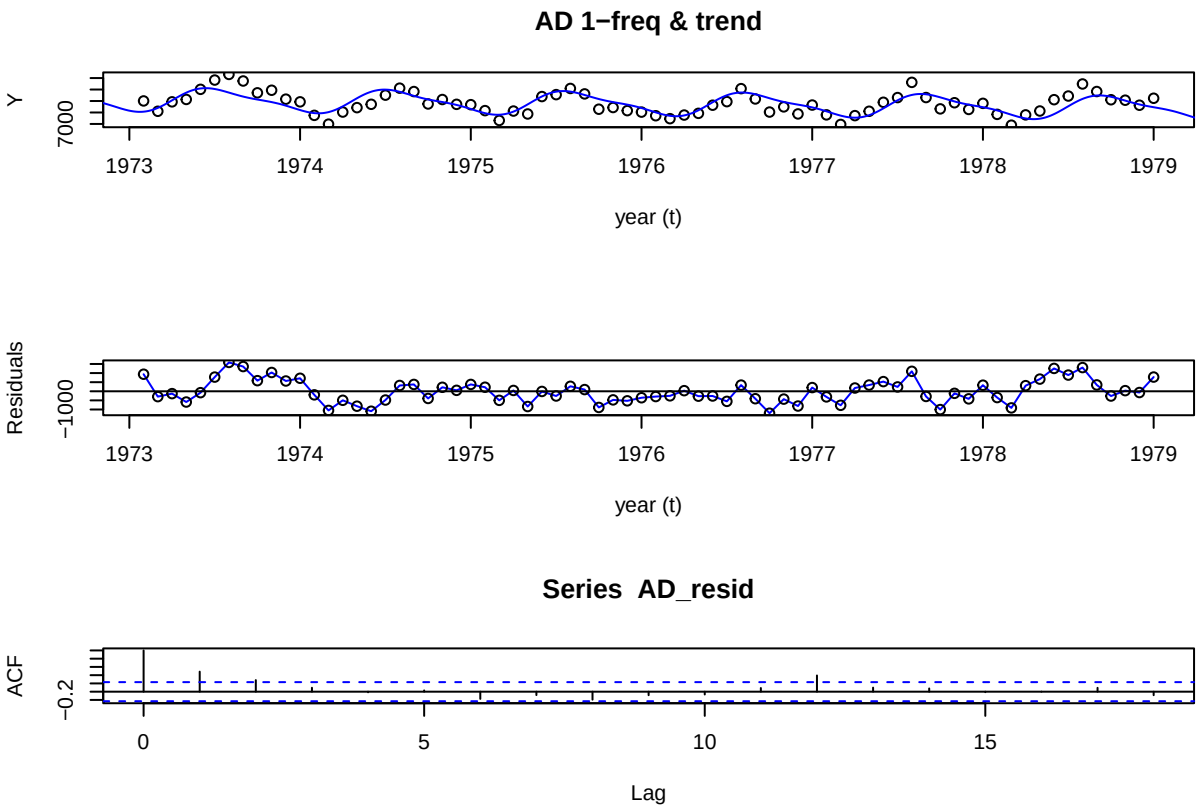
```r
#par(mfrow=c(1,1))
par(mfrow=c(3,1))

# fit harmonic regression model hat{s}_t
# Recall I() allows you to apply functions and create variables on the fly
# Plot sigmoidal function using predict() function on estimated hat{s}_t model
harmonic_model = lm(Y~I(sin(6*time))+I(cos(6*time)),data=AD)

plot_time = seq(1970,1980,length=1000)
plot_y =  predict(harmonic_model,
                  newdata = data.frame(time=plot_time))

plot(AD$time, AD$Y,
     main="AD 1-freq", ylab="Y", xlab="year (t)")
lines(plot_time,plot_y,
      col="blue")
abline(v=c(1974, 1974+2*pi/6), col="red")

# define residuals (de-seasonalized time series Y_t = X_t - hat(s)_t)
# Plot of residuals does not resemble white noise (perhaps we could use ARMA model)
# ACF is not too bad - there is some temporal dependence and a lag at 12 (due to yearly seasonality)
AD_resid = residuals(harmonic_model)
plot(AD$time, AD_resid,
     type="l", col="blue",
     ylab="Residuals", xlab="year (t)")
points(AD$time,AD_resid)
abline(h=0)
acf(AD_resid)
```




```r
# fit second harmonic regression model hat{s}_t
# Recall I() allows you to apply functions and create variables on the fly
# Plot sigmoidal function using predict() function on estimated hat{s}_t model
harmonic_model = lm(Y~I(sin(12*time))+I(cos(12*time))+I(sin(6*time))+I(cos(6*time)),data=AD)

plot_time = seq(1970,1980,length=1000)
plot_y =  predict(harmonic_model,
                  newdata = data.frame(time=plot_time))

plot(AD$time, AD$Y,
     main="AD 2-freq", ylab="Y", xlab="year (t)")
lines(plot_time,plot_y,col="blue")

# define residuals (de-seasonalized time series Y_t = X_t - hat(s)_t)
# Plot of residuals does not resemble white noise
# ACF is similar to the previous
AD_resid = residuals(harmonic_model)
plot(AD$time, AD_resid,
     type="l", col="blue", ylab="Residuals", xlab="year (t)")
points(AD$time,AD_resid)
abline(h=0)
acf(AD_resid)
```




```r
# Looking at the raw data - there appears to be a linear trend
# Define harmonic model + trend by introducing time to the harmonic regression model hat{s}_t
# Plot sigmoidal function using predict() function on estimated hat{s}_t + hat{m}_t model
harmonic_model = lm(Y~time+I(sin(12*time))+I(cos(12*time))+I(sin(6*time))+I(cos(6*time)),data=AD)

plot_time = seq(1970,1980,length=1000)
plot_y =  predict(harmonic_model,
                  newdata = data.frame(time=plot_time))

plot(AD$time, AD$Y,
     main="AD 1-freq & trend", ylab="Y", xlab="year (t)")
lines(plot_time,plot_y,col="blue")

# define residuals (de-trended and de-seasonalized time series Y_t)
# Plot of residuals does not resemble white noise
# ACF is similar to the previous
AD_resid = residuals(harmonic_model)
plot(AD$time,AD_resid,type="l",col="blue",ylab="Residuals",xlab="year (t)")
points(AD$time,AD_resid)
abline(h=0)
acf(AD_resid)
```




```
# Smoothers could be used to de-trend and de-seasonalize, though they are less interpretable
par(mfrow=c(1,1))
```

Example: HW 1 #4 - Removing Seasonality and Trend from Time Series

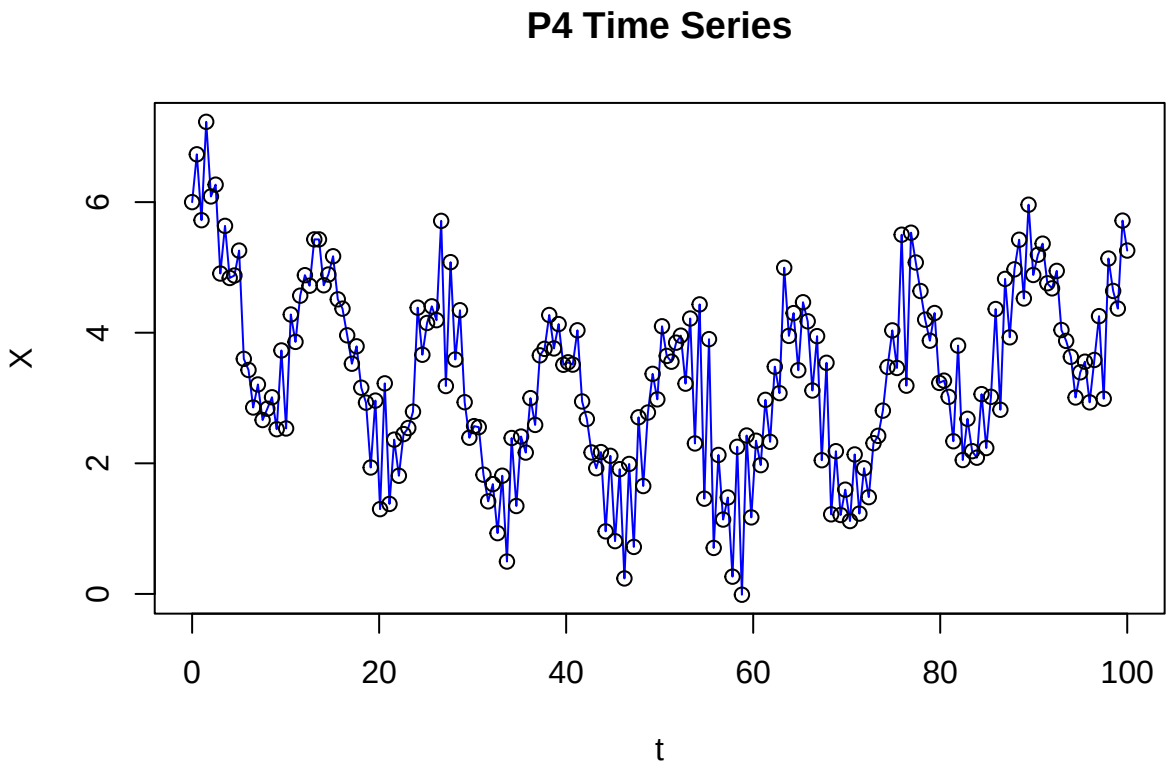
Consider the CSV file `HW1_problem4.csv`, which includes the realized time series $\{x_t : t = 1, 2, \ldots, 200\}$.
Assume the classical decomposition

$$X_t = m_t + s_t + Y_t$$

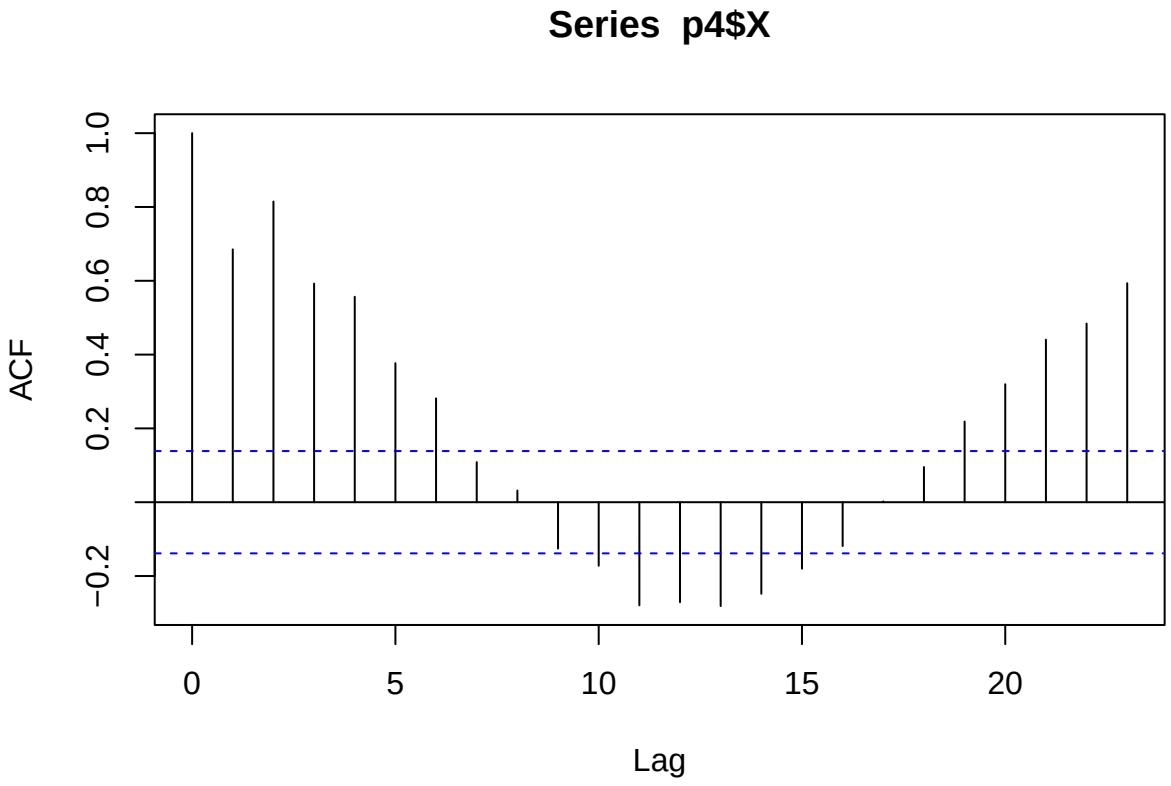
where m_t is the deterministic trend, s_t is the deterministic seasonal component, and $\{Y_t\}$ is the de-trended and de-seasonalized time series.

a) Display the raw time series $\{x_t : t = 1, 2, \ldots, 200\}$ in a scatterplot and plot its sample ACF.
Do you see any notable features based on these two plots?

```
p4 = read.csv("HW1_problem4.csv")

# Plot raw time series X_t
# There is a clear periodic structure
plot(p4$t, p4$X,
     type="l", col="blue",
     main="P4 Time Series", ylab="X", xlab="t")

points(p4$t,p4$X)
```




```
acf(p4$X)
```



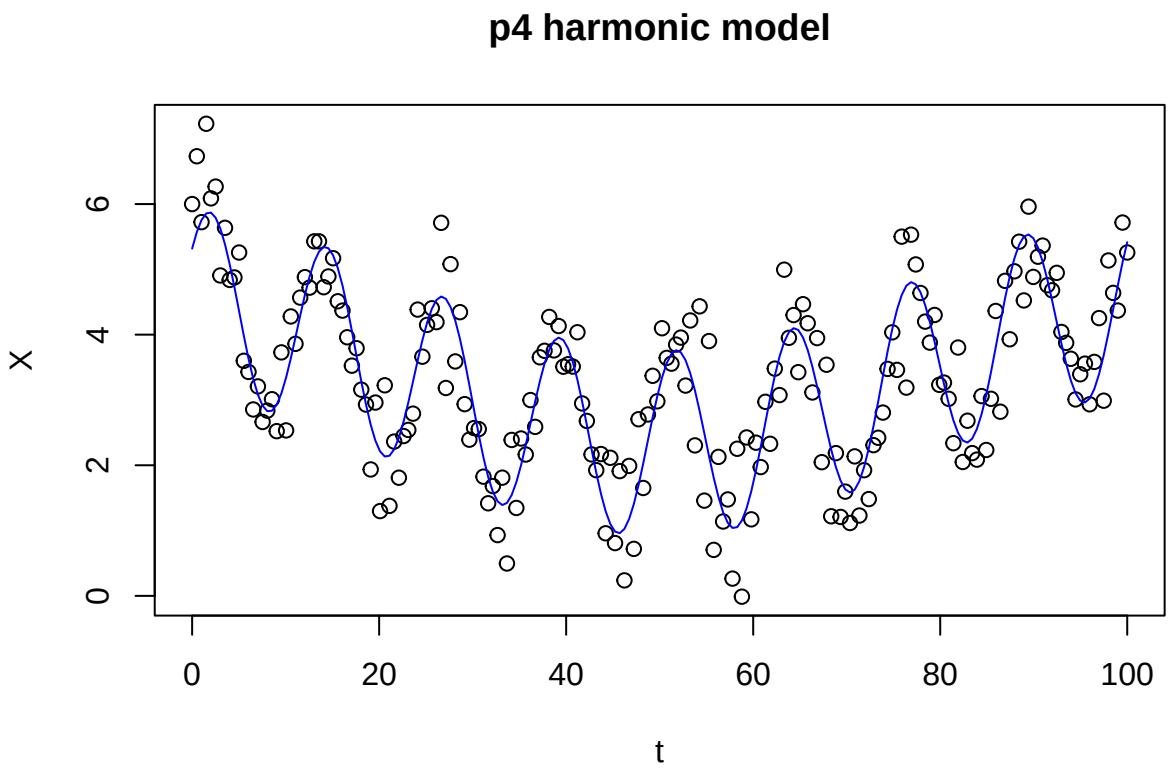
The time series demonstrates periodicity of 10-12 time units, a U-shaped structure over time.

The sample ACF plot demonstrates temporal behavior with periodic lag dependence roughly every 8 lags. Additionally, the ACF plot is not decreasing smoothly and regularly exceeds the confidence bands.

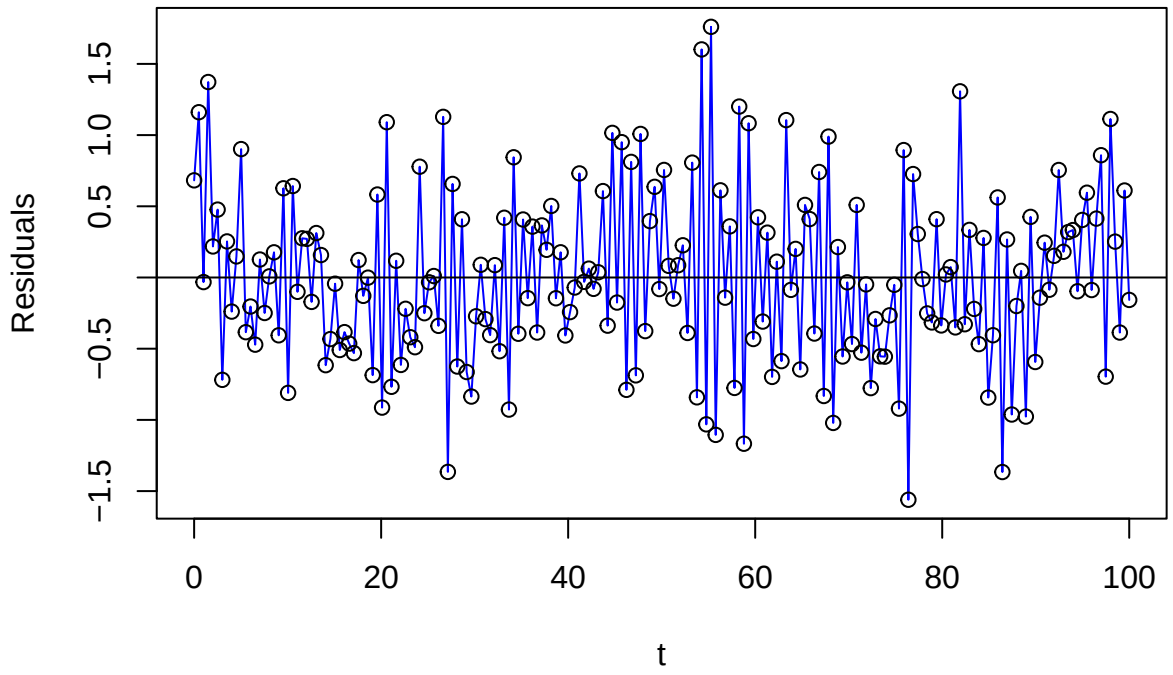
To accommodate this highly periodic structure we could attempt harmonic regression.

b) De-trend and de-seasonalize $\{x_t\}$, i.e., $y_t \approx x_t - \hat{m}_t - \hat{s}_t$.
Show your resulting residual plot of y_t and its sample ACF.
Does this plot show an iid error structure and why?

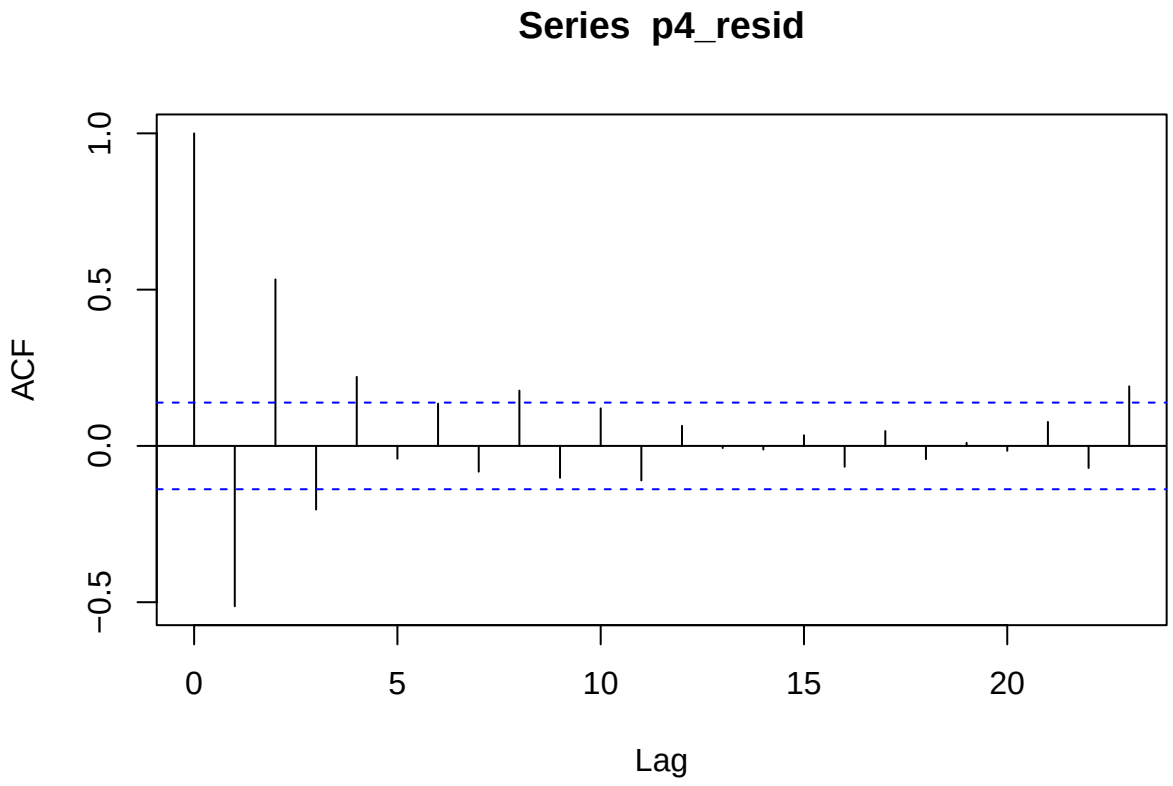
```
# fit harmonic regression model hat{m}_t + hat{s}_t
# Recall I() allows you to apply functions and create variables on the fly
# Plot sigmoidal function using predict() function on estimated hat{m}_t + hat{s}_t model
harmonic_model = lm(p4$X ~ I(cos(12*p4$t)) + I(sin(12*p4$t)) + I(cos(200*p4$t)) + I(sin(200*p4$t)) )

plot_t = seq(0,100,length=200)
plot_x =  predict(harmonic_model,
                  newdata = data.frame(time=plot_t))

plot(p4$t, p4$X,
     main="p4 harmonic model", ylab="X", xlab="t")

lines(plot_t,plot_x,
      col="blue")
```




```
# define residuals (de-trended and de-seasonalized time series Y_t = X_t - (hat(m)_t + hat(s)_t) )
# Plot of residuals does not resemble white noise (perhaps we could use ARMA model)
p4_resid = residuals(harmonic_model)
plot(p4$t, p4_resid,
     type="l", col="blue",
     ylab="Residuals", xlab="t")
points(p4$t,p4_resid)
abline(h=0)
```




```
acf(p4_resid)
```



The residual plot and ACF plot do not demonstrate IID structure.
There is still some level of temporal dependence, notably among the first 5 lags (and occassionally beyond this as well),

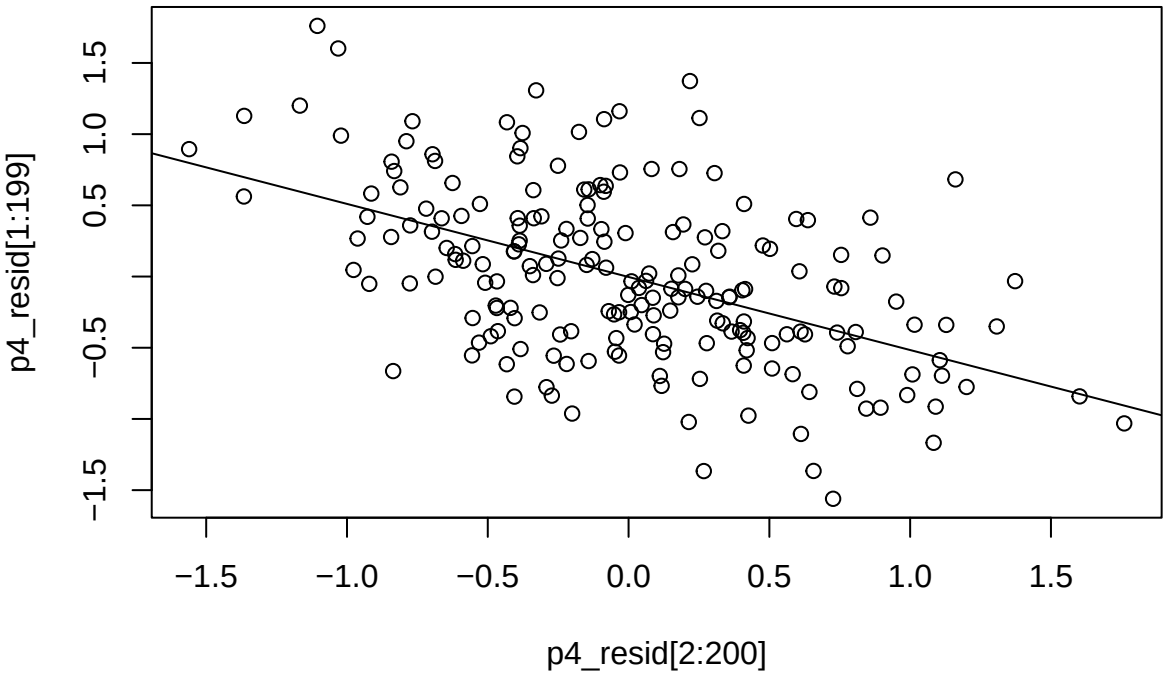
but overall the correlation structure is generally decreasing over the lags.

c) Assume that $\{Y_t\}$ follows an AR(1) process, i.e.,

$$Y_t = \phi Y_{t-1} + Z_t$$

where $Z_i \overset{iid}{\sim} WN(0, \sigma^2)$, $|\phi| \leq 1$, and $X_0 = 0$.
Compute the ordinary least squares (OLS) estimator of ϕ by regressing `Y[2:200]~Y[1:199]`.
What is the estimated AR(1) coefficient $\hat{\phi}$?

```
# Y_t= phi Y_{t-1} + Z_t
plot(p4_resid[2:200],p4_resid[1:199])
p4_ar1 = lm(p4_resid[2:200]~p4_resid[1:199])
abline(p4_ar1)
```




```
phi_est = round(coef(p4_ar1),3)
phi_est
```

```
##     (Intercept) p4_resid[1:199]
##          -0.003          -0.513
```

$$\hat{\phi} = -0.513$$

d) Estimate the noise variance σ^2 using techniques from Linear Regression.

```
x = p4$X[2:200]
x_bar = mean(x)
x_hat = fitted(p4_ar1)
SSE = sum((x-x_hat)^2)
```

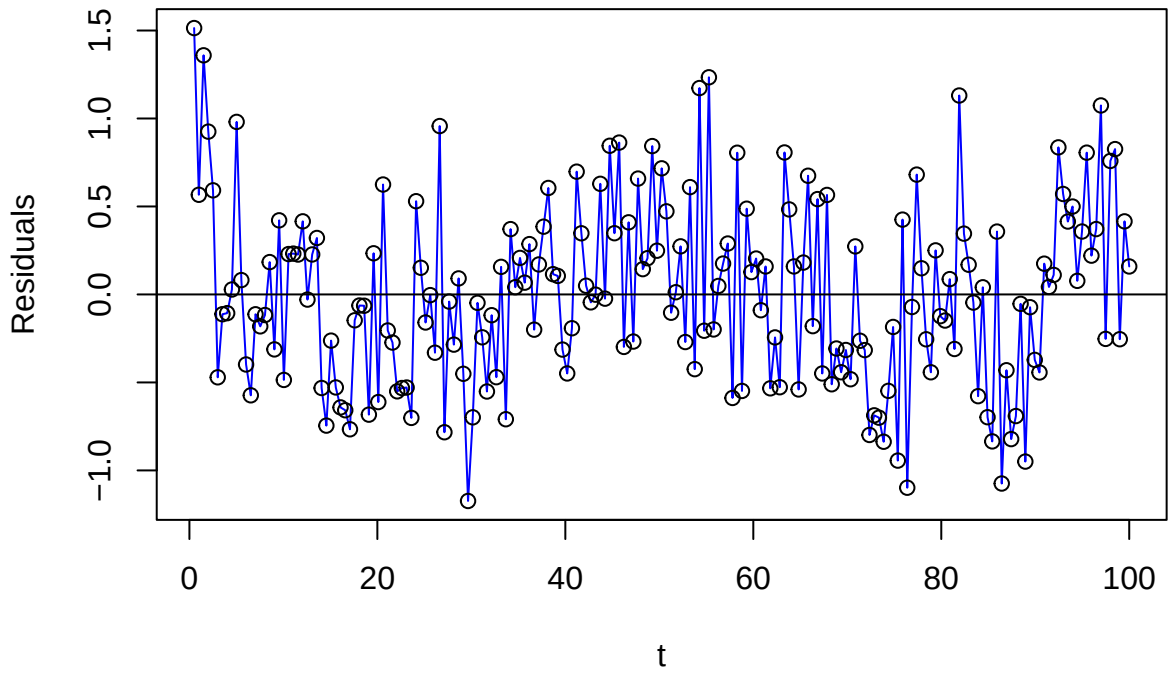
```
n = 200-2+1
p = 2

MSE = (SSE)/(n-p)
MSE
```

```
## [1] 13.00702
```

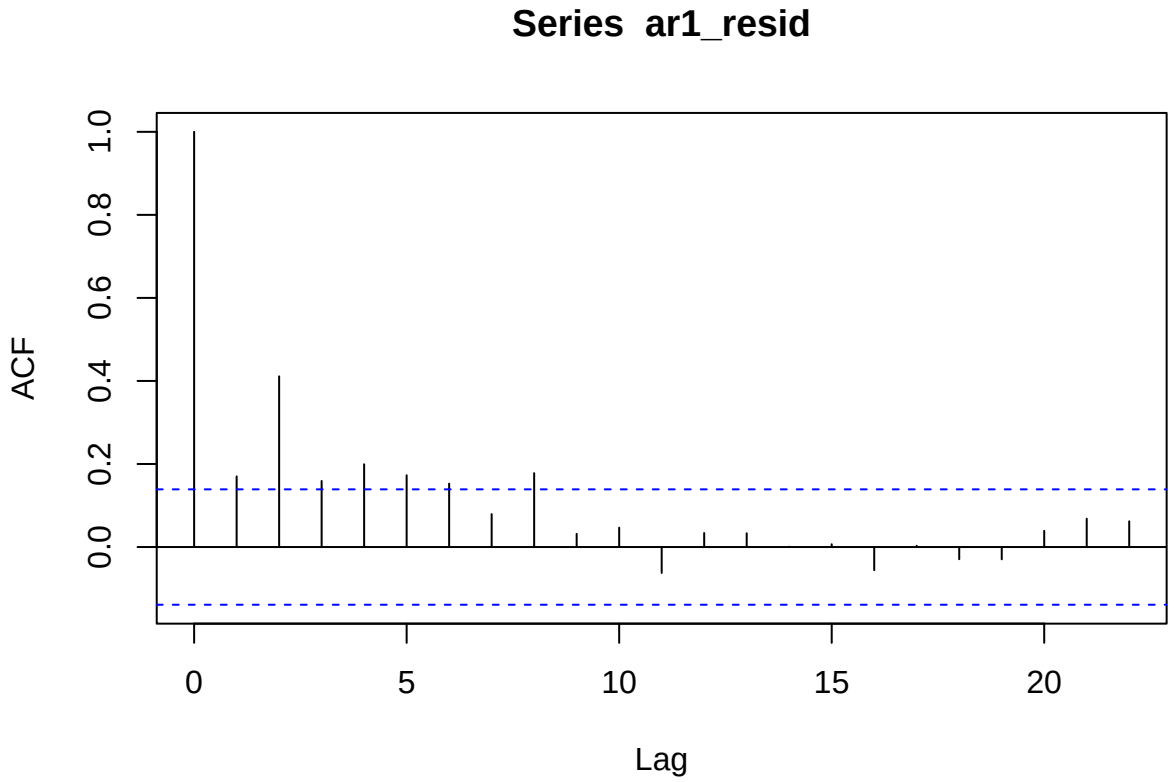
$$\hat{\sigma}^2 = 13.007$$

e) Display the sample ACF of the residuals on your AR(1) model Y_t.
Does this plot show an iid error structure and why?

```
# Z_t = Y_t - phi Y_{t-1}
ar1_resid = residuals(p4_ar1)
plot(p4$t[2:200], ar1_resid,
     type="l", col="blue",
     ylab="Residuals", xlab="t")
points(p4$t[2:200],ar1_resid)
abline(h=0)
```




```
acf(ar1_resid)
```



The residual plot and ACF plot do not demonstrate IID structure.
The plots demonstrate clear temporal dependence with a curved structure throughout the plot.
The ACF plot is similar to before (with temporal dependence among the first 5-7 lags), but the correlation structure has decreased significantly from before.
Perhaps a model other than AR(1) would yield better results - in particular, we could consider an AR(p) model which controls for dependence between more time steps.

STATIONARY TIME SERIES - LINEAR PROCESSES

Stationary Models:

Loosely speaking, $\{X_t : t \in \mathbb{Z}\}$ is stationary if $X_t \overset{d}{\approx} X_{t+h}$ for $h > 0$ (if X_t and X_{t+h} are similar in distribution).
In other words, $X_t \approx$ "$shifted\ RV$".

We will restrict our attention to first and second order models (the mean and the covariance functions).

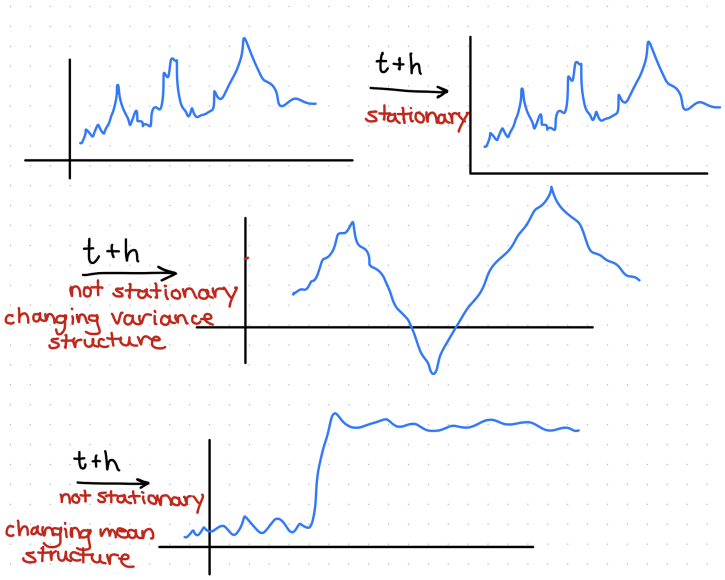
$$E[X_t] = \mu_t, \quad \gamma(t,s) = Cov(X_t, X_s)$$

Note that in both a theoretical and applied perspective, we check the first and second order properties of a time series model to evaluate (weak) stationarity.

Weak Stationarity:

$\{X_t\}$ is weakly stationary if
1. $E[X_t] = \mu_X(t)$ is independent of t (does not depend on t)
2. $\gamma_X(t+h, t) = \gamma(h)$ is independent of t for each h (for a fixed value of h, it should not depend on t).

Whenever we say '$\{X_t\}$ is stationary,' we mean weakly stationary, unless otherwise indicated.



Strict Stationarity:

$\{X_t\}$ is strictly stationary if

$$\mathbf{X} = (X_1, \ldots, X_n) \overset{d}{=} (X_{1+h}, \ldots, X_{n+h}), \quad \forall\ h \in \mathbb{Z},\ n \geq 1$$

Note that equality in distribution of the finite dimensional distributions implies equality of the marginal distributions.

We would like to work with strict stationarity for its mathematical properties (e.g. iid sequence is strictly stationary).
However many properties are easier to handle for weak stationarity.
Thus, whenever we say '$\{X_t\}$ is stationary,' we mean weakly stationary, unless otherwise indicated.

Properties:

Suppose we have a strictly stationary time series $\{X_t\}$.

1a. X_t are identically distributed (not iid).
1b. $(X_t, X_{t+h}) = (X_1, X_{1+h})$

Any two random variables in the random vector have the same (marginal) distribution.
Thus, the time point t does not affect the distribution of the variables.

2. X_t is weakly stationary for $E[X_t^2] < \infty$.

Strict stationarity implies weak stationarity for $E[X_t^2] < \infty$.

3. Weak stationarity does not imply strict stationarity.

4. An iid sequence is strictly stationary.

Properties of the Autocovariance Function for Stationary Processes:
The definition of stationarity is primarily focused on the autocovariance function.

Let $\{X_t\}$ be a stationary time series.
The ACVF of $\{X_t\}$ at lag h is

$$\gamma_X(h) = Cov(X_{t+h}, X_t)$$

The ACF of $\{X_t\}$ at lag h is

$$\rho_X(h) = Corr(X_{t+h}, X_t) = \frac{\gamma_X(h)}{\gamma_X(0)}$$

Properties:

1. For stationary time series, $\gamma_X(h) \equiv \gamma_X(t+h, t)$, because stationary time series do not depend on t.

2. $\gamma(0) = Var(X_t) \geq 0$

3. $|\gamma(h)| \leq \gamma(0), \quad \forall\ h$

Proof:

Holds by the Cauchy-Schwartz inequality.

4. $\gamma(\cdot)$ is an even function, $\implies \gamma(h) = \gamma(-h), \quad \forall\ h$

Proof:

The covariance function is symmetric.

$$\gamma(h) = Cov(X_{t+h}, X_t) = Cov(X_t, X_{t+h}) = \gamma(-h)$$

Theorem - The ACVF of a Stationary Process is Even and Non-Negative Definite:
A real-valued function defined on the integers is the ACVF of a stationary process if and only if it is even and non-negative definite.

A real-valued function $k(\cdot)$ on the integers is non-negative definite if

$$\sum_{i=1}^{n}\sum_{j=1}^{n} a_i k(i-j) a_j \geq 0$$

For all positive values of n and real vectors $\mathbf{a} = (a_1, \ldots, a_n)^T \in \mathbb{R}^n$.

Non-negative definite implies that the eigenvalues of a matrix are non-negative.
Note that a covariance matrix is non-negative definite.
Non-negative definiteness also ensures that a quadratic form is convex.

Proof (Half-Proof):

An "if and only if" statement must be proven in both directions.
We will only demonstrate one direction of the proof.

Suppose $\gamma(h)$ is the ACVF of a stationary process.
Let $\mathbf{a} \in \mathbb{R}^n$, and set $\mathbf{X}_n = (X_1, \cdots, X_n)^T$.
Then

$$Var(\mathbf{a}^T\mathbf{X}_n) = \mathbf{a}^T Var(\mathbf{X}_n)\mathbf{a} = \sum_{i=1}^{n}\sum_{j=1}^{n} a_i \Gamma(i-j) a_j \geq 0, \ \textit{because variance} \geq 0$$

For $Var(\mathbf{X}_n) = \Gamma_n$, a covariance matrix.

For $n = 2$,

$$Var(a_1X_1 + a_2X_2) = a_1^2 Var(X_1) + a_1a_2Cov(X_1, X_2) + a_2a_1Cov(X_2, X_1) + a_2^2 Var(X_2)$$

Properties:

1. To show that a given function is non-negative definite, it's often simpler to find a stationary process that has the specified function as its ACVF ($\gamma(h)$), than to verify non-negative definiteness directly.

q-Dependence of a Time Series:
From a theoretical perspective, we often require a certain error structure in the models that we work with - a q-dependent structure on the errors is a common assumption to provide us with certain useful properties.

A simple way to construct a strictly stationary time series is to "filter" (take a real-valued function of) an iid sequence of random variables.

Let $\{Z_t\}$ be an iid sequence of random variables.
By the properties of a strictly stationary time series, $\{Z_t\}$ is strictly stationary.
Define a filter (real-valued function) g such that

$$X_t = g(Z_t, Z_{t-1}, \ldots, Z_{t-q})$$

For example, g could be a sum (e.g. random walk) or linear combination, e.g.,

$$X_t = g(Z_t, Z_{t-1}, \ldots, Z_{t-q}) = Z_t + \theta_1 Z_{t-1} + \cdots + \theta_q Z_{t-q}$$

Because $\{Z_t\}$ is iid, it is strictly stationary by the properties of a strictly stationary process.
By definition,

$$(Z_t, \ldots, Z_{t-q}) \stackrel{d}{=} (Z_{t+h}, \ldots, Z_{t+h-q})$$

It follows that $\{X_t\}$ is strictly stationary because it is a function of strictly stationary noise.
This can be shown explicitly.

$$(X_1, X_2, \ldots, X_n) = (g(Z_1, Z_0, \ldots, Z_{1-q}), g(Z_2, Z_1, \ldots, Z_{2-q}), g(Z_n, Z_{n-1}, \ldots, Z_{n-q}))$$

$$\stackrel{d}{=} (g(Z_{1+h}, Z_{1+h}, \ldots, Z_{h+1-q}), g(Z_{2+h}, Z_{1+h}, \ldots, Z_{h+2-q}), g(Z_{n+h}, Z_{n-1+h}, \ldots, Z_{h+n-q})) = (X_{1+h}, X_{2+h}, \ldots, X_{n+h})$$

Properties:

1. A time series $\{X_t\}$ is q-dependent if X_s and X_t are independent whenever $|t-s| > q$.

   The q-dependent series $\{X_t\}$ is strictly stationary and has a dependence structure for $\leq q$ lags, but will be independent for $> q$ lags.

2. A time series $\{X_t\}$ is q-correlated if $\gamma(h) = 0$ whenever $|h| > q$.

   Recall that WN indicates an uncorrelated error structure.
   Recall that a correlation of 0 does not imply independence, so it relates to weak stationarity.

   The q-correlated series $\{X_t\}$ is weakly stationary, and has $\gamma(h) = 0$ whenever $|h| > q$.

MA(q) Process:
$\{X_t\}$ is a moving average process of order q, MA(q) if

$$X_t = Z_t + \theta_1 Z_{t-1} + \cdots + \theta_q Z_{t-q}, \quad Z_t \sim WN(0, \sigma^2)$$

For constants $\theta_1, \ldots, \theta_q$.

Note that the MA(q) process is a linear combination of past noise.

Properties:

1. If $Z_t \sim WN(0, \sigma^2)$, the MA(q) process is q-correlated and weakly stationary.
2. If $Z_t \sim IID(0, \sigma^2)$, the MA(q) process is q-dependent and strictly stationary.

Proposition:

1. If $\{X_t\}$ is a stationary q-correlated time series with mean 0, it can be represented as the MA(q) process.

Linear Process (Class of Stationary Time Series Models - Including ARMA):
The class of linear time series models (which includes ARMA models) provides a general framework for studying stationary processes.

Every second-order stationary process (i.e. with well-defined covariances) is either a linear process or can be transformed into a linear process (i.e. of noise) by subtracting some deterministic (e.g. trend, seasonality) component.

The time series $\{X_t\}$ is a linear process if it has the (infinite sum of WN) representation

$$X_t = \sum_{j=-\infty}^{\infty} \Psi_j Z_{t-j}, \quad Z_t \sim WN(0, \sigma^2), \quad \forall\, t$$

Where Ψ_j is a sequence of constants such that

$$\sum_{j=-\infty}^{\infty} |\Psi_j| < \infty$$

i.e., they demonstrate absolute convergence.

In other words, X_t is an infinite sum of past and future observations (which are composed of WN) - we will later restrict models to consider only past observations (WN) - causal models.

Note that

$$\sum_{j=-\infty}^{\infty} |\Psi_j| < \infty \implies \sum_{j=-\infty}^{\infty} \Psi_j^2 < \infty$$

Properties:

1. Suppose we would like to express a linear process in terms of backshift operators.

$$X_t = \Psi(B) Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad \forall\, t$$

$$\Psi(B) = \sum_{j=-\infty}^{\infty} \Psi_j B^j$$

Where Ψ_j is a sequence of constants such that

$$\sum_{j=-\infty}^{\infty} |\Psi_j| < \infty$$

Proposition:

1. A linear process is called a MA(∞) if $\Psi_j = 0$ for all $j < 0$, i.e., if

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}$$

Theorem - Convergence in Mean (Existence) of Linear Processes:
Recall that we have discussed convergence of random variables in terms of convergence in probability, almost surely convergence, and convergence in mean.
We will discuss existence in terms of convergence in mean - existence implies that expectation exists, which in turn implies the expectation of the absolute value of the mean is finite.

Provided that $\sum_{j=-\infty}^{\infty} |\Psi_j| < \infty$, how do we know that $\sum_{j=-\infty}^{\infty} \Psi_j Z_{t-j}$ converges?

Let $\{X_t : t \in \mathbb{Z}\}$ be the time series defined by $X_t = \sum_{j=-\infty}^{\infty} \Psi_j Z_{t-j}$, where $Z_t \sim WN(0, \sigma^2)$.
If $\sum_{j=-\infty}^{\infty} |\Psi_j| < \infty$ then $E|X_t| < \infty$.

Note that $E|X_t| < \infty$ resembles a LLN type result, $E|X_t - 0| < \infty$, indicating that this result concerns mean convergence.

In other words, a time series expressed as a (infinite) linear process of WN has a finite expectation provided the restriction on the constants Ψ_j.
This is how we can justify discussing the random variable defined as an infinite linear process.

Note that another way to demonstrate convergence of this series is through the use of Cauchy sequences.
Set $v_n = \sum_{j=-n}^{n} \Psi_j Z_{t-j}$, a sequence of partial sums.

Provided that $\sum_{j=-\infty}^{\infty} \Psi_j^2 < \infty$, it can be argued that

$$\lim_{n,m\to\infty} E|v_n - v_m|^2 = 0$$

Proof:

$$E|X_t| = E\left|\sum_{j=-\infty}^{\infty} \Psi_j Z_{t-j}\right|$$

Noting that the absolute value is inside the expectation, using the triangle inequality and Jensen's inequality,

$$\leq \sum_{j=-\infty}^{\infty} |\Psi_j| E|Z_{t-j}|$$

$$\overset{HW\ Result}{\leq} \sum_{j=-\infty}^{\infty} |\Psi_j|\sigma$$

$$= \sigma\left(\sum_{j=-\infty}^{\infty} |\Psi_j|\right) < \infty$$

Assuming finite-ness of the absolute value of the sum of the constants, Ψ_j, the expectation of the (infinite) linear process $\{X_t\}$ must be finite.

Linear Filter $\Psi(B)$ - Applications (Generalizing the Characterization of Stationary TS):
Recall that we can express a linear process in terms of back-shift operators.

$$X_t = \Psi(B)Z_t, \quad Z_t \sim WN(0,\sigma^2), \quad \forall\ t$$

$$\Psi(B) = \sum_{j=-\infty}^{\infty} \Psi_j B^j$$

Where Ψ_j is a sequence of constants such that

$$\sum_{j=-\infty}^{\infty} |\Psi_j| < \infty$$

The operator $\Psi(B) = \sum_{j=-\infty}^{\infty} \Psi_j B^j$ can be thought of as a (infinite) linear filter.
If Y_t is stationary, then $X_t = \Psi(B)Y_t$ is also stationary.

Theorem - Stationarity and Characterization of Resulting Time Series Given Use of Linear Filter $\Psi(B)$:
Let $\{Y_t\}$ be a stationary time series with mean function $E[Y_t] = 0$ and covariance function γ_Y.
If $\sum_{j=-\infty}^{\infty} |\Psi_j| < \infty$, then the time series

$$X_t = \sum_{j=-\infty}^{\infty} \Psi_j Y_{t-j} = \Psi(B)Y_t$$

i. is stationary with mean $E[X_t] = 0$,
ii. and covariance function

$$\gamma_X(h) = \sum_{j=-\infty}^{\infty}\sum_{k=-\infty}^{\infty} \Psi_j\Psi_k\gamma_Y(h+k-j)$$

In particular, recall that if $\{Y_t\}$ is
1. Weakly stationary (e.g., WN), $\{X_t\}$ is a linear process and weakly stationary by the properties of q-correlation.
2. Strictly stationary (e.g., iid), $\{X_t\}$ is a linear process and strictly stationary by the properties of q-dependence.

For example, suppose we have an AR(1) process $\{Y_t\}$ - its ϕ values are known.
Upon applying a linear filter to $\{Y_t\}$, this theorem indicates the covariance function of the resulting time series.

Remark: Properties Hold for the Application of Successive Operators - Inverse Operator Justification:
The absolute convergence of $X_t = \sum_{j=-\infty}^{\infty} \Psi_j Y_{t-j}$ (the linear process of a stationary process) implies that filters of the form

$\alpha(B) = \sum_{j=-\infty}^{\infty} \alpha_j B^j$ and $\beta(B) = \sum_{j=-\infty}^{\infty} \beta_j B^j$, with $\sum_{j=-\infty}^{\infty} |\alpha_j| < \infty$ and $\sum_{j=-\infty}^{\infty} |\beta_j| < \infty$, can be applied successively to the stationary time series $\{Y_t\}$ to generate a new stationary time series, whose constants have the form

$$\Psi_j = \sum_{k=-\infty}^{\infty} \alpha_k \beta_{j-k} = \sum_{k=-\infty}^{\infty} \beta_k \alpha_{j-k}$$

These relations are equivalent to

$$W_t = \Psi(B) Y_t$$
$$\Psi(B) = \alpha(B)\beta(B) = \beta(B)\alpha(B)$$

In other words, we can apply multiple operators successively (in any order - commutativity) and still generate a linear process. Recall that we cannot divide by operators - instead we must define a new inverse operator.
This remark justifies the creation of a stationary process given the application of multiple operators.

Theorem Application - ACVF of a Linear Process:
In the special case where $\{X_t\}$ is a linear process,

$$\gamma_X(h) = \sigma^2 \sum_{j=-\infty}^{\infty} \Psi_j \Psi_{j+h}$$

In other words, if we identify that $\{X_t\}$ is a linear process, we can use this formula to determine the covariance function of $\{X_t\}$ instead of deriving it directly.

ARMA(1,1):
The time series process $\{X_t\}$ is an ARMA(1,1) process if it satisfies

$$X_t - \phi X_{t-1} = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad \phi + \theta \neq 0$$

Considering the structure of the AR(1) and MA(1) model, the ARMA(1,1) indicates serial dependence in both the observations and the noise terms.

The ideal solution of the ARMA(1,1) model is stationary, causal, and invertible, characterized by $|\phi| < 1$, $|\theta| < 1$.

Parameter Constraint Properties:

1. A stationary solution exists if and only if $|\phi| \neq 1 \implies \phi \neq \pm 1$, i.e., the AR(1) process is not a random walk.

   Note that for $|\phi| = 1 \implies \phi = \pm 1$, the ARMA(1,1) model introduces a random walk in the AR(1) component, which is not stationary.
   For $|\phi| < 1$, there is stationarity of the causal model.
   For $|\phi| > 1$, there is stationarity of the non-causal model.
   Note that the magnitude of θ does not affect stationarity.

2. If $|\phi| < 1$, then the unique stationary solution is

$$X_t = Z_t + (\phi + \theta) \sum_{j=1}^{\infty} \phi^{j-1} Z_{t-j}$$

   In this case, we say that $\{X_t\}$, the ARMA(1,1) time series model, is causal - i.e., $\{X_t\}$ can be expressed in terms of current and past WN terms, Z_s, $s \leq t$.

3. If $|\phi| > 1$, then the unique stationary solution is

$$X_t = -\theta \phi^{-1} Z_t - (\phi + \theta) \sum_{j=1}^{\infty} \phi^{-j-1} Z_{t+j}$$

   In this case, we say that $\{X_t\}$, the ARMA(1,1) time series model is non-causal, i.e., $\{X_t\}$ can be expressed in terms of current and future WN terms, Z_s, $s \geq t$.

4. If $|\theta| = 1 \implies \theta = \pm 1$, then the ARMA(1,1) reduces to an AR(1) process (although it is still invertible).

   For $\theta = 1$,

$$X_t - \phi X_{t-1} = Z_t + Z_{t-1}$$

Recall that the sum of independent noise terms causes the resulting variance to be the sum of their variances, i.e. $Z_t + Z_{t-1} \sim WN(0, 2\sigma^2)$ - an AR(1) process with double variance.

5. If $|\theta| < 1$, then the unique stationary solution is

$$Z_t = X_t - (\phi + \theta)\sum_{j=1}^{\infty}(-\theta)^{j-1}X_{t-j}$$

In this case, we say that $\{X_t\}$, the ARMA(1,1) time series model is invertible, i.e., Z_t can be expressed in terms of current and past observations, X_s, $s \leq t$.

6. If $|\theta| > 1$, then the unique stationary solution is

$$Z_t = -\phi\theta^{-1}X_t + (\phi + \theta)\sum_{j=1}^{\infty}(-\theta)^{-j+1}X_{t+j}$$

In this case, we say that $\{X_t\}$, the ARMA(1,1) time series model is non-invertible, i.e., Z_t can be expressed in terms of current and future observations, X_s, $s \geq t$.

Linear Processes of ARMA(1,1) Cases:
Case 1 (Causal): $|\phi| < 1$

$$X_t - \phi X_{t-1} = Z_t + \theta Z_{t-1}$$
$$(1 - \phi B)X_t = (1 + \theta B)Z_t$$

Let $\Phi(B) = (1 - \phi(B))$, $\Theta(B) = (1 + \theta B)$.

$$\Phi(B)X_t = \Theta(B)Z_t$$

Solve for X_t.
Note that the constraint $|\phi| < 1$ allows us to take the inverse of the LHS operator because the geometric series converges. In particular, because $|\phi| < 1$, the power series expansion of $\frac{1}{\Phi(B)}$ is $\sum_{j=0}^{\infty}\phi^j B^j = \sum_{j=0}^{\infty}(\phi B)^j = \frac{1}{1-\phi B}$, which has absolutely summable coefficients.
To invert the LHS and solve for X_t, we define inverse operator $\Pi(B) = \sum_{j=0}^{\infty}\phi^j B^j$.

$$\Pi(B)\Phi(B)X_t = \Pi(B)\Theta(B)Z_t$$
$$X_t = \Pi(B)\Theta(B)Z_t$$
$$= (1 + \phi B + \phi^2 B^2 + \cdots)(1 + \theta B)Z_t$$
$$= [(1 + \phi B + \phi^2 B^2 + \cdots) + (\theta B + \phi\theta B^2 + \phi^2\theta B^3 + \cdots)]Z_t$$
$$= [1 + (\phi + \theta)B + (\phi + \theta)\phi B^2 + (\phi + \theta)\phi^2 B^3 + \cdots]Z_t$$
$$= [\Psi_0 + \Psi_1 B + \Psi_2 B^2 + \Psi_3 B^3 + \cdots]Z_t$$

For $\Psi_0 = 1$, $\Psi_j = (\phi + \theta)\phi^{j-1}$, $\quad j \geq 1$.

We conclude that this is a representation of the MA(∞) process, i.e.,
For $|\phi| < 1$, the unique stationary linear process solution of the ARMA(1,1) is

$$X_t = Z_t + (\phi + \theta)\sum_{j=1}^{\infty}\phi^{j-1}Z_{t-j}$$

Note that for $\phi + \theta = 0$, the series will become degenerate WN, hence the constraint.
Recall that using the linear process expression of the ARMA(1,1), we can calculate the ACVF.

Case 2 (Non-Causal): $|\phi| > 1$

We conclude that this is a representation of the MA(∞) process, i.e.,
For $|\phi| > 1$, the unique stationary linear process solution of the ARMA(1,1) is

$$X_t = -\theta\phi^{-1}Z_t - (\phi + \theta)\sum_{j=1}^{\infty}\phi^{-j-1}Z_{t+j}$$

Note that for $\phi + \theta = 0$, the series will become degenerate WN, hence the constraint.
Recall that using the linear process expression of the ARMA(1,1), we can calculate the ACVF.

Case 3 (Invertible): $|\theta| < 1$

$$X_t - \phi X_{t-1} = Z_t + \theta Z_{t-1}$$
$$(1 - \phi B)X_t = (1 + \theta B)Z_t$$

Let $\Phi(B) = (1 - \phi(B))$, $\Theta(B) = (1 + \theta B)$.

$$\Phi(B)X_t = \Theta(B)Z_t$$

Solve for Z_t.
What conditions do we have to place on $|\theta|$ that will allow us to take the inverse of the RHS operator (to ensure the geometric series converges).
In particular, we require a power series expansion of $\frac{1}{\Theta(B)} = \frac{1}{1+\theta B}$.
To obtain a converging geometric series, we require a negative in the denominator,

$$\frac{1}{\Theta(B)} = \frac{1}{1 + \theta B} = \frac{1}{1 - (-\theta B)}$$

For $|\theta| < 1$, define $\xi(B) = \sum_{j=0}^{\infty}(-\theta)^j B^j = \sum_{j=0}^{\infty}(-\theta B)^j = \frac{1}{1+\theta B}$.

$$\Phi(B)X_t = \Theta(B)Z_t$$
$$\Theta(B)Z_t = \Phi(B)X_t$$
$$\xi(B)\Theta(B)Z_t = \xi(B)\Phi(B)X_t$$
$$Z_t = \xi(B)\Phi(B)X_t$$
$$\vdots$$

For $|\theta| < 1$, the unique stationary linear process solution of the ARMA(1,1) is

$$Z_t = X_t - (\phi + \theta)\sum_{j=1}^{\infty}(-\theta)^{j-1}X_{t-j}$$

Note that for $\phi + \theta = 0$, the series will become degenerate, hence the constraint.
Recall that using the linear process expression of the ARMA(1,1), we can calculate the ACVF.

Case 4 (Non-Invertible): $|\theta| > 1$

For $|\theta| > 1$, the unique stationary linear process solution of the ARMA(1,1) is

$$Z_t = -\phi\theta^{-1}X_t + (\phi + \theta)\sum_{j=1}^{\infty}(-\theta)^{-j+1}X_{t+j}$$

Note that for $\phi + \theta = 0$, the series will become degenerate, hence the constraint.
Recall that using the linear process expression of the ARMA(1,1), we can calculate the ACVF.

Causality:
Causality indicates that $\{X_t\}$ can be expressed in terms of current and past WN terms, Z_s, $s \leq t$.
Non-causality indicates that $\{X_t\}$ can be expressed in terms of current and future WN terms, Z_s, $s \geq t$.

Note that in the ARMA(p,q) models, causality is determined by the ϕ parameter of the AR(p) portion of the model.
The MA(q) process is an AR(∞) process (infinite linear combination of past WN terms Z_t) and is therefore always causal and stationary - it has no roots that violate the causality and stationarity conditions.

Invertibility:
Invertibility indicates that Z_t can be expressed in terms of current and past observations, X_s, $s \leq t$.
Non-invertibility indicates that Z_t can be expressed in terms of current and future observations, X_s, $s \geq t$.

Note that in the ARMA(p,q) models, invertibility is determined by the θ parameter of the MA(q) portion of the model. The AR(p) process is a MA(∞) process (infinite linear combination of past observations X_t) and is therefore always invertible - it has no roots that violate the invertibility conditions.

Applications - Forecasting MA Models:
Invertibility requirements are not as prevalent as causality requirements, but they do appear - for example, when estimating MA(q) coefficients with data, invertibility constraints on the series may be necessary to ensure that the series converges.
In particular, consider the MA(1) and ARMA(1,1) models,

$$X_t = Z_t + \theta Z_{t-1}$$
$$X_t = \phi X_{t-1} + Z_t + \theta Z_{t-1}$$

How are we meant to forecast $\hat{X}_{t+1}$ in the MA(1) model when only the X_t's are observable?
By invertibility, we can express Z_t in terms of X_t to estimate the noise.
The estimated noise terms can be used to forecast.

Properties of the Sample Mean for Dependent Data, Long Run Variance:
Motivation:
Consider the iid case:
Let $X_1, \ldots, X_n \overset{iid}{\sim} Dist(\mu, \sigma^2)$.
Let the statistic (random variable) $\bar{X}_n = \frac{1}{n}\sum_{i=1}^n X_i$ be an estimator of $E[X_i] = \mu$.

LLN: $Var(\bar{X}_n) = \frac{\sigma^2}{n} \to 0, \quad as \quad n \to \infty$
LLN (Preferred): $nVar(\bar{X}_n) = \sigma^2 \to \sigma^2, \quad as \quad n \to \infty$

CLT (Informal): For large n, $\bar{X}_n \overset{d}{\approx} N\left(\mu, \frac{\sigma^2}{n}\right)$

CLT: $\frac{\bar{X}_n - \mu}{\sigma/\sqrt{n}} \overset{d}{\to} N(0,1)$

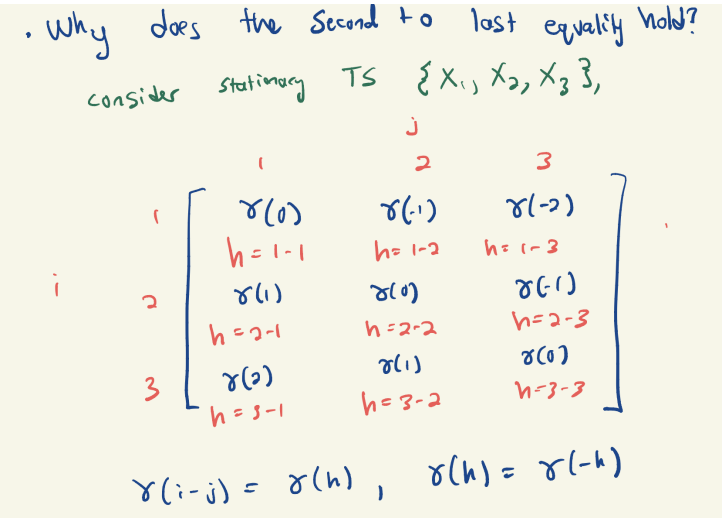
CLT (Preferred): $\sqrt{n}(\bar{X}_n - \mu) \overset{d}{\to} N(0, \sigma^2)$

Variance of Sample Mean (General Case - Dependent Data):
Now suppose $\{X_t\}$ is a stationary time series:

$$Var(\bar{X}_n) = E[(\bar{X}_n - \mu)^2] = \frac{1}{n^2}\sum_{i=1}^n\sum_{j=1}^n Cov(X_1, X_j) = \frac{1}{n^2}\sum_{i-j=-n}^{n}(n - |i-j|)\gamma_X(i-j)$$
$$= \frac{1}{n}\sum_{h=-n}^{n}\left(1 - \frac{|h|}{n}\right)\gamma_X(h)$$

Note that this expression holds in the iid case - all covariance terms are 0 ($\gamma_X(h) = 0$, $h \neq 0$), and the only term that survives is the $h = 0$ term, σ^2.

For example,



$$Var(\bar{X}_n) = \frac{1}{3^2}\sum_{i=1}^{3}\sum_{j=1}^{3} Cov(X_i, X_j) = \frac{1}{3^2}\sum_{i-j=-3}^{3}(3-|i-j|)\gamma_X(i-j) = \frac{1}{3^2}\sum_{h=-3}^{3}(3-|h|)\gamma_X(h)$$

$$= \frac{1}{3^2}(0 + \gamma_X(-2) + 2\gamma_X(-1) + 3\gamma_X(0) + 2\gamma_X(1) + \gamma_X(2) + 0)$$

Motivation - Long-Run Variance of Sample Mean (General Case - Dependent Data):
Define the long-run variance.

$$Var(\bar{X}_n) = \frac{1}{n}\sum_{h=-n}^{n}\left(1 - \frac{|h|}{n}\right)\gamma_X(h)$$

$$\lim_{n\to\infty} nVar(\bar{X}_n) = \sum_{h=-\infty}^{\infty}\gamma_X(h), \quad for \quad \sum_{h=-\infty}^{\infty}|\gamma_X(h)| < \infty$$

Note that these expressions holds in the iid case - all covariance terms are 0 ($\gamma_X(h) = 0$, $h \neq 0$), and the only term that survives is the $h = 0$ term, σ^2.

When incorporating a dependence structure between observations, the long-run variance will not simply reduce to the variance of the noise.
Thus, when performing inferences on the mean (e.g. a testing procedure), we must divide by an estimator of the long run variance of the sample mean.

For example, in the iid case,

$$H_0 : \mu = \mu_0$$
$$H_A : \mu \neq \mu_0$$

For ν known, the test statistic is

$$Z = \frac{\bar{X}_n - \mu_0}{\sqrt{\nu/n}}$$

For ν unknown, we must estimate $\hat{\nu}$ as the infinite sum of sample covariances resulting in

$$t = \frac{\bar{X}_n - \mu_0}{\sqrt{\hat{\nu}/n}}$$

Recall that a stationary time series $\{X_t\}$ can be expressed as a linear process.

$$X_t = \sum_{j=-\infty}^{\infty}\Psi_j Z_{t-j}$$

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n}\sum_{j=-\infty}^{\infty}\Psi_j Z_{t-j}$$

Thus it is reasonable that the long-run variance is expressed as an infinite sum of covariances.

It is not possible to sum over all lags, so a natural estimator for the long-run variance is the sum over the ACVF for available lags in sample data.

Formal Definition of Long-Run Variance of the Sample Mean (General Case - Dependent Data):
Assume $\{X_t\}$ is a stationary time series with mean μ and covariance function $\gamma_X(h)$.

The long-run variance ν of stationary time series $\{X_t\}$ is

$$\nu = \sum_{h=-\infty}^{\infty}\gamma_X(h)$$

Provided the limit exists.

Properties:
Due to symmetry of the covariance function, note that alternative expressions for the long-run variance are

1. $\nu = \sum_{|h|<\infty} \gamma_X(h) = \sum_{h=-\infty}^{\infty} \gamma_X(h) = \gamma(0) + 2\sum_{h=1}^{\infty} \gamma_X(h)$

This expression is particularly useful for stationary models, to reduce the number of necessary $\gamma_X(h)$ calculations.

2. $\nu = \sum_{h=-\infty}^{\infty} \gamma_X(h) = \sigma^2(\sum_{j=-\infty}^{\infty} \Psi_j)^2, \quad for \quad Z_t \sim WN(0, \sigma^2)$

This expression is particularly useful for models expressed as linear processes.

Proof (2):

Recall that for a linear process

$$\gamma_X(h) = \sum_{j=-\infty}^{\infty} \Psi_j \Psi_{j+h} \sigma^2$$

$$\therefore \nu = \sum_{h=-\infty}^{\infty} \gamma_X(h) = \sum_{h=-\infty}^{\infty} \sum_{j=-\infty}^{\infty} \Psi_j \Psi_{j+h} \sigma^2$$

Because we are summing over infinity for h,

$$= \left(\sum_{j=-\infty}^{\infty} \Psi_j \right)^2 \sigma^2$$

LLN and CLT for Stationary Time Series (General Case - Dependent Data):
Assume $\{X_t\}$ is a stationary time series with mean μ and covariance function $\gamma_X(h)$.

LLN: $Var(\bar{X}_n) \to 0, \quad as \quad n \to \infty$, if $\gamma_X(n) \to 0, \quad as \quad n \to \infty$
LLN (Preferred): $nVar(\bar{X}_n) \to \nu = \sum_{|h|<\infty} \gamma_X(h), \quad if \quad \sum_{h=-\infty}^{\infty} |\gamma_X(h)| < \infty$
The constraint on the covariance function indicates that the covariance should diminish as n gets large.

Note that for the CLT of stationary time series $\{X_t\}$, we require that the noise of the time series (sequence) $\{X_t\}$ is iid (that $\{X_t\}$ is strictly stationary).

CLT (Informal): For large n,

$$\sqrt{n}(\bar{X}_n - \mu) \overset{d}{\approx} N\left(0, \sum_{h=-n}^{n} \left(1 - \frac{|h|}{n}\right) \gamma_X(h)\right)$$

Note that we do not use the long-run variance in this informal statement, because convergence has not yet occurred.

To formally express the CLT in terms of the long-run variance, we must write the CLT in terms of a linear process (with iid noise).

CLT (for a Strictly Stationary Linear Process with iid Noise):

Note that our time series $\{X_t\}$ has mean μ.
Assume $\{X_t\}$ can be expressed as

$$X_t = \mu + \sum_{j=-\infty}^{\infty} \Psi_j Z_{t-j}, \quad Z_t \sim IID(0, \sigma^2), \quad \sum_{j=-\infty}^{\infty} |\Psi_j| < \infty, \quad \sum_{j=-\infty}^{\infty} \Psi_j \neq 0$$

The noise terms are iid to ensure strict stationarity.
We require absolute convergence of the parameters Ψ_j and that that they have a non-zero sum (to prevent a degenerate 0-variance time series).

Then,

$$\sqrt{n}(\bar{X}_n - \mu) \xrightarrow{d} N(0, \nu)$$

Note that these expressions hold in the iid case as well.

Uses of the CLT for Stationary Time Series (General Case - Dependent Data):
Suppose we have the time series model

$$X_t = \mu + \sum_{j=-\infty}^{\infty} \Psi_j Z_{t-j}, \quad \sum_{j=-\infty}^{\infty} |\Psi_j| < \infty$$

In practice, we have data $X_1, \ldots, X_n$.
Suppose that we would like to test the hypotheses

$$H_0 : \mu = \mu_0$$
$$H_A : \mu \neq \mu_0$$

We must compute a test statistic given by

$$\frac{\bar{X}_n - \mu}{\sqrt{\nu/n}} = \frac{\bar{X}_n - \mu}{\frac{\sigma\sqrt{\left(\sum_{j=-\infty}^{\infty} \Psi_j\right)}}{n}}$$

Examples:

Example: HW 1 #5 - Expectation Exercise

Let $Z \sim WN(0, \sigma^2)$. Show that $E[Z] \leq \sigma$.
Hint: Apply Jensen's inequality on the concave function $h(u) = \sqrt{u}, \quad u \geq 0$. Also note that $|u| = \sqrt{u^2}$

Jensen's inequality states the following.
If $g(x)$ is a convex function on R_x, and $E[g(X)]$ and $g(E[X])$ are finite, then

$$E[g(X)] \geq g(E[X])$$

Let $X = |Z_t|$, $g(X) = X^2$,

$$E[g(X)] = E[X^2] = E[Z_t^2] \geq g(E[X]) = (E[\ |Z_t|\ ])^2$$

Note that $Var(Z_t) = E[Z_t^2] = \sigma^2$.

$$\therefore (E[\ |Z_t|\ ])^2 \leq \sigma^2$$
$$E[Z_t] \leq \sigma$$

Example: HW 1 #7 - Characterizing a Time Series Process, Non-Negative Definiteness of a Function

Let $\{X_t : t \in \mathbb{Z}\}$ be a time series defined by

$$X_t = Z_1 \cos(\omega t) + Z_t \sin(\omega t), \quad Z_1, Z_2 \sim WN(0, 1)$$

Where ω is a fixed frequency in the interval $[0, \pi]$.

a) Derive the mean function $\mu_X(t)$ of $\{X_t : t \in \mathbb{Z}\}$.

$$\mu_X(t) = E[X_t] = E[Z_1 \cos(\omega t) + Z_2 \sin(\omega t)] = \cos(\omega t)E[Z_1] + \sin(\omega t)E[Z_2] = 0$$

b) Derive the covariance function $\gamma_X(t + h, t)$ of $\{X_t : t \in \mathbb{Z}\}$.

$$\gamma_X(t + h, t) = Cov(X_{t+h}, X_t) = E[X_{t+h}X_t] = E[(Z_1 \cos(\omega t) + Z_2 \sin(\omega t))(Z_1 \cos(\omega(t + h)) + Z_2 \sin(\omega(t + h)))]$$

$$= E[Z_1^2 \cos(\omega t)\cos(\omega t + \omega h) + Z_1 Z_2 \cos(\omega t)\sin(\omega t + \omega h) + Z_1 Z_2 \sin(\omega t)\cos(\omega t + \omega h) + Z_2^2 \sin(\omega t)\sin(\omega t + \omega h)]$$

$$= E[Z_1^2]\cos(\omega t)\cos(\omega t + \omega h) + E[Z_1 Z_2]\cos(\omega t)\sin(\omega t + \omega h) + E[Z_1 Z_2]\sin(\omega t)\cos(\omega t + \omega h) + E[Z_2^2]\sin(\omega t)\sin(\omega t + \omega h)$$

Noting that covariance terms are 0, and squared terms are variances, for $Var(Z_1) = Var(Z_2) = 1$.

$$= \cos(\omega t)\cos(\omega t + \omega h) + \sin(\omega t)\sin(\omega t + \omega h)$$
$$= \cos(\omega t)\left(\cos(\omega t)\cos(\omega h) - \sin(\omega t)\sin(\omega h)\right) + \sin(\omega t)\left(\sin(\omega t)\cos(\omega h) + \cos(\omega t)\sin(\omega h)\right)$$
$$= \cos^2(\omega t)\cos(\omega h) - \cos(\omega t)\sin(\omega t)\sin(\omega h) + \sin^2(\omega t)\cos(\omega h) + \sin(\omega t)\cos(\omega t)\sin(\omega h)$$
$$= \cos^2(\omega t)\cos(\omega h) + \sin^2(\omega t)\cos(\omega h)$$
$$= \cos(\omega h)(\cos^2(\omega t) + \sin^2(\omega t))$$

$$= \cos(\omega h)$$

$$\gamma_X(t+h,t) = \cos(\omega h)$$

c) Is $\{X_t : t \in \mathbb{Z}\}$ a stationary time series?

$\{X_t : t \in \mathbb{Z}\}$ is a stationary time series because $\mu_X(t)$ and $\gamma_X(t+h,t)$ do not depend on t.

d) Prove that the function $\kappa(h)$ is non-negative definite, where

$$\kappa(h) = \cos(\omega h), \quad h \in \mathbb{Z}$$

We note that a real-valued function defined on the integers is the ACVF of a stationary process if and only if it is even and non-negative definite.

Note that the ACVF of the stationary process defined above is given by $\gamma_X(t+h,t) = \cos(\omega h)$.

Therefore, $\kappa(h) = \cos(\omega h)$, $h \in \mathbb{Z}$ is non-negative definite.

Example: AR(1) Process, $|\phi| < 1$ - Deriving the Linear Process, ACVF (Linear Process), Uniqueness of Stationary Solution, Long-Run Variance

Consider the AR(1) process for which the stationarity condition $|\phi| < 1$ holds.

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad |\phi| < 1$$

a) Derive the linear process for an AR(1).

Define the operators

$$\Phi(B) = 1 - \phi(B)$$
$$\Pi(B) = \sum_{j=0}^{\infty} \phi^j B^j = \sum_{j=0}^{\infty} (\phi B)^j$$

It is not possible to divide by an operator, but we note that

$$\sum_{j=0}^{\infty} (\phi B)^j \overset{geom\ sum}{=} \frac{1}{1-\phi B} \overset{!}{=} \frac{1}{\Phi(B)}$$

Note that

$$\Pi(B)\Phi(B) = \Phi(B)\Pi(B) = 1$$

$$X_t = \phi X_{t-1} + Z_t$$
$$X_t - \phi X_{t-1} = Z_t$$
$$(1 - \phi B) X_t = Z_t$$
$$\Phi(B) X_t = Z_t$$
$$\Pi(B)\Phi(B) X_t = \Pi(B) Z_t$$
$$X_t = \Pi(B) Z_t$$
$$X_t = \sum_{j=0}^{\infty} \phi^j B^j Z_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}$$

b) Recall that when $\{X_t\}$ is a linear process (because Y_t is WN or iid),

$$\gamma_X(h) = \sigma^2 \sum_{j=-\infty}^{\infty} \Psi_j \Psi_{j+h}$$

Given the unique stationary solution, $X_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}$.
Then,

$$\gamma_X(h) = \sigma^2 \sum_{j=-\infty}^{\infty} \Psi_j \Psi_{j+h}$$

Using the sum of a geometric series,

$$= \sigma^2 \sum_{j=0}^{\infty} \phi^j \phi^{j+h} = \sigma^2 \phi^h \sum_{j=0}^{\infty} \left(\phi^2\right)^j = \frac{\sigma^2 \phi^h}{1-\phi^2}, \quad h \geq 0$$

c) Show that $X_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}$ is the (only) unique stationary solution of the AR(1) process.

To prove uniqueness, you suppose that there is another solution and you demonstrate that the solutions are the same. We must show that these linear processes end up converging at infinity.

Suppose $\{Y_t\}$ is another stationary solution to the AR(1) process. Then

$$\begin{aligned} Y_t &= Z_t + \phi Y_{t-1} \\ &= Z_t + \phi Z_{t-1} + \phi^2 Y_{t-2} \\ &= Z_t + \phi Z_{t-1} + \phi^2 Z_{t-2} + \phi^3 Y_{t-3} \\ &= \cdots \\ &= Z_t + \phi Z_{t-1} + \cdots + \phi^k Z_{t-k} + \phi^{k+1} Y_{t-(k+1)} \end{aligned}$$

We will now apply a convergence-type result to the expectation.

$$E\left[Y_t - \sum_{j=0}^{k} \phi^j Z_{t-j}\right]^2$$

The term $\sum_{j=0}^{k} \phi^j Z_{t-j}$ represents a sequence of partial sums that characterize X_t (which can be expressed as a sequence of partial sums because it is a linear process).
Taking the limit of the expression to show that the linear processes converge at infinity, we obtain

$$= E[\phi^{k+1} Y_{t-(k+1)}]^2 = \phi^{2k-2} E[Y_{t-(k+1)}^2] \to 0 \;\; as \; k \to \infty$$

Because $E[Y_{t-(k+1)}^2] < \infty$ and $|\phi| < 1$.

d) Find the long-run variance of $\{X_t\}$ with mean μ.

X_t must be replaced with $X_t - \mu$ to guarantee mean 0.
View this as subtracting a (constant) trend prior to analyzing the errors.
We restate the model as

$$X_t - \mu = \phi(X_{t-1} - \mu) + Z_t$$

Note that the covariance function $\gamma_X(h)$ does not change because covariances are invariant under shifts. Recall

$$\gamma_X(h) = \frac{\sigma^2 \phi^{|h|}}{1-\phi^2}, \quad \forall \; h$$

We will restrict our attention $h \geq 0$ for convenience,

$$\gamma_X(h) = \frac{\sigma^2 \phi^h}{1-\phi^2}, \quad h \geq 0$$

Note that this is a causal process, so all $\Psi_j = 0$ for $j < 0$.

$$\nu = \sigma^2 \left(\sum_{j=-\infty}^{\infty} \Psi_j\right)^2$$

$$\sum_{j=-\infty}^{\infty} \Psi_j = \sum_{j=0}^{\infty} \phi^j = \frac{1}{1-\phi}, \quad |\phi| < 1$$

$$\therefore \nu = \sigma^2 \left(\sum_{j=-\infty}^{\infty} \Psi_j \right)^2 = \left(\frac{1}{1-\phi} \right)^2 \sigma^2$$

Alternatively,

$$\nu = \sum_{h=-\infty}^{\infty} \gamma_X(h) = \gamma(0) + 2\sum_{h=1}^{\infty} \gamma_X(h) = \frac{\sigma^2}{1-\phi^2} + 2\sum_{h=1}^{\infty} \frac{\sigma^2 \phi^h}{1-\phi^2}$$

$$= \frac{\sigma^2}{1-\phi^2} \left(1 + 2\sum_{h=1}^{\infty} \phi^h \right)$$

Note that for the geometric series starting at 1, we have the following

$$= \frac{\sigma^2}{1-\phi^2} \left(1 + 2\left(\frac{\phi}{1-\phi} \right) \right)$$

$$= \frac{\sigma^2}{(1-\phi)(1+\phi)} \left(\frac{1+\phi}{1-\phi} \right)$$

$$= \sigma^2 \left(\frac{1}{1-\phi} \right)^2$$

Example: HW 2 #1 - AR(1) Process, $|\phi| > 1$ - Deriving the Linear Process, Uniqueness of Stationary Solution

Let $\{X_t : t \in \mathbb{Z}\}$ be a time series defined by the First-Order Autoregressive Model AR(1).

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| > 1$$

a) Derive the unique stationary solution (linear process) of the above AR(1) process for $|\phi| > 1$.

$$X_t = \phi X_{t-1} + Z_t$$

In order for the geometric series sum to converge, the exponentiated term must be < 1.
We will multiply by ϕ^{-1}.

$$\phi^{-1} X_t = X_{t-1} + \phi^{-1} Z_t$$

$$X_{t-1} = \phi^{-1} X_t - \phi^{-1} Z_t$$

For the change in variables $t = t-1$, we observe that the past noise depends on the future observations (by non-causality).

$$X_t = \phi^{-1} X_{t+1} - \phi^{-1} Z_{t+1}$$

$$X_t - \phi^{-1} X_{t+1} = -\phi^{-1} Z_t$$

Define the forward shift operator B^{-1}, the inverse of the backward shift operator such that

$$BX_t = X_{t-1}$$

$$B^{-1} B X_t = B^{-1} X_{t-1}$$

$$X_t = B^{-1} X_{t-1}$$

We express our model in terms of operators.

$$X_t - \phi^{-1} X_{t+1} = -\phi^{-1} Z_t$$

$$(1 - \phi^{-1} B^{-1}) X_t = -\phi^{-1} Z_{t+1}$$

Define the operator $\Gamma(B^{-1}) = (1 - \phi^{-1}B^{-1})$.
The constraint $|\phi| > 1 \implies |\phi^{-1}| < 1$.
Because we meet the conditions for a converging power series, define the inverse operator

$$\Delta(B^{-1}) = \sum_{j=0}^{\infty} \phi^{-j}B^{-j} = \sum_{j=0}^{\infty}(\phi^{-1}B^{-1}) \stackrel{!}{=} \frac{1}{1 - \phi^{-1}B^{-1}}$$

Although taking the inverse of an operator is not well defined, note the sum of the geometric series.

$$(1 - \phi^{-1}B^{-1})X_t = -\phi^{-1}Z_{t+1}$$
$$\Gamma(B^{-1})X_t = -\phi^{-1}Z_{t+1}$$
$$\Delta(B^{-1})\Gamma(B^{-1})X_t = \Delta(B^{-1})(-\phi^{-1}Z_{t+1})$$
$$X_t = \left(\sum_{j=0}^{\infty} \phi^{-j}B^{-j}\right)(-\phi^{-1}B^{-1}Z_t)$$
$$X_t = -\left(\sum_{j=0}^{\infty} \phi^{-j-1}B^{-j-1}\right)Z_t$$

For the change in variables, $j = j - 1$,

$$X_t = -\sum_{j=1}^{\infty} \phi^{-j}B^{-j}Z_t = -\sum_{j=1}^{\infty} \phi^{-j}Z_{t+j}$$

Thus the linear process is given by

$$X_t = -\sum_{j=1}^{\infty} \phi^{-j}Z_{t+j}$$

This is the unique stationary solution in the non-causal case - X_t is correlated with future terms.
It is customary to restrict our attention to $|\phi| < 1$.

b) Briefly describe (in one or two sentences) why the solution obtained in part (a) is not very useful in practice.

The linear solution for the linear process of the AR(1) model for $|\phi| > 1$ is not useful in practice because $\{X_t : t \in \mathbb{Z}\}$ is non-causal, i.e. X_t can be expressed in terms of future WN terms Z_t which is not tractable in practice.

c) Show that the stationary solution derive in part (a) is unique.
To solve this problem, suppose that $\{Y_t\}$ is another stationary solution that satisfies the AR(1) process, and show that the solution in part (a), which can be written as an infinite sum, converges in mean square to Y_t.

Suppose $\{Y_t : t \in \mathbb{Z}\}$ is another stationary solution to the AR(1) process. Then,

$$Y_t = -\phi^{-1}Z_{t+1} + \phi^{-1}Y_{t+1}$$
$$Y_t = -\phi^{-1}Z_{t+1} + \phi^{-2}Z_{t+2} + \phi^{-2}Y_{t+2}$$
$$= \cdots$$
$$= -\phi^{-1}Z_{t+1} - \cdots - \phi^{-k}Z_{t+k} + \phi^{-k}Y_{t+k}$$

$$E\left[Y_t - \left(-\sum_{j=1}^{k} \phi^{-j}Z_{t+j}\right)\right]^2 = E\left[\phi^{-k}Y_{t+k}\right]^2 = \phi^{-2k}E\left[Y_{t+k}^2\right]$$

If Y_t is stationary, then $E[Y_t] < \infty$ is finite and independent of t.
Additionally, $|\phi| < 1$.

$$\therefore \phi^{-2k}E\left[Y_{t+k}^2\right] \to \quad as \quad k \to \infty$$

Because Y_t is equal to the mean square limit of X_t, the defined linear process is the unique stationary solution.

Example: MA(1) Process - Deriving the Linear Process

Because the MA(1) process is AR(∞) and causal,

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}$$

$$X_t = Z_t + \theta Z_{t-1}$$
$$\psi_0 = 1$$
$$\psi_1 = \theta$$
$$\psi_j = 0, \quad j > 1$$

Example: HW 2 #5 - MA(2) Process with Non-Zero Mean - Long-Run Variance, CLT

Let $\{Y_t : t \in \mathbb{Z}\}$ be a time series defined by the Second-Order Moving Average Model with non-zero mean given below.

$$Y_t - \mu = Z_t + \theta_1 Z_{t-1} + \theta_2 Z_{t-2}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

Assume that $\{Y_t\}$ is invertible.

a) Show that $\{Y_t\}$ is a linear process of the form

$$Y_t = \mu + \sum_{j=-\infty}^{\infty} \psi_j Z_{t-j}$$

That is, identify the coefficients ψ_j.

$$\psi_0 = \theta_0 = 1$$
$$\psi_1 = \theta_1$$
$$\psi_2 = \theta_2$$
$$\psi_j = 0, \quad for\ j > 2$$

b) Is the process $\{Y_t\}$ causal? Explain your reasoning in one or two sentences.

Yes. The MA(2) process $\{Y_t\}$ is expressed in terms of current and past noise terms Z_t by definition - hence it is causal.

c) Derive the covariance function $\gamma_Y(h)$.
Note that you can use the following formula which computes the covariance function $\gamma(h)$ for a generic MA(q) model.

$$\gamma(h) = \begin{cases} \sigma^2 \sum_{j=0}^{q-|h|} \theta_j \theta_{j+|h|} & if\ |h| \leq q \\ 0 & if\ |h| > q \end{cases}$$

For $q = 2$,

$$\gamma(0) = \sigma^2 \sum_{j=0}^{2} \theta_j^2 = (1 + \theta_1^2 + \theta_2^2)\sigma^2$$

$$\gamma(1) = \gamma(-1) = \sigma^2 \sum_{j=0}^{1} \theta_j \theta_{j+1} = (\theta_1 + \theta_1\theta_2)\sigma^2$$

$$\gamma(2) = \gamma(-2) = \sigma^2 \sum_{j=0}^{0} \theta_j \theta_{j+2} = \theta_2 \sigma^2$$

$$\gamma_Y(h) = \begin{cases} (1 + \theta_1^2 + \theta_2^2)\sigma^2 & if\ |h| = 0 \\ (\theta_1 + \theta_1\theta_2)\sigma^2 & if\ |h| = 1 \\ \theta_2 \sigma^2 & if\ |h| = 2 \\ 0 & if\ |h| > q \end{cases}$$

d) Compute the long-run variance of $\{Y_t\}$. Simplify the result as much as possible.

$$\nu = \sum_{h=-\infty}^{\infty} \gamma_Y(h) = \gamma_Y(0) + 2\gamma_Y(1) + 2\gamma_Y(2)$$
$$= (1 + \theta_1^2 + \theta_2^2)\sigma^2 + 2(\theta_1 + \theta_1\theta_2)\sigma^2 + 2\theta_2\sigma^2$$
$$= \sigma^2(1 + \theta_1^2 + \theta_2^2 + 2\theta_1\theta_2 + 2\theta_1 + 2\theta_2)$$
$$= \sigma^2(1 + \theta_1 + \theta_2)^2$$

Alternatively,

$$\nu = \sigma^2 \left(\sum_{j=-\infty}^{\infty} \Psi_j \right)^2 = \sigma^2(1 + \theta_1 + \theta_2)^2$$

e) Consider the sample average

$$\bar{Y}_n = \frac{1}{n} \sum_{i=1}^{n} Y_t$$

Where the Y_t's come from the MA(2) model defined in equation above.
Further assume that the noise structure is IID, i.e. $Z_t \sim IID(0, \sigma^2)$.
Identify the limiting distribution of

$$\sqrt{n}(\bar{Y}_n - \mu)$$

You may simply reference the appropriate theorem and compute the limiting distribution's mean and variance.

With the necessary assumption that the noise structure is iid, for Y_t's from the MA(2) process defined above, for the sample average specified as $\bar{Y}_n = \frac{1}{n}\sum_{i=1}^{n} Y_t$, the CLT for a strictly stationary linear process with iid noise indicates that the following limiting distribution holds.

$$\sqrt{n}(\bar{Y}_n - \mu) \xrightarrow{d} N(0, \nu) \equiv N(0, \sigma^2(1 + \theta_1 + \theta_2)^2)$$

FORECASTING STATIONARY TIME SERIES

Forecasting Stationary Time Series - Prediction Operator:
Consider the problem of predicting the values X_{n+h}, $h > 0$, of a stationary time series with known mean μ and (covariances) autocovariance function $\gamma_X(h)$, in terms of $\{X_n, \ldots, X_1\}$ up to time n (i.e. with predictions depending on more recent cases before prior cases).
Note that h indicates the number of steps ahead that we are forecasting.

The goal of forecasting is to find the linear combination of $(1, X_n, X_{n-1}, \ldots, X_1)$ that forecasts X_{n+h} with minimum mean square error.

Denote the best linear predictor based on a linear combination of $(1, X_n, X_{n-1}, \ldots, X_1)$ as $P_n(X_{n+h}|1, X_n, X_{n-1} \ldots, X_1) = P_n(X_{n+h}|\mathbf{X}_n) = P_n X_{n+h}$, where P_n is the prediction operator (i.e. filtration) used to forecast future values.
The subscript n of P_n indicates that n values $X_n, \ldots, X_1$ are used in the prediction.

$$\hat{X}_{n+h} = P_n X_{n+h} = a_0 + a_1 X_n + \cdots + a_n X_1$$

Note that the 1 term in the linear combination is used to estimate the mean μ, to introduce some bias into the estimator (akin to the intercept in regression).

Note that the prediction operator P_n is not an expression of conditional probability, but it is an operator that can be expressed in terms of conditional expectation.

Recall the classical additive time series decomposition.

$$Y_t = m_t + s_t + X_t$$
$$X_t \approx Y_t - \hat{m}_t - \hat{s}_t$$

For trend m_t, seasonality s_t and stationary time series X_t.
To forecast $\hat{Y}_{n+h}$, we compute

$$\hat{Y}_{n+h} = \hat{m}_{n+h} + \hat{s}_{n+h} + P(X_{n+h}|\mathbf{X}_n)$$

Minimizing Mean Square Error (Squared Loss), Forecasting Equations, Best Linear Predictor Under Squared Loss:
The objective function is given by

$$S(a_0, a_1, \ldots, a_n) = E\left[(X_{n+h} - a_0 - a_1 X_n - \cdots - a_n X_1)^2\right]$$

Note that the conditional expression above (involving the prediction operator) is expressed as a joint distribution.

Construct the gradient (partial derivatives) and set them to 0.

$$\frac{\delta}{\delta a_j} S(a_0, a_1, \ldots, a_n) = 0, \quad j = 0, \ldots, n$$

$$\frac{\delta}{\delta a_0} S(a_0, a_1, \ldots, a_n) = 2E\left[(X_{n+h} - a_0 - a_1 X_n - \cdots - a_n X_1)\right](-1) \stackrel{set}{=} 0$$

$$\frac{\delta}{\delta a_j} S(a_0, a_1, \ldots, a_n) = 2E\left[(X_{n+h} - a_0 - a_1 X_n - \cdots - a_n X_1)(-X_{n+1-j})\right] \stackrel{set}{=} 0, \quad j = 1, \ldots, n$$

$$E\left[X_{n+h} - a_0 - \sum_{i=1}^{n} a_i X_{n+1-i}\right] \stackrel{set}{=} 0$$

$$E\left[\left(X_{n+h} - a_0 - \sum_{i=1}^{n} a_i X_{n+1-i}\right)(X_{n+1-j})\right] \stackrel{set}{=} 0, \quad j = 1, \ldots, n$$

Note that the second normal equation will result in terms which are the expectation of product of X terms - i.e., covariances (gammas).
This is why stationarity is necessary - the stationary covariance functions will be used as the solution.

Let
i. $\mathbf{a}_n = (a_1, \ldots, a_n)^T$
ii. $\mathbf{\Gamma}_n = [\gamma(i-j)]_{i,j=1}^n$, the covariance matrix

iii. $\gamma_n(h) = [\gamma(h), \gamma(h+1), \ldots, \gamma(h+n-1)]^T$
Recall that the covariance matrix is symmetric, non-negative definite, and invertible.

Following the optimization procedure, we obtain a_0, $\mathbf{a}_n$ that satisfy the (forecasting) equations

$$a_0 = \mu\left(1 - \sum_{i=1}^{n} a_i\right)$$
$$\mathbf{\Gamma}_n \mathbf{a}_n = \gamma_n(h)$$

Note that a_0 relates to the mean, and $\mathbf{a}_n$ relates to the covariances.

The solution is given by

$$\mathbf{a}_n = \mathbf{\Gamma}_n^{-1}\gamma_n(h)$$

Note that this resembles a least squares solution - recall that the least squares equation in regression is given by $(X^TX)^{-1}X^TY$, where $\sigma^2(X^TX)^{-1}$ is the covariance matrix.

Suppose that we are given a specified stationary time series.
We can use the derived covariance function of the stationary time series to collect the terms necessary to construct the γ_n matrix and $\gamma_n(h)$ vector.

Thus, the best linear predictor is given by

$$\hat{X}_{n+h} = P_n X_{n+h} = a_0 + a_1 X_n + \cdots + a_n X_1$$

For a_0 and $\mathbf{a}_n$ that satisfy the forecasting equations

$$a_0 = \mu\left(1 - \sum_{i=1}^{n} a_i\right)$$
$$\mathbf{a}_n = \mathbf{\Gamma}_n^{-1}\gamma_n(h)$$

Note that if $\mu = 0$, we only need to solve $\mathbf{a}_n = \mathbf{\Gamma}_n^{-1}\gamma_n(h)$.

Note that other loss functions other than squared loss could be used (e.g., because squared loss is sensitive to outliers), but numerical methods would have to be used due to the lack of a closed form solution.

Mean Square Prediction Error (MSE):
The best linear predictor $\hat{X}_{n+h} = P_n(X_{n+h}|1, X_n, X_{n-1}\ldots, X_1) = P_n X_{n+h}$ is an estimator of X_{n+h}, but we require a variance-like measure to assess the quality of our forecast.

$$MSE = E\left[\left(X_{n+h} - \hat{X}_{n+h}\right)^2\right] = E\left[\left(X_{n+h} - P_n X_{n+h}\right)^2\right] = \gamma_X(0) - \mathbf{a}_n^T\gamma_n(h)$$

This is loosely analogous to a bias-variance decomposition - we are comparing our prediction to the lag-0 measure $\gamma_X(0)$ with an adjustment indicated by $\mathbf{a}_n^T\gamma_n(h)$.

Note that other loss functions other than squared loss could be used (e.g., because squared loss is sensitive to outliers), but numerical methods would have to be used due to the lack of a closed form solution.

Properties of $\mathbf{P_n X_{n+h}}$:

1. Expression for Best Linear Prediction Operator

   $P_n X_{n+h} = \mu + \sum_{i=1}^{n} a_i(X_{n+1-i} - \mu)$, where

   $$\mathbf{a}_n = \mathbf{\Gamma}_n^{-1}\gamma_n(h)$$

   Note that if $\mu = 0$, we only need to solve $\mathbf{a}_n = \mathbf{\Gamma}_n^{-1}\gamma_n(h)$.

2. The Best Linear Prediction Operator is Unbiased

   $$E[X_{n+h} - P_n X_{n+h}] = 0$$

3. The Innovations ("Residuals") are Orthogonal to X_j, $j = 1, 2, \ldots, n$ (predictors).

   $$E[(X_{n+h} - P_n X_{n+h})X_j] = 0, \quad j = 1, 2, \ldots, n$$

4. MSE (Mean Squared Prediction Error)

$$MSE = E\left[\left(X_{n+h} - \hat{X}_{n+h}\right)^2\right] = E\left[(X_{n+h} - P_n X_{n+h})^2\right] = \gamma_X(0) - \mathbf{a}_n^T \gamma_n(h)$$

General Case - Prediction of (Forecasting) Second-Order Random Variables, Best Linear Predictor (Prediction Operator) Under Squared Loss:
Let Y (a target value) and $W_n, \ldots, W_1$ (a filtration, known linear combination) be any random variables with finite second moments.

Assume that the following quantities are known.
i. $E[Y] = \mu_Y$
ii. $E[W_i] = \mu_{W_i}$
iii. $Cov(Y,Y)$, $Cov(Y, W_i)$, $Cov(W_i, W_j)$

Define

$$\mathbf{W}_n = (W_n, \ldots, W_1)^T$$
$$\mu_{\mathbf{W}_n} = E[\mathbf{W}_n] = (\mu_{W_n}, \ldots, \mu_{W_1})$$
$$\gamma_n = Cov(Y, \mathbf{W}_n) = [Cov(Y, W_n), Cov(Y, W_{n-1}, \ldots, Cov(Y, W_1)]^T$$
$$\mathbf{\Gamma}_{n\times n} = Cov(\mathbf{W}_n, \mathbf{W}_n) = [Cov(W_{n+1-i}, W_{n+1-j})]_{i,j=1}^n$$

The best linear predictor of Y in terms of the collection $(1, W_n, \ldots, W_1)$ is given by

$$\hat{Y} = P_n(Y|\mathbf{W}_n) = \mu_Y + \mathbf{a}^T(\mathbf{W}_n - \mu_{\mathbf{W}_n})$$

Where $\mathbf{a}$ is any solution of

$$\mathbf{\Gamma a} = \gamma_n$$

Given by

$$\mathbf{a} = \mathbf{W}\gamma_n$$

This formula encompasses more situations than the previous forecasting equations.
Note that for $\mu_Y = \mu_{\mathbf{W}_n} = 0$, then $\mathbf{a}^T\mathbf{W}_n$ is a linear combination as in the previous case.

Note that other loss functions other than squared loss could be used (e.g., because squared loss is sensitive to outliers), but numerical methods would have to be used due to the lack of a closed form solution.

General Case - Mean Square Prediction Error (MSE):
The best linear predictor $\hat{Y} = P(Y|\mathbf{W}_n) = \mu_Y + \mathbf{a}^T(\mathbf{W}_n - \mu_{\mathbf{W}_n})$ is an estimator of Y, but we require a variance-like measure to assess the quality of our forecast.

$$MSE = E\left[\left(Y - \hat{Y}\right)^2\right] = E\left[(Y - P_n(Y|\mathbf{W}_n)^2\right] = Var(Y) - \mathbf{a}^T\gamma_n$$

This is loosely analogous to a bias-variance decomposition - we are comparing our prediction to the measure $Var(Y)$ with an adjustment indicated by $\mathbf{a}^T\gamma_n$.

Note that other loss functions other than squared loss could be used (e.g., because squared loss is sensitive to outliers), but numerical methods would have to be used due to the lack of a closed form solution.

General Case - Properties of the Prediction Operator $\mathbf{P}(\cdot|\mathbf{W}_n)$:
Suppose that we have finite second moments of the random variables U and V, i.e. $E[U^2] < \infty$, $E[V^2] < \infty$.
Suppose also we have constants $\beta_1, \alpha_1, \ldots, \alpha_n$, and the covariance matrix,

$$\mathbf{\Gamma}_{n\times n} = Cov(\mathbf{W}_n, \mathbf{W}_n) = [Cov(W_{n+1-i}, W_{n+1-j})]_{i,j=1}^n$$

Note that $P_n X_{n+h} = P(X_{n+h}|\mathbf{X}_n)$

1. Expression for Best Linear Prediction Operator

$P_n(U|\mathbf{W}_n) = E[U] + \mathbf{a}^T(\mathbf{W}_n - E[\mathbf{W}_n])$, where

$$\mathbf{\Gamma a} = Cov(U, \mathbf{W}_n)$$

2. $P_n(U|\mathbf{W}_n) = E[U]$ if $Cov(U, \mathbf{W}_n) = \mathbf{0}_n$

   In the AR(p) model, this is the property that allows us to set $P(Z_j|\mathbf{X}_n) = 0 \; \forall \; j > n$ because Z_j terms are uncorrelated with past $X's$.

3. The Best Linear Prediction Operator is Unbiased

$$E[U - P_n(U|\mathbf{W}_n)] = 0$$

4. The Residuals are Orthogonal to $\mathbf{W}_n$

$$E[(U - P_n(U|\mathbf{W}_n))\mathbf{W}_n] = \mathbf{0}_n$$

   Recall that the residuals in regression are always orthogonal to X's (predictors).
   This is an analogous property related to the prediction operator, where $U - P(U|\mathbf{W}_n)$ can be thought of as a residual.

5. MSE (Mean Squared Prediction Error)

$$MSE = E[(U - P_n(U|\mathbf{W}_n))^2] = Var(U) - \mathbf{a}^T Cov(U, \mathbf{W}_n)$$

6. Linearity of the Prediction Operator (Conditional Expectations)

$$P_n(\alpha_1 U + \alpha_2 V + \beta|\mathbf{W}_n) = \alpha_1 P(U|\mathbf{W}_n) + \alpha_2 P(V|\mathbf{W}_n) + \beta$$

7. Filtration Property of the Prediction Operator - Predicting Training Values Yields Training Values

$$P_n\left(\sum_{i=1}^{n} \alpha_i W_i + \beta \middle| \mathbf{W}_n\right) = \sum_{i=1}^{n} \alpha_i W_i + \beta$$

8. Law of Iterated Predictions (Conditional Expectations)

   $P_n(U|\mathbf{W}_n) = P_n(P_n(U|\mathbf{W}_n, \mathbf{V}_n)|\mathbf{W}_n)$ if $\mathbf{V}_n$ is a random vector such that the components of $E\left[\mathbf{V}_n \mathbf{V}_n^T\right]$ are all finite.

   In other words, if we have another random vector $\mathbf{V}_n$ analogous to $\mathbf{W}_n$, prediction (conditioning on $\mathbf{W}_n$) of the prediction (conditioning on $\mathbf{W}_n$, $\mathbf{V}_n$).
   Recall that the prediction operator is related to conditional expectations - thus the law of iterated expectations holds.

Note:

While the following properties are useful for determining forecasts, they do not make calculating mean square prediction error more amenable.
Thus, it is still necessary to solve

$$\mathbf{\Gamma a} = Cov(U, \mathbf{W}_n)$$

if we would like to derive other quantities (e.g. the variance).

Computational Complexity of Forecasting - Recursive Algorithms (Durbin-Levinson, Innovations):
Recall that in regression, the computation of $(X^T X)^{-1} X^T Y$ involves computing a $p \times p$ inverse.

Note that solving

$$\mathbf{\Gamma}_{n \times n} \mathbf{a}_n = \gamma_n$$
$$\mathbf{a}_n = \mathbf{\Gamma}_{n \times n}^{-1} \gamma_n$$

Requires computing an $n \times n$ inverse, where n indicates the number of observations.

To avoid this computation, perhaps we can use a recursive algorithm, e.g. using the one-step forecast $P_n X_{n+1}$ to forecast the next step $P_{n+1} X_{n+2}$.

The two algorithms that are useful in this context are the Durbin-Levinson algorithm and the innovations algorithm.

Durbin-Levinson Algorithm - Recursively Solve AR(p) Prediction Operator:
The Durbin-Levinson algorithm is a recursive algorithm used to estimate the prediction operator of AR(p) models using recursion to simplify the matrix inversion.

Suppose we have the forecasting equations

$$\mathbf{\Gamma}_{n\times n}\phi_n = \gamma_n$$
$$\phi_n = \mathbf{\Gamma}_{n\times n}^{-1}\gamma_n$$

The Durbin-Levinson algorithm will be used to recursively estimate

$$\mathbf{a}_n \equiv \phi_n = (\phi_{n_1}, \phi_{n_2}, \ldots, \phi_{n_n})^T$$

for

$$P_n X_{n+1} = \phi_{n_1} X_n + \phi_{n_2} X_{n-1} + \cdots + \phi_{n_n} X_1$$

Innovations Algorithm - Recursively Solve MA(q), ARMA(p,q) Prediction Operator:
The innovations algorithm is a recursive algorithm used to estimate the prediction operator of MA(q) and ARMA(p,q) models using recursion to simplify the matrix inversion.

Define an innovation as $(X_n - \hat{X}_n)$, where

$$\hat{X}_n = \begin{cases} 0 & for\ n = 1 \\ P_{n-1}X_n & for\ n = 2, 3, \ldots \end{cases}$$

Note that an innovation resembles differencing observations to obtain noise terms (residuals).
The linear combinations will be orthogonal, like residuals in regression.
We use the innovations algorithm to recursively estimate

$$\theta_n = (\theta_{n_1}, \theta_{n_2}, \ldots, \theta_{n_n})^T$$

$$\hat{X}_{n+1} = \begin{cases} 0 & for\ n = 0 \\ \sum_{j=1}^{n} \theta_{n_j}(X_{n+1-j} - \hat{X}_{n+1-j}) & for\ n = 1, 2, \ldots \end{cases}$$

Note that in the linear combination above, θ is weighted by the innovations.

It turns out that all θ terms end up vanishing except for the most recent term.

$$\sum_{j=1}^{n} \theta_{n_j}(X_{n+1-j} - \hat{X}_{n+1-j}) = \theta_{n_1}(X_n - \hat{X}_n)$$

Examples:

Example: AR(1) One-Step Ahead Forecast, Matrix Method

a) Find the one-step ahead forecast for the AR(1) model.

Assume $\mu = 0$.
Consider the AR(1) process given by

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| < 1$$

Recall the covariance function is given by

$$\gamma_X(h) = \frac{\sigma^2 \phi^{|h|}}{1 - \phi^2}, \quad h \in \mathbb{Z}$$

Set up the forecasting equation(s).
This is a zero-mean process, so we only need to solve for $\mathbf{a}_n$.

$$\mathbf{a}_n = \mathbf{\Gamma}_n^{-1}\gamma_n(h)$$

$$\gamma_n(h) = [\gamma(h), \gamma(h+1), \ldots, \gamma(h+n-1)]^T$$

$$\gamma_n(h=1) = [\gamma(1), \gamma(2), \ldots, \gamma(n)]^T = \frac{\sigma^2}{1-\phi^2}\left[\phi, \phi^2, \ldots, \phi^h\right]^T$$

$$\mathbf{\Gamma}_n = [\gamma(i-j)]_{i,j=1}^n = \begin{bmatrix} \gamma(1-1) & \gamma(1-2) & \cdots & \gamma(1-n) \\ \gamma(2-1) & \gamma(2-2) & \cdots & \gamma(2-n) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(n-1) & \gamma(n-2) & \cdots & \gamma(n-n) \end{bmatrix}$$

Recall that the covariance matrix is symmetric, $\gamma_X(h) = \gamma_X(-h)$.
Pull out $\gamma(0)$ (the common factor) to obtain

$$\frac{\sigma^2}{1-\phi^2}\begin{bmatrix} 1 & \phi & \phi^2 & \cdots & \phi^{n-1} \\ \phi & 1 & \phi & \cdots & \phi^{n-2} \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ \phi^{n-1} & \phi^{n-2} & \phi^{n-3} & \cdots & 1 \end{bmatrix}$$

$$\mathbf{\Gamma}_n \mathbf{a}_n = \gamma_n(h=1)$$

$$\frac{\sigma^2}{1-\phi^2}\begin{bmatrix} 1 & \phi & \phi^2 & \cdots & \phi^{n-1} \\ \phi & 1 & \phi & \cdots & \phi^{n-2} \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ \phi^{n-1} & \phi^{n-2} & \phi^{n-3} & \cdots & 1 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \frac{\sigma^2}{1-\phi^2}\begin{bmatrix} \phi \\ \phi^2 \\ \vdots \\ \phi^n \end{bmatrix}$$

$$\begin{bmatrix} 1 & \phi & \phi^2 & \cdots & \phi^{n-1} \\ \phi & 1 & \phi & \cdots & \phi^{n-2} \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ \phi^{n-1} & \phi^{n-2} & \phi^{n-3} & \cdots & 1 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{bmatrix} = \begin{bmatrix} \phi \\ \phi^2 \\ \vdots \\ \phi^n \end{bmatrix}$$

By inspection, $a_1 = \phi$, $a_j = 0$, $1 < j \leq n$.

The one-step ahead forecast is given by

$$\hat{X}_{n+1} = P_n X_{n+1} = a_0 + a_1 X_n + \cdots + a_n X_1 = \phi X_n$$

Recall that the AR(1) process depends only on the previous case.

b) Find the mean square prediction error of the AR(1) one-step ahead forecast.

Note that

$$\gamma_X(0) = \frac{\sigma^2}{1-\phi^2}$$

$$\mathbf{a}_n^T = [\phi, 0, \cdots, 0]$$

$$\gamma_n(h=1) = [\gamma(1), \gamma(2), \ldots, \gamma(n)]^T = \frac{\sigma^2}{1-\phi^2}\left[\phi, \phi^2, \ldots, \phi^h\right]^T$$

$$E\left[X_{n+1} - P_n X_{n+1})^2\right] = \gamma_X(0) - \mathbf{a}_n^T \gamma_n(1) = \sigma^2$$

$$= \frac{\sigma^2}{1-\phi^2} - \frac{\sigma^2}{1-\phi^2}[\phi, 0, \ldots, 0]\left[\phi, \phi^2, \ldots, \phi^h\right]^T$$

$$= \frac{\sigma^2}{1-\phi^2}\left(1-\phi^2\right) = \sigma^2$$

In other words, the noise variance is the result of the mean square prediction error.

c) Find the one-step ahead forecast for the AR(1) model using the properties of the prediction operator P_n.

Assume $\mu = 0$.
Consider the AR(1) process given by

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| < 1$$

Recall the covariance function is given by

$$\gamma_X(h) = \frac{\sigma^2 \phi^{|h|}}{1 - \phi^2}, \quad h \in \mathbb{Z}$$

If we assume that we have a dataset, we can rewrite the AR(1) process as follows.

$$X_t = \phi X_{t-1} + Z_t$$
$$X_{n+1} = \phi X_n + Z_{n+1}$$

We can apply the prediction operator to the LHS and RHS, yielding

$$\begin{aligned} P_n(X_{n+1}|\mathbf{X}_n) &= P_n(\phi X_n + Z_{n+1}|\mathbf{X}_n) \\ &\overset{linearity}{=} \phi P_n(X_n|\mathbf{X}_n) + P_n(Z_{n+1}|\mathbf{X}_n) \\ &\overset{filtration}{=} \phi X_n \end{aligned}$$

Note that $P_n(Z_{n+1}|\mathbf{X}_n) = 0$ because white noise is uncorrelated with previous observations, given causality of the model.

Example: AR(1) - Two Step Ahead Forecast, Prediction Operator

Assume $\mu = 0$.
Consider the AR(1) process given by

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| < 1$$

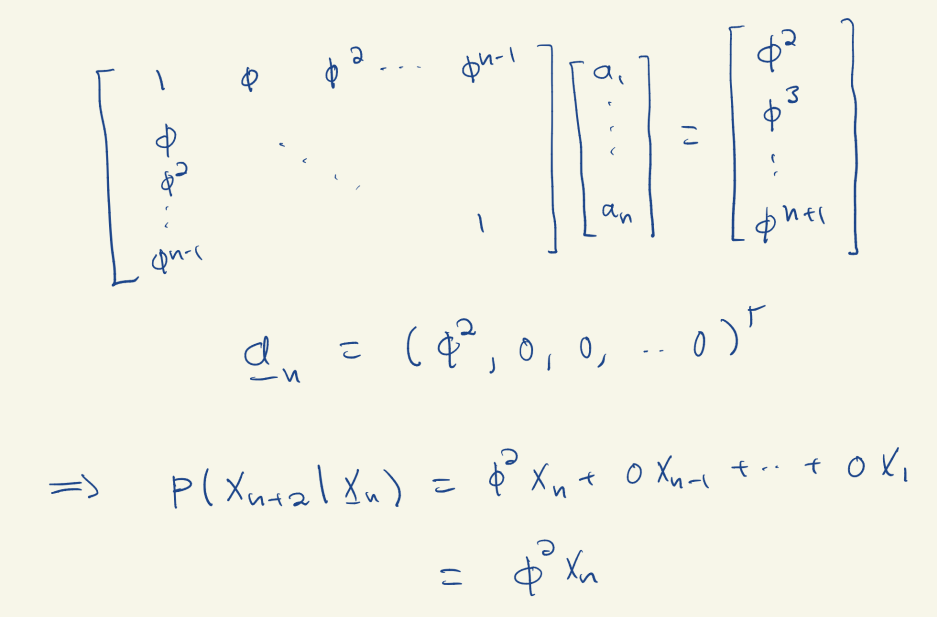
Find the two-step ahead forecast for the AR(1) process.

Because the AR(1) process is true for any $t \in \mathbb{Z}$, if we assume that we have a dataset, we can rewrite the AR(1) process as follows.

$$X_{n+2} = \phi X_{n+1} + Z_{n+2}$$

$$\begin{aligned} P_n(X_{n+2}|\mathbf{X}_n) &= P_n(\phi X_{n+1} + Z_{n+2}|\mathbf{X}_n) \\ &\overset{linearity}{=} \phi P_n(X_{n+1}|\mathbf{X}_n) \\ &= \phi P_n(\phi X_n + Z_{n+1}|\mathbf{X}_n) \\ &\overset{linearity}{=} \phi^2 X_n \end{aligned}$$

Note that the following matrix computation omits the factor $\frac{\sigma^2}{1-\phi^2}$ because it is cancelled on both sides.



Example: AR(1) - h Step Ahead Forecast, Prediction Operator

Find the h-step ahead forecast for the $AR(1)$.

Assume $\mu = 0$.
Consider the AR(1) process given by

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| < 1$$

Recall the covariance function is given by

$$\gamma_X(h) = \frac{\sigma^2 \phi^{|h|}}{1 - \phi^2}, \quad h \in \mathbb{Z}$$

Because the AR(1) process is true for any $t \in \mathbb{Z}$, if we assume that we have a dataset, we can rewrite the AR(1) process as follows.

$$X_{n+h} = \phi X_{n+h-1} + Z_{n+h}$$

We can apply the prediction operator to the LHS (and RHS).

$$\begin{aligned} P_n(X_{n+h}|\mathbf{X}_n) &= P_n(\phi X_{n+h-1} + Z_{n+h}|\mathbf{X}_n) \\ &\overset{linearity}{=} \phi P_n(\phi X_{n+h-1}|\mathbf{X}_n) + P(Z_{n+h}|\mathbf{X}_n) \\ &= \phi P_n(\phi X_{n+h-2} + Z_{n+h-1}|\mathbf{X}_n) \\ &\overset{linearity}{=} \phi^2 P_n(\phi X_{n+h-2}|\mathbf{X}_n) \\ &= \cdots \\ &\overset{filtration}{=} \phi^h P_n(X_{n+h-h}|\mathbf{X}_n) = \phi^h X_n \end{aligned}$$

Note that $P_n(Z_{n+j}|\mathbf{X}_n) = 0 \ \forall j$ because white noise is uncorrelated with previous observations under the assumption of causality.

If $\phi < 1$ (causal model), then ϕ^h will be small for large h.
In other words, we do not have much information for h-step ahead predictions for large h.
The best forecast will be for the first step.

Example: AR(p) - One Step Ahead Forecast, Prediction Operator

Find the one-step ahead forecast for the AR(p) process.

Suppose that $\{X_t\}$ is a stationary time series satisfying

$$X_t = \phi_1 X_{t-1} + \phi_2 X_{t-2} + \cdots + \phi_p X_{t-p} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

Assume that Z_t is uncorrelated with X_s, $s \leq t$.

Because the AR(p) process is true for any $t \in \mathbb{Z}$, if we assume that we have a dataset, we can rewrite the AR(p) process as follows.

$$X_{n+1} = \phi_1 X_n + \phi_2 X_{n-1} + \cdots + \phi_p X_{n+1-p} + Z_t$$

Assume $n > p$.

$$\begin{aligned} P_n(X_{n+1}|\mathbf{X}_n) &= P_n(\phi_1 X_n + \cdots + \phi_p X_{n+1-p} + Z_{n+1}|\mathbf{X}_n) \\ &= P_n(\phi_1 X_n + \cdots + \phi_p X_{n+1-p}|\mathbf{X}_n) + P_n(Z_{n+1}|\mathbf{X}_n) \\ &\overset{linearity,\ filtration}{=} \phi_1 X_n + \cdots + \phi_p X_{n+1-p} \end{aligned}$$

Example: Missing Value of an AR(1), Matrix Method

Assume $\mu = 0$.
Consider the AR(1) process given by

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| < 1$$

Recall the covariance function is given by

$$\gamma_X(h) = \frac{\sigma^2 \phi^{|h|}}{1 - \phi^2}, \quad h \in \mathbb{Z}$$

Suppose we observe X_1, X_3 and we want to predict X_2, where

$$P_n(X_2|X_3, X_1) = a_1 X_3 + a_2 X_1$$

$$\mathbf{\Gamma a} = \gamma_n$$
$$\mathbf{a} = \mathbf{\Gamma}^{-1}\gamma_n$$

$$\begin{bmatrix} Cov(X_1, X_1) & Cov(X_1, X_3) \\ Cov(X_3, X_1) & Cov(X_3, X_3) \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} Cov(X_2, X_3) \\ Cov(X_2, X_1) \end{bmatrix}$$

For convenience, we drop the component $\frac{\sigma^2}{1-\phi^2}$ from the covariance expressions, since all terms share the common component.

$$\begin{bmatrix} 1 & \phi^2 \\ \phi^2 & 1 \end{bmatrix} \begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \phi \\ \phi \end{bmatrix}$$

$$\begin{bmatrix} 1 & \phi^2 \\ \phi^2 & 1 \end{bmatrix}^{-1} = \frac{1}{1-\phi^4} \begin{bmatrix} 1 & -\phi^2 \\ -\phi^2 & 1 \end{bmatrix} = \frac{1}{(1+\phi^2)(1-\phi^2)} \begin{bmatrix} 1 & -\phi^2 \\ -\phi^2 & 1 \end{bmatrix}$$

$$\mathbf{a} = \frac{1}{(1+\phi^2)(1-\phi^2)} \begin{bmatrix} 1 & -\phi^2 \\ -\phi^2 & 1 \end{bmatrix} \begin{bmatrix} \phi \\ \phi \end{bmatrix} = \frac{1}{(1+\phi^2)(1-\phi^2)} \begin{bmatrix} \phi(1-\phi^2) \\ \phi(1-\phi^2) \end{bmatrix} = \frac{\phi}{(1+\phi^2)} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$$

$$\therefore P_n(X_2|x_1, x_3) = \frac{\phi}{1+\phi^2}(X_1 + X_3)$$

This is analogous to an average, depending on the weight of ϕ.

$$E[(X_2 - P_n(X_2|x_1, x_3))^2] = Var(X_2) - \mathbf{a}^T \gamma_n$$

$$= \frac{\sigma^2}{1-\phi^2} - \left[\frac{\phi}{1+\phi^2}, \quad \frac{\phi}{1+\phi^2}\right] \begin{bmatrix} \frac{\sigma^2\phi}{1-\phi^2} \\ \frac{\sigma^2\phi}{1-\phi^2} \end{bmatrix}$$

$$= \frac{\sigma^2}{1-\phi^2} - \frac{\phi}{1+\phi^2}\left(\frac{2\phi\sigma^2}{1-\phi^2}\right)$$

$$= \frac{(1+\phi^2)\sigma^2 - 2\phi^2\sigma^2}{(1-\phi^2)(1+\phi^2)}$$

$$= \frac{\sigma^2(1-\phi^2)}{(1-\phi^2)(1+\phi^2)}$$

$$= \frac{\sigma^2}{1+\phi^2}$$

Example: HW 2 #2 - AR(1) Process with Non-Zero Mean, $|\phi| < 1$ - Prediction Operator Properties, MSPE

Let $\{Y_t : t \in \mathbb{Z}\}$ be a time series defined by the First-Order Autoregressive Model with non-zero mean.

$$Y_t - \mu = \phi(Y_{t-1} - \mu) + Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| < 1$$

Note that $E[Y_t] = \mu \implies E[Y_t - \mu] = 0$.

a) Using properties of the prediction operator $P(\cdot|Y_n, Y_{n-1}, \ldots, Y_1)$, $h > 0$.

$$P_n[(Y_{n+h} - \mu|\mathbf{Y}_n)] = P_n[\phi(Y_{n+h-1} - \mu) + Z_t|\mathbf{Y}_n]$$

$$\overset{linearity}{=} \phi P_n[(Y_{n+h-1} - \mu)|\mathbf{Y}_n] + P_n(Z_t|\mathbf{Y}_n) = P_n[(Y_{n+h-1} - \mu)|\mathbf{Y}_n]$$

We note that Z_t is uncorrelated with past Y_t terms, meaning that $Cov(Z_t|\mathbf{Y}_n) = 0 \implies P_n(Z_t|\mathbf{Y}_n) = E[Z_t] = 0$.

$$= \phi P_n[\phi(Y_{n+h-2} - \mu) + Z_t|\mathbf{Y}_n] \overset{linearity}{=} \phi^2 P_n[(Y_{n+h-2} - \mu)|\mathbf{Y}_n] + 0$$

$$= \cdots$$

$$\therefore P_n[(Y_{n+h} - \mu)|\mathbf{Y}_n] = \phi^h P_n[(Y_n - \mu|\mathbf{Y}_n)] = \phi^h(Y_n - \mu)$$

$$P_n[(Y_{n+h} - \mu)|\mathbf{Y}_n] \overset{linearity}{=} P_n(Y_{n+h}|\mathbf{Y}_n) - \mu = \phi^h(Y_n - \mu)$$

$$\therefore P_n(Y_{n+h}|\mathbf{Y}_n) = \phi^h(Y_n - \mu) + \mu$$

In other words, we can apply the prediction operator to the de-trended data and add the trend (mean) back later.

b) Compute the mean square prediction error using the following formula.

$$MSPE = E[Y_{n+h} - P(Y_{n+h}|\mathbf{Y}_n) = \gamma(0) - \mathbf{a}_n^T\gamma_n$$

$$\mathbf{\Gamma}_{n\times n}\mathbf{a}_n = \gamma_n$$

The covariance function of Y_n will be the same as $X_n = Y_n - \mu$, the AR(1) process with zero mean, because covariances are not affected by scalar shifts.
Thus the covariance function of Y_n is given by

$$\gamma_Y(h) = \frac{\sigma^2\phi^{|h|}}{1-\phi^2}, \quad h \in \mathbb{Z}$$

$$\gamma_n(h = h) = [\gamma(h), \gamma(h+1), \ldots, \gamma(h+n-1)]^T = \frac{\sigma^2}{1-\phi^2}\left[\phi^h, \phi^{h+1}, \ldots, \phi^{h+n-1}\right]^T$$

$$\mathbf{\Gamma}_n = [\gamma(i-j)]_{i,j=1}^n = \begin{bmatrix} \gamma(1-1) & \gamma(1-2) & \cdots & \gamma(1-n) \\ \gamma(2-1) & \gamma(2-2) & \cdots & \gamma(2-n) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma(n-1) & \gamma(n-2) & \cdots & \gamma(n-n) \end{bmatrix}$$

Recall that the covariance matrix is symmetric, $\gamma_X(h) = \gamma_X(-h)$.
Pull out $\gamma(0)$ (the common factor) to obtain

$$\frac{\sigma^2}{1-\phi^2}\begin{bmatrix} 1 & \phi & \phi^2 & \cdots & \phi^{n-1} \\ \phi & 1 & \phi & \cdots & \phi^{n-2} \\ \vdots & \vdots & \cdots & \ddots & \vdots \\ \phi^{n-1} & \phi^{n-2} & \phi^{n-3} & \cdots & 1 \end{bmatrix}$$

$$\mathbf{\Gamma}_{n\times n}\mathbf{a}_n = \gamma_n$$

By inspection, $\mathbf{a}_n^T = [\phi^h, 0, 0, \ldots, 0]$.

$$\therefore MSPE = E[(Y_{n+h} - P_n(Y_{n+h}|\mathbf{Y}_n))^2] = \gamma_Y(0) - \mathbf{a}_n^T\gamma_{\mathbf{n}} = \frac{\sigma^2}{1-\phi^2}(1-\phi^{2h})$$

Example: HW 2 #3 - MA(1) Process - One-Step Ahead Forecast, MSPE

Let $\{X_t : t \in \mathbb{Z}\}$ be a time series defined by the First-Order Moving Average Model MA(1).

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0,\sigma^2), \quad t \in \mathbb{Z}$$

a) Derive the one-step ahead forecast X_3 based on X_2, X_1.
More specifically, derive $P(X_3|X_2, X_1)$, assuming the MA(1) process above.
To solve this problem, find coefficients $\mathbf{a}$ that satisfy $\mathbf{\Gamma}_n\mathbf{a}_n = \gamma_n$.

$$P_n = a_1X_2 + a_2X_1$$
$$\mathbf{\Gamma}_2\mathbf{a}_2 = \gamma_2$$

$$\mathbf{\Gamma}_2 = \begin{bmatrix} Cov(X_1,X_1) & Cov(X_1,X_2) \\ Cov(X_2,X_1) & Cov(X_2,X_2) \end{bmatrix} = \begin{bmatrix} \gamma(1-1)=\gamma(0) & \gamma(1-2)=\gamma(-1)=\gamma(1) \\ \gamma(2-1)=\gamma(1) & \gamma(2-2)=\gamma(0) \end{bmatrix}$$

The covariance function of the MA(1) process is given by

$$\gamma(h) = \begin{cases} \sigma^2(1+\theta^2) & if\ h=0 \\ \sigma^2\theta & if\ |h|=1 \\ 0 & if\ |h|>1 \end{cases}$$

$$\mathbf{\Gamma}_2 = \sigma^2\begin{bmatrix} 1+\theta^2 & \theta \\ \theta & 1+\theta^2 \end{bmatrix}$$
$$\gamma_2(h=1) = \begin{bmatrix} Cov(X_3,X_2) \\ Cov(X_3,X_1) \end{bmatrix} = \begin{bmatrix} \gamma(h=1) \\ \gamma(h+1=2) \end{bmatrix} = \sigma^2\begin{bmatrix} \theta \\ 0 \end{bmatrix}$$

$$\mathbf{\Gamma}_2\mathbf{a}_2 = \gamma_2$$

$$\sigma^2\begin{bmatrix} 1+\theta^2 & \theta \\ \theta & 1+\theta^2 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \sigma^2\begin{bmatrix} \theta \\ 0 \end{bmatrix}$$
$$\begin{bmatrix} 1+\theta^2 & \theta \\ \theta & 1+\theta^2 \end{bmatrix}\begin{bmatrix} a_1 \\ a_2 \end{bmatrix} = \begin{bmatrix} \theta \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} 1+\theta^2 & \theta \\ \theta & 1+\theta^2 \end{bmatrix}^{-1} = \frac{1}{(1+\theta^2)^2-\theta^2}\begin{bmatrix} 1+\theta^2 & -\theta \\ -\theta & 1+\theta^2 \end{bmatrix} = \frac{1}{(1+\theta^2+\theta)(1+\theta^2-\theta)}\begin{bmatrix} 1+\theta^2 & -\theta \\ -\theta & 1+\theta^2 \end{bmatrix}$$

$$\therefore \mathbf{a}_2 = \frac{1}{(1+\theta^2+\theta)(1+\theta^2-\theta)}\begin{bmatrix} 1+\theta^2 & -\theta \\ -\theta & 1+\theta^2 \end{bmatrix}\begin{bmatrix} \theta \\ 0 \end{bmatrix} = \frac{1}{(1+\theta^2+\theta)(1+\theta^2-\theta)}\begin{bmatrix} (1+\theta^2)(\theta) \\ (-\theta)(\theta) \end{bmatrix}$$
$$= \frac{\theta}{(1+\theta^2+\theta)(1+\theta^2-\theta)}\begin{bmatrix} 1+\theta^2 \\ -\theta \end{bmatrix}$$

b) Derive the mean square prediction error of $P(X_3|X_2, X_1)$.

$$MSPE = E[(X_3 - P(X_3|\mathbf{X}_2))^2] = Var(X_3) - \mathbf{a}_2^T\gamma_2 = \gamma(0) - \mathbf{a}_2^2\gamma_2$$

$$\mathbf{a}_2^T\gamma_2 = \frac{\theta\sigma^2}{(1+\theta^2+\theta)(1+\theta^2-\theta)}[(1+\theta^2),\ -\theta]\begin{bmatrix}\theta\\0\end{bmatrix} = \frac{\theta\sigma^2}{(1+\theta^2+\theta)(1+\theta^2-\theta)}(\theta(1+\theta^2)) = \frac{\theta^2(1+\theta^2)\sigma^2}{(1+\theta^2+\theta)(1+\theta^2-\theta)}$$

$$MSPE = E[(X_3 - P(X_3|\mathbf{X}_2))^2] = Var(X_3) - \mathbf{a}_2^T\gamma_2 = \gamma(0) - \mathbf{a}_2^2\gamma_2 = \sigma^2(1+\theta^2) - \frac{\theta^2(1+\theta^2)\sigma^2}{(1+\theta^2+\theta)(1+\theta^2-\theta)}$$

Example: HW 2 #4 - MA(1) Process - Simulation, Mean Square Prediction Error

Let $X_t : t \in \mathbb{Z}$ be a time series defined by

$$X_t = Z_t + 0.5Z_{t-1}, \quad Z_t \sim WN(0, 2^2), \quad t \in \mathbb{Z}$$

a) Consider using $n = 10$ cases $\{X_1, \ldots, X_{10}\}$ to perform a one-step ahead forecast $P(X_{11}|X_{10}, \ldots, X_1)$.
Find the coefficients $\mathbf{a}$ that satisfy $\Gamma_{10}\mathbf{a}_{10} = \gamma_{10}$.

i) Display the 10×10 matrix $\mathbf{\Gamma}_{10\times10}$ and the 10×1 vector γ_{10}.
You will have to write down each entry manually based on the true ACVF of the MA(1) process.

The ACVF of the MA(1) process is given by

$$\gamma(h) = \begin{cases} \sigma^2(1+\theta^2) = (4)(1.25) = 5 & if\ h = 0 \\ \sigma^2\theta = (4)(0.5) = 2 & if\ |h| = 1 \\ 0 & if\ |h| > 1 \end{cases}$$

```
2^2*(1+0.5^2)
## [1] 5
2^2*0.5
## [1] 2
```

```
r1 = c( c(5,2), rep(0,8) )
r2 = c( c(2,5,2), rep(0,7) )
r3 = c( rep(0,1), c(2,5,2), rep(0,6) )
r4 = c( rep(0,2), c(2,5,2), rep(0,5) )
r5 = c( rep(0,3), c(2,5,2), rep(0,4) )
r6 = c( rep(0,4), c(2,5,2), rep(0,3) )
r7 = c( rep(0,5), c(2,5,2), rep(0,2) )
r8 = c( rep(0,6), c(2,5,2), rep(0,1) )
r9 = c( rep(0,7), c(2,5,2) )
r10 = c( rep(0,8), c(2,5) )

( matGamma = rbind(r1,r2,r3,r4,r5,
                   r6,r7,r8,r9,r10) )
```

```
##      [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9] [,10]
## r1      5    2    0    0    0    0    0    0    0     0
## r2      2    5    2    0    0    0    0    0    0     0
## r3      0    2    5    2    0    0    0    0    0     0
## r4      0    0    2    5    2    0    0    0    0     0
## r5      0    0    0    2    5    2    0    0    0     0
## r6      0    0    0    0    2    5    2    0    0     0
## r7      0    0    0    0    0    2    5    2    0     0
## r8      0    0    0    0    0    0    2    5    2     0
## r9      0    0    0    0    0    0    0    2    5     2
## r10     0    0    0    0    0    0    0    0    2     5
```

```
( vecGamma = c(2,rep(0,9)) )
```

```
##  [1] 2 0 0 0 0 0 0 0 0 0
```

ii) Solve for $\mathbf{a}_{10}$ numerically.

```
( vecA = solve(matGamma) %*% vecGamma )
```

```
##               [,1]
##  [1,]  0.499999642
##  [2,] -0.249999106
##  [3,]  0.124998122
##  [4,] -0.062496200
##  [5,]  0.031242378
##  [6,] -0.015609745
##  [7,]  0.007781984
##  [8,] -0.003845216
##  [9,]  0.001831055
## [10,] -0.000732422
```

b) Compute the (theoretical) mean square prediction error of $P(X_{11}|X_{10},\ldots,X_1)$ numerically.

$$MSPE = E[(X_{11} - P(X_{11}|X_{10},\ldots,X_1))^2] = Var(X_{11}) - \mathbf{a}_{10}^T\gamma_{10} = \gamma(0) - \mathbf{a}_{10}^T\gamma_{10} = 4$$

```
5 - (t(vecA) %*% vecGamma)
```

```
##          [,1]
## [1,] 4.000001
```

c) Run a simulation to verify that the (theoretical) mean square prediction error from part (b) matches the empirical mean square prediction error.
To solve this problem, write a loop using 10000 iterations.

Simulate the MA(1) time series X_t with $n = 11$ observations.
Forecast the 11th observation using $X_{10}, X_9, \ldots, X_1$, i.e. compute $P_n(X_{11}|\mathbf{X}_{10})$.
Store both the forecasted 11th case and the simulated X_{11} for each iteration k.

Use the stored values to compute the empirical mean square prediction error and compare this result with the mean square prediction error from part (b).

```
set.seed(101)

# Define vectors of simulated values and forecasted values
sim11 = NULL
fore11 = NULL

for(k in 1:1e4) {

  # Simulate iid Zt N(0,4)
  vecZ = rnorm(51, 0, 2)

  # Define X vector
  vecX = NULL


  # Determine X vector
  for(i in 2:length(vecZ)) {
```

```
    vecX[i-1]  = vecZ[i] + 0.5*vecZ[i-1]
  }

  # Remove burn-in period
  vecX = vecX[35:45]

  # Save simulated and forecasted values
  sim11[k] = vecX[11]
  fore11[k] = t(vecA) %*%vecX[10:1]
}


mean((sim11 - fore11)^2)
```

```
## [1] 3.977532
```

The empirical mean squared prediction error is 3.978 is approximately equal to the theoretical mean squared prediction error of 4.000.

Example: HW 2 #6 - Prove the Linearity Property of the Second-Order Prediction Operator

Suppose that U and V are random variables such that $E[U^2] < \infty$ and $E[V^2] < \infty$.
Suppose that $\mathbf{\Gamma} = Cov(\mathbf{W}, \mathbf{W})$ and that β, α_1, and α_2 are constants.
Prove that

$$P_n(\alpha_1 U + \alpha_2 V + \beta|\mathbf{W}) = \alpha_1 P_n(U|\mathbf{W}) + \alpha_2 P_n(V|\mathbf{W}) + \beta$$

Assume the defining properties of the projection operator for $P_n(U|\mathbf{W})$ and $P_n(V|\mathbf{W}_n)$.
More specifically, you can assume,

$$P_n(U|\mathbf{W}) = E[U] + \mathbf{a}_1^T(\mathbf{W} - E[\mathbf{W}])$$
$$P_n(V|\mathbf{W}) = E[V] + \mathbf{a}_2^T(\mathbf{W} - E[\mathbf{W}])$$

Where,

$$\mathbf{\Gamma}\mathbf{a}_1 = Cov(U, \mathbf{W}) \implies \mathbf{a}_1 = \mathbf{\Gamma}^{-1} Cov(U, \mathbf{W})$$
$$\mathbf{\Gamma}\mathbf{a}_2 = Cov(U, \mathbf{W}) \implies \mathbf{a}_2 = \mathbf{\Gamma}^{-1} Cov(U, \mathbf{W})$$

Use the defining formula of the projection operator to simplify $P_n(\alpha_1 U + \alpha_2 V + \beta|\mathbf{W})$.

$$P_n(\alpha_1 U + \alpha_2 V + \beta|\mathbf{W}) = E[\alpha_1 U + \alpha_2 V + \beta] + \mathbf{a}^T(\mathbf{W} - E[\mathbf{W}])$$

For $\mathbf{\Gamma}\mathbf{a} = Cov(\alpha_1 U + \alpha_2 V + \beta, \mathbf{W})$.

Solve for $\mathbf{a}$ and apply linearity of covariance $Cov(\alpha_1 U + \alpha_2 V + \beta, \mathbf{W})$.

$$\mathbf{\Gamma}\mathbf{a} = Cov(\alpha_1 U + \alpha_2 V + \beta, \mathbf{W}) \overset{linearity}{=} \alpha_1 Cov(U, \mathbf{W}) + \alpha_2 Cov(V, \mathbf{W}) + Cov(\beta, \mathbf{W})$$

Because β is a constant, $Cov(\beta, \mathbf{W}) = 0$.

$$\overset{def\ \mathbf{a}_1,\ \mathbf{a}_2}{=} \alpha_1 \mathbf{\Gamma}\mathbf{a}_1 + \alpha_2 \mathbf{\Gamma}\mathbf{a}_2 = \mathbf{\Gamma}(\alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2)$$
$$\implies \mathbf{a} = (\alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2)$$

$$\begin{aligned} P_n(\alpha_1 U + \alpha_2 V + \beta|\mathbf{W}) &= E[\alpha_1 U + \alpha_2 V + \beta] + \mathbf{a}^T(\mathbf{W} - E[\mathbf{W}]) \\ &\overset{linearity}{=} \alpha_1 E[U] + \alpha_2 E[V] + \beta + (\alpha_1 \mathbf{a}_1 + \alpha_2 \mathbf{a}_2)^T(\mathbf{W} - E[\mathbf{W}]) \\ &= \alpha_1 E[U] + \alpha_2 E[V] + \beta + (\alpha \mathbf{a}_1 + \alpha_2 \mathbf{a}_2)^T(\mathbf{W} - E[\mathbf{W}]) \\ &= \alpha_1 E[U] + \alpha_2 E[V] + \beta + \left[\alpha_1 \mathbf{a}_1^T(\mathbf{W} - E[\mathbf{W}]) + \alpha_2 \mathbf{a}_2^T(\mathbf{W} - E[\mathbf{W}])\right] \end{aligned}$$

$$= \alpha_1 \left[E[U] + \mathbf{a}_1^T(\mathbf{W} - E[\mathbf{W}])\right] + \alpha_2 \left[E[V] + \mathbf{a}_2^T(\mathbf{W} - E[\mathbf{W}])\right] + \beta$$
$$= \alpha_1 P(U|\mathbf{W}) + \alpha_2 P(V|\mathbf{W}) + \beta$$

Example: Midterm #1

Consider the AR(1) model with $|\phi| < 1$.

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim IID(0, \sigma^2)$$

And consider the sample average

$$\bar{X}_n = \frac{1}{n}\sum_{i=1}^{n} X_t$$

Where the X_t's come from the AR(1) defined above.

a) Derive the unique stationary solution of the AR(1), ie derive the linear process of the form

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}$$

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \sim IID(0, \sigma^2), \quad |\phi| < 1$$
$$(1 - \phi B) X_t = Z_t$$

Define $\Phi(B) = 1 - \phi B$.

$$\Phi(B) X_t = Z_t$$

For $|\phi| < 1$ (which guarantees absolute convergence of constants), define the inverse operator

$$\Pi(B) = \sum_{j=0}^{\infty} (\phi B)^j \stackrel{!}{=} \frac{1}{1 - \phi B}$$

Although we note that division by an operator is not well-defined.

$$\Pi(B)\Phi(B) = \Phi(B)\Pi(B) = 1$$

$$\Pi(B)\Phi(B) X_t = \Pi(B) Z_t$$
$$X_t = \Pi(B) Z_t = \sum_{j=0}^{\infty} (\phi B)^j Z_t = \sum_{j=0}^{\infty} \phi^j B^j Z_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}$$

b) Derive the autocovariance function of the AR(1) using the linear process formula.

$$\gamma_X(h) = \sigma^2 \sum_{j=0}^{\infty} \psi_j \psi_{j+|h|}$$

We know that $\Psi_j = \phi^j$ for $j \geq 0$.

$$\gamma_X(h) = \sigma^2 \sum_{j=0}^{\infty} \psi_j \psi_{j+|h|} = \sigma^2 \sum_{j=0}^{\infty} \phi^j \phi^{j+|h|} = \sigma^2 \phi^{|h|} \sum_{j=0}^{\infty} (\phi^2)^j$$

Because $|\phi| < 1$, this implies that $|\phi^2| < 1$.

$$= \frac{\sigma^2 \phi^{|h|}}{1 - \phi^2}, \quad \forall\ h \in \mathbb{Z}$$

c) Derive the long-run variance of the causal AR(1) process.

$$\nu = \sigma^2 \left(\sum_{j=0}^{\infty} \psi_j \right)^2 = \sigma \left(\sum_{j=0}^{\infty} \phi^j \right)^2 = \frac{\sigma^2}{(1-\phi)^2}$$

Alternatively,

$$\nu = \sum_{h=-\infty}^{\infty} \gamma_X(h) = \gamma_X(0) + 2\sum_{h=1}^{\infty} \gamma_X(h) = \frac{\sigma^2}{1-\phi^2} + \frac{2\sigma^2}{1-\phi^2} \sum_{h=1}^{\infty} \phi^h = \frac{\sigma^2}{1-\phi^2} + \frac{2\sigma^2}{(1-\phi^2)} \cdot \frac{\phi}{(1-\phi)}$$

$$= \frac{(1-\phi)\sigma^2 + 2\sigma^2\phi}{(1-\phi)(1-\phi^2)} = \frac{\sigma^2(1+\phi)}{(1-\phi)(1-\phi)(1+\phi)} = \frac{\sigma^2}{(1-\phi)^2}$$

d) Identify the limiting distribution of $\sqrt{n}\bar{X}_n$.
You may simply reference the appropriate theorem and compute the limiting distribution's mean and variance.

The given AR(1) model is a 0 mean process.
Note also that because $Z_t \sim IID(0, \sigma^2)$, by the linear process filter, $\{X_t\}$ is a strictly stationary process.
Therefore,

$$CLT: \quad \sqrt{n}\bar{X}_n \xrightarrow{d} N(0, \nu), \quad \nu = \frac{\sigma^2}{(1-\phi)^2}$$

e) Derive the one-step ahead best linear predictor of the AR(1) process, which has the form

$$P(X_{n+1}|X_n, X_{n-1}, \ldots, X_1) = a_1 X_n + a_2 X_{n-1} + \cdots + a_n X_1$$

Also derive the mean square prediction error of the AR(1)'s one-step ahead best linear predictor.

$$P(X_{n+1}|\mathbf{X}_n) = P_n(\phi X_n + Z_{n+1}|\mathbf{X}_n)$$
$$\overset{linearity}{=} \phi P_n(X_n|\mathbf{X}_n) + P_n(Z_{n+1}|\mathbf{X}_n)$$

Note that by causality of the process, $Cov(Z_n, \mathbf{X}_n) = 0 \implies P_n(Z_{n+1}|\mathbf{X}_n) = E[Z_{n+1}] = 0$.

$$= \phi P_n(X_n|\mathbf{X}_n)$$
$$\overset{filtration}{=} \phi X_n$$

$$\therefore \mathbf{a}_n^T = \begin{bmatrix} a_1 & a_2 & \cdots & a_n \end{bmatrix} = \begin{bmatrix} \phi & 0 & \cdots & 0 \end{bmatrix}$$
$$\gamma_n = \begin{bmatrix} \gamma(1) & \gamma(2) & \cdots & \gamma(n) \end{bmatrix}^T$$

$$MSPE = E\left[(X_{n+1} - P(X_{n+1}|\mathbf{X}_n))^2\right] = \gamma_X(0) - \mathbf{a}_n^T \gamma_n$$
$$= \frac{\sigma^2}{1-\phi^2} - \phi\gamma_X(1)$$
$$= \frac{\sigma^2}{1-\phi^2} - \phi\frac{\sigma^2\phi}{1-\phi^2}$$
$$= \frac{\sigma^2(1-\phi^2)}{(1-\phi^2)}$$
$$= \sigma^2$$

ARMA MODELS

ARMA(p,q) Process:
The time series $\{X_t\}$ is an ARMA(p,q) process if $\{X_t\}$ is stationary and if and only if the process can be written in the form

$$X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \cdots + \theta_q Z_{t-q}, \quad Z_t \sim WN(0, \sigma^2)$$

For which the AR polynomial ($\Phi(u) = 1 - \phi_1 u - \cdots - \phi_p u^p$) and the MA polynomial ($\Theta(u) = 1 + \theta_1 u + \cdots + \theta_q u^q$) have no common factors.

Note that it is only constructive to talk about stationary ARMA(p,q) processes.

Parameter Redundancy:
If the ARMA(p,q) process $\{X_t\}$,

$$\Phi(B)X_t = \Theta(B)Z_t$$

has common roots in the AR and MA characteristic polynomials, we can "cancel" the roots out and consider a lower-order ARMA model.

For the following characteristic polynomials,

$$\Phi(u) = (1 - au)^k \Phi^*(u)$$
$$\Theta(u) = (1 - au)^k \Theta^*(u)$$

The lower order model is given by $\Phi^*(u)$ and $\Theta^*(u)$.

Thus, $\{X_t\}$ can be reduced to

$$\Phi^*(B)X_t = \Theta^*(B)Z_t$$

Which is an ARMA(p^*, q^*) process with orders $p^* = p - k$, and $q^* = q - k$.

For example, an ARMA(1,1) process would reduce to WN if the characteristic polynomials are the same, $X_t = Z_t$.
For example, an ARMA(3,2) process with one common factor would reduce to an ARMA(2,1) process.

Stationarity, Causality, and Invertibility of the ARMA(p,q) Process Using the Characteristic Polynomials:
Recall that we have used the backshift operator with the AR polynomial in the AR(1) and ARMA(1,1) processes for $\phi \neq \pm 1$ to find a power series representation of the process - i.e. by reciprocating $(1 - \phi B)$ to find an inverse operator in terms of a geometric series.

The stationarity condition $\phi \neq \pm 1$ can be restated as
i. $\Phi(u) \neq 0$ for $u = \pm 1$

$$\Phi(u) = 1 - \phi u = 0 \implies \phi = \frac{1}{u} = \pm 1 \quad for \quad u = \pm 1 \implies \phi = \pm 1$$

$$\therefore \Phi(u) \neq 0 \quad for \quad u = \pm 1 \implies \phi \neq \pm 1$$

ii. Equivalently, $u = \pm 1$ are not roots of $\Phi(u)$

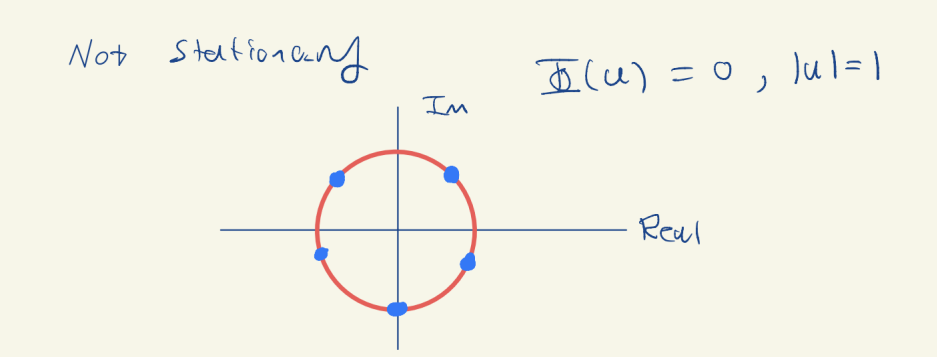
For higher-order ARMA(p,q) processes,

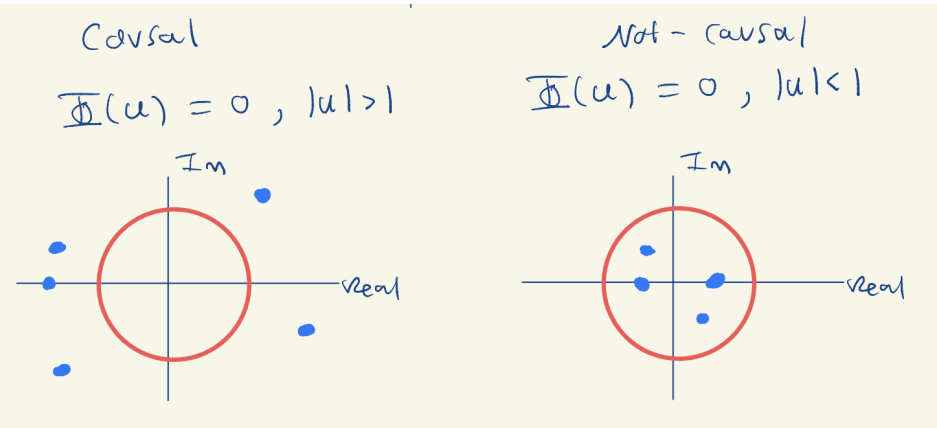
Note that for a complex number $u = x + yi$, $|u| = \sqrt{x^2 + y^2}$.

1. Supposing we have a non-zero AR polynomial for all complex u with the properties
i. $|u| = 1$, i.e., for point on the unit circle in the complex plane - the ARMA(p,q) process is stationary.
ii. $|u| < 1$, i.e., for points within the unit circle in the complex plane - the ARMA(p,q) process is causal.
Recall that the modulus in the complex numbers is akin to a distance formula.

2. If all complex roots u of $\Phi(u)$ have the properties
i. $|u| \neq 1$, i.e., roots are not on the unit circle in the complex plane - the ARMA(p,q) process is stationary.
ii. $|u| > 1$, i.e., roots fall outside the unit circle in the complex plane - the ARMA(p,q) process is causal.
Recall that the modulus in the complex numbers is akin to a distance formula.

Note that if the roots u have the property $|u| = 1$, the process is non-stationary.
Note that if the roots u have the property $|u| < 1$, the process is non-causal.

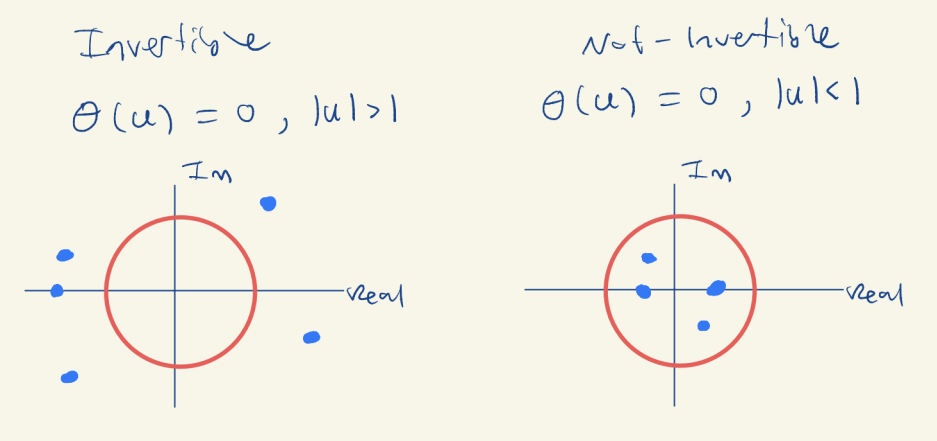




1. Supposing we have a non-zero MA polynomial for all complex u with the properties
i. $|u| < 1$, i.e., for points within the unit circle in the complex plane - the ARMA(p,q) process is invertible.

2. If all complex roots u of $\Theta(u)$ have the properties
i. $|u| > 1$, i.e., roots fall outside the unit circle in the complex plane - the ARMA(p,q) process is invertible.

Note that if the roots u have the property $|u| < 1$, the process is non-invertible.



Stationarity of the ARMA(p,q) Process:
The ARMA(p,q) process can be expressed as

$$\Phi(B)X_t = \Theta(B)Z_t$$

Where

$$\Phi(u) = 1 - \phi_1 u - \cdots - \phi_p u^p$$
$$\Theta(u) = 1 + \theta_1 u + \cdots + \theta_q u^q$$

The ARMA(p,q) process is stationary if

$$\Phi(u) = 1 - \phi_1 u - \cdots - \phi_p u^p \neq 0$$

for all complex u with $|u| = 1$.

In other words, if the AR polynomial does not have complex roots on the unit circle, the ARMA(p,q) process is stationary. Note that the set $\{u \in \mathbb{C} : |u| = 1\}$ indicates all u in the complex numbers such that we lie on the unit circle, i.e. $|u| = 1$.

Linear Process of the ARMA(p,q) Model, Unique Stationary Solution:

Theorem - Conditions for Establishing an Inverse Operator and Linear Process Representation of the ARMA(p,q):
If $\Phi(u) \neq 0$ for all u on the unit circle (i.e. the ARMA(p,q) process is stationary), then there exists a value $\delta > 0$ such that the AR polynomial $\Phi(u)$ can be inverted and expressed as an infinite series,

$$\frac{1}{\Phi(u)} = \sum_{j=-\infty}^{\infty} \eta_j u^j, \quad for \;\; 1 - \delta < |u| < 1 + \delta$$

for which absolute convergence of the η_j terms is guaranteed,

$$\sum_{j=-\infty}^{\infty} |\eta_j| < \infty$$

In other words, if the AR polynomial does not have complex roots on the unit circle, then we can invert the AR polynomial and identify a power series representation of the AR polynomial.
While it is not trivial to solve for this expression, its existence is guaranteed if the conditions are met.

Theorem Application - Defining the Inverse Operator and Linear Process of the ARMA(p,q) Model, Unique Stationary Solution:
Based on the conditions in the preceding theorem, we can (arbitrarily) define an inverse operator $\frac{1}{\Phi(u)}$ as the linear filter with absolutely summable coefficients as

$$\Pi(u) = \frac{1}{\Phi(u)} = \sum_{j=-\infty}^{\infty} \eta_j u^j$$

Then the ARMA(p,q) process can be expressed as

$$\Phi(B)X_t = \theta(B)Z_t$$
$$\Pi(B)\Phi(B)X_t = \Pi(B)\Theta(B)Z_t$$
$$X_t = \Pi(B)\Theta(B)Z_t$$

We know that $\Pi(B)$ (the inverse AR polynomial) and $\Theta(B)$ (the MA polynomial) have power series representations - thus there must be a linear process representation of the ARMA(p,q) model.

Generally, this linear process representation is too complex to be identified in most situations.
Thus we will define the linear process of the ARMA(p,q) model as

$$\Psi(u) = \Pi(u)\Theta(u) = \sum_{j=-\infty}^{\infty} \Psi_j u^j$$

Then the unique stationary solution X_t is given by

$$X_t = \Pi(B)\Theta(B)Z_t = \Psi(B)Z_t = \sum_{j=-\infty}^{\infty} \Psi_j Z_{t-j}$$

Note that this expression is not causal - causal solutions start summing at $j = 0$.

Existence and Uniqueness of the Stationary Solution of the ARMA(p,q) Process:
A stationary solution $\{X_t\}$ of the ARMA(p,q) process,

$$X_t - \phi X_t - 1 - \cdots - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \cdots + \theta_q Z_{t-q}, \quad Z_t \sim WN(0, \sigma^2)$$

exists (and is also the unique stationary solution) if and only if

$$\Phi(u) = 1 - \phi_1 u - \cdots - \phi_p u^p \neq 0$$

for all $|u| = 1$.

In other words, if the AR polynomial does not have complex roots on the unit circle, then there exists a unique stationary solution $\{X_t\}$ to the ARMA(p,q) process.

Causality of the ARMA(p,q) Process:
An ARMA(p,q) process is causal if there exists (absolutely summable coefficients of a linear process) $\{\Psi_j\}$ such that $\sum_{j=-\infty}^{\infty}|\Psi_j| < \infty$ and $X_t = \sum_{j=0}^{\infty}\Psi_j Z_{t-j}$, $\forall\ t$.
Note that the linear process representation of a causal X_t starts at $j = 0$.

Causality is equivalent to the condition

$$\Phi(u) = 1 - \phi_1 u - \cdots - \phi_p u^p \neq 0$$

for all $|u| \leq 1$.

In other words, if the AR polynomial does not have complex roots on or within the unit circle, then the ARMA(p,q) process is stationary and causal.

Invertibility of the ARMA(p,q) Process:
An ARMA(p,q) process is invertible if there exists (absolutely summable constants) $\{\pi_j\}$ such that $\sum_{j=0}^{\infty}|\pi_j| < \infty$ and

$$Z_t = \sum_{j=0}^{\infty}\pi_j X_{t-j}, \quad \forall\ t$$

Note that the linear process representation of an invertible Z_t starts at $j = 0$.

Invertibility is equivalent to the condition

$$\Theta(u) = 1 + \theta u + \cdots + \theta_q u^q \neq 0$$

for all $|u| \leq 1$.

In other words, if the MA polynomial does not have complex roots on or within the unit circle, then the ARMA(p,q) process is invertible.

Note that this is a similar idea as causality, but we represent Z_t as AR(∞).

Characterizing the Linear Process Coefficients ($\mathbf{\Psi_j}'s$) of the Causal ARMA(p,q) Process:
Assume that we have a causal ARMA(p,q) process,

$$X_t - \phi X_{t-1} - \cdots - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \cdots + \theta_q Z_{t-q}, \quad Z_t \sim WN(0, \sigma^2)$$

Because we know X_t is causal (guaranteeing absolute convergence of the ϕ terms), it can be expressed as the following linear process,

$$X_t = \sum_{j=0}^{\infty}\Psi_j Z_{t-j}, \quad \forall\ t$$

Substituting the linear process form for $\{X_t\}$ into the operator expression,

$$\Phi(B)X_t = \Theta(B)Z_t$$

$$\Phi(B)\sum_{j=0}^{\infty}\Psi_j Z_{t-j} = \Theta(B)Z_t$$

Instead of dealing with this expression directly, we will work with the polynomials (of u instead of the backshift operator B) and combine like terms.

$$(1 - \phi_1 u - \cdots - \phi_p u^p)(\psi_0 u^0 + \psi_1 u^1 + \psi_2 u^2 + \cdots) = (1 + \theta_1 u + \cdots + \theta_q u^q)$$

$$1 = \psi_1$$

$$\theta_1 = \psi_1 - \psi_0\phi_1$$
$$\theta_2 = \psi_2 - \psi_1\phi_1 - \psi_0\phi_2$$
$$\vdots$$

Thus the causal ARMA(p,q) linear process coefficients are given by

$$\psi_j = 0, \quad for\ j < 0$$
$$\psi_j - \sum_{k=1}^{p} \phi_k \psi_{j-k} = \theta_j, \quad for\ j = 0, 1, \ldots$$
$$\theta_0 := 1$$
$$\theta_j = 0, \quad for\ j > q$$

Given an abstract ARMA(p,q) model, we now have an algorithm to identify the linear process coefficients.

Non-Uniqueness of MA Models - Invertibility Model Constraint:
Consider the MA(1) process.

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

The ACF is given by

$$\rho_X(h) = \begin{cases} 1 & for\ h = 0 \\ \frac{\theta}{(1+\theta^2)} & for\ |h| = 1 \\ 0 & for\ |h| > 1 \end{cases}$$

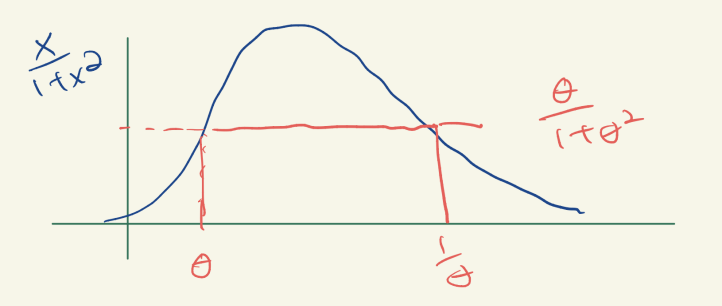
Consider a similar time series model with θ replaced by $\frac{1}{\theta}$,

$$X_t = Z_t + \frac{1}{\theta} Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

The ACF is identical to the ACF of the previous model.

$$\rho_X(h) = \begin{cases} 1 & for\ h = 0 \\ \frac{\left(\frac{1}{\theta}\right)}{\left(1+\frac{1}{\theta^2}\right)} \cdot \frac{\theta^2}{\theta^2} = \frac{\theta}{1+\theta^2} & for\ |h| = 1 \\ 0 & for\ |h| > 1 \end{cases}$$

That is, $\rho(h)$ of the MA(1) model is the same for θ and $1/\theta$ - we can find two pairs of (θ, σ^2) that lead to the same $\gamma_X(h)$, making it impossible to distinguish between the two models if we are only provided observed data X_t.



Thus, if we would like to consider the ACVF / ACF to be a fingerprint of a model, now two models have the same fingerprint.

Given an option, we would prefer to work with the invertible MA model, which has an AR(∞) representation.

ARMA(p,q) Modeling Constraints - Consider Causal, Invertible Models to Ensure Utility of ACVF / ACF:
To guarantee uniqueness of the ACVF / ACF (i.e., a one-to-one correspondence between a time series and an ACF), we must place constraints on the models we consider.

In practice, we focus on
i. stationary,

ii. causal, and
iii. invertible
ARMA(p,q) models.

ACVF / ACF and PACF of the ARMA(p,q) Process - Diagnostic Tools:

ACVF of the Causal ARMA(p,q) Processes:
Suppose that we have a causal ARMA(p,q) process X_t.
Because X_t is causal, it can be expressed as the following linear process,

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}, \quad \forall\ t$$

Recall that for any linear process, the ACVF is given by

$$\gamma_X(h) = \sigma^2 \sum_{j=-\infty}^{\infty} \Psi_j \Psi_{j+h}$$

For causal processes, $\psi_j = 0$, $\forall\ j < 0$, so the ACVF of a causal ARMA(p,q) processes is given by

$$\gamma_X(h) = E[X_{t+h} X_t] = \sigma^2 \sum_{j=0}^{\infty} \Psi_j \Psi_{j+|h|}$$

For instance, the ACVF of the ARMA(1,1) process is

$$\gamma_X(h) = \begin{cases} \sigma^2 \left[1 + \frac{(\phi+\theta)^2}{1-\phi^2}\right] & for\ h = 0 \\ \sigma^2 \left[(\phi+\theta) + \frac{(\phi+\theta)^2 \phi}{1-\phi^2}\right] & for\ |h| = 1 \\ \phi^{|h|-1} \gamma_X(1) & for\ |h| > 1 \end{cases}$$

Recall that the ACVF of an MA process eventually goes to zero because it is q-correlated.
Thus, past lag 1, the ACVF of the ARMA(1,1) process becomes more simplistic because it is primarily composed of the AR(1) component (i.e., note that a $\phi^{|h|-1}$ term appears in the ACVF past lag 1).

ACVF of the MA(q) Model - Identification via ACVF / ACF:
Consider the MA(q) model.

$$X_t = Z_t + \theta_1 Z_{t-1} + \cdots + \theta_q Z_{t-q}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

Recall that the MA(q) process is a zero-mean, stationary, causal process (because it is AR(∞)) - implying that it has a linear process representation.
Thus, the ACVF of the MA(q) process is

$$\gamma_X(h) = \begin{cases} \sigma^2 \sum_{j=0}^{q-|h|} \theta_j \theta_{j+|h|} & for\ |h| \leq q \\ 0 & for\ |h| > q \end{cases}$$

Note that terms start vanishing after $q - |h|$ (because the MA(q) process is q-correlated).
Thus the ACVF / ACF is a useful tool for identifying the order q of a pure MA(q) process.

Because the MA(q) process is q-correlated, a simulated MA(q) ACF will have peaks (in the positive or negative direction depending on the sign of the coefficients),
i. That are beyond the confidence bands up to lag $h = q$
ii. That are ≈ 0 (i.e., peaks below the confidence bands due to chance correlation structure) for lags $h > q$

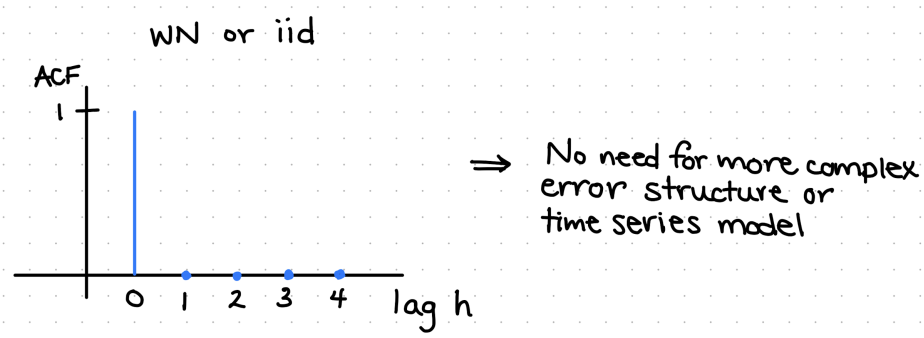
Note that the introduction of an AR component complicates the identification of q.

ACVF / ACF - Identifying Order q of a Pure MA(q) Process (and WN/IID Processes):

The ACVF / ACF is a useful tool for the identification of models and model order.

i. WN or IID - ACVF / ACF

WN or IID structure should demonstrate lag-0 correlation of 1, and no correlation structure beyond lag-0 (except for any random chance correlation structure).



ii. Pure MA(q) - ACVF / ACF

The ACVF / ACF is a useful tool for identifying the order q of a pure MA(q) process.

Because the MA(q) process is q-correlated, a simulated MA(q) ACF will have peaks (in the positive or negative direction depending on the sign of the coefficients),
i. That are beyond the confidence bands up to lag $h = q$
ii. That are ≈ 0 (i.e., peaks below the confidence bands due to chance correlation structure) for lags $h > q$

Note that the introduction of an AR component complicates the identification of q.

iii. AR(1) in ARMA(p,q)

When mixed in an ARMA(1,q) model, the AR(1) structure demonstrates heavy serial dependence that decreases gradually over lags, which causes the ACVF / ACF to taper down.
An AR(p) structure in an ARMA(p,q) will taper down even more slowly (and can be difficult to identify).

For instance, a random walk (for $|\phi| = 1$) will have an ACVF that is essentially uniform - each case will be perfectly correlated with the previous case, meaning that it never tapers off.

iv. ARMA(1,1)

Depending on the size and order of the parameters, there can be vastly different structures in the ACF.

v. Difficulties in Identification via ACVF / ACF - Higher Order AR(p) Processes

How do we identify higher order AR(p) processes (or ARMA(p,q) process with an AR(p) contribution)?
We require another tool to study the covariance structure of an AR process.
Additionally, we require methods to identify processes beyond a pure AR(p) or pure MA(p) process.

Partial Autocorrelation Function (PACF) - Identifying Order p of a Pure AR(p) Process:
Recall that the best linear predictor of X_{n+1} based on $\mathbf{X}_n$ $(X_n, X_{n-1}, \ldots, X_1)$ is of the form

$$P(X_{n+1}|X_n, \cdots, X_1) = a_1 X_n + \cdots + a_n X_1$$

Where $\mathbf{a}$ satisfies $\mathbf{\Gamma a} = \gamma$.

Suppose we have an AR(k) autoregressive model where the order of the model, k, is less than or equal to p, the unknown "true" order of the autoregressive process.
For an AR(k) model $k < p < n$, the BLP is given by

$$P(X_{n+1}|\mathbf{X}_n) = \phi_1 X_n + \phi_2 X_{n-1} + \cdots + \phi_k X_{n+1-k} + 0$$

Note that the BLP of the AR(k) model uses k ϕ's to predict X_{n+1}, i.e., the BLP of the AR(k) process truncates after k terms ($\phi_j = 0$ for $j > k$).

The PACF utilizes this structure to identify the "true" order p of an autoregressive process by calculating the BLP for all lags.

The partial autocorrelation function (PACF) of an ARMA(p,q) process is the function $\alpha(\cdot)$ defined by the equations

$$\alpha(0) = 1$$
$$\alpha(h) = \phi_{hh}, \quad h \geq 1$$

Where ϕ_{hh} is the last component of the solution (related to the lag h or $\gamma(h)$, term) of the following equation (analogous to the BLP formula)

$$\phi_h = \mathbf{\Gamma}_h^{-1}\gamma_h$$

For

$$\mathbf{\Gamma}_h = [\gamma(i-j)]_{i,j=1}^h$$
$$\gamma_h = [\gamma(1),\ \gamma(2), \cdots,\ \gamma(h)]^T$$

The PACF formula identifies the element $\phi_k \equiv \phi_{hh}$ in the best linear prediction for lag $h < p < n$ for all $h \in [1, n]$.
Note that a non-causal AR(p) will not produce the above previous result.
In R, we use the `pacf()` function.

Note that the PACF ϕ_{hh} for a causal ARMA $\{X_t\}$ is given by the BLP formula,

$$\phi_{hh} = Corr(X_h - P(X_h|X_{h-1}, \ldots, X_1), X_0 - P(X_0|X_{h-1}, \ldots, X_1))$$

PACF of a Causal AR(p) Process:
Consider the causal AR(p) process.

$$X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = Z_t, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}, \quad |\phi| < 1$$

If $h \geq p$, then the BLP of X_{h+1} in terms of $\{X_h, \ldots, X_1\}$ is

$$P(X_{h+1}|\mathbf{X}_h) = \phi_1 X_h + \phi_2 X_{h-1} + \cdots + \phi_p X_{h+1-p}$$

In other words, for $h \geq p$ the BLP will use up to p ϕ terms in its prediction.
Because we know that the PACF of a pure causal AR(p) process is 0 for all lags $h > p$, we can use the PACF to identify the order p of a pure AR(p) process.

We conclude that the PACF $\alpha(\cdot)$ of a pure AR(p) process (evaluated at lag p) has the properties

$$\alpha(h) = \phi_{hh}, \quad h < p$$
$$\alpha(p) = \phi_p, \quad for\ h = p$$
$$\alpha(h) = 0, \quad for\ h > p$$

Summary - Interpreting the ACF and PACF of Pure AR(p), MA(q), ARMA(p,q) Processes:
The symbol $\searrow 0$ indicates $\gamma(h) \to 0$ as $h \to \infty$.
The symbol $\searrow 0$, *exp* indicates an exponential decrease.

	AR(p)	MA(q)	ARMA(p,q)
ACF	$\searrow 0$, exp	Cut-off at $h = q$	$\searrow 0$
PACF	Cut-off at $h = p$	$\searrow 0$	$\searrow 0$

The ACF of a pure MA(q) is trivial because it is 0 at lag $h > q$.
The PACF of a pure AR(p) is trivial because it is 0 at lag $h > p$.

Forecasting Stationary ARMA Models:
The h-step ahead BLP can be computed for causal/non-causal, and invertible/non-invertible stationary ARMA models.

However, we will restrict our attention to causal and invertible stationary ARMA models to ensure
i. A unique ACF (invertibility)
ii. A meaningful PACF (causality)

Examples:

Example: Parameter Redundancy

Consider an ARMA(1,1) process.

$$X_t = 0.5X_{t-1} + Z_t - 0.5Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2)$$

Note that the characteristic polynomials are identical.

$$\phi(u) = 1 - 0.5u = 0$$
$$\theta(u) = 1 - 0.5u = 0$$

Both polynomials have the common complex root $u = 2$.

Thus the model is behaving like a lower-order model, (in this case) a WN process.

$$(1 - 0.5B)X_t = (1 - 0.5B)Z_t$$
$$X_t = Z_t$$

Check that $X_t = Z_t$ is a solution to the ARMA(1,1) model above.
Based on this relationship, substituting Z_{t-1} in for X_{t-1}, we obtain

$$X_t = 0.5Z_{t-1} + Z_t - 0.5Z_{t-1} = Z_t$$

From a data perspective - you would only detect a WN ACF.
But if you are given the expression of the model, you might initially consider it to be causal and invertible, though it is simply a degenerate WN process.

Example: Stationarity and Causality of the ARMA(2,1) Process

Determine if the following AR(2) process is stationary and causal.

$$X_t - 0.5X_{t-1} - 0.5X_{t-2} = Z_t + 0.9Z_{t-1}$$

$$\Phi(u) = 1 - 0.5u - 0.5u^2 = 0$$
$$(\Phi(u) = 1 - 0.5u - 0.5u^2 = 0)(-2)$$
$$u^2 + u - 2 = (u - 1)(u + 2) = 0$$

$$\Phi(u) = 1 - 0.5u - 0.5u^2 = (-0.5)(u - 1)(u + 2) = 0$$
$$u = \{1, -2\}$$
$$|u| = \{1, 2\} \geq 1$$

$|u| = 1$ is on the unit circle.
This ARMA(2,1) process is non-stationary.

On inspection, the two AR coefficients that are less than 1 make this ARMA(2,1) appear to be stationary, but it is not.

$$\theta(u) = 1 + 0.9u = (0.9)\left(u + \frac{10}{9}\right) = 0$$
$$u = -\frac{1}{0.9} = -\frac{10}{9}$$
$$|u| = \frac{10}{9}$$

The ARMA(2,1) process is invertible.

```
(roots = polyroot(c(1,-0.5, -0.5)))
## [1]  1-0i -2+0i
Mod(roots)
## [1] 1 2
```

Example: Stationarity and Causality of the AR(2) Process, Linear Process Coefficients

a) Determine if the following AR(2) process is stationary and causal.

$$X_t - 0.5X_{t-1} - 0.25X_{t-2} = Z_t$$

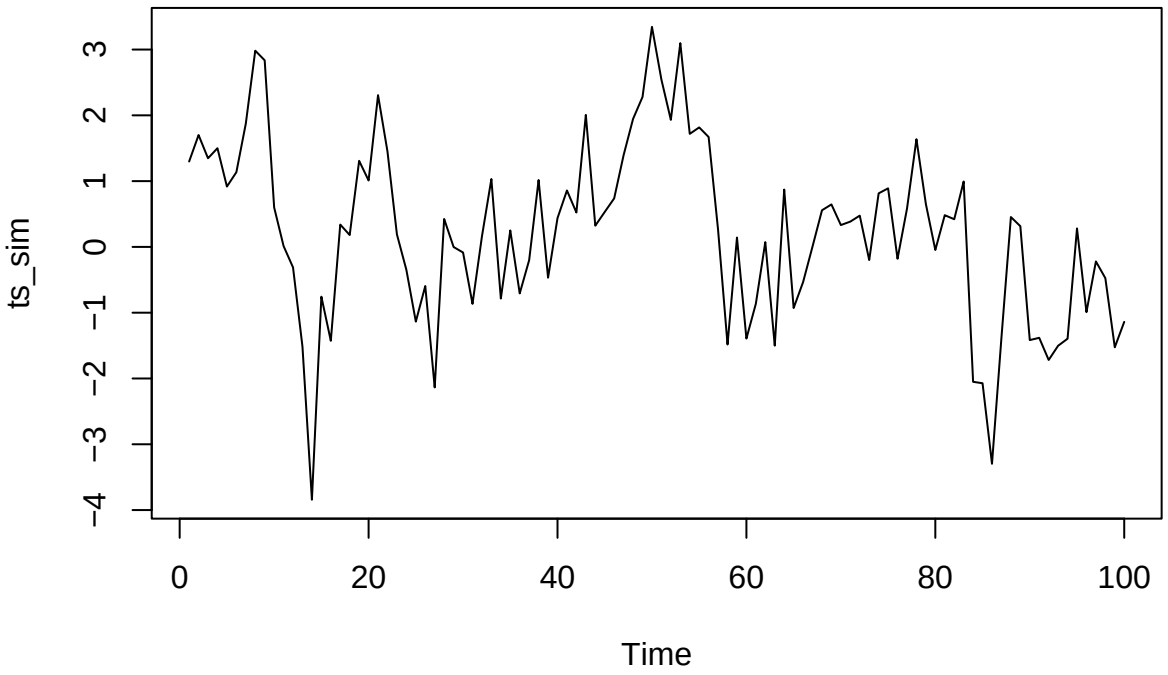
$$\Phi(u) = 1 - 0.5u - 0.25u^2 = (-0.25)(u - 1.1236)(u + 3.3236) = 0$$
$$u = \{1.1236, -3.236\}$$
$$|u| = \{1.1236, 3.236\} > 1$$

This AR(2) process is stationary and causal.

Intuitively, the coefficients are decreasing at a certain rate, which seems to indicate that the AR(2) process is stationary.

```
(roots = polyroot(c(1,-0.5, -0.25)))
## [1]  1.236068+0i -3.236068-0i
Mod(roots)
## [1] 1.236068 3.236068
```

```
# arima.sim requires ARIMA to be stationary and causal
# ar and ma parameters always positive
n = 100
ar_sim = c(0.5, 0.25)
ma_sim = c(1.25)

# not including ma = ma_sim
model_params = list(ar = ar_sim)
ts_sim = arima.sim(model = model_params, n = n)
plot(ts_sim, type = "l")
```



b) Find the linear process coefficients of a generic AR(2) process.

The causal ARMA(p,q) linear process is given by

$$\psi_j = 0, \quad for\ j < 0$$

$$\psi_j - \sum_{k=1}^{p} \phi_k \psi_{j-k} = \theta_j, \quad for\ j = 0, 1, \ldots$$

$$\theta_0 := 1$$

$$\theta_j = 0, \quad for\ j > q$$

For the AR(2) process, $p = 2$, $q = 0$,

For $j = 0$,

$$\psi_0 - \sum_{k=1}^{2} \phi_k \psi_{j-k} = \theta_0 := 1$$

$$\psi_0 - (\phi_1 \psi_{-1} + \phi_2 \psi_{-2}) = \psi_0 - (\phi_1(0) + \phi_2(0)) = 1$$

$$\psi_0 = 1$$

For $j = 1$,

$$\psi_1 - \sum_{k=1}^{2} \phi_k \psi_{j-k} = \theta_1 = 0$$

$$\psi_1 - (\phi_1 \psi_0 + \phi_2 \psi_{-1}) = (\phi_1(1) + \phi_2(0)) = 0$$

$$\psi_1 - \phi_1 = 0$$

$$\psi_1 = \phi_1$$

For $j = 2$,

$$\psi_2 - \sum_{k=1}^{2} \phi_k \psi_{j-k} = \theta_2 = 0$$

$$\psi_2 - (\phi_1\psi_1 + \phi_2\psi_0) = \psi_2 - (\psi_1^2 + \phi_2(1)) = 0$$

$$\psi_2 = \phi_1^2 + \phi_2$$

For $j = 3$,

$$\psi_3 - \sum_{k=1}^{2} \phi_k \psi_{j-k} = \theta_3 = 0$$

$$\psi_3 - (\phi_1\psi_2 - \phi_2\psi_1) = 0$$

$$\psi_3 - \phi_1(\phi_1^2 + \phi_2) - \phi_2\phi_1 = 0$$

$$\psi_3 = \phi_1(\phi_1^2 + \phi_2) + \phi_2\phi_1 = \phi^3 + 2\phi_2\phi_1$$

$$\psi_j = \phi_1\psi_{j-1} + \phi_2\psi_{j-2}, \quad j = 2, 3, \ldots$$

Note that attempting to derive the ACF of the generic AR(2) process would be difficult - this would involve an infinite sum of a recursive definition.
It is much easier to work with order 1 models, but the solution exists in principle.

Example: Stationarity and Causality of the AR(2) Process

Determine if the following AR(2) process is stationary and causal.

$$X_t = X_{t-1} + X_{t-2} + Z_t$$

$$\Phi(u) = 1 - u - u^2 = (-1)(u - 0.618)(u + 1.618) = 0$$

$$u = \{0.618, -1.618\}$$

$$|u| = \{0.618, 1.618\} < 1$$

Note that this model appears to resemble a random walk, e.g.,

$$X_t = X_{t-1} + Z_t$$

However it is stationary and non-causal.

```
(roots = polyroot(c(1,-1, -1)))
## [1]  0.618034+0i -1.618034-0i
Mod(roots)
## [1] 0.618034 1.618034
```

Example: ARMA(1,1) - Deriving the ACVF

Derive the ACVF of the ARMA(1,1) process.

Consider the ARMA(1,1) process.

$$X_t - \phi X_{t-1} = Z_t - \theta Z_{t-1}, \quad |\phi| < 1$$

Recall that the causal stationary solution of the ARMA(1,1) is

$$X_t = Z_t + (\phi + \theta) \sum_{j=1}^{\infty} \phi^{j-1} Z_{t-j}$$

Therefore,

$$\Psi_0 = 1$$
$$\Psi_j = (\phi+\theta)\phi^{j-1}, \quad for \;\; j \geq 1$$

$$\gamma_X(0) = \sigma^2 \sum_{j=0}^{\infty} \Psi_j^2 = \sigma^2 + \sigma^2 \sum_{j=1}^{\infty} (\phi+\theta)^2 \phi^{2(j-1)} = \sigma^2 + \sigma^2(\phi+\theta)^2 \sum_{j=1}^{\infty} \phi^{2j}\phi^{-2}$$
$$= \sigma^2 + \sigma^2(\phi+\theta)^2 \left(\frac{1}{\phi^2}\right) \sum_{j=1}^{\infty} (\phi^2)^j$$
$$= \sigma^2 \left[1 + \frac{(\phi+\theta)^2}{1-\phi^2}\right]$$

$$\gamma_X(1) = \sigma^2 \sum_{j=0}^{\infty} \Psi_j \Psi_{j+1} = \sigma^2 \Psi_0 \Psi_1 + \sigma^2 \sum_{j=1}^{\infty} \Psi_j \Psi_{j+1}$$
$$= \sigma^2(\phi+\theta) + \sigma^2 \sum_{j=1}^{\infty} (\phi+\theta)^2 \phi^{2j-1}$$
$$= \sigma^2(\phi+\theta) + \sigma^2(\phi+\theta)^2 \frac{1}{\phi} \sum_{j=1}^{\infty} (\phi^2)^j$$
$$= \sigma^2 \left[(\phi+\theta) + \frac{(\phi+\theta)^2\phi}{1-\phi^2}\right]$$

$$\gamma_X(h) = \sigma^2 \sum_{j=0}^{\infty} \Psi_j \Psi_{j+h} = \sigma^2 \left[(\phi+\theta))\phi^{(h-1)} + \sum_{j=1}^{\infty} (\phi+\theta)^2 \phi^{j-1}\phi^{j+h-1}\right]$$
$$= \sigma^2 \left[(\phi+\theta)\phi^{h-1} + (\phi+\theta)^2\phi^h \sum_{j=1}^{\infty} \phi^{2(j-1)}\right]$$
$$= \sigma^2 \left[(\phi+\theta)\phi^{h-1} + (\phi+\theta)^2\phi^h \left(\frac{1}{1-\phi^2}\right)\right]$$

Recall that the ACVF of an MA process eventually goes to zero because it is q-correlated.
Thus, past lag 1, the ACVF of the ARMA(1,1) process becomes more simplistic because it is primarily composed of the AR(1) component (i.e., note that a $\phi^{|h|-1}$ term appears in the ACVF past lag 1).

$$= \phi^{|h|-1}\gamma_X(1)$$

Thus, the ACVF of the ARMA(1,1) is

$$\gamma_X(h) = \begin{cases} \sigma^2 \left[1 + \frac{(\phi+\theta)^2}{1-\phi^2}\right] & for \; h = 0 \\ \sigma^2 \left[(\phi+\theta) + \frac{(\phi+\theta)^2\phi}{1-\phi^2}\right] & for \; |h| = 1 \\ \phi^{|h|-1}\gamma_X(1) & for \; |h| > 1 \end{cases}$$

Example: Non-Uniqueness of the MA(1) Model

Consider an MA(1) process.

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

For the non-invertible case $\theta = 5 > 1$, $\sigma^2 = 1$,

$$\gamma_X(h) = \begin{cases} \sigma^2(1+\theta^2) = 26 & for \; h = 0 \\ \sigma^2\theta = 5 & for \; |h| = 1 \\ 0 & for \; |h| > 1 \end{cases}$$

For the invertible case $\theta = \frac{1}{5} < 1$, $\sigma^2 = 25$,

$$\gamma_X(h) = \begin{cases} \sigma^2(1+\theta^2) = 26 & for\ h = 0 \\ \sigma^2\theta = 5 & for\ |h| = 1 \\ 0 & for\ |h| > 1 \end{cases}$$

Example: PACF of MA(1) Process - Atypical Application of PACF

Consider the MA(1) process given by

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

The PACF is given by

$$\alpha(h) = \phi_{hh} = \frac{-(-\theta)^h}{(1 + \theta^2 + \cdots + \theta^{2h})}$$

Note that the PACF decreases for increasing lag h if the model is invertible ($|\theta| < 1$).

Example: ARMA(2,2) - Comprehensive Example

Consider an ARMA(2,2) process given by

$$X_t = 0.4X_{t-1} + 0.45X_{t-2} + Z_t + Z_{t-1} + 0.25Z_{t-2}, \quad Z_t \sim WN(0, \sigma^2), \quad t \in \mathbb{Z}$$

a) Determine if the process is stationary, causal, and invertible.

The characteristic polynomials are given by

$$\Phi(u) = 1 - 0.4u - 0.45u^2 = (-0.45)(1 + 0.5u)(1 - 0.9u)$$

$$\Phi(u) = 1 - 0.4u - 0.45u^2 \overset{*-20}{=} 9u^2 + 8u - 20$$

$$u = \frac{-8 \pm \sqrt{64 - (4)(9)(-20)}}{2(9)} = \frac{-8 \pm \sqrt{784}}{2(9)} = \frac{-8 \pm 28}{18} = \left\{\frac{10}{9}, -2\right\}$$

$$\Theta(u) = 1 + u + 0.25u^2 = (1 + 0.5u)^2$$

$$\Theta(u) = 1 + u + 0.25u^2 \overset{*4}{=} u^2 + 4u + 4$$

$$u = \frac{-4 \pm \sqrt{16 - (4)(1)(4)}}{2(1)} = \frac{-4}{2} = \{-2, -2\}$$

The characteristic polynomials have the common root -2, and this part of the polynomial will cancel. Thus the ARMA(2,2) reduces to an ARMA(1,1) with

$$\Phi^*(u) = (-0.45)(1 - 0.9u)$$

$$u = \frac{10}{9}$$

$$|u| = \frac{10}{9} > 1$$

Thus the process is causal ($u > 1$) and stationary ($u \neq 1$).

$$\Theta^*(u) = (1 + 0.5u)$$

$$u = -2$$

$$|u| = 2 > 1$$

Thus the process is invertible ($u > 1$).

Example: PACF / ACF Interpretation via Simulation

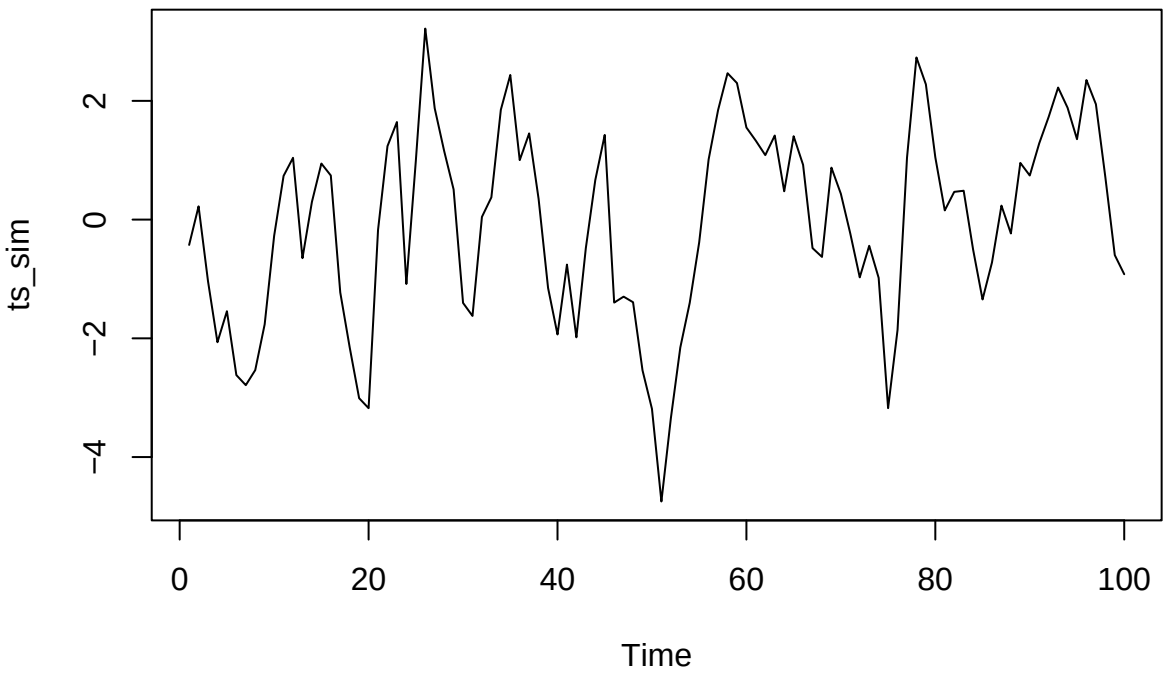
a) AR(2)

```r
# non-stationary
n = 100
ar_sim = c(0.5, 0.5)

( roots = polyroot(c(1,-ar_sim)) )
## [1]  1-0i -2+0i
Mod(roots)
## [1] 1 2
```

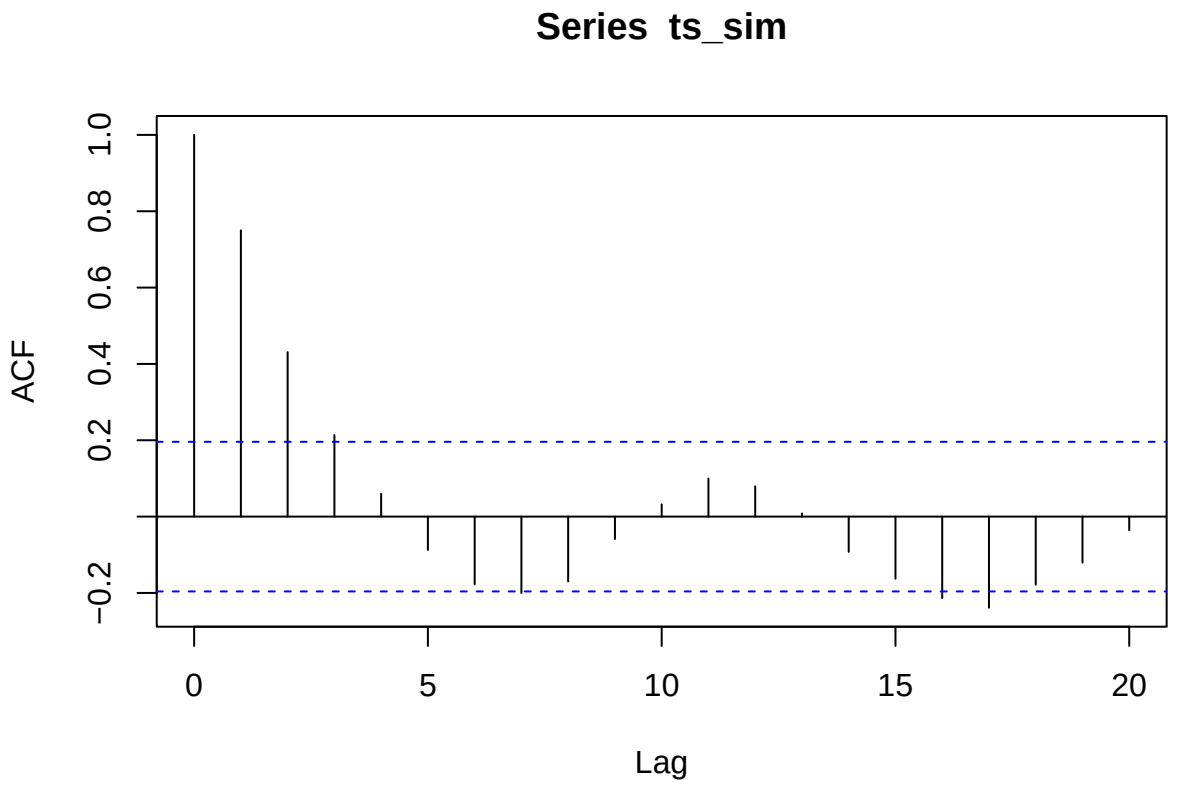
b) ARMA(2,2)

```r
n = 100
ar_sim = c(0.5, 0.1)
ma_sim = c(0.5, 0.1)

(roots = polyroot(c(1,-ar_sim)))
## [1]  1.531129+0i -6.531129-0i
Mod(roots)
## [1] 1.531129 6.531129

(roots = polyroot(c(1,ma_sim)))
## [1] -2.5+1.936492i -2.5-1.936492i
Mod(roots)
## [1] 3.162278 3.162278
```

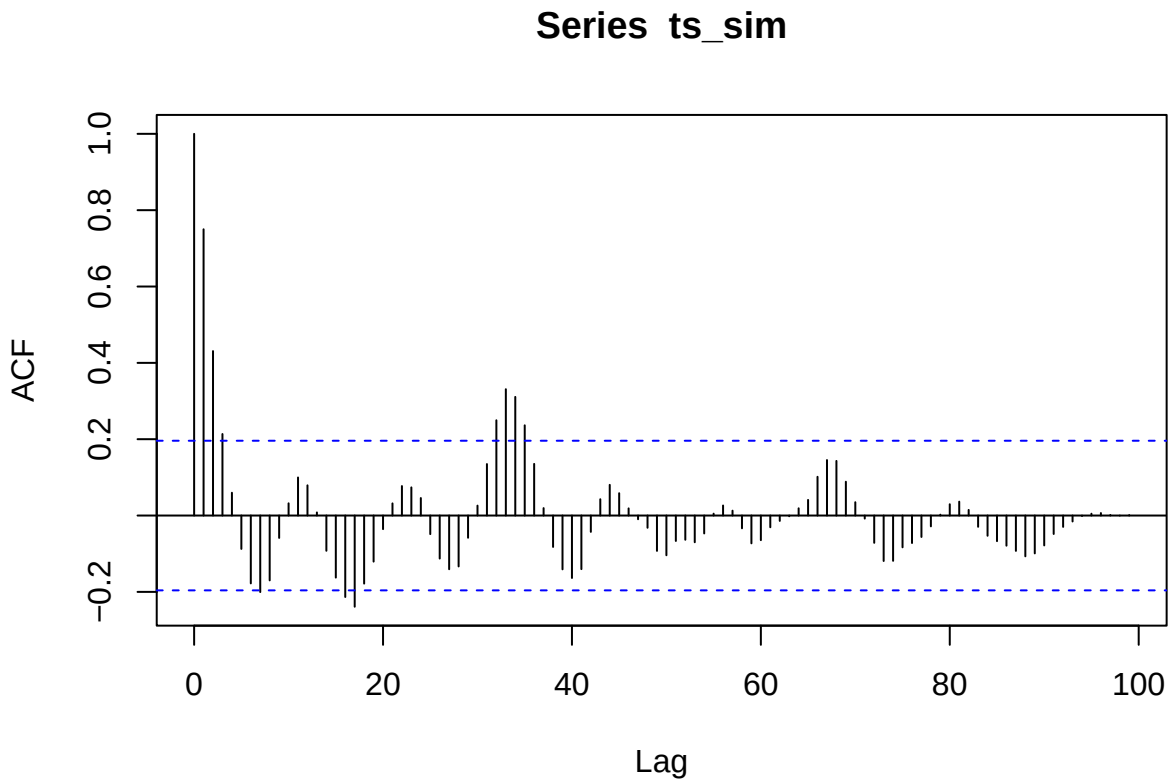
```r
model_params = list(ar = ar_sim, ma = ma_sim)
ts_sim = arima.sim(model = model_params, n = n)
plot(ts_sim, type = "l")
```



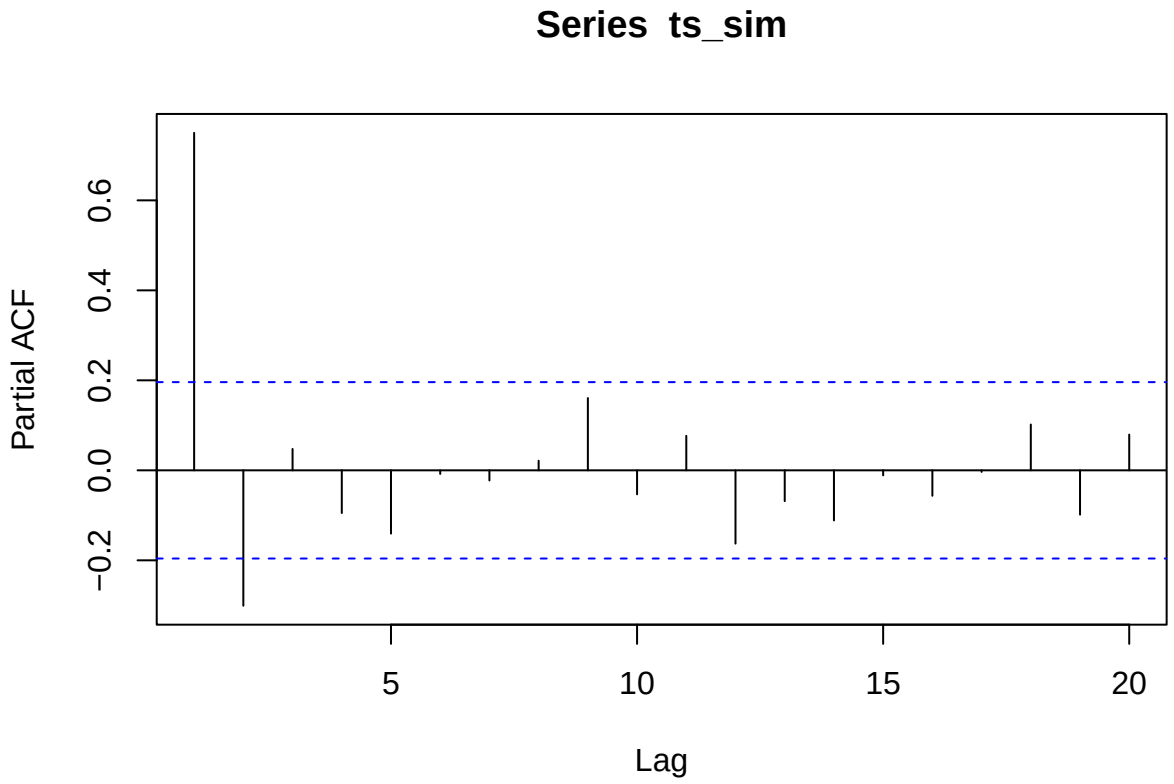
```r
# detecting 3-4 lags although only ARMA(2,2)
# the taper and periodicity (which reduces asymptotically) implies an AR component
acf(ts_sim)
```




```r
acf(ts_sim,
    lag.max=100)
```




```
# detecting lag 1,2 contribution
# Note that lag 1 contribution (coefficient) stronger
# the lag 2 component = phi_2
pacf(ts_sim)
```



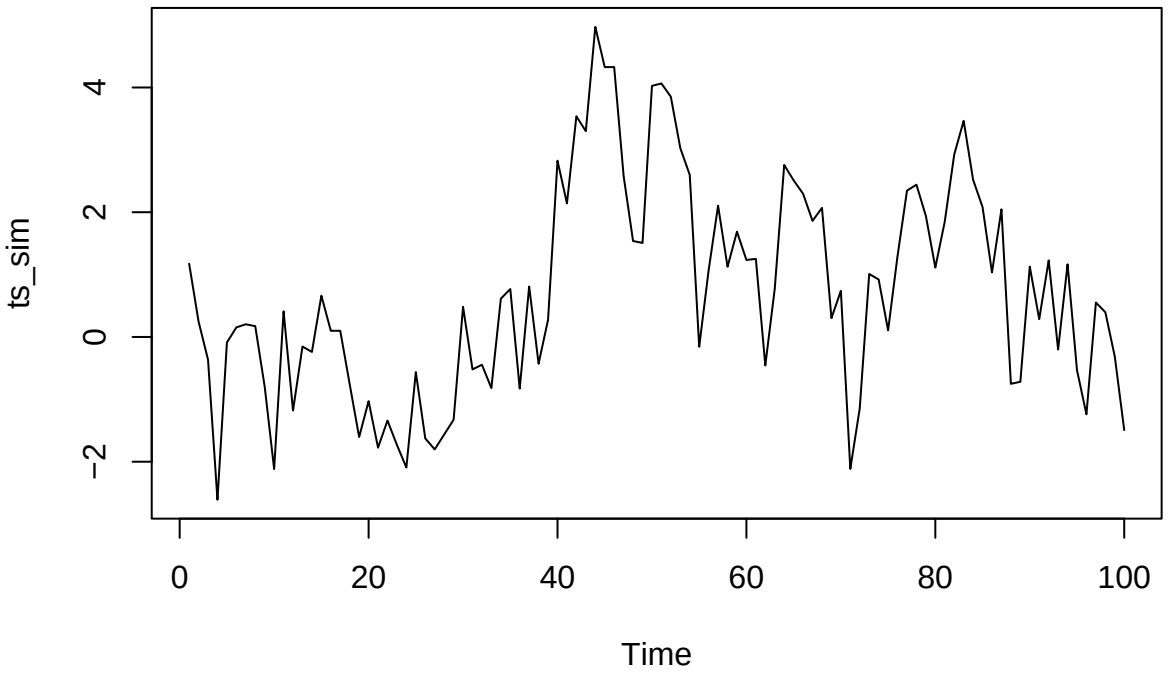
c) ARMA(3,2)

```
# non-stationary
n = 100
ar_sim = c(0.5, 0.4, 0.1)
ma_sim = c(0.5, 0.1)

(roots = polyroot(c(1,-ar_sim)))
## [1]  1.0+0.000000i -2.5+1.936492i -2.5-1.936492i
Mod(roots)
## [1] 1.000000 3.162278 3.162278

(roots = polyroot(c(1,ma_sim)))
## [1] -2.5+1.936492i -2.5-1.936492i
Mod(roots)
## [1] 3.162278 3.162278
```

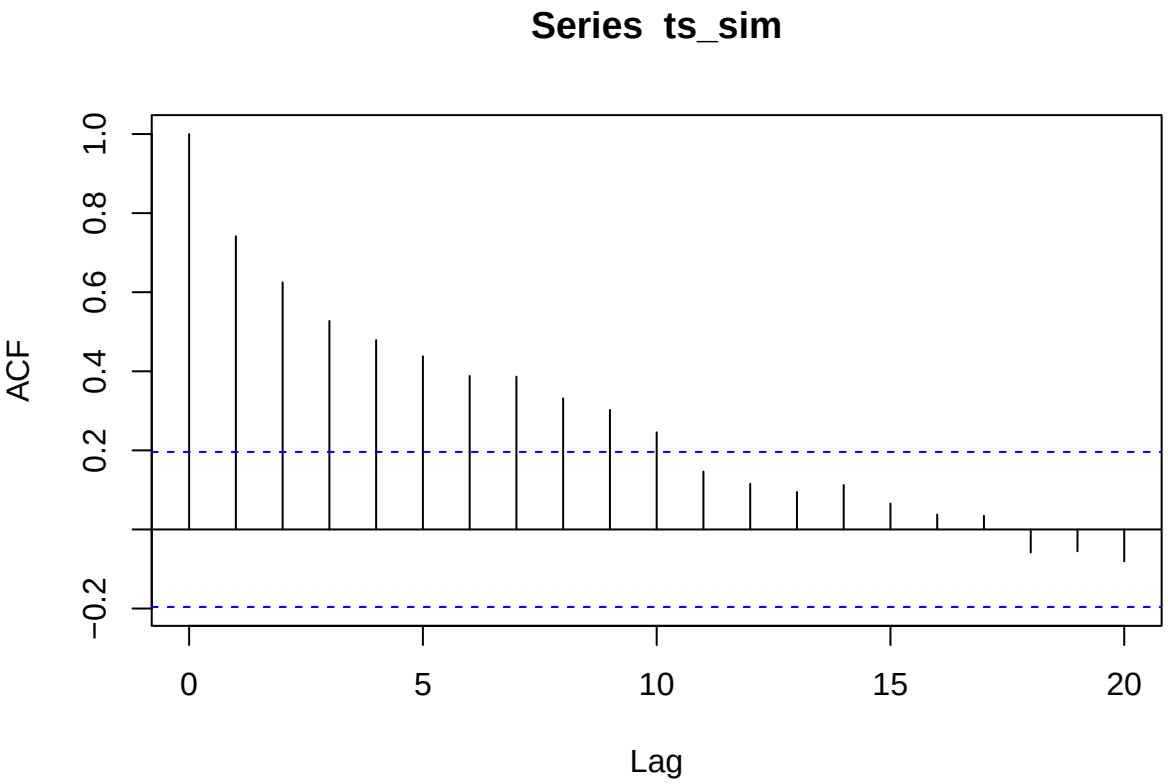
d) ARMA(3,2)

```
n = 100
ar_sim = c(0.3, 0.4, 0.1)
ma_sim = c(0.5, 0.1)

(roots = polyroot(c(1,-ar_sim)))
## [1]  1.133743+0.000000i -2.566871+1.493826i -2.566871-1.493826i
Mod(roots)
## [1] 1.133743 2.969906 2.969906

(roots = polyroot(c(1,ma_sim)))
## [1] -2.5+1.936492i -2.5-1.936492i
Mod(roots)
## [1] 3.162278 3.162278
```

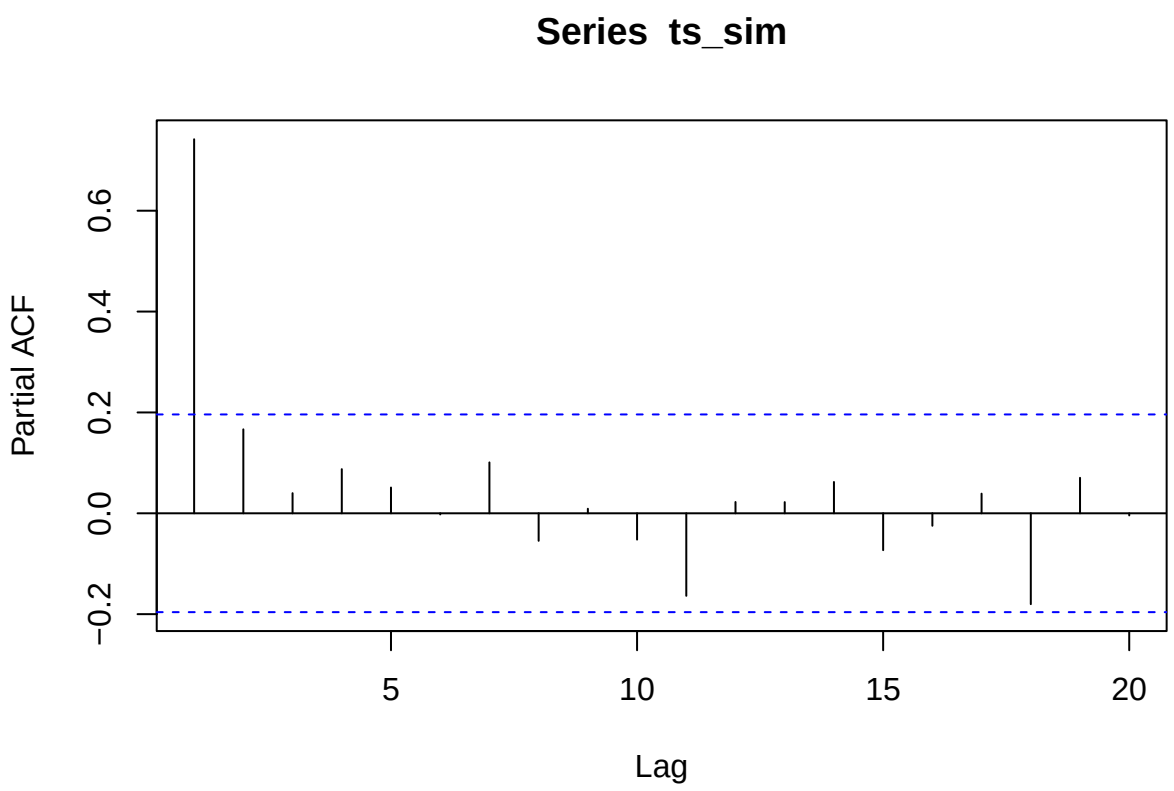
```
model_params = list(ar = ar_sim, ma = ma_sim)
ts_sim = arima.sim(model = model_params, n = n)
plot(ts_sim, type = "l")
```




```
# Tapers off - seems like AR, not MA
acf(ts_sim)
```



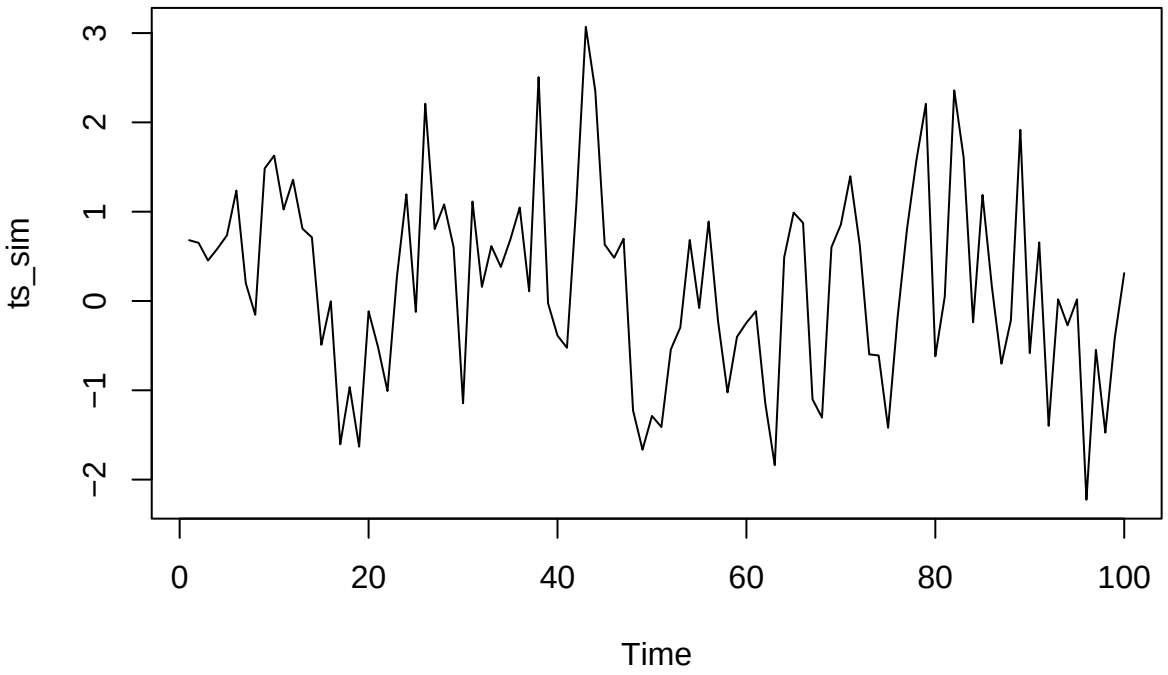
```r
# PACF within confidence bounds seems to indicate MA dominance, but this is ARMA(3,2)
# Not detecting contribution beyond lag 1 - maybe lag 2,3 coefficients too small
# Depends on luck with data set if it is detected
pacf(ts_sim)
```



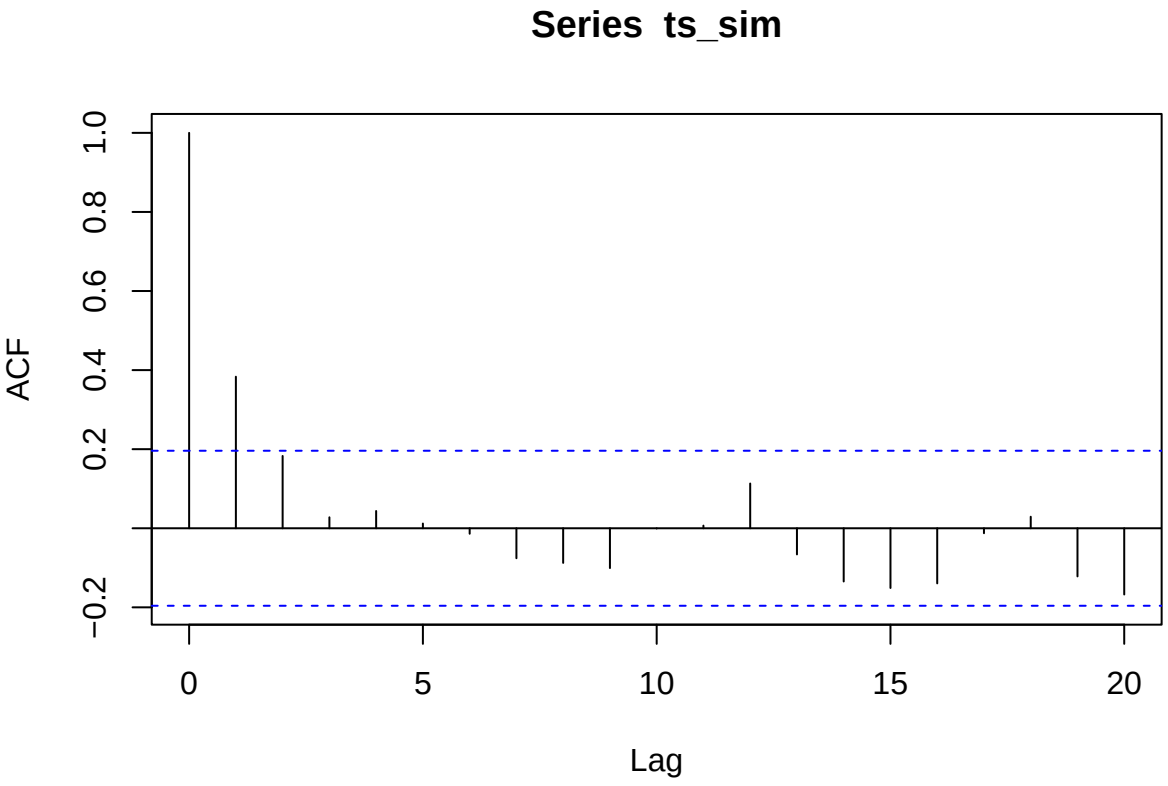
e) AR(1)

```r
# Pure AR
n = 100
ar_sim = c(0.5)

(roots = polyroot(c(1,-ar_sim)))
## [1] 2+0i
Mod(roots)
## [1] 2
```

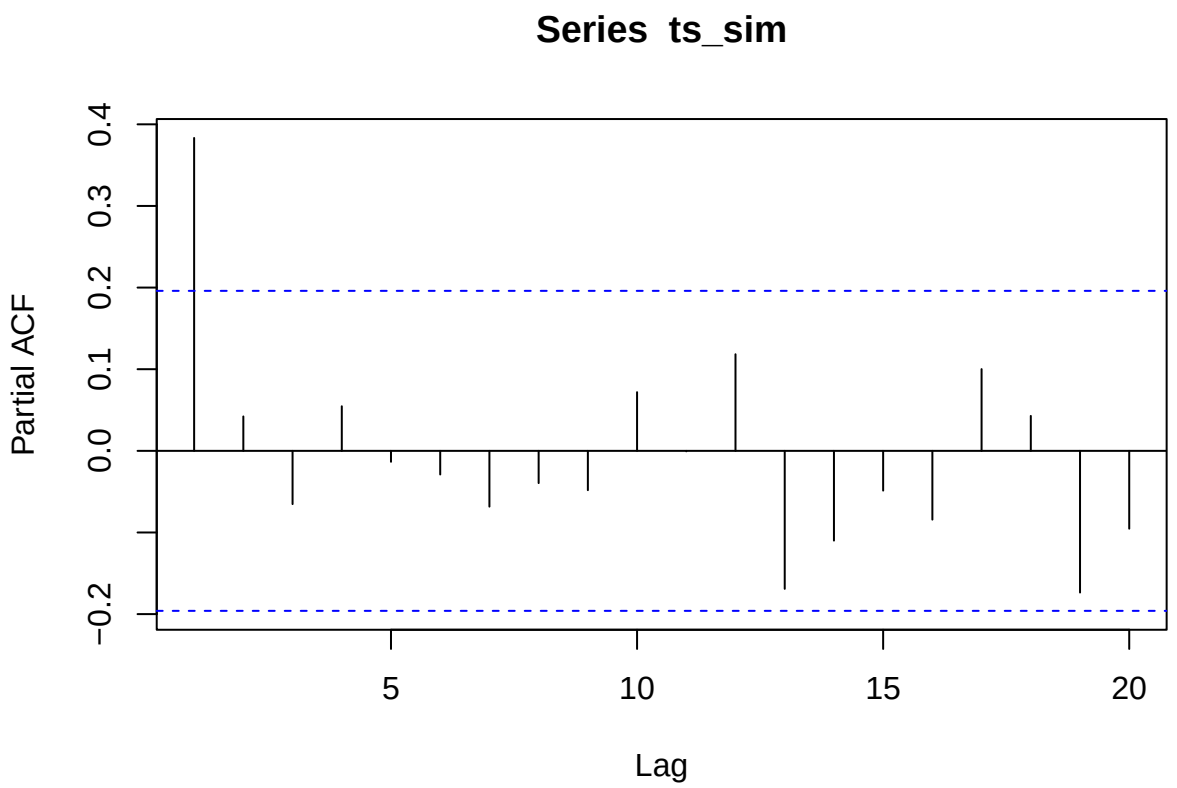
```r
model_params = list(ar = ar_sim)
ts_sim = arima.sim(model = model_params, n = n)
plot(ts_sim, type = "l")
```




```
# Periodicity although there is no periodicity in the model
# The tapering/periodicity indicates AR
# But for p=1 almost looks like MA, but there is a taper
acf(ts_sim)
```



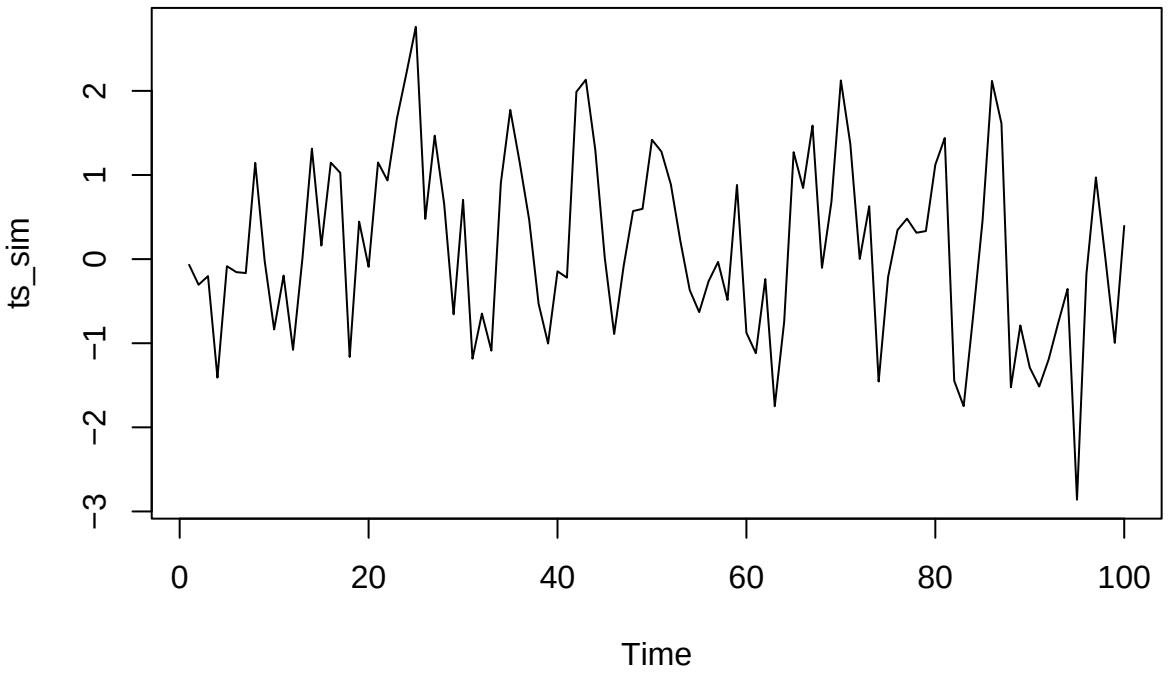
```r
# lag 1 dependence detected, p = 1
pacf(ts_sim)
```



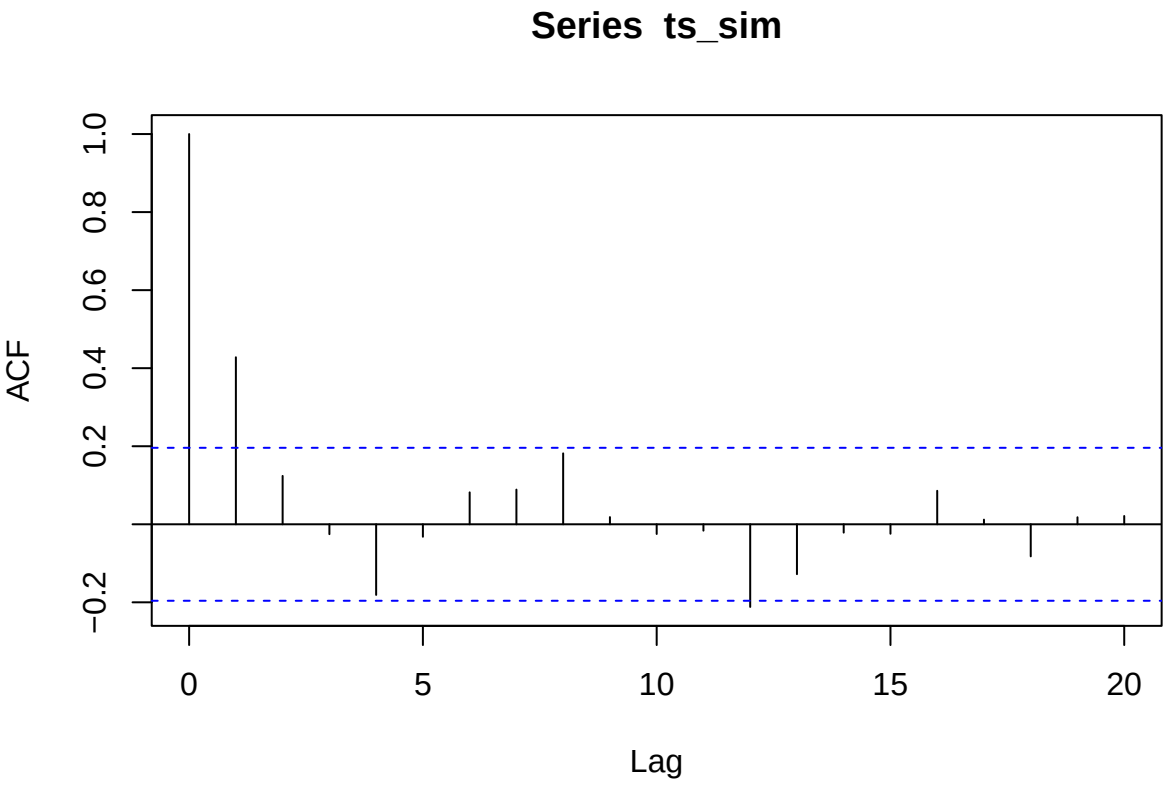
f) MA(2)

```r
# Pure MA
n = 100
ma_sim = c(0.5, 0.1)

(roots = polyroot(c(1,ma_sim)))
## [1] -2.5+1.936492i -2.5-1.936492i
Mod(roots)
## [1] 3.162278 3.162278
```

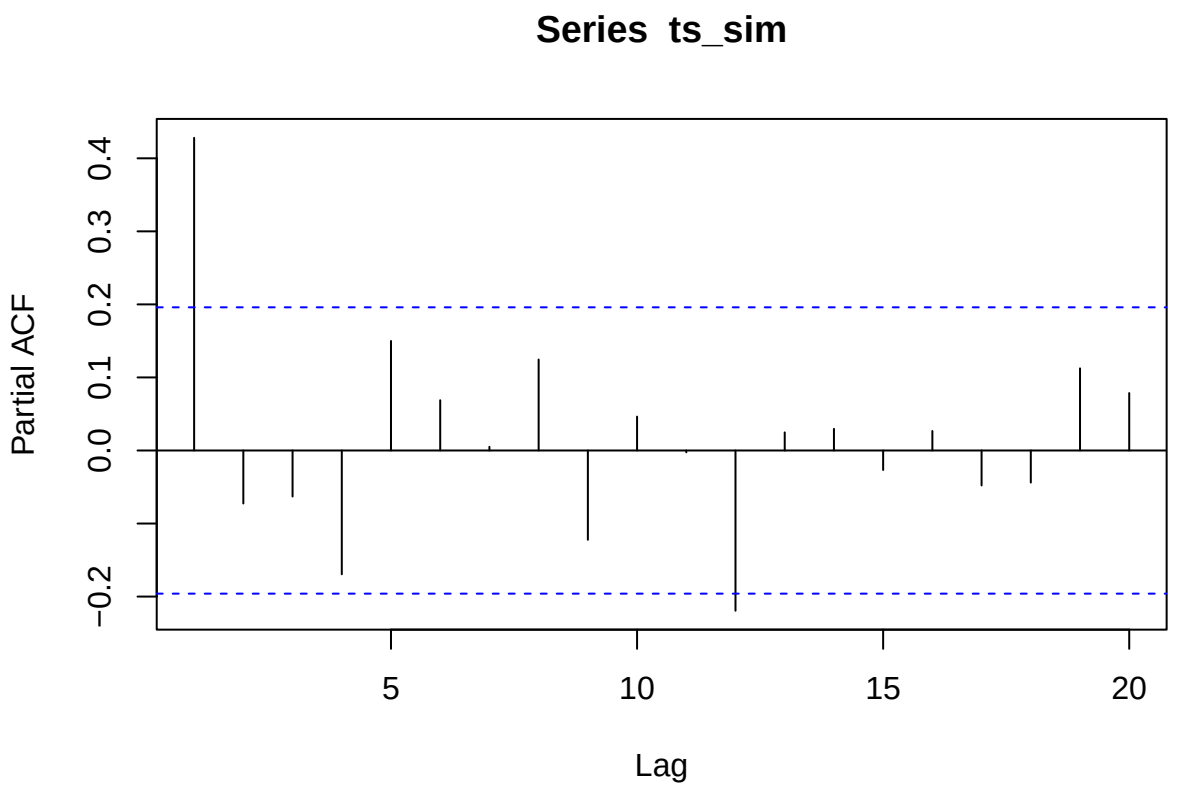
```r
model_params = list(ma = ma_sim)
ts_sim = arima.sim(model = model_params, n = n)
plot(ts_sim, type = "l")
```




```
# 1,2 coefficient detected, q = 2
acf(ts_sim)
```



```
# not really useful for MA
# lag 1,2 dependence detected - AR(2)?
pacf(ts_sim)
```



g) `armia.sim` Sample Code

```
# AR(p) simulation function w/ burn-in period
my_AR_sim <- function(n=189, ar_coeff=c(.5), sigma=1) {
    n.start = n + floor(.2/(1-.2)*n)
    burnin = floor(.20*n.start)
    model_params = list(ar = ar_coeff)
    ts_sim = arima.sim(model = model_params, n = n.start, sd=sigma)
    ts_sim = ts_sim[(burnin+1):n.start]
    return(ts_sim)
}

n=100 # 1000, 10000
sigma <- 1
ar_coeff <- c(.5,.4)
main_exp <- paste("AR sim: ",paste(ar_coeff,collapse=", "), paste(" (n=",n,")",sep=""))

(roots = polyroot(c(1,-1*ar_coeff)) )
```

```
## [1]  1.075184-0i -2.325184+0i
```

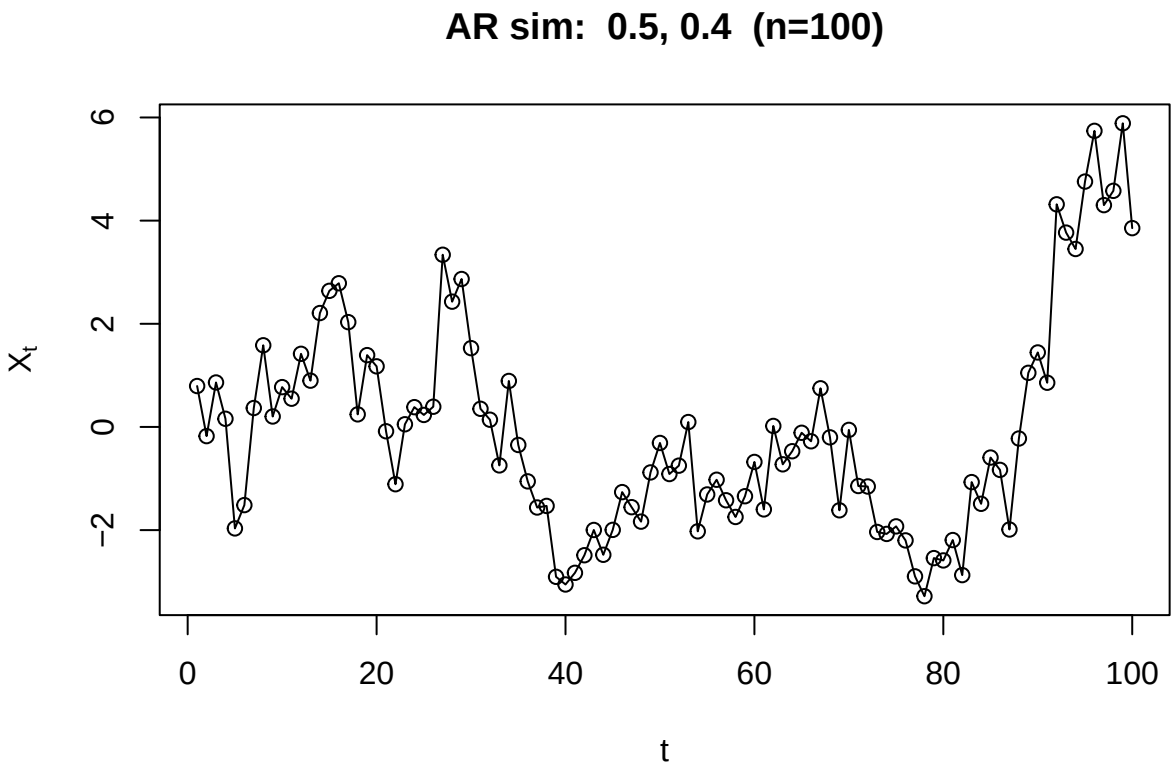
```
Mod(roots)
```

```
## [1] 1.075184 2.325184
```

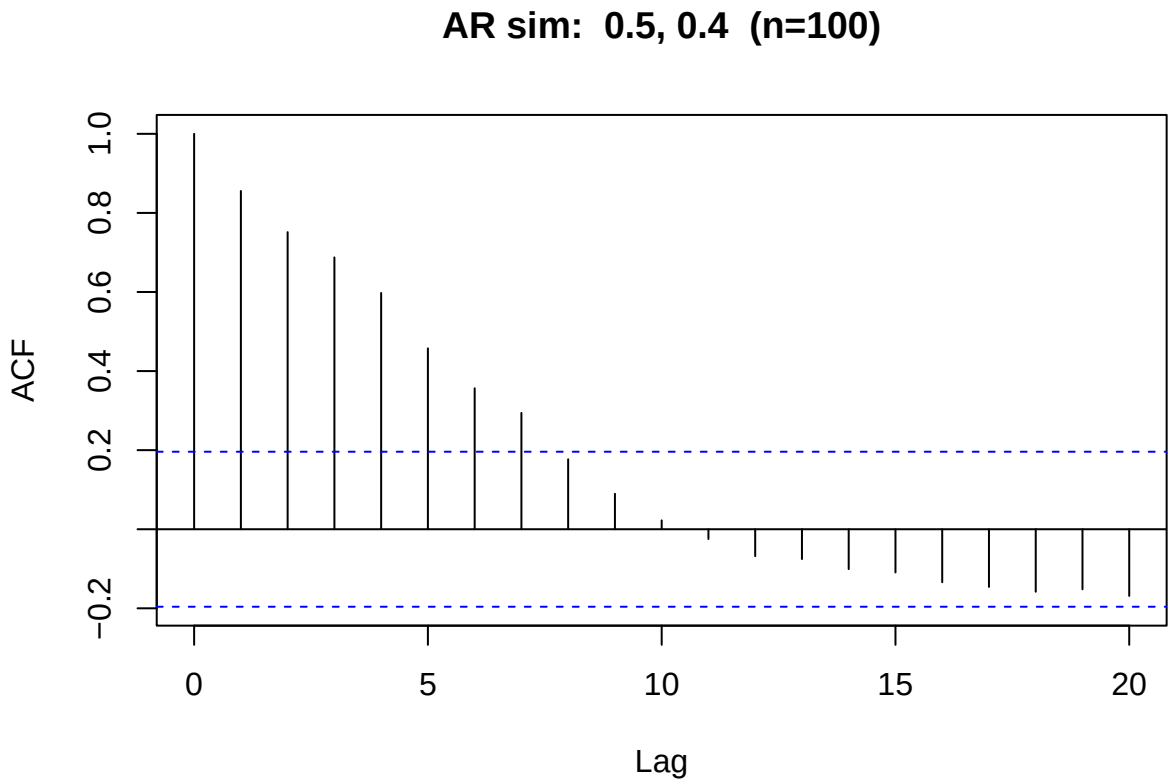
```r
ar_sim = my_AR_sim(n=n, ar_coeff=ar_coeff, sigma=sigma)
length(ar_sim)
```

```
## [1] 100
```

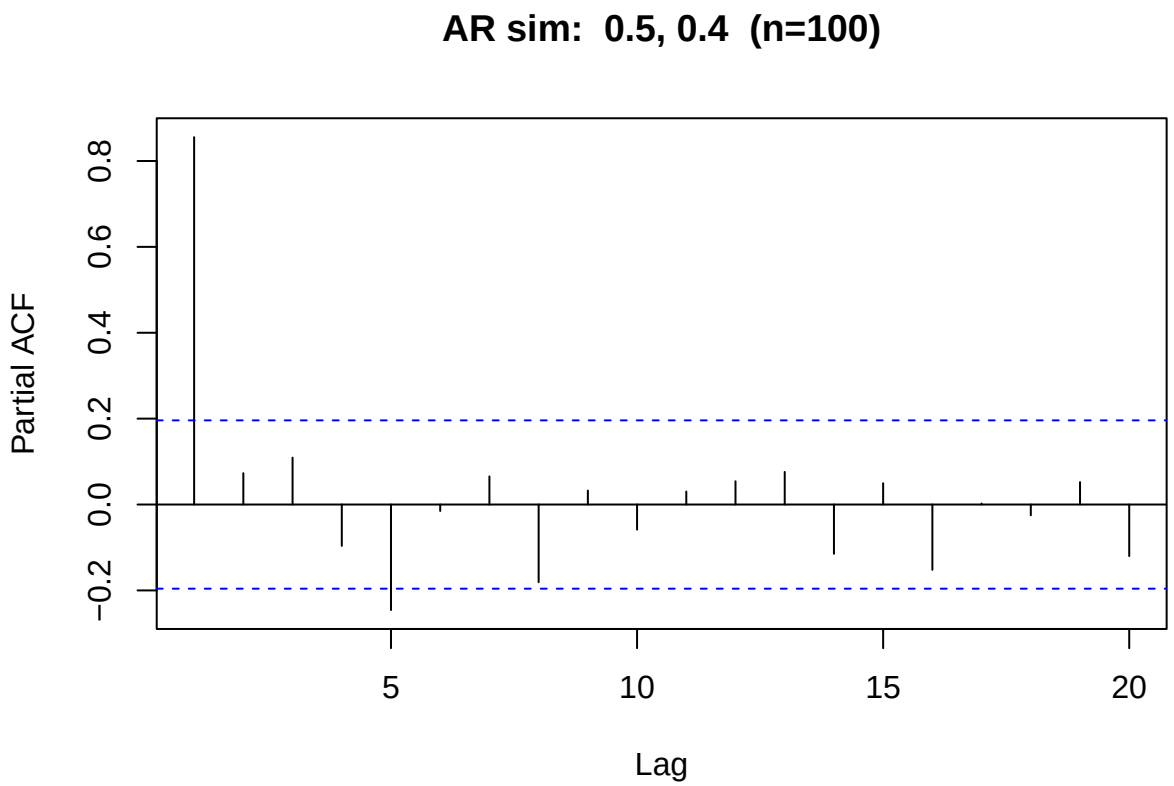
```r
plot(ar_sim,type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
```




```r
acf(ar_sim, main=main_exp)
```



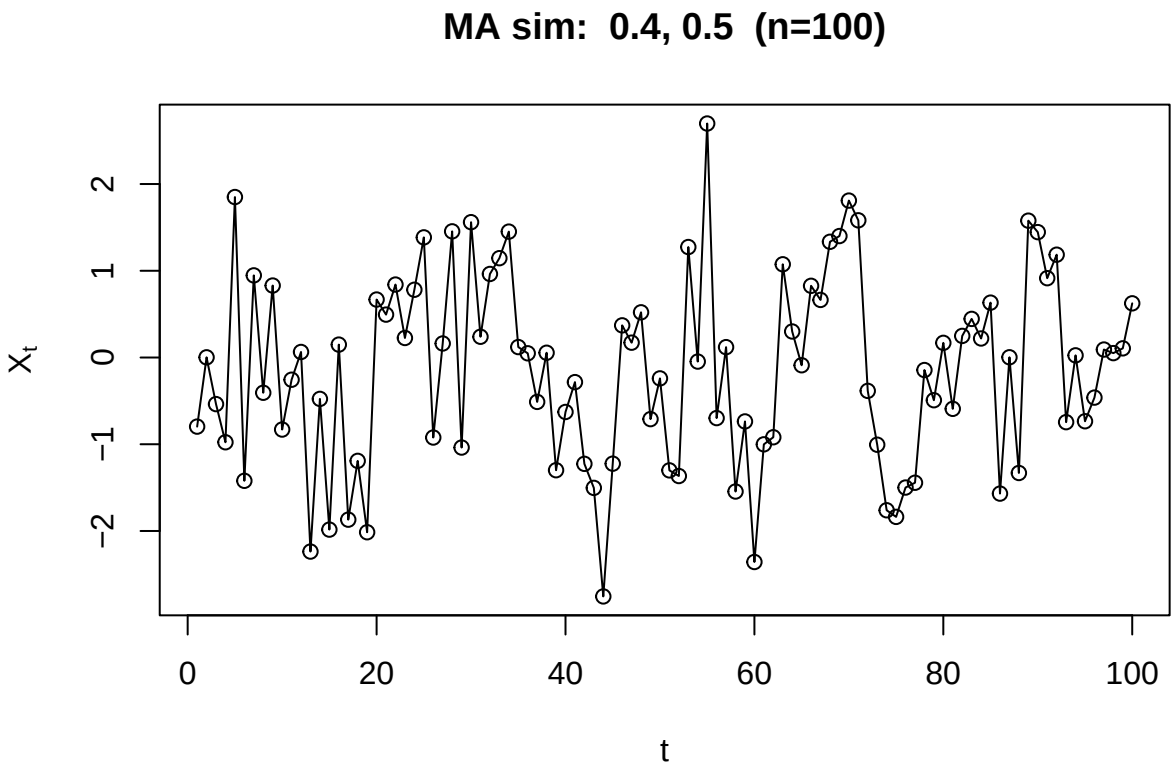

```
pacf(ar_sim, main=main_exp)
```



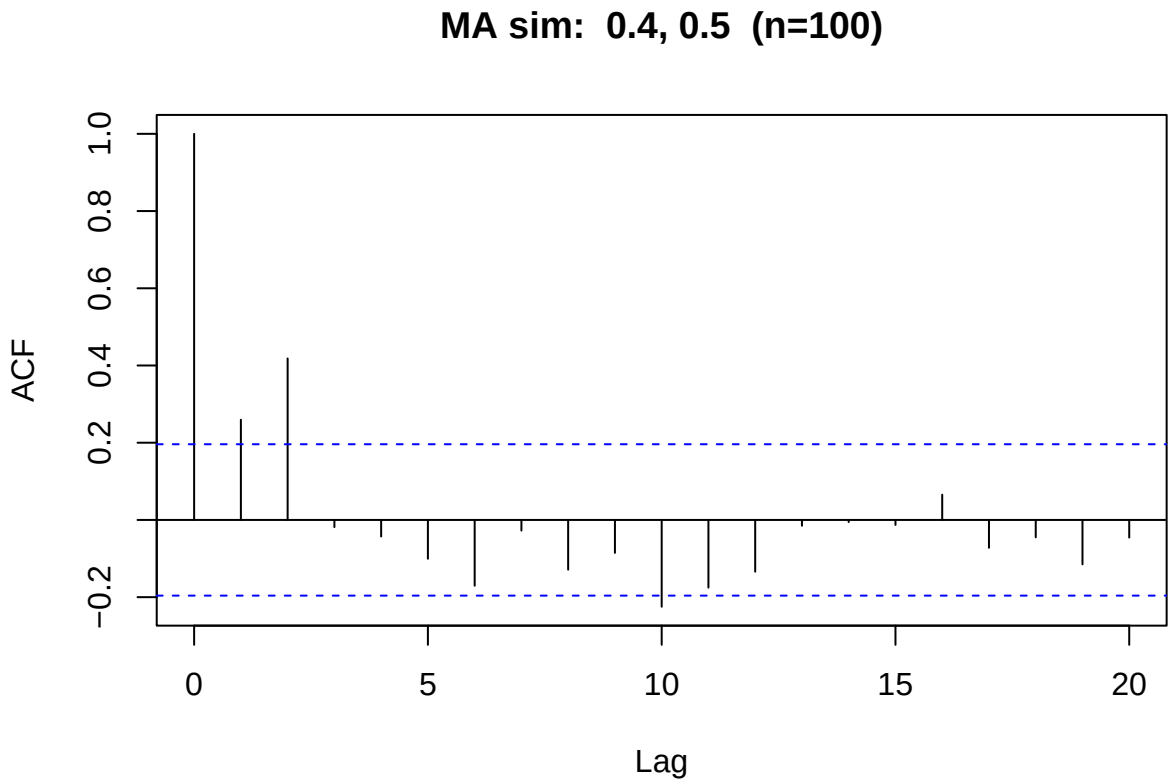
```r
# MA(q) simulation function w/ burn-in period
my_MA_sim = function(n=189, ma_coeff=c(.5), sigma=1) {
  n.start = n + floor(.2/(1-.2) * n)
  burnin = floor(.20 * n.start)
  model_params <- list(ma = ma_coeff)
  ts_sim = arima.sim(model = model_params, n = n.start, sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
  return(ts_sim)
}

n=100 # 10000
ma_coeff=c(.4,.5)
sigma = 1
main_exp = paste("MA sim: ", paste(ma_coeff,collapse=", "), paste(" (n=",n,")", sep=""))

ma_sim = my_MA_sim(n=n, ma_coeff=ma_coeff, sigma=sigma)
length(ma_sim)
```

```
## [1] 100
```

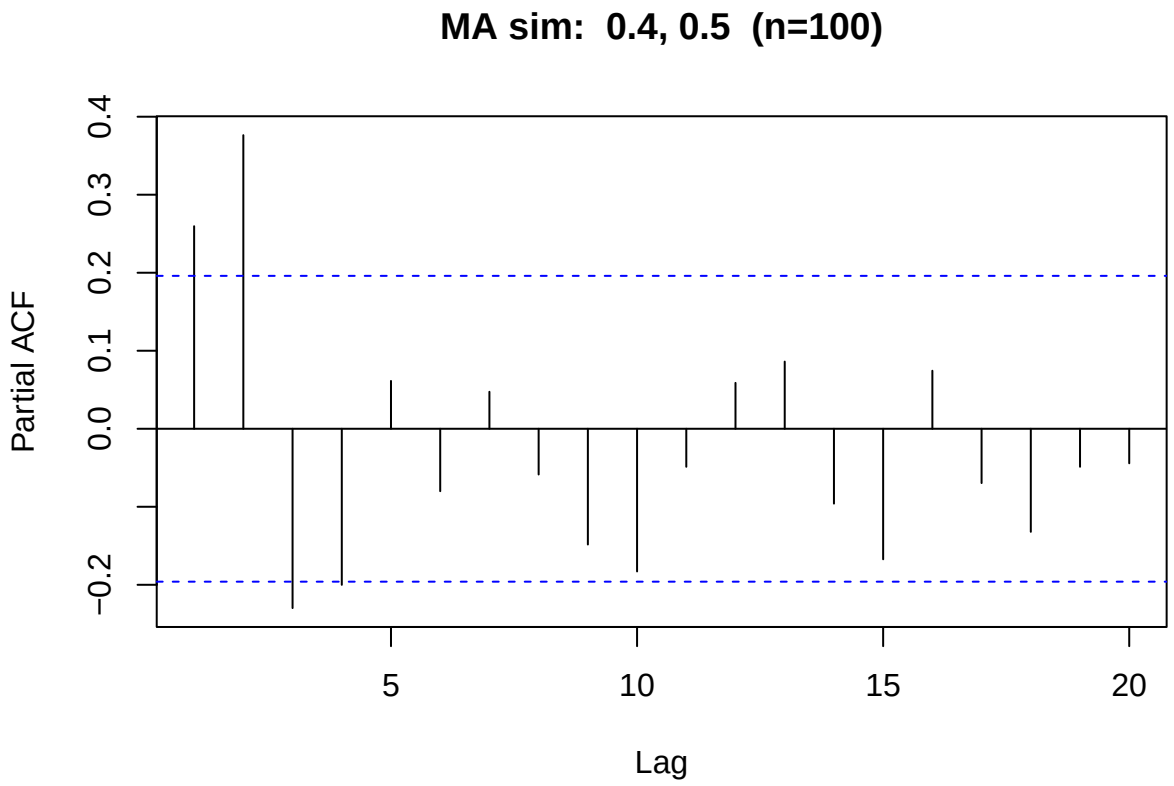
```r
plot(ma_sim,type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
```




```r
acf(ma_sim, main = main_exp)
```




```
pacf(ma_sim, main = main_exp)
```



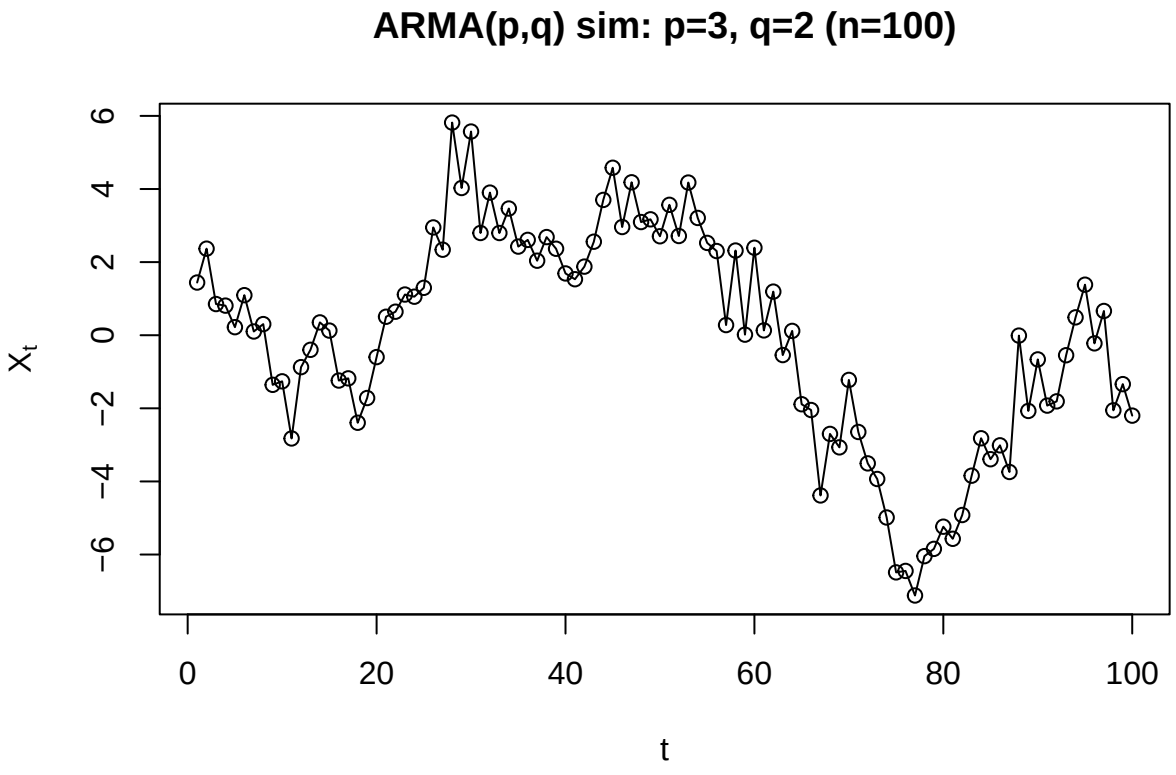
```r
# ARMA(p,q) simulation function w/ burn-in period
my_ARMA_sim = function(n=189, ma_coeff=c(.5), ar_coeff=c(.5), sigma=1) {
  n.start = n + floor(.2/(1-.2) * n)
  burnin = floor(.20*n.start)
  model_params = list(ma = ma_coeff, ar=ar_coeff)
  ts_sim = arima.sim(model = model_params, n = n.start, sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
  return(ts_sim)
}

n=100 # 1000, 10000, 100000
ar_coeff=c(.4,.4,.1)
ma_coeff=c(.4,.5)
sigma = 1
paste("p=",length(ar_coeff),", ","q=",length(ma_coeff),sep="")
```

```
## [1] "p=3, q=2"
```

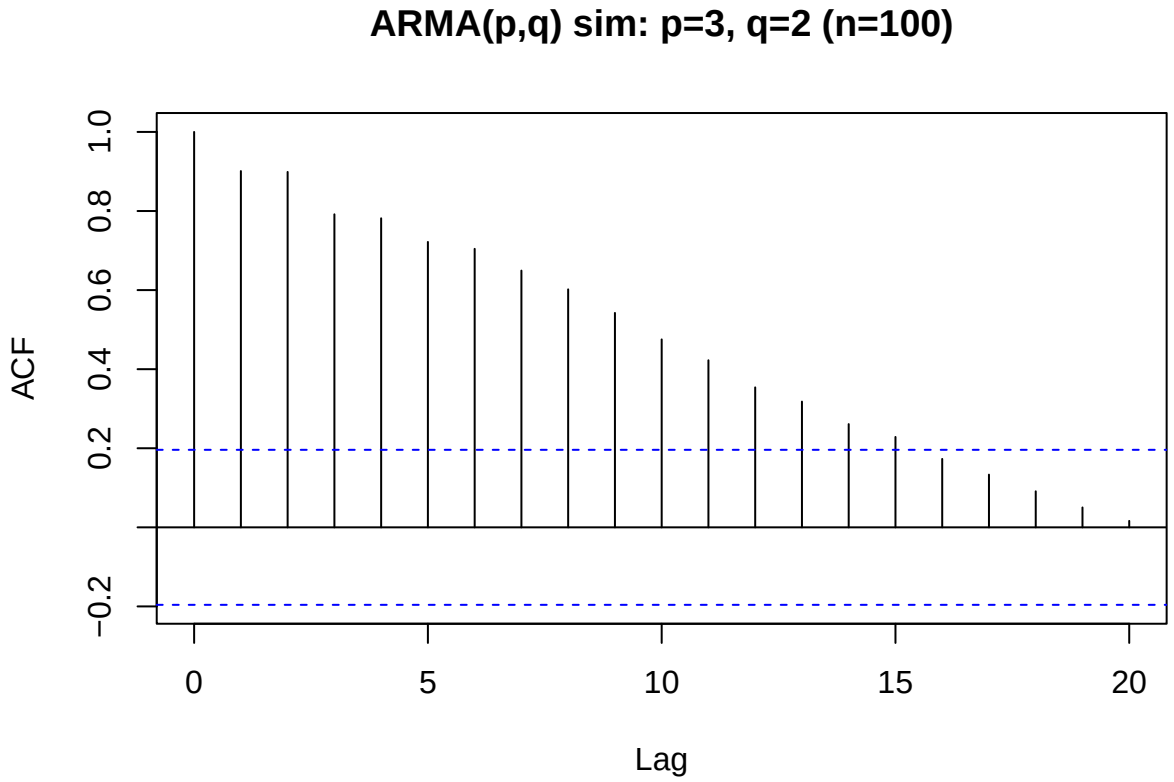
```r
main_exp <- paste("ARMA(p,q) sim:",paste("p=", length(ar_coeff),", ","q=", length(ma_coeff),sep=""), paste("(n=

ARMA_sim = my_ARMA_sim(n=n, ma_coeff=ma_coeff, ar_coeff=ar_coeff, sigma=sigma)
length(ARMA_sim)
```

```
## [1] 100
```

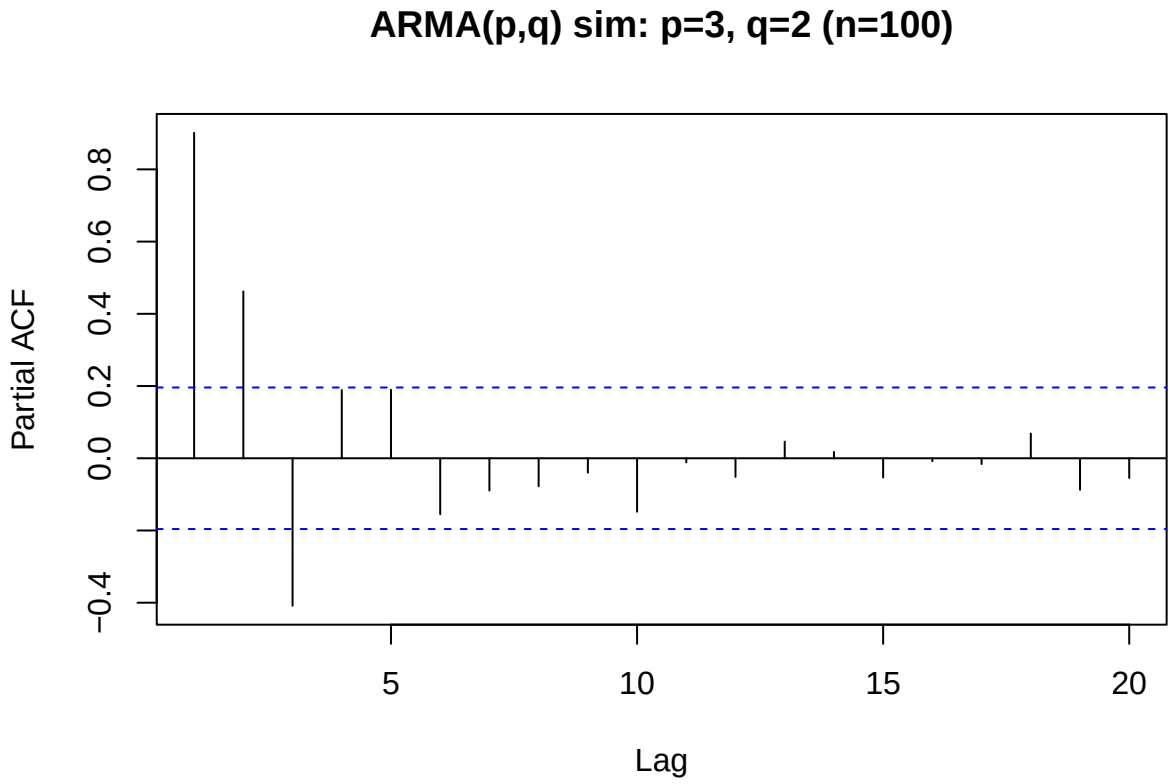
```r
plot(ARMA_sim,type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
```



```
acf(ARMA_sim, main = main_exp)
```




```
pacf(ARMA_sim, main = main_exp)
```



Example: HW 3 #1 - Construct Sample PACF Function

Goal: Explore the details of the sample partial autocorrelation function.

Assume that the observed data `HW3_AR_Data1.csv` and `HW3_AR_Data2.csv` come from pure autoregressive models with respective orders $p_1 > 0$ and $p_2 > 0$.

a) Build you own sample PACF using `R` and test the function on datasets `HW3_AR_Data1.csv` and `HW3_AR_Data2.csv`. Report the sample PACF values for lags $h = 0, 1, \ldots, 20$.
You may extract the lag-correlations $\hat{\rho}(h)$ and/or the lag-covariances $\hat{\gamma}(h)$ directly from the `R` function `acf()`.

The partial autocorrelation function (PACF) of an ARMA(p,q) process is the function $\alpha(\cdot)$ defined by the equations

$$\alpha(0) = 1$$
$$\alpha(h) = \phi_{hh}, \quad h \geq 1$$

Where ϕ_{hh} is the last component of the solution (related to the lag h or $\gamma(h)$, term) of the following equation (analogous to the best linear predictor's formula)

$$\phi_h = \mathbf{\Gamma}_h^{-1}\gamma_h$$

For

$$\mathbf{\Gamma}_h = [\gamma(i-j)]_{i,j=1}^{h}$$
$$\gamma_h = [\gamma(1),\ \gamma(2), \cdots,\ \gamma(h)]^T$$

In other words, we know that the best linear predictor of an AR(p) process relates up to the ϕ_k term for $k < p < n$, and will have 0 for all other terms.

```
# read in data
ARdat1 = read.csv("HW3_AR_Data1.csv")
ARdat2 = read.csv("HW3_AR_Data2.csv")
```

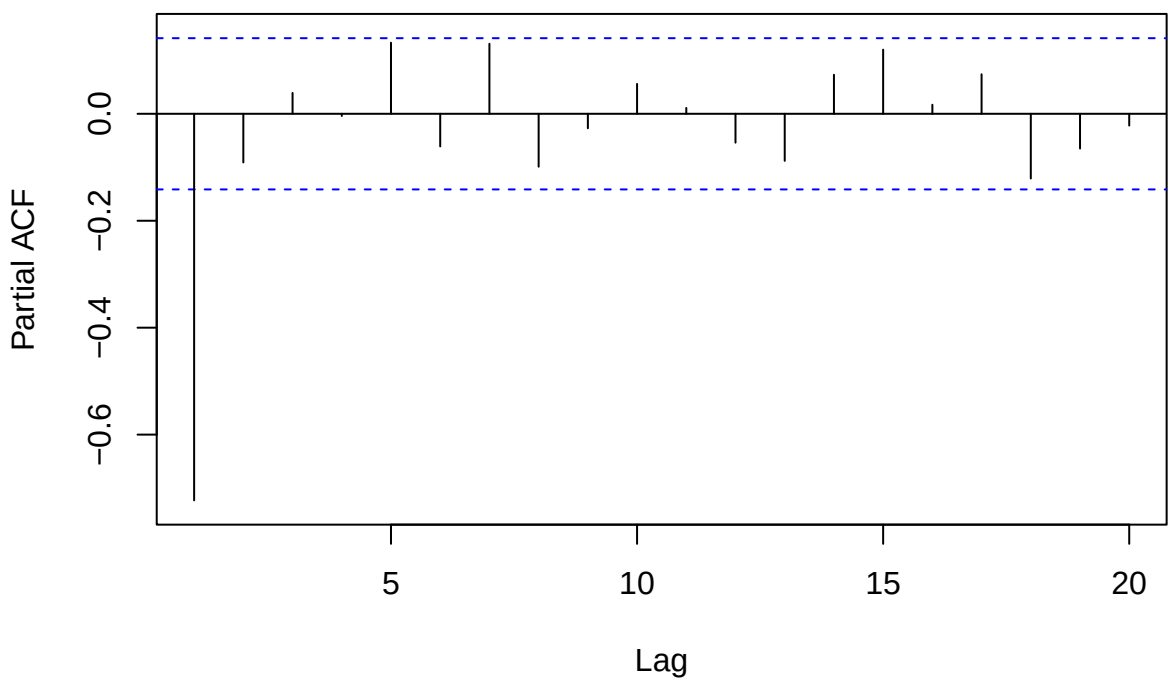
```
# sample PACF function
samplePACF = function(data) {
  vecPACF = rep(NA, 20)

  # extract sample autocorrelations hat{rho}(h) from acf function
  rho = c(acf(data, plot=F)$acf)
  len = length(rho)

  # setup correlation matrix of hat{rho}(h) values
  Corr = matrix(NA, nrow = len, ncol = len)

  # construct correlation matrix of hat{rho}(h) values
  for (i in 1:len) {
    for (j in 1:len) {

        index = abs(i-j)
        Corr[i,j] = c(rho)[index+1]
    }
  }

  for (i in 1:20) {

    # Yule-Walker hat{phi} vector for order i
    # rho vector should start with hat{rho}(1) not hat{rho}(0) (so start with index 2)
    # pacf functions takes ith value of Yule-Walker hat{phi} vector
    vecPhi = solve(Corr[1:i,1:i])%*%rho[2:(i+1)]
    vecPACF[i] = round(vecPhi[i],3)
  }
```

```
  return(vecPACF)
}

# Plot PACF wrapper function using my sample PACF function
plotPACF = function(data, name) {

  # use my sample PACF function to generate sample PACF values
  vecPACF = samplePACF(data)

  # calculate plot limits
  n = length(data)
  vecLimits = c(-2/sqrt(n),2/sqrt(n),vecPACF)
  limUpper = max(vecLimits)
  limLower = min(vecLimits)

  # construct empty plot
  plot(vecPACF,
       xlab = "Lag",
       ylab = "Partial ACF",
       main = paste(name),
       type = "n",
       ylim = c(limLower-0.01,limUpper+0.01))

  # x-axis
  abline(h=0)

  # 95% confidence bands given by \pm 2/sqrt(n) based on (Box, Jenkins, and Reinsel 1970)
  abline(h=2/sqrt(n),
         lty = 2,
         col = "blue")

  abline(h=-2/sqrt(n),
         lty = 2,
         col = "blue")

  # Plot line segments corresponding to sample PACF values
  for (i in 1:20) {

    segments(x0 = i,
             y0 = 0,
             x1 = i,
             y1 = vecPACF[i])
  }
}
```

```
samplePACF(ARdat1[,2])
```

```
##  [1] -0.723 -0.091  0.039 -0.004  0.133 -0.061  0.131 -0.099 -0.027  0.056
## [11]  0.011 -0.054 -0.088  0.073  0.120  0.017  0.074 -0.121 -0.065 -0.022
```

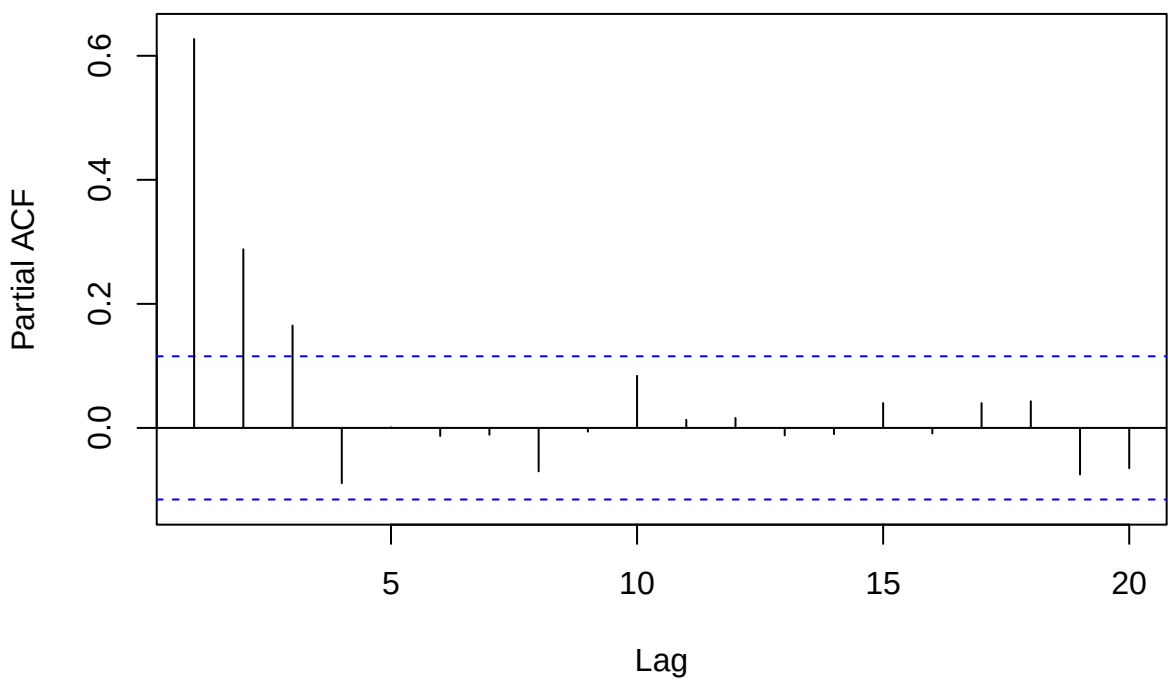
```
plotPACF(ARdat1[,2], "AR Data 1")
```




```
samplePACF(ARdat2[,2])
```

```
##  [1]  0.627  0.288  0.165 -0.089  0.001 -0.013 -0.011 -0.070 -0.006  0.084
## [11]  0.013  0.016 -0.012 -0.010  0.040 -0.009  0.040  0.043 -0.075 -0.065
```

```
plotPACF(ARdat2[,2], "AR Data 2")
```



b) Use the `R` function `pacf()` to compute the sample partial autocorrelations of datasets `HW3_AR_Data1.csv` and `HW3_AR_Data2.csv` and compare the results with part (a).
The output should be the same.

```
pacf(ARdat1[,2], plot = FALSE, lag.max = 20)
```

```
##
## Partial autocorrelations of series 'ARdat1[, 2]', by lag
##
##      1      2      3      4      5      6      7      8      9     10     11
## -0.723 -0.091  0.039 -0.004  0.133 -0.061  0.131 -0.099 -0.027  0.056  0.011
##     12     13     14     15     16     17     18     19     20
## -0.054 -0.088  0.073  0.120  0.017  0.074 -0.121 -0.065 -0.022
```

```
samplePACF(ARdat1[,2])
```

```
##  [1] -0.723 -0.091  0.039 -0.004  0.133 -0.061  0.131 -0.099 -0.027  0.056
## [11]  0.011 -0.054 -0.088  0.073  0.120  0.017  0.074 -0.121 -0.065 -0.022
```

```
pacf(ARdat2[,2], plot = FALSE, lag.max = 20)
```

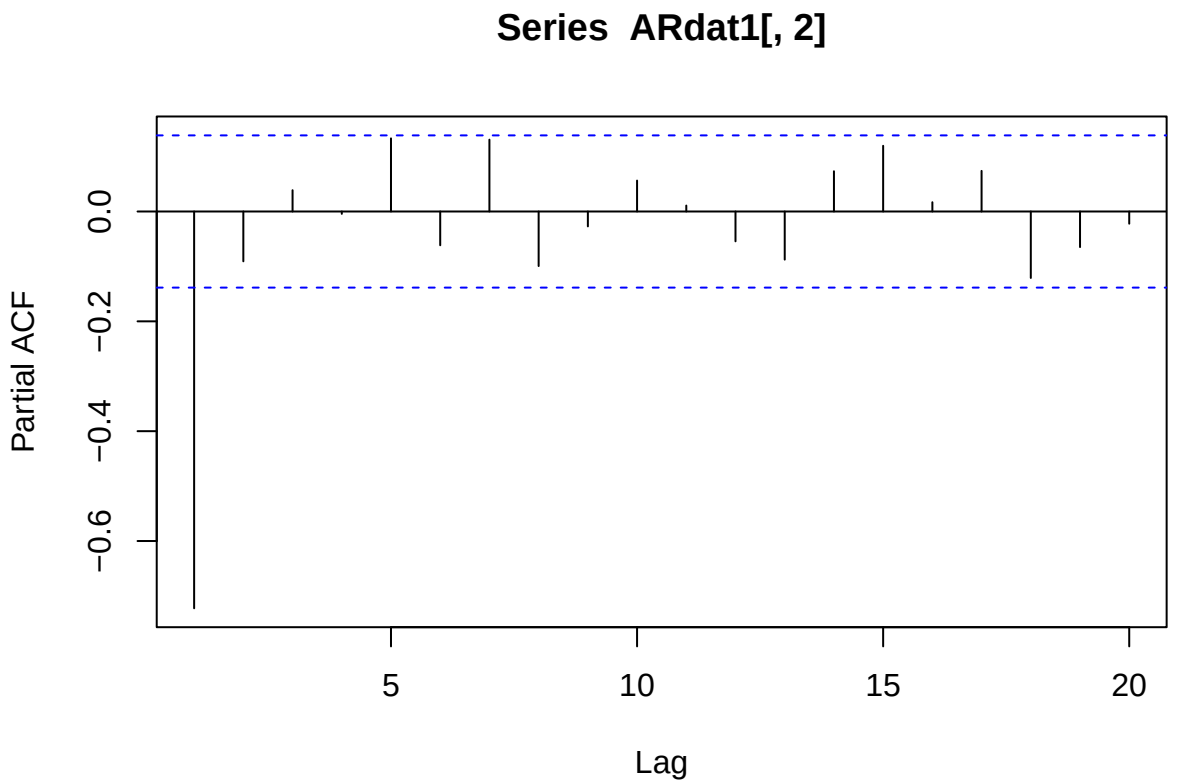
```
##
## Partial autocorrelations of series 'ARdat2[, 2]', by lag
##
##      1      2      3      4      5      6      7      8      9     10     11
##  0.627  0.288  0.165 -0.089  0.001 -0.013 -0.011 -0.070 -0.006  0.084  0.013
##     12     13     14     15     16     17     18     19     20
##  0.016 -0.012 -0.010  0.040 -0.009  0.040  0.043 -0.075 -0.065
```

```
samplePACF(ARdat2[,2])
```

```
##  [1]  0.627  0.288  0.165 -0.089  0.001 -0.013 -0.011 -0.070 -0.006  0.084
## [11]  0.013  0.016 -0.012 -0.010  0.040 -0.009  0.040  0.043 -0.075 -0.065
```

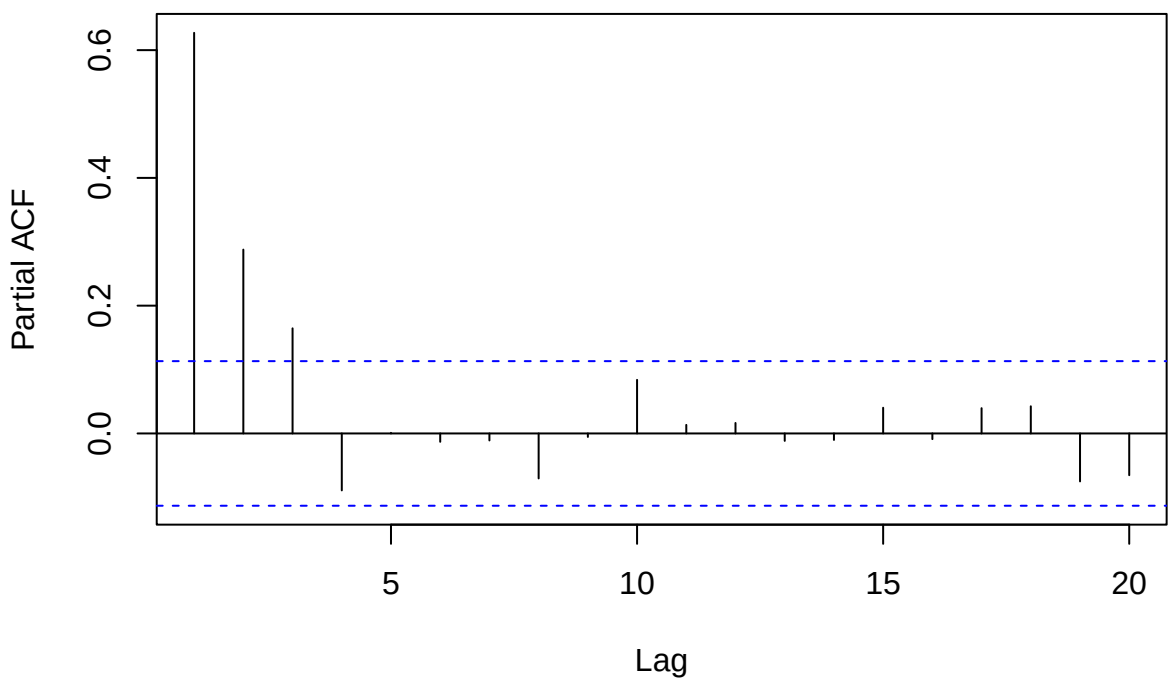
c) Based on the R function `pacf()`, identify the autoregressive orders ($p_1 > 0$ and $p_2 > 0$) for datasets `HW3_AR_Data1.csv` and `HW3_AR_Data2.csv`.
Briefly explain how you chose the AR model orders from the $PACF$ plots.
For this question, you will use the 95% confidence bands provided by the `pacf()` output.

```
pacf(ARdat1[,2], lag.max = 20)
```



The PACF of a pure AR(p) model will exceed the confidence bands for lag h values up to the model order p_1.
For the dataset `HW3_AR_Data1.csv`, the AR order is $p_1 = 1$.

```
pacf(ARdat2[,2], lag.max = 20)
```



The PACF of a pure AR(p) model will exceed the confidence bands for lag h values up to the model order p_2. For the dataset `HW3_AR_Data2.csv`, the AR order is $p_2 = 3$.

MODELING (ESTIMATING PARAMETERS), AND FORECASTING ARMA MODELS

Modeling ARMA(p,q) Processes:
To model an ARMA(p,q) process, we must
1. Determine the model order (via model selection): p, q

2. Estimate model parameters: $(\phi_1, \ldots, \phi_p)$, $(\theta_1, \ldots, \theta_q)$, σ^2

3. Determine goodness of fit

We will first focus on estimation of $\hat{\phi}_p$, $\hat{\theta}_q$, $\hat{\sigma}^2$ when p and q are known.

Yule Walker (MOM) Estimation of a Pure, Causal AR(p) Processes - Yule-Walker Equations, Method of Moments Estimation (MOM):

Yule-Walker (MOM) Equations - Estimating ϕ_p:
Consider the pure, causal AR(p) process.

$$X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = Z_t$$

Recall that MOM estimation requires substituting sample moments for population moments, and solving a system of equations for an expression of the parameters.
To achieve this, we will generate p equations and p unknowns by

1. Multiplying each side of the equation by an X_{t-j} term for $j = 1, \ldots, p$,

$$\begin{cases} X_{t-1}X_t - \phi_1 X_{t-1}X_{t-1} - \cdots - \phi_p X_{t-1}X_{t-p} = X_{t-1}Z_t, & j = 1 \\ X_{t-2}X_t - \phi_1 X_{t-2}X_{t-1} - \cdots - \phi_p X_{t-2}X_{t-p} = X_{t-2}Z_t, & j = 2 \\ \vdots \\ X_{t-p}X_t - \phi_1 X_{t-p}X_{t-1} - \cdots - \phi_p X_{t-p}X_{t-p} = X_{t-p}Z_t, & j = p \end{cases}$$

2. Taking the expected value of both sides of the equations,

$$\begin{cases} E[X_{t-1}X_t] - \phi_1 E[X_{t-1}^2] - \cdots - \phi_p E[X_{t-1}X_{t-p}] = E[X_{t-1}Z_t] = 0, & j = 1 \\ E[X_{t-2}X_t] - \phi_1 E[X_{t-2}X_{t-1}] - \cdots - \phi_p E[X_{t-2}X_{t-p}] = E[X_{t-2}Z_t] = 0, & j = 2 \\ \vdots \\ E[X_{t-p}X_t] - \phi_1 E[X_{t-p}X_{t-1}] - \cdots - \phi_p E[X_{t-p}^2] = E[X_{t-p}Z_t] = 0, & j = p \end{cases}$$

The expectation (covariance) terms can be replaced by $\gamma(h)$ terms for the appropriate lag h.

Note that causal and stationary processes $\{X_t\}$, can be expressed as a linear process of past error terms,

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}$$

Thus, past observations are linear combinations of past error terms, and are therefore uncorrelated with future error terms. Therefore, $E[X_{t-j}Z_t] = 0$.
Thus causality is a key assumption in estimation of model parameters.

Proof:

If X_t is causal and stationary, we know that X_t can be expressed as the linear process,

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}$$

Therefore, X_{t-1} can be expressed as a linear process of past WN terms,

$$X_{t-1} \sum_{j=0}^{\infty} \Psi_j Z_{(t-1)-j}$$

Because WN terms are uncorrelated with past WN terms,

$$E[Z_t X_{t-1}] = E\left[Z_t \sum_{j=0}^{\infty} \Psi_j Z_{(t-1)-j}\right] = \sum_{j=0}^{\infty} \Psi_j E[Z_t Z_{(t-1)-j}] = \sum_{j=0}^{\infty} \Psi_j(0) = 0$$

3. Expressing the expectation terms as autocovariance $\gamma(h)$ terms,

$$\begin{cases} \gamma(1) - \phi_1\gamma(0) - \cdots - \phi_p\gamma(1-p) = 0, & j = 1 \\ \gamma(2) - \phi_1\gamma(1) - \cdots - \phi_p\gamma(2-p) = 0, & j = 2 \\ \vdots \\ \gamma(p) - \phi_1\gamma(p-1) - \cdots - \phi_p\gamma(0) = 0, & j = p \end{cases}$$

$$\begin{cases} \phi_1\gamma(0) + \cdots + \phi_p\gamma(1-p) = \gamma(1), & j = 1 \\ \phi_1\gamma(1) + \cdots + \phi_p\gamma(2-p) = \gamma(2), & j = 2 \\ \vdots \\ \phi_1\gamma(p-1) + \cdots + \phi_p\gamma(0) = \gamma(p), & j = p \end{cases}$$

Therefore,

$$\mathbf{\Gamma}_p\phi_p = \gamma_p$$
$$\phi_p = \mathbf{\Gamma}_p^{-1}\gamma_p$$

i. $\mathbf{\Gamma}_p = [\gamma(i-j)]_{i,j=1}^p$
ii. $\phi_p = [\phi_1, \phi_2, \ldots, \phi_p]^T$
iii. $\gamma_p = [\gamma(1), \gamma(2), \ldots, \gamma(p)]^T$

Yule-Walker (MOM) Equations - Estimating σ^2:
Consider the pure, causal AR(p) process.

$$X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = Z_t$$

We generate an equation for the variance term by

1. Multiplying each side of the equation by X_t,

$$X_t X_t - \phi_1 X_t X_{t-1} - \cdots - \phi_p X_t X_{t-p} = X_t Z_t$$

2. Taking the expected value of both sides of the equations,

$$E[X_t^2] - \phi_1 E[X_t X_{t-1}] - \cdots - \phi_p E[X_t X_{t-p}] = E[X_t Z_t]$$

3. Expressing the expectation terms as autocovariance $\gamma(h)$ terms,

$$\gamma(0) - \phi_1\gamma(1) - \cdots - \phi_p\gamma(p) = \sigma^2$$

Proof:

If X_t is causal and stationary, we know that X_t can be expressed as the linear process

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}$$

Because $\psi_1 = 1$ for the causal AR(p) process,

$$E[Z_t X_t] = E\left[Z_t \sum_{j=0}^{\infty} \Psi_j Z_{t-j}\right] = \sum_{j=0}^{\infty} \Psi_j E[Z_t Z_{t-j}] = \psi_1\sigma^2 = \sigma^2$$

Note that only the $j = 0$ in the sum survives (WN terms are uncorrelated with past noise terms).

Therefore,

$$\sigma^2 = \gamma(0) - \phi_p^T\gamma_p$$

Note that $\gamma(0) = Var(X_t)$.

Yule-Walker Equations:
1. $\mathbf{\Gamma}_p \phi_p = \gamma_p \implies \phi_p = \mathbf{\Gamma}_p^{-1} \gamma_p$
i. $\mathbf{\Gamma}_p = [\gamma(i-j)]_{i,j=1}^p$
ii. $\phi_p = [\phi_1, \phi_2, \ldots, \phi_p]^T$
iii. $\gamma_p = [\gamma(1), \gamma(2), \ldots, \gamma(p)]^T$

2. $\sigma^2 = \gamma(0) - \phi_p^T \gamma_p$
i. $\gamma(0) = Var(X_t)$

Sample Yule-Walker Equations:
To compute the Yule-Walker (MOM) estimator from data, we substitute sample autocovariances into the Yule-Walker equations,
1. $\hat{\phi}_p = \hat{\mathbf{\Gamma}}_p^{-1} \hat{\gamma}_p$
2. $\hat{\sigma}^2 = \hat{\gamma}(0) - \hat{\phi}_p^T \hat{\gamma}_p$
Which yields $\hat{\phi}_p$ and subsequently $\hat{\sigma}^2$.

We can equivalently express the sample Yule-Walker equations in terms of autocorrelations.
1. $\hat{\phi}_p = \hat{\mathbf{R}}_p^{-1} \hat{\rho}_p$
2. $\hat{\sigma}^2 = \hat{\gamma}(0) \left(1 - \hat{\rho}_p^T \hat{\mathbf{R}}_p^{-1} \hat{\rho}_p\right)$
i. $\hat{\mathbf{R}}_p = [R(i-j)]_{i,j=1}^p$
ii. $\hat{\rho}_p = [\hat{\rho}(1), \hat{\rho}(2), \ldots, \hat{\rho}(p)]^T = \frac{1}{\hat{\gamma}(0)} \hat{\gamma}_p$

Proof (2):

$$\hat{\phi}_p = \hat{\mathbf{R}}_p^{-1} \hat{\rho}_p$$

Because the autocorrelation matrix (and its inverse) are symmetric, the transpose operation has no effect,

$$\hat{\sigma}^2 = \hat{\gamma}(0) - \hat{\phi}_p^T \hat{\gamma}_p = \hat{\gamma}(0) - \hat{\rho}_p^T \left(\hat{\mathbf{R}}_p^{-1}\right)^T \hat{\gamma}_p = \hat{\gamma}(0) \left(1 - \hat{\rho}_p^T \hat{\mathbf{R}}_p^{-1} \hat{\rho}_p\right)$$

Yule-Walker Estimator - MOM Estimator, Requires Causality Assumption, Produces Stationary and Causal Solution:
1. The sample Yule-Walker estimator $\hat{\phi}_p$ is a method of moments (MOM) estimator.

   Recall that for IID samples, MOM estimators have convenient asymptotic properties - invariance, consistency, etc.
   Thus, if the noise $Z_t \sim IID$, then the Yule-Walker estimator $\hat{\phi}_p$ has many useful properties.
   Additionally, recall that the CLT for dependent data requires strict stationarity, which is satisfied by IID noise.

2. Causality of $\{X_t\}$ is a critical assumption when formulating the Yule-Walker equations / estimator.

   In particular, causality ensures that the desired quantities to be estimated ϕ_p and σ^2 result from taking expectations of the AR(p) model multiplied by X_{t-j}, $j = 0, \ldots, p$.

3. The Yule-Walker estimator $\hat{\phi}_p$ produces a stationary and causal solution.

Asymptotic (Large Sample) Distribution of Yule-Walker Estimator $\hat{\phi}_p$:
Consider a time series $\{X_t\}$ coming from a causal AR(p) process.

$$X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = Z_t, \quad Z_t \sim IID(0, \sigma^2)$$

Further consider a (non-IID) random sample $\{X_1, X_2, \ldots, X_n\}$ from $\{X_t\}$ - although we have a random sample, the time series has a dependency structure so the sample is not IID.

Then for large n, Γ_p, σ^2 known,

$$\sqrt{n}(\hat{\phi}_p - \phi_p) \overset{d}{\approx} N(\mathbf{0}_p, \sigma^2 \Gamma_p^{-1})$$

The asymptotic (large sample) distribution of the Yule-Walker estimator $\hat{\phi}_p$ is multivariate normal.

Then for large n, Γ_p, σ^2 unknown,

$$\sqrt{n}\left(\hat{\phi}_p - \phi_p\right) \overset{d}{\approx} N\left(\mathbf{0}_p, \hat{\sigma}^2 \hat{\Gamma}_p^{-1}\right)$$

Where

$$\hat{\sigma}^2 = \hat{\gamma}(0) - \hat{\phi}_p^T \hat{\gamma}_p = \hat{\gamma}(0)\left(1 - \hat{\rho}_p^T \hat{\mathbf{R}}_p^{-1} \hat{\rho}_p\right)$$

The asymptotic (large sample) distribution of the Yule-Walker estimator $\hat{\phi}_p$ is multivariate normal.
In the non-asymptotic case, the distribution would be multivariate t.

Equivalently,

$$\left(\hat{\phi}_p - \phi_p\right)^T \hat{\mathbf{\Gamma}}_p \left(\hat{\phi}_p - \phi_p\right) \overset{d}{\approx} \left(\frac{1}{n}\hat{\sigma}^2\right) \chi^2(p)$$

Recall that squaring a standard normal (in this case, a quadratic form) results in a chi-squared distribution with p degrees of freedom.
Note that the $1/n$ term results from the squared $\sqrt{n}$ term from the LHS being brought to the RHS.

Recall that in linear regression,

$$Y_i = \beta_0 + \beta_1 X_{i1} + \cdots + \beta_p X_{ip} + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$$

We can write the matrix form,

$$\mathbf{Y}_p = \mathbf{X}_p \beta_p + \varepsilon_p, \quad \varepsilon_p \sim N(\mathbf{0}_p, \mathbf{\Sigma}_p)$$

Whose solution is given by

$$\hat{\beta}_p = (\mathbf{X}_p^T \mathbf{X}_p)^{-1} \mathbf{X}_p^T \mathbf{Y}_p$$
$$E\left[\hat{\beta}_p\right] = \beta_p$$
$$Var\left(\hat{\beta}\right) = \sigma^2 \left(\mathbf{X}^T \mathbf{X}\right)^{-1}$$

Which resembles the asymptotic solution above.

Asymptotic (Large Sample) Confidence Interval Using the Yule-Walker Estimator $\hat{\phi}_p$:
The asymptotic $100(1-\alpha)\%$ confidence interval for the Yule-Walker estimator ϕ_j

$$\hat{\phi}_{p_j} \pm z_{1-\alpha/2} \frac{1}{\sqrt{n}} \sqrt{\hat{\nu}_j}$$

Where
i. $\hat{\phi}_{p_j}$ is the jth scalar in $\hat{\phi}_p$, and
ii. $\hat{\nu}_j$ is the jth diagonal element of $\hat{\sigma}^2 \hat{\mathbf{\Gamma}}_p^{-1}$

Yule Walker Estimation of a Pure, Causal MA(q) Processes or Causal ARMA(p,q) Processes (q>0) - Limitations of Method of Moments Estimation:
Consider the causal ARMA(p,q) process,

$$X_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p} = Z_t + \theta_1 Z_{t-1} + \cdots + \theta_q Z_{t-q}$$

Multiply each side by X_{t-k}, for $k = 0, 1, \ldots$

Following a similar procedure as for the Yule-Walker equations of the causal AR(p) process, we obtain

$$\gamma(k) - \phi_1 \gamma(k-1) - \cdots \phi_p \gamma(k-p) = \sigma^2 \sum_{j=k}^{q} \theta_j \Psi_{j-k}, \quad 0 \le k \le p+q$$

In particular, because we have a causal model, we can express X_t as a linear combination of Z_t's.
This yields ϕ terms multiplied by the covariance structure on the LHS (as before), and linear process terms on the RHS.

This result allows us to solve for both ϕ and θ terms provided that the Ψ terms are known - we have a recursive formula to determine these values.

Limitations:
However, due to restrictions on the range of $\gamma_X(h)$ and $\rho_X(h)$ values required to obtain real solutions, Yule-Walker (MOM) estimators are a more effective tool for pure AR(p) models than for pure MA(q) or ARMA(p,q) models with $q > 0$.

Maximum Likelihood Estimation Review - SLR:
Maximum likelihood estimation identifies the parameter value θ_{MLE} that produces the highest likelihood of observing the observed sample data $\{x_1, \ldots, x_n\}$.
For $X_1, \ldots, X_n \overset{iid}{\sim} f(X|\theta)$.

$$L(\theta) = L(\theta|\mathbf{x}_n) = \prod_{i=1}^{n} f(x_i|\theta)$$

$$\hat{\theta}_{MLE} = \arg\max_{\theta} L(\theta|x_1, \ldots, x_n)$$

In SLR (assuming X is fixed),

$$Y_i = \beta_0 + \beta_1 x_i + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} N(0, \sigma^2), \quad i = 1, \ldots, n$$

To construct the likelihood function of the parameters $\theta = (\beta_0, \beta_1, \sigma^2)$, we must identify the marginal PDF's of Y_i.

$$Y \overset{ind}{\sim} N(\beta_0 + \beta_1 x_i, \sigma^2)$$
$$E[Y_i] = \beta_0 + \beta_1 x_i$$
$$Var(Y_i) = \sigma^2$$

$$f(Y_i|\beta_0, \beta_1, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(Y_i - \beta_0 - \beta_1 x_i)^2\right)$$

$$L(\beta_0, \beta_1, \sigma^2|\mathbf{Y}_n) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(Y_i - \beta_0 - \beta_1 x_i)^2\right)$$

$$l(\beta_0, \beta_1, \sigma^2|\mathbf{Y}_n) = -\frac{n}{2}\log(2\pi) - \frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(Y_i - \beta_0 - \beta_1 x_i)^2$$

Instead of maximizing the log-likelihood, we can consider minimizing the negative log-likelihood.
A negative log-likelihood resembles a loss (cost) function.

$$Q(\beta_0, \beta_1, \sigma^2|\mathbf{Y}_n) = -\log L(\beta_0, \beta_1, \sigma^2|\mathbf{Y}_n)$$
$$= \frac{n}{2}\log(2\pi) + \frac{n}{2}\log(\sigma^2) + \frac{1}{2\sigma^2}\sum_{i=1}^{n}(Y_i - \beta_0 - \beta_1 x_i)^2$$

Recall that for a Gaussian error structure, maximum likelihood estimation is equivalent to the method least squares, because maximizing the likelihood reduces to minimizing the squared loss in the exponential term.

ML Estimation of a Pure AR(1) Process (Requires Gaussian Error Structure):
Suppose $\{X_0, X_1, \ldots, X_n\}$ are a random sample from an AR(1) process with an IID normal error structure.

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \overset{iid}{\sim} N(0, \sigma^2)$$

Note that we require the assumption of a normal error structure to obtain a closed-form Gaussian PDF.

(Approximate) Conditional (Gaussian) Likelihood of the Pure AR(1) Process:
The simplest method to construct the likelihood is to specify the conditional PDF

$$P(X_1, \ldots, X_n|X_0, \phi, \sigma^2)$$

Where we condition on the value of X_0, which is technically an incorrect method of formulating the likelihood.
Technically, we should account for the auto-regressive structure of X_0 instead of treating it as a fixed value.
Nonetheless, we will proceed with this assumption.

Note that the AR(1) process can be equivalently expressed as

$$Z_t = X_t - \phi X_{t-1}, \quad Z_t \overset{iid}{\sim} N(0, \sigma^2)$$

Using the Gaussian structure, we can express the PDF of the AR(1) process as,

$$f(Z_i|X_0, \phi, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}Z_i^2\right)$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x_i - \phi x_{i-1})^2\right)$$

Note that because they share the same underlying structure through an AR(1) relationship,

$$P(X_1, \ldots, X_n | X_0, \phi, \sigma^2) = P(Z_1, \ldots, Z_n | X_0, \phi, \sigma^2)$$

In other words, the joint density of $\{X_1, \ldots, X_n\}$ conditional on $\{X_0, \theta = (\phi, \sigma^2)\}$ is equivalent to the joint density of $\{Z_1, \ldots, Z_n\}$ conditional on $\{X_0, \theta = (\phi, \sigma^2)\}$.

Note that writing X_0 with the parameters (as above) is technically an abuse of notion - because X_0 is a known quantity and not a parameter to be estimated, we do not express the likelihood in terms of X_0, only the parameters ϕ and σ^2.

Therefore,

$$L(\phi, \sigma^2) = \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \phi x_{i-1})^2\right)$$

Note that this likelihood resembles the likelihood of SLR through the origin (without an intercept) term.

The negative log-likelihood is

$$Q(\phi, \sigma^2) = \frac{n}{2}\log(2\pi) + \frac{n}{2}\log(\sigma^2) + \frac{1}{2\sigma^2}\sum_{i=0}^{n-1}(x_{i+1} - \phi x_i)^2$$

Note that for convenience, we have re-indexed the sum.

$$\theta = \left(\hat{\phi}, \hat{\sigma}^2\right) = \arg\min_{\phi, \sigma^2} Q(\phi, \sigma^2)$$

The maximum likelihood estimators of the AR(1) conditional likelihood are

$$\hat{\phi} = \frac{\sum_{i=0}^{n-1} X_i X_{i+1}}{\sum_{i=0}^{n-1} X_i^2}$$

$$\hat{\sigma}^2 = \frac{1}{n}\sum_{i=0}^{n-1}(X_{i+1} - \hat{\phi} X_i)^2$$

Recall that maximum likelihood estimators are biased, dividing by n, not $n-1$.

The maximum likelihood estimators of the conditional likelihood synthesize with the Yule-Walker estimates for pure AR(p) processes.

However, while the conditional MLE has a closed form solution, it is only an approximation of the exact MLE which requires a numerical approximation and cannot be solved analytically.
Thus the conditional maximum likelihood estimators are useful because they allow us to grasp how the estimators behave.

Exact (Gaussian) Likelihood of a Pure AR(1) Process:
Consider observing only 4 cases: $\{X_0, X_1, X_2, X_3\}$
By the chain rule of conditional probability,

$$\begin{aligned} P(X_0, X_1, X_2, X_3) &= P(X_3 | X_2, X_1, X_0) P(X_2, X_1, X_0) \\ &= P(X_3 | X_2, X_1, X_0) P(X_2 | X_1, X_0) P(X_1, X_0) \\ &= P(X_3 | X_2, X_1, X_0) P(X_2 | X_1, X_0) P(X_1 | X_0) P(X_0) \end{aligned}$$

To construct a likelihood, consider an AR(1) process with $Z_t \overset{iid}{\sim} N(0, \sigma^2)$,

$$E[X_3 | X_2, X_1, X_0] = E[\phi X_2 + Z_3 | X_2, X_1, X_0] \overset{filtration}{=} \phi X_2 + E[Z_3 | X_2, X_1, X_0] = \phi X_2$$

Error terms are independently distributed, and X_i is a sum of noise terms, which implies $E[Z_3 | X_2, X_1, X_0] = E[Z_3] = 0$,

$$Var(X_3 | X_2, X_1, X_0) = Var(\phi X_2 + Z_3 | X_2, X_1, X_0) = Var(\phi X_2 | X_2, X_1, X_0) + Var(Z_3 | X_2, X_1, X_0) == Var(Z_3 | X_2, X_1, X_0) = \sigma^2$$

Given X_2, ϕX_2 is a constant, and $Var(\phi X_2) = 0$.
Therefore the marginal (conditional) density of $X_3|X_2, X_1, X_0$ is given by

$$f(X_3|X_2, X_1, X_0) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(X_3 - \phi X_2)^2\right)$$

This represents only $P(X_3|X_2, X_1, X_0)$ in the chain rule above.

In general, for any $X_t|X_{t-1}, \ldots, X_0$,

$$E[X_t|X_{t-1}, \ldots, X_0] = \phi X_{t-1}$$
$$Var(X_t|X_{t-1}, \ldots, X_0) = \sigma^2$$
$$\therefore X_t|X_{t-1}, \ldots, X_0 \sim N(\phi X_{t-1}, \sigma^2)$$

This allows us to construct all marginal (conditional) densities except for $P(X_0)$.

Because X_0 comes from an AR(1) process, we know that

$$E[X_0] = 0$$
$$Var(X_0) = \gamma_{X_0}(0) = \frac{\sigma^2}{1-\phi^2}$$
$$\therefore f(x_0) = \frac{1}{\sqrt{2\pi\frac{\sigma^2}{(1-\phi^2)}}} \exp\left(-\frac{1}{2\left(\frac{\sigma^2}{(1-\phi^2)}\right)} X_0^2\right)$$

Thus,

$$L(\phi, \sigma^2|x_0, \ldots, x_n) = P(X_0, X_1, \ldots, X_n|\phi, \sigma^2)$$
$$= f(X_0|\phi, \sigma^2) \prod_{t=1}^{n} f(X_t|x_{t-1}, \ldots, x_0; \phi, \sigma^2) = \frac{1}{\sqrt{2\pi\left(\frac{\sigma^2}{(1-\phi^2)}\right)}} \exp\left(-\frac{(1-\phi^2)}{2\sigma^2} x_0^2\right) \cdot \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_i - \phi x_{i-1})^2\right)$$

Note that the exact MLE does not have a closed form solution when minimizing the negative log-likelihood.
The negative log-likelihood can be minimized in R using `nlm()` (a non-linear minimizer).

Asymptotics of the (Approximate) Conditional (Gaussian) MLE's of the Pure AR(1) Process:
Recall that the (approximate) conditional maximum likelihood estimators are useful because they synthesize with Yule-Walker estimator results, and they allow us to grasp how the estimators behave.
Recall that

$$\hat{\phi}_{MLE} = \hat{\phi}_{YW,MOM} = \frac{\sum_{i=0}^{n-1} X_i X_{i+1}}{\sum_{i=0}^{n-1} X_i^2}$$

Recall that in inference, we specify a null value for the parameter and assess how the asymptotics behave (e.g. behavior of the test statistic).
Assume X_t is a random walk, i.e. that $H_0: \ \phi = 1$ is true. Under the null hypothesis,

$$X_t = X_{t-1} + Z_t$$
$$\implies X_i X_{i+1} - X_i^2 = X_i(X_{i+1} - X_i) = X_i Z_{i+1}$$

Then if we consider formulating a rudimentary test statistic for testing against the random walk $\phi = 1$ (under the null), we obtain

$$\hat{\phi} - 1 = \frac{\sum_{i=0}^{n-1} X_i X_{i+1} - \sum_{i=0}^{n-1} X_i^2}{\sum_{i=0}^{n-1} X_i^2} = \frac{\sum_{i=0}^{n-1} X_i Z_{i+1}}{\sum_{i=0}^{n-1} X_i^2}$$

Note that this expression resembles a Riemann sum - an integral divided by an integral.

Skipping many details, we arrive at the following results.
Assuming $H_0 : \phi = 1$ is true,

$$\frac{1}{n}\sum_{i=0}^{n-1} X_i Z_{i+1} \xrightarrow{d} \frac{1}{2}\sigma^2\left([W(1)]^2 - 1\right)$$

$$\frac{1}{n^2}\sum_{i=0}^{n-1} X_i^2 \xrightarrow{d} \sigma^2 \int_0^1 [W(r)]^2 \, dr$$

Where $W(t)$ is standard Brownian motion.

Then under the null,

$$n(\hat{\phi} - 1) = \frac{\frac{1}{n}\sum_{i=0}^{n-1} X_i Z_{i+1}}{\frac{1}{n^2}\sum_{i=0}^{n-1} X_i^2} \xrightarrow{d} \frac{\frac{1}{2}\sigma^2((W(1))^2 - 1)}{\sigma^2 \int_0^1 [W(r)]^2 \, dr} = \frac{((W(1))^2 - 1)}{2\int_0^1 [W(r)]^2 \, dr}$$

We have a pivotal distribution.

Thus, we can test the hypothesis that series is a random walk, given access to this distribution.
However, this process consists of squared Wiener processes (Brownian motion) squared, and must be simulated using Monte Carlo techniques.

ML Estimation of a Pure MA(1) Process (Requires Gaussian Error Structure):
Suppose $\{X_0, X_1, \ldots, X_n\}$ are a random sample from an MA(1) process with an IID normal error structure.

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \overset{iid}{\sim} N(0, \sigma^2)$$

Note that we require the assumption of a normal error structure to obtain a closed-form Gaussian PDF.

(Approximate) Conditional (Gaussian) Likelihood of the Pure MA(1) Process:
The simplest method to construct the likelihood is to specify the conditional PDF

$$P(X_1, \ldots, X_n | Z_0, \theta, \sigma^2)$$

Where we condition on the value of $Z_0 = 0$, which is technically an incorrect method of formulating the likelihood.
Technically, we should account for the normal structure of Z_0 instead of treating it as a fixed value.
Nonetheless, we will proceed with this assumption.

Given $Z_0 = 0$, the sequence $\{Z_1, Z_2, \ldots, Z_n\}$ can be calculated explicitly from the sequence $\{X_1, X_2, \ldots, X_n\}$ recursively (making progress to the base case $Z_0 = 0$) using the following system,

$$Z_t = X_t - \theta Z_{t-1}$$
$$Z_0 = 0$$

For example,

$$Z_2 = X_2 - \theta Z_1 = X_2 - \theta(X_1 - \theta Z_0) = X_2 - \theta X_1$$

Using this recursion,

$$X_1 = Z_1 + \theta Z_0 \implies Z_1 = X_1 - \theta X_0 = X_1$$
$$f_{X_2|X_1,Z_0}(X_2, |X_1, Z_0; \theta, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(X_2 - \theta Z_1)^2\right) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(X_2 - \theta x_1)^2\right)$$

Thus, a component of the likelihood function has been expressed in terms of observable quantities.

Note that the MA(1) process can be equivalently expressed as

$$Z_t = X_t - \theta Z_{t-1}, \quad Z_t \overset{iid}{\sim} N(0, \sigma^2)$$

Using the Gaussian structure, we can express the PDF of the MA(1) process as,

$$f(Z_i | Z_0, \phi, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2} Z_i^2\right)$$

$$= \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{1}{2\sigma^2}(x_i - \theta z_{i-1})^2\right)$$

Note that because they share the same underlying structure through an MA(1) relationship,

$$P(X_1, \ldots, X_n | Z_0, \phi, \sigma^2) = P(Z_1, \ldots, Z_n | Z_0, \theta, \sigma^2)$$

In other words, the joint density of $\{X_1, \ldots, X_n\}$ conditional on $\{Z_0, \theta = (\theta, \sigma^2)\}$ is equivalent to the joint density of $\{Z_1, \ldots, Z_n\}$ conditional on $\{Z_0, \theta = (\theta, \sigma^2)\}$.

The conditional likelihood for the pure MA(1) process is therefore given by

$$f_{\mathbf{X}_n|Z_0=0}(\mathbf{X}_n | z_0 = 0; \theta, \sigma^2) = f_{X_1|z_0=0}(X_1 | z_0 = 0) \prod_{t=2}^{n} f_{X_t|\mathbf{x}_{t-1}, z_0=0}(X_t | \mathbf{x}_{t-1}, z_0 = 0; \theta, \sigma^2)$$

Note that we have separated the term associated with the base case $Z_0 = 0$ because it lacks a conditioning set in $\mathbf{x}$.

We seek to maximize $l(\theta, \sigma^2)$ with respect to (θ, σ^2).

Note that instead of the argument above, we may construct a filtration object by conditioning on information observed up to a particular time point.

$$E[X_t | Z_{t-1}] = E[Z_t + \theta Z_{t-1} | Z_{t-1}] = E[Z_t | Z_{t-1}] + \theta E[Z_{t-1} | Z_{t-1}] = \theta Z_{t-1}$$

Error terms are independently distributed, which implies $E[Z_t | Z_{t-1}] = E[Z_t] = 0$.

$$Var(X_t | Z_{t-1}) = Var(Z_t + \theta Z_{t-1} | Z_{t-1}) = Var(Z_t | Z_{t-1}) + Var(\theta Z_{t-1} | Z_{t-1}) = \sigma^2$$

Because Z_{t-1} is in the conditioning set, it is a fixed constant, which implies $Var(\theta Z_{t-1} | Z_{t-1}) = 0$.
Error terms are independently distributed, which implies $Var(Z_t | Z_{t-1}) = Var(Z_t) = \sigma^2$.

Exact Likelihood of an ARMA(p,q) Process (Requires Gaussian Error Structure) Using the Multivariate Normal Distribution:
Recall that the scalar form of multiple linear regression is given by

$$Y_i = \beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip} + \varepsilon_i, \quad \varepsilon_i \overset{iid}{\sim} N(0, \sigma^2)$$
$$\therefore Y_i \sim N(\beta_0 + \beta_1 x_{i1} + \cdots + \beta_p x_{ip}, \sigma^2)$$

The matrix form of multiple linear regression is given by

$$\mathbf{Y}_n = \mathbf{X}_n \beta_n + \varepsilon_n, \quad \varepsilon_n \sim N(\mathbf{0}_n, \sigma^2 \mathbf{I}_n)$$
$$\therefore \mathbf{Y}_n \sim N(\mathbf{X}_n \beta_n, \sigma^2 \mathbf{I}_n)$$

Note that to construct a normal distribution, we must specify the parameters $\mu_n, \mathbf{\Sigma}_{n \times n}$.

In time series, we can use the mean function and ACVF to construct the mean vector μ_n and covariance matrix $\mathbf{\Sigma}_{n \times n} \equiv \mathbf{\Gamma}_{n \times n}$, and assume that the time series is multivariate normal given a normal error structure.

Consider the random vector $\mathbf{X}_n$.

$$\mathbf{X}_n = \begin{bmatrix} X_1 & \cdots & X_n \end{bmatrix}^T$$

The random vector $\mathbf{X}_n$ has the multivariate normal distribution if it has probability density given by

$$f(\mathbf{X}_n | \mu_n, \mathbf{\Sigma}_{n \times n}) = (2\pi)^{-n/2} \det\left(\mathbf{\Sigma}_{n \times n}^{-1}\right)^{1/2} \exp\left(-\frac{1}{2}(\mathbf{X}_n - \mu_n)^T \mathbf{\Sigma}_{n \times n}^{-1} (\mathbf{X}_n - \mu_n)\right)$$

With support

$$R_{\mathbf{X}_n} = -\infty < X_1, X_2, \ldots, X_n < \infty$$

Note that

$$\mu_n = E[\mathbf{X}_n] = \begin{bmatrix} \mu_1 & \cdots & \mu_n \end{bmatrix}^T$$
$$\mathbf{\Sigma}_{n \times n} = E[(\mathbf{X}_n - \mu_n)(\mathbf{X}_n - \mu_n)^T]$$

We assume that the time series $\{X_1, \ldots, X_n\}$ with normal error structure comes from a multivariate normal distribution.
1. Compute μ_n, which is often $\mathbf{0}_n$.
2. Compute $\mathbf{\Sigma}_{n\times n} \equiv \mathbf{\Gamma}_{n\times n}$ based on the ACVF $\gamma_X(h)$.
3. Use $f(X_1, \ldots, X_n|\mu_n, \mathbf{\Sigma}_{n\times n})$ as the likelihood function (when interpreted as a function of the parameters).

Estimating Parameters - Discussion:
We have many tools for estimating the AR(p) ϕ coefficients.
i. Yule-Walker (MOM)
ii. MLE
iii. Multivariate Normal Likelihood
iv. BLP

Recall that the best linear predictor (BLP) is given by

$$\hat{X}_{n+h} = P_n(X_{n+h}|1, X_n, X_{n-1} \ldots, X_1) = P_n X_{n+h} = a_0 + a_1 X_n + \cdots + a_n X_1$$
$$\mathbf{a}_n = \mathbf{\Gamma}_n^{-1}\gamma_n(h)$$

In particular, the BLP of an AR(p) process is given by

$$P(X_{n+1}|X_n, \ldots, X_1) = \phi_1 X_n + \phi_2 X_{n-1} + \cdots + \phi_p X_{n-(p-1)}$$

Note that $\phi_j = 0$ for $j > p$.

Thus, the BLP should also yield the $\hat{\phi}$'s if we replace $\mathbf{\Gamma}_n$ by $\hat{\mathbf{\Gamma}}_n$ and $\gamma_n(h)$ by $\hat{\gamma}_n$.

$$\hat{\phi}_n = \hat{\mathbf{\Gamma}}_n^{-1}\hat{\gamma}_n(h)$$

Note that these are the Yule-Walker equations - the BLP, Yule-Walker, and PACF equations are closely related.

In addition we can use various algorithms including
i. Durbin-Levinson Algorithm
ii. Burg's Algorithm
iii. Hannan-Rissanen Algorithm (Least Squares)

There are fewer methods for estimating MA(q) or ARMA(p,q) θ coefficients because the Z_t noise terms are unobservable.
i. MLE
ii. Multivariate Normal Likelihood
iii. Innovations Algorithm

Estimating MA(q) Parameters Using the Innovations Algorithm:
Recall that finding the best linear predictor requires the inversion of an $n \times n$ matrix when using data from $\{X_1, \ldots, X_n\}$. The innovations algorithm is a recursive algorithm used to simplify the maxtrix inversion and estimate the prediction operator of MA(q) and ARMA(p,q) models.

However, the innovations algorithm also allows us to
1. Directly express the maximum likelihood PDF, and
2. Estimate MA(q) θ parameters directly
Without the use of a likelihood expression or a Gaussian error structure.

Additionally, the innovations algorithm does not require stationarity of a time series, and is thus applicable to all time series with finite second moments regardless of stationarity.
Thus the innovations algorithm can be used for stock data, which is likely not stationary.

Innovations Representation:
We will use the following notation.
i. $E[X_i X_j] = \kappa(i, j)$

Note that this is an expression of the true covariance function, which can be used for non-stationary time series (unlike $\gamma(h)$).
In particular, note that $\kappa(i, i) = \gamma(i)$ for a stationary process.

Thus the innovations algorithm does not require stationarity of time series to find the BLP, and is thus applicable to all time series with finite second moments regardless of stationarity.

ii.

$$\hat{X}_n = \begin{cases} 0 & for\ n = 1 \\ P(X_n|X_{n-1}, \ldots, X_1) & for\ n = 2, 3, \ldots \end{cases}$$

Note that $\hat{X}_n$ is not an estimator, it is the BLP of X_n.
In particular, it is a linear combination of prior data points.

Note that $\hat{X}_1$ is the BLP of X_1 for which we have no prior data, so $\hat{X}_1 = 0$.
This will be the base case from which the model will be constructed recursively.

iii. $v_n = E[(X_{n+1} - \hat{X}_{n+1})^2] = E[(X_{n+1} - P(X_{n+1}|X_n, \ldots, X_1))]^2$

Note that ν_n is the MSPE of the BLP.

iv. The nth innovation is given by $U_n = X_n - \hat{X}_n$.

Note that while error terms are not observable, $U_n = X_n - \hat{X}_n$ effectively approximates Z_n.

For $n \geq 2$, consider expressing the innovations in terms of BLP's with coefficients β (in place of a's),

$$U_n = X_n - \hat{X}_n = X_n - \beta_{n-1,1}X_{n-1} - \cdots - \beta_{n-1,n-1}X_1$$

Note that the β_{ij} terms have two subscripts corresponding to the $i+1$th innovation and jth previous case.

We can then write U_i for $i = 1, 2, \ldots, n$,

$$\begin{aligned}
U_1 &= X_1 - \hat{X}_1 = X_1 - 0 \\
U_2 &= X_2 - \hat{X}_2 = X_2 - \beta_{11}X_1 \\
U_3 &= X_3 - \hat{X}_3 = X_3 - \beta_{21}X_1 - \beta_{22}X_1 \\
&\vdots \\
X_n - \hat{X}_n &= X_n - \beta_{n-1,1}X_{n-1} - \beta_{n-1,2}X_{n-2} - \cdots - \beta_{n-1,n-1}X_1
\end{aligned}$$

Define vectors
i.

$$\mathbf{U}_n = \begin{bmatrix} U_1 & \cdots & U_n \end{bmatrix}^T$$

ii.

$$\mathbf{X}_n = \begin{bmatrix} X_1 & \cdots & X_n \end{bmatrix}^T$$

iii. Define the vector of one-step predictions as

$$\hat{\mathbf{X}}_n = \begin{bmatrix} 0 & P(X_2|X_1) & \cdots & P(X_n|X_{n-1}, \ldots, X_1) \end{bmatrix}^T = \begin{bmatrix} 0 & P_1X_2 & \cdots & P_{n-1}X_n \end{bmatrix}^T$$

We can write $\mathbf{U}_n$ as a linear transformation of $\mathbf{X}_n$.

$$\mathbf{U}_n = \mathbf{A}_{n\times n}\mathbf{X}_n$$

Where

$$\mathbf{A}_{n\times n} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ a_{11} & 1 & 0 & \cdots & 0 & 0 \\ a_{22} & a_{21} & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ a_{n-1,n-1} & a_{n-1,n-2} & a_{n-1,n-3} & \cdots & a_{n-1,1} & 1 \end{bmatrix}$$

Note that the ith row of $\mathbf{A}_{n\times n}$ corresponds to U_i.
Note that $a_{ij} = -\beta_{ij}$.

Because the linear predictors are expressed in terms of X's, it is intuitive that the innovations could be expressed by a linear transformation of the X's by collecting terms appropriately.

Recall that a triangular matrix is invertible iff no element on its principal diagonal is 0.
Thus $\mathbf{A}_{n\times n}$ is non-singular with inverse $\mathbf{C}_{n\times n}$ given by

$$\mathbf{C}_{n\times n} = \begin{bmatrix} 1 & 0 & 0 & \cdots & 0 & 0 \\ \theta_{11} & 1 & 0 & \cdots & 0 & 0 \\ \theta_{22} & \theta_{21} & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \theta_{n-1,n-1} & \theta_{n-1,n-2} & \theta_{n-1,n-3} & \cdots & \theta_{n-1,1} & 1 \end{bmatrix}$$

Using $\mathbf{C}_{n\times n} = \mathbf{A}_{n\times n}^{-1}$, we can express $\mathbf{X}_n$ as a linear transformation of $\mathbf{U}_n$,

$$\mathbf{X}_n = \mathbf{C}_{n\times n}\mathbf{U}_n$$

Therefore,

$$\begin{aligned}\hat{\mathbf{X}}_n &= \mathbf{X}_n - (\mathbf{X}_n - \hat{\mathbf{X}}_n)\\ &\mathbf{X}_n - \mathbf{U}_n\\ &= \mathbf{C}_{n\times n}\mathbf{U}_n - \mathbf{U}_n\\ &= (\mathbf{C}_{n\times n} - \mathbf{I}_{n\times n})\mathbf{U}_n\end{aligned}$$

Note that $\mathbf{C}_{n\times n} - \mathbf{I}_{n\times n}$ simply eliminates the diagonal 1 terms leaving only the θ terms.
Thus define $\theta_{n\times n} = \mathbf{C}_{n\times n} - \mathbf{I}_{n\times n}$.

$$\begin{aligned}&= \theta_{n\times n}\mathbf{U}_n\\ &= \theta_{n\times n}(\mathbf{X}_n - \hat{\mathbf{X}}_n)\end{aligned}$$

Thus we have derived the following.

$$\hat{\mathbf{X}}_n = \theta_{n\times n}(\mathbf{X}_n - \hat{\mathbf{X}}_n)$$

Where

$$\theta_{n\times n} = \begin{bmatrix} 0 & 0 & 0 & \cdots & 0 & 0\\ \theta_{11} & 0 & 0 & \cdots & 0 & 0\\ \theta_{22} & \theta_{21} & 0 & \cdots & 0 & 0\\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots\\ \theta_{n-1,n-1} & \theta_{n-1,n-2} & \theta_{n-1,n-3} & \cdots & \theta_{n-1,1} & 0\end{bmatrix}$$

Thus we can write the forecasted case $\mathbf{X}_n$ in terms of θ_n times the nth innovation.
This is useful because this demonstrates that the $n+1$th BLP (forecasted value) depends on the prior innovations as opposed to simply X terms.

Note that the above satisfies

$$\hat{X}_{n+1} = \begin{cases} 0 & for\ n = 0\\ \sum_{j=1}^{n}\theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j}) & for\ n = 1, 2, \ldots\end{cases}$$

The linear combinations will be orthogonal (like residuals in regression).

As a system of equations we have

$$\begin{aligned}\hat{X}_1 &= 0\\ \hat{X}_2 &= \theta_{11}(X_1 - \hat{X}_1) = \theta_{11}X_1\\ \hat{X}_3 &= \theta_{21}(X_2 - \hat{X}_2) + \theta_{22}(X_1 - \hat{X}_1) = \theta_{21}(X_2 - \hat{X}_2) + \theta_{22}X_1\\ \hat{X}_4 &= \theta_{31}(X_3 - \hat{X}_3) + \theta_{32}(X_2 - \hat{X}_2) + \theta_{33}(X_1 - \hat{X}_1) = \theta_{31}(X_3 - \hat{X}_3) + \theta_{32}(X_2 - \hat{X}_2) + \theta_{33}X_1\\ &\vdots\end{aligned}$$

These expressions provide us with another method for determining the BLP and for estimating the θ MA coefficients.

Interpretation of the Innovations Algorithm:
Note that while error terms are not observable, $U_n = X_n - \hat{X}_n$ effectively approximates Z_n.
We use the innovations algorithm to recursively estimate

$$\theta_n = (\theta_{n_1}, \theta_{n_2}, \ldots, \theta_{n_n})^T$$

Note that the innovations algorithm generates the linear process coefficients θ_i for an MA(q) model.

Represent X_{n+1} Using Innovations:
Define $\theta_{n0} \equiv 1$.

$$X_{n+1} = X_{n+1} - \hat{X}_{n+1} + \hat{X}_{n+1}$$

$$= \theta_{n0}(X_{n+1} - \hat{X}_{n+1}) + \sum_{j=1}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})$$

$$= \sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})$$

Theorem - Innovations are Uncorrelated:
Innovations U_j, $j = 1, \ldots, n-1$ are uncorrelated with U_n.

$$E[U_n U_j] = 0, \quad j = 1, 2, \ldots, n-1$$

Prove in HW 3.

Deriving an Expression for v_n:
Recall that $v_n = E[(X_{n+1} - \hat{X}_{n+1})^2]$.

$$v_n = E[[(X_{n+1} - \hat{X}_{n+1})^2]$$

Substituting our earlier expression for $\hat{X}_{n+1}$,

$$= E\left[\left(X_{n+1} - \sum_{j=1}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})\right)^2\right]$$

$$= E[X_{n+1}^2] - 2\sum_{j=1}^{n} \theta_{nj} E[(X_{n+1-j} - \hat{X}_{n+1-j})X_{n+1}] + E\left[\left(\sum_{j=1}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})\right)^2\right]$$

Consider the cross product term.
Recall that $X_{n+1} = \sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})$.

$$\sum_{j=1}^{n} \theta_{nj} E[(X_{n+1-j} - \hat{X}_{n+1-j})X_{n+1}] = \sum_{j=1}^{n} \theta_{nj} E\left[(X_{n+1-j} - \hat{X}_{n+1-j}) \sum_{i=0}^{n} \theta_{ni}(X_{n+1-i} - \hat{X}_{n+1-i})\right]$$

Pulling out the $i = 0$ term from the second sum, and noting that $\theta_{n0} = 1$, we obtain

$$= \sum_{j=1}^{n} \theta_{nj} \left(E[(X_{n+1-j} - \hat{X}_{n+1-j})(X_{n+1} - \hat{X}_{n+1})] + \sum_{i=1}^{n} \theta_{ni} E[(X_{n+1-j} - \hat{X}_{n+1-j})(X_{n+1-i} - \hat{X}_{n+1-i})]\right)$$

Distributing the sum in terms of j, we obtain

$$= \sum_{j=1}^{n} \theta_{nj} E[(X_{n+1-j} - \hat{X}_{n+1-j})(X_{n+1} - \hat{X}_{n+1})] + \sum_{j=1}^{n}\sum_{i=1}^{n} \theta_{nj}\theta_{ni} E[(X_{n+1-j} - \hat{X}_{n+1-j})(X_{n+1-i} - \hat{X}_{n+1-i})]$$

Noting that innovations U_j, $j = 1, \ldots, n-1$ are uncorrelated with U_n, the first sum disappears entirely and the only terms that remain when performing the double sum occur when the indices are the same.

$$= \sum_{j=1}^{n} \theta_{nj}^2 E[(X_{n+1-j} - \hat{X}_{n+1-j})^2]$$

Consider the last term.

$$E\left[\left(\sum_{j=1}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})\right)^2\right]$$

When expanding the square, we note that (as before) the only terms that remain in the double sum occur when the indices are the same.

$$= \sum_{j=1}^{n} \theta_{nj}^2 E\left[(X_{n+1-j} - \hat{X}_{n+1-j})^2\right]$$

Thus, these are like terms.

$$v_n = E[X_{n+1}^2] - 2\sum_{j=1}^{n} \theta_{nj} E[(X_{n+1-j} - \hat{X}_{n+1-j})^2] + \sum_{j=1}^{n} \theta_{nj} E[(X_{n+1-j} - \hat{X}_{n+1-j})^2]$$

$$= E[X_{n+1}^2] - \sum_{j=1}^{n} \theta_{nj} E[(X_{n+1-j} - \hat{X}_{n+1-j})^2]$$

$$= \kappa(n+1, n+1) - \sum_{j=1}^{n} \theta_{nj}^2 v_{n-j}$$

Changing the indices, we obtain

$$v_n = \kappa(n+1, n+1) - \sum_{j=0}^{n-1} \theta_{n,n-j}^2 v_j$$

Prove the update step for θ in HW 3.

Recursive Innovations Algorithm Expressions for v_n, θ_n:
The coefficients $\theta_{n1}, \ldots, \theta_{nn}$ can be computed recursively from the equations

$$v_0 = \kappa(1,1)$$

$$\theta_{n,n-k} = v_k^{-1}\left(\kappa(n+1, k+1) - \sum_{j=0}^{k-1} \theta_{k,k-j} - \theta_{n,n-j} v_j\right), \quad 0 \le k \le n$$

$$v_n = \kappa(n+1, n+1) - \sum_{j=0}^{n-1} \theta_{n,n-j}^2 v_j$$

Follow the form of the $\theta_{n\times n}$ matrix.
i. $v_0 = \kappa(1,1)$
ii. Next, find θ_{11}, then v_1
iii. Next, find θ_{22}, θ_{21}, then v_2
iv. Next, find $\theta_{33}, \theta_{32}, \theta_{31}$, then v_3
etc.

Recall that the innovations algorithm technically relates to derivation of the BLP of true (population parameter) models, not estimation.
To estimate θ_n and v_n using the innovations algorithm, simply replace the ACVF $\kappa(i,j)$ by the sample ACVF $\hat{\gamma}(h = |i-j|)$ (for a stationary model).
In `R`, use the `acf()` function with the argument `type = covariance`.

Fitted Innovations MA(m) Model:
The fitted innovations MA(m) model is given by

$$X_t = Z_t + \hat{\theta}_{m1} Z_{t-1} + \cdots + \hat{\theta}_{mm} Z_{t-m}, \quad Z_t \sim WN(0, \hat{v}_m)$$

Note that $\hat{\theta}_{mj}$ (with two subscripts) and $\hat{v}_m$ denote the recursive formulas of the innovations algorithm.
In particular, the vector $\hat{\theta}_m$ and scalar $\hat{v}_m$ are obtained from the innovations algorithm with the ACVF $\kappa(i,j)$ replaced by the sample ACVF $\hat{\gamma}(h = i-j)$.

Note that we express the model as order m because, in practice, the model order will not be known ahead of time.

Estimating ARMA(p,q) Parameters (p>0, q>0) Using the Innovations Algorithm:
Recall that if an ARMA(p,q) model is causal, then

$$X_t = \sum_{j=0}^{\infty} \Psi_j Z_{t-j}$$

i. $\Psi_j = \theta_j + \sum_{i=1}^{\min(j,p)} \phi_i \Psi_{j-i}$, $j = 0, 1, \ldots$
ii. $\theta_0 = 1$
iii. $\theta_j = 0$, $j > q$

Note that the innovations algorithm generates the linear process coefficients.
We can use the innovations algorithm to estimate Ψ_j, ie find $\hat{\Psi}_j$'s.

The above relationship can then be used to find $\hat{\theta}_i$ and $\hat{\phi}_i$.
However, note that the solution is not guaranteed to be causal.

Suppose $q = 2$, and $p = 3$.
In other words, we would like to estimate $\{\theta_1, \theta_2, \phi_1, \phi_2, \phi_3\}$.
Let $j = 1, 2, \ldots, p + q = 5$.

For $j = 0$,

$$\Psi_0 = 1$$

For $j = 1$,

$$\Psi_1 = \theta_1 + \phi_1 \Psi_0 = \theta_1 + \phi_1$$

For $j = 2$,

$$\Psi_2 = \theta_2 + \phi_1 \Psi_1 + \phi_2 \Psi_0 = \theta_2 + \phi \Psi_1 + \phi_2$$

For $j = 3$,

$$\Psi_3 = \theta_3 + \phi_1 \Psi_2 + \phi_2 \Psi_1 + \phi_3 = \phi_1 \Psi_2 + \phi_2 \Psi_1 + \phi_3$$

For $j = 4$,

$$\Psi_4 = \phi_1 \Psi_3 + \phi_2 \Psi_2 + \phi_3 \Psi_1$$

For $j = 5$,

$$\Psi_5 = \phi_1 \Psi_4 + \phi_2 \Psi_3 + \phi_3 \Psi_2$$

Note that the θ terms have been exhausted for the last three equations.
These equations can be written as a matrix expression that allows us to solve for the $\hat{\phi}_i$ terms directly.
Using these values, we can then solve for the θ_i terms.

$$\begin{bmatrix} \Psi_3 \\ \Psi_4 \\ \Psi_5 \end{bmatrix} = \begin{bmatrix} \Psi_2 & \Psi_1 & 1 \\ \Psi_3 & \Psi_2 & \Psi_1 \\ \Psi_4 & \Psi_3 & \Psi_2 \end{bmatrix} \begin{bmatrix} \phi_1 \\ \phi_2 \\ \phi_3 \end{bmatrix}$$

However, note that the solution is not guaranteed to be causal.

Forecasting ARMA(p,q) Models Using the Innovations Algorithm - Expressing the ARMA(p,q) Model in Terms of Innovations:
Let $\{X_t\}$ be an ARMA(p,q) process,

$$\Phi(B) X_t = \Theta(B) Z_t, \quad Z_t \sim WN(0, \sigma^2)$$

Define the transformed time series process

$$W_t = \begin{cases} \sigma^{-1} X_t & for \;\; t = 1, \ldots, m \\ \sigma^{-1} \Phi(B) X_t & for \;\; t > m \end{cases}$$

For $m = \max(p, q)$.

Suppose $p > q$ which implies that $m = p$.
The transformed process W_t indicates that the first $m = p$ terms will be defined in terms of X_t, $\sigma^{-1}X_t$.
The remaining terms will incorporate an autoregressive structure through the incorporation of the $\Phi(B)$ term, $\sigma^{-1}\Phi(B)X_{t-1}$.

This transformed process allows us to conveniently express the innovations algorithm in terms of an ARMA(p,q) model.

The autocovariance function of W_t is given by

$$\kappa_W(i,j) = \begin{cases} \sigma^{-2}\gamma_X(i-j) & for \;\; 1 \leq i,j \leq m \\ \sigma^2\left(\gamma_X(i-j) - \sum_{r=1}^{p}\phi_r\gamma_X(r-|i-j|)\right) & for \;\; \min(i,j) \leq m < \max(i,j) \leq 2n \\ \sum_{r=0}^{q}\theta_r\theta_{r+|i-j|} & for \;\; \min(i,j) > m \\ 0 & otherwise \end{cases}$$

Applying linearity of the prediction operator $P(\cdot|X_n)$, we obtain

$$\hat{W}_t = \begin{cases} \sigma^{-1}\hat{X}_t & for \;\; t = 1,\ldots,m \\ \sigma^{-1}\left(\hat{X}_t - \phi_1 X_{t-1} - \cdots - \phi_p X_{t-p}\right) & for \;\; t > m \end{cases}$$

For $m = \max(p,q)$.

Thus we can write innovations for the ARMA(p,q) process for all $t \geq 1$ as

$$X_t - \hat{X}_t = \sigma\left(W_t - \hat{W}_t\right)$$

Thus the ARMA(p,q) model expressed in terms of the innovations algorithm is given by the one-step ahead best linear predictor (in terms of θ's and ϕ's),

$$\hat{X}_{n+1} = \begin{cases} \sum_{j=1}^{n}\theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j}) & for \;\; 1 \leq n < m \\ \phi_1 X_n + \cdots + \phi_p X_{n+1-p} + \sum_{j=1}^{q}\theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j}) & n \geq m \end{cases}$$

For $m = \max(p,q)$.

The MSPE is given by

$$E[(X_{n+1} - \hat{X}_{n+1})^2] = \sigma^2 E[(W_{n+1} - \hat{W}_{n+1})^2] = \sigma^2 r_n$$

Recall that $v_n = E[(X_{n+1} - \hat{X}_{n+1})^2]$.
By convention, we write $r_n = E[(W_{n+1} - \hat{W}_{n+1})^2]$.

Thus, we have written the one-step ahead best linear predictor (using the innovations representation $X_n - \hat{X}_n$) in terms of θ's and ϕ's if we assume an underlying ARMA(p,q) process.
This will allow us to make predictions using estimated $\hat{\phi}$ and $\hat{\theta}$ values.
Recall that the innovations algorithm can be used for stationary and non-stationary processes.

For any $n < m$ (small n less than (close to) the order of the model), the 1-step prediction resembles the MA(q) BLP.
For $n > m$ (large n greater than the order of the model), the 1-step prediction resembles the AR(p) (autoregressive) BLP plus the MA(q) BLP.

Expressing the ARMA(p,q) model in terms of innovations is important because it allows us to express the maximum likelihood function of the ARMA(p,q) in terms of innovations.

Innovations and Maximum Likelihood Estimation of Gaussian Time Series:

Motivating (Multivariate) Gaussian Time Series - Bivariate Normal Time Series:
Consider the random vector $\mathbf{X}_2 = (X_1, X_2)^T$, where $\mathbf{X} \sim BN(\mathbf{0}_2, \mathbf{\Sigma_{2\times 2}})$.
Note that

$$\mathbf{\Sigma}_{n\times n} = \begin{bmatrix} \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 \end{bmatrix} = \sigma^2 \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}$$

The PDF of $\mathbf{X}_2$ is given by

$$f(X_1, X_2; \mathbf{\Sigma}_{n\times n}) = \frac{1}{2\pi\sigma^2(1-\rho^2)}\exp\left(-\frac{1}{2(1-\rho^2)\sigma^2}\left(X_1^2 - 2\rho X_1 X_2 + X_2^2\right)\right)$$

Note that for a time series $\{X_t\}$,

$$\mathbf{\Sigma}_{n\times n} = \begin{bmatrix} \sigma^2 & \rho\sigma^2 \\ \rho\sigma^2 & \sigma^2 \end{bmatrix} \equiv \mathbf{\Gamma}_{n\times n} = \begin{bmatrix} \gamma(0) & \gamma(1) \\ \gamma(1) & \gamma(0) \end{bmatrix}$$

Additionally, $\rho = \frac{\gamma(1)}{\gamma(0)}$, $\sigma^2 = \gamma(0)$.

Note that
i. $E[X_2|X_1 = x_1] = E[X_2] + \rho\frac{\sigma}{\sigma}(x_1 - (E[X_1])^2) = \frac{\gamma(1)}{\gamma(0)}x_1$

Recall that in regression, the LOBF is given by

$$\hat{\beta}_0 + \hat{\beta}_1 x = (\bar{y} - \hat{\beta}_1\bar{x}) - \hat{\beta}_1 x = \bar{y} + \hat{\beta}_1(x - \bar{x})$$
$$= \bar{y} + r\frac{S_Y}{S_X}(x - \bar{x})$$

In particular, $E[X_2|X_1 = x_1] = \frac{\gamma(1)}{\gamma(0)}x_1$ is a linear combination of past terms (i.e., x_1), resembling a BLP.
The BLP for this application is in fact,

$$\hat{X}_2 = P(X_2|X_1) = P_1 X_2 = E[X_2|X_1] = \frac{\gamma(1)}{\gamma(0)}X_1$$

ii. $E[(X_2 - (E[X_2|X_1])^2] = \gamma(0)(1 - \rho(1))^2$

In other words, the conditional mean square error is the same as the conditional variance.
Note that $E[X_2|X_1] \equiv \hat{X}_2$.

(Multivariate) Gaussian Time Series:
Let $\{X_t\}$ be a (Multivariate) Gaussian time series with zero mean and ACVF $\gamma(i,j) = E[X_i X_j]$.
Let

$$\mathbf{X}_n = \begin{bmatrix} X_1 & \cdots & X_n \end{bmatrix}^T$$
$$\hat{\mathbf{X}}_n = \begin{bmatrix} \hat{X}_1 & \cdots & \hat{X}_n \end{bmatrix}^T$$

For
i. $\hat{X}_1 = 0$
ii. $\hat{X}_j = E[X_j|X_1, \ldots, X_{j-1}] \equiv P(X_j|X_{j-1}, \ldots, X_1) = P_{j-1}X_j$
Note that the equivalence of the conditional average and the best linear predictor holds by generalization of the above discussion regarding bivariate time series.

Let $\{X_t\} \sim MN(\mathbf{0}_n, \mathbf{\Gamma}_{n\times n})$, for
i. $\mu_n = \mathbf{0}_n$.
ii. $\mathbf{\Gamma}_{n\times n} = E[\mathbf{X}_n\mathbf{X}_n]^T$

The likelihood of the Gaussian time series is given by

$$L(\mathbf{\Gamma}_{n\times n}; x_1, \ldots, x_n) = (2\pi)^{-n/2}(\det(\mathbf{\Gamma}_{n\times n}))^{-1/2}\exp\left(-\frac{1}{2}\mathbf{X}_n^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_n\right)$$

Innovations Form of the Likelihood of a (Multivariate) Gaussian Time Series:
Recall that the likelihood of the Gaussian time series is given by

$$L(\mathbf{\Gamma}_{n\times n}; \mathbf{x}_n) = (2\pi)^{-n/2}(\det(\mathbf{\Gamma}_{n\times n}))^{-1/2}\exp\left(-\frac{1}{2}\mathbf{X}_n^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_n\right)$$

Recall that $\mathbf{X}_n = \mathbf{C}_{n\times n}(\mathbf{X}_n - \hat{\mathbf{X}}_n)$.

Therefore,

$$\mathbf{\Gamma}_{n\times n} = E[\mathbf{X}_n\mathbf{X}_n^T]$$
$$= E[\mathbf{C}_{n\times n}(\mathbf{X}_n - \hat{\mathbf{X}}_n)(\mathbf{X}_n - \hat{\mathbf{X}}_n)^T\mathbf{C}_{n\times n}^T]$$

Because $\mathbf{C}_{n\times n}$ is a fixed matrix of constants,

$$= \mathbf{C}_{n\times n}E[(\mathbf{X}_n - \hat{\mathbf{X}}_n)(\mathbf{X}_n - \hat{\mathbf{X}}_n)^T]\mathbf{C}_{n\times n}^T$$

Note that $E[(\mathbf{X}_n - \hat{\mathbf{X}}_n)(\mathbf{X}_n - \hat{\mathbf{X}}_n)^T]$ is the covariance matrix of the innovations.
Because the innovations are uncorrelated unless they have the same index, the covariance matrix of the innovations consists of only diagonal elements $\{v_0, \ldots, v_{n-1}\}$ where the element v_i indicates the ith squared prediction error.

Define $\mathbf{D}_{n\times n} = diag\{v_0, \ldots, v_{n-1}\}$.

$$= \mathbf{C}_{n\times n}\mathbf{D}_{n\times n}\mathbf{C}_{n\times n}^T$$

Thus, the covariance matrix has been expressed as a linear transformation of the innovations times a diagonal matrix of squared prediction errors.

Using this expression for the covariance matrix, we can solve $\mathbf{X}_n^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_n$.

$$\mathbf{X}_n^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_n = (\mathbf{X}_n - \mathbf{X}_n^T)\mathbf{C}_{n\times n}^T(\mathbf{C}_{n\times n}\mathbf{D}_{n\times n}\mathbf{C}_{n\times n}^T)^{-1}\mathbf{C}_{n\times n}(\mathbf{X}_n - \hat{\mathbf{X}}_n)$$

Note that $(\mathbf{C}_{n\times n}\mathbf{D}_{n\times n}\mathbf{C}_{n\times n}^T)^{-1} = (\mathbf{C}_{n\times n}^T)^{-1}\mathbf{D}_{n\times n}^{-1}\mathbf{C}_{n\times n}^{-1}$.

$$= (\mathbf{X}_n - \mathbf{X}_n^T)\mathbf{C}_{n\times n}^T(\mathbf{C}_{n\times n}^T)^{-1}\mathbf{D}_{n\times n}^{-1}\mathbf{C}_{n\times n}^{-1}\mathbf{C}_{n\times n}(\mathbf{X}_n - \hat{\mathbf{X}}_n)$$
$$= (\mathbf{X}_n - \mathbf{X}_n^T)\mathbf{D}_{n\times n}^{-1}(\mathbf{X}_n - \hat{\mathbf{X}}_n)$$

Note that $\det(\mathbf{\Gamma_{n\times n}}) = \det(\mathbf{C}_{n\times n})^2\det(\mathbf{D}_{n\times n}) = v_0\cdots v_{n-1}$ Therefore,

$$L(\mathbf{\Gamma}_{n\times n}; x_1, \ldots, x_n) = \frac{1}{\sqrt{(2\pi)^n v_0\cdots v_{n-1}}}\exp\left(-\frac{1}{2}\sum_{j=1}^{n}(x_j - \hat{x}_j)^2/v_{j-1}\right)$$

Thus we have parameterized the likelihood of a multivariate gaussian using the innovations parameters (as opposed to $\mathbf{\Gamma}_{n\times n}$), allowing us to estimate the parameters ϕ_n and θ_n.

Exact Likelihood of an ARMA(p,q) Process (Requires Gaussian Error Structure) Using Innovations:
Recall that the ARMA(p,q) model is given by

$$\Phi(B)X_t = \Theta(B)Z_t, \quad Z_t \sim N(0, \sigma^2)$$

Recall that the innovations algorithm expressed in terms of the ARMA(p,q) model is given by

$$\hat{X}_{n+1} = \begin{cases} \sum_{j=1}^{n}\theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j}) & for \;\; 1 \le n < m \\ \phi_1 X_n + \cdots + \phi_p X_{n+1-p} + \sum_{j=1}^{q}\theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j}) & n \ge m \end{cases}$$

For $m = \max(p, q)$.

The MSPE is given by

$$E[(X_{n+1} - \hat{X}_{n+1})^2] = \sigma^2 E[(W_{n+1} - \hat{W}_{n+1})^2] = \sigma^2 r_n$$

Recall that $v_n = E[(X_{n+1} - \hat{X}_{n+1})^2]$.
By convention, we write $r_n = E[(W_{n+1} - \hat{W}_{n+1})^2]$.

Expressing the likelihood of a (multivariate) Gaussian ARMA(p,q) time series in the form of innovations, we have

$$L(\phi_n, \theta_n, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2 r_0\cdots r_n}}\exp\left(-\frac{1}{2\sigma^2}\sum_{j=1}^{n}(x_j - \hat{x}_j)^2/r_{j-1}\right)$$

Recall that we were able to express Z_t's in terms of X's for the MA(q) process when deriving the conditional likelihood using the unrealistic assumption $Z_0 = 0$ and a recursive expression, $Z_t = X_t - \theta Z_{t-1}$.
However, it is extremely cumbersome to express an ARMA(p,q) process in terms of X's using traditional methods because the the recursive dependency structure inherent in the models.

In this case, we have noted that the innovations allowed us to derive a (recursive) expression of the BLP (X's) in an ARMA(p,q) model.
Consequently, the innovations algorithm allows us to estimate the parameters ϕ_n and θ_n by re-expressing the likelihood in terms of innovations.

Equivalence of Least Squares and Maximum Likelihood for an ARMA(p,q) Process (Requires Gaussian Error Structure):
In particular, suppose $\hat{\phi}_n$ and $\hat{\theta}_n$ are the values of the parameters the minimize

$$-l(\phi_n, \theta_n) = \log\left(\frac{1}{n}S(\phi_n, \theta_n)\right) + \frac{1}{n}\sum_{j=1}^{n}\log(r_{j-1})$$

Where $S(\phi_n, \theta_n)$ is the least squares objective (expressed in terms of innovations), which resembles SSE from linear regression.

$$S(\hat{\phi}_n, \hat{\theta}_n) = \sum_{j=1}^{n} (x_j - \hat{x}_j)^2 / r_{j-1}$$

Using this structure, $\hat{\sigma}^2$ is given by

$$\hat{\sigma}^2 = \frac{1}{n} S(\hat{\phi}_n, \hat{\theta}_n)$$

Note that the least squares objective $S(\phi_n, \theta_n)$ resembles SSE from linear regression, and that minimizing the least squares objective is equivalent to maximizing likelihood under a Gaussian error structure.

Model Order Selection of ARMA(p,q) Models:

1. Exploratory Data Analysis

   For pure AR(p) or pure MA(q) models, the PACF or ACF may be used to determine the order respectively.
   In practice, you will not know if you have a pure AR(p) or pure MA(q) process.

2. AIC-Corrected (AICC)

   Even if the noise structure is not normal, we can still use MLE quantities to measure the goodness-of-fit and compute relevant metrics (AIC, BIC).

   Recall that the Kulback-Liebler divergence (derived from cross-entropy in information theory) measures the "distance" between the true distribution $\mathbb{P}(x)$ and the observed distribution $\mathbb{Q}(x)$, i.e. between two probability functions.

   $$KL(\mathbb{P}|\mathbb{Q}) = \int \mathbb{P}(x) \log\left(\frac{\mathbb{P}(x)}{\mathbb{Q}(x)}\right) \, dx$$

   In particular, we choose p, q, ϕ_p, θ_q which minimizes AICC,

   $$AICC = -2l(\phi_p, \theta_q, S(\phi_p, \theta_q)/n) + \frac{2(p+q+1)n}{(n-p-q-2)}$$

   Where the addition term is the penalty term for excess model parameters.

3. Data Split Cross-Validation

   A classic data split can be used to determine model order (p, q) based on minimizing test error, or some innovations-related quantity using cross-validation.

   Be mindful on how the data is split, because we cannot simply resample due to the dependency structure in the data.

Examples:

Example: Yule-Walker Estimation (MOM) Limitation - MA(1)

Consider the MA(1) process given by

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2)$$

Recall

$$\gamma_X(h) = \begin{cases} \sigma^2(1+\theta^2) & for \quad h = 0 \\ \sigma^2\theta & for \quad |h| = 1 \\ 0 & for \quad |h| > 1 \end{cases}$$

Find the Yule-Walker (MOM) estimators of θ and σ^2.

To solve for the 2 parameters (unknowns) θ and σ^2, we require 2 equations.

$$\gamma(0) = \sigma^2(1+\theta^2)$$
$$\gamma(1) = \sigma^2(\theta)$$

Equivalently, we can write

$$\gamma(0) = \sigma^2(1+\theta^2)$$

$$\rho(1) = \frac{\theta}{1+\theta^2}$$

Note that there is no real solution for $|\rho(1)| > 0.5$.

$$\rho(1) + \rho(1)\theta^2 - \theta = 0$$

To obtain a real solution, we must have a non-negative discriminant,

$$1 - 4\rho(1)^2 \geq 0$$

Equivalently,

$$\rho(1) \leq \sqrt{\frac{1}{4}} \implies |\rho(1)| \leq \frac{1}{2}$$

Thus, we can only solve for the Yule-Walker (MOM) estimator if the MA(1) ACF has a lag-1 autocorrelation less than 1/2.
In this case,

$$\hat{\theta} = \frac{1 - \sqrt{1 - 4\hat{\rho}^2(1)}}{2\hat{\rho}(1)}$$

$$\hat{\sigma}^2 = \frac{\hat{\gamma}(0)}{1 + \hat{\theta}_1^2}$$

Thus, due to restrictions on the range of $\gamma_X(h)$ and $\rho_X(h)$ values required to obtain real solutions, Yule-Walker (MOM) estimators are a more effective tool for pure AR(p) models than for pure MA(q) or ARMA(p,q) models with $q > 0$.

Example: Exact Likelihood of the AR(1) Process

Consider the causal AR(1) process given by

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \overset{iid}{\sim} N(0, \sigma^2)$$

Consider a sample $\{X_1, \ldots, X_n\}$ derived from the above AR(1) process.

Recall that the ACVF of the AR(1) process is given by

$$\gamma_X(h) = \frac{\sigma^2}{1-\phi^2}\phi^{|h|}$$

$$\mu = E[\mathbf{X}_n] = \mathbf{0}_n$$

$$\begin{aligned}
\mathbf{\Sigma}_{n\times n} &= E[(\mathbf{X_n} - \mu_\mathbf{n})(\mathbf{X}_n - \mu_n)^T] = E[\mathbf{X}_n\mathbf{X}_n^T] \\
&= E\begin{bmatrix} X_1X_1 & X_1X_2 & \cdots & X_1X_n \\ X_2X_1 & X_2X_2 & \cdots & X_2X_n \\ \vdots & \vdots & \ddots & \vdots \\ X_nX_1 & X_nX_2 & \cdots & X_nX_n \end{bmatrix} \\
&= \begin{bmatrix} E[X_1X_1] & E[X_1X_2] & \cdots & E[X_1X_n] \\ E[X_2X_1] & E[X_2X_2] & \cdots & E[X_2X_n] \\ \vdots & \vdots & \ddots & \vdots \\ E[X_nX_1] & E[X_nX_2] & \cdots & E[X_nX_n] \end{bmatrix} \\
&= \begin{bmatrix} \gamma_X(0) & \gamma_X(1) & \cdots & \gamma_X(n-1) \\ \gamma_X(1) & \gamma_X(0) & \cdots & \gamma_X(n-2) \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_X(n-1) & \gamma_X(n-2) & \cdots & \gamma_X(0) \end{bmatrix} \\
&= \frac{\sigma^2}{1-\phi^2}\begin{bmatrix} 1 & \phi & \cdots & \phi^{n-1} \\ \phi & 1 & \cdots & \phi^{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ \phi^{n-1} & \phi^{n-2} & \cdots & 1 \end{bmatrix}
\end{aligned}$$

Example: True MA(1) Model Estimation - Innovations Algorithm

Consider a true MA(1) model, with population parameter θ known.

$$X_t = Z_t + \theta Z_{t-1}, \quad Z_t \sim WN(0, \sigma^2)$$

Recall that we can find a BLP using previously derived methods involving minimization of squared loss.
We will now use the innovations algorithm.

i. $\kappa(i,i) \overset{stationary}{=} \gamma(0) = \sigma^2(1+\theta)^2$
ii. $\kappa(i,i+1) \overset{stationary}{=} \gamma(1) = \theta\sigma^2$
iii. $\kappa(i,j) \overset{stationary}{=} \gamma(i-j) = 0$, for $|i-j| > 1$

Using the innovations algorithm we obtain
i. $\theta_{nj} = 0$, for $2 \leq j \leq n$
ii. $\theta_{n1} = v_{n-1}^{-1}\theta\sigma^2$
iii. $v_0 = (1+\theta^2)\sigma^2$
iv. $v_n = [1 + \theta^2 - v_{n-1}^{-1}\theta^2\sigma^2]\sigma^2$

We will simulate $X_t = Z_t - 0.9Z_{t-1}$ and attempt to estimate $\theta = -0.9$ using the innovations algorithm equations.

```
# Assume the true MA(1) is known, ie parameters theta, sigma^2 known
# Not realistic in practice

# Loop instead of recursion
# Note that values converge to the true theta and sigma^2

R = 75
sigma2 = 1
theta = -0.9

v0 = sigma2*(1+theta^2)

theta_n1 = NULL
v_n = NULL
v_n[1] = v0

# We calculate only the coefficient theta_n1 because we know the model order q=1
for (n in 2:R) {
  theta_n1[n] = 1/(v_n[n-1])*theta*sigma2
  v_n[n] = (1+theta^2-1/(v_n[n-1])*theta^2)*sigma2
}

tail(cbind(theta_n1,v_n))
```

```
##         theta_n1 v_n
## [70,] -0.8999999   1
## [71,] -0.8999999   1
## [72,] -0.8999999   1
## [73,] -0.9000000   1
## [74,] -0.9000000   1
## [75,] -0.9000000   1
```

Example: Yule-Walker Estimation - Build Yule-Walker Equations

```
# AR Simulation function - 20% burn in period
my_AR_sim = function(n=189, ar_coeff=c(.5), sigma=1) {
  n.start = n + floor(.2/(1-.2) * n)
  burnin = floor(.20 * n.start)
  model_params = list(ar = ar_coeff)
```

```
  ts_sim = arima.sim(model = model_params, n = n.start, sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
  return(ts_sim)
}

# Yule-Walker function, specify AR order p
my_yule_walker = function(ts_data, AR_order=2) {

  # rho_p vector - extract autocorrelations up to lag p
  rho_p = c(acf(ts_data,plot=F,lag.max=AR_order)$acf)

  # index matrix - add column of lag index to rho_p
  rho_p_indices = cbind(seq(0,length(rho_p)-1),rho_p) # combine gamma_h with lag index

  # Setup R (Corr) matrix
  Corr = matrix(NA,nrow=length(rho_p)-1,ncol=length(rho_p)-1)

  # nested loop to create R matrix
  for (i in 1:(length(rho_p)-1)) {
    for (j in 1:(length(rho_p)-1)) {

      # calculate lag, extract element of rho_p with lag h, store element
      h = abs(i-j)
      select_acvf = rho_p_indices[which(rho_p_indices[,1]==h),2]
      Corr[i,j] = c(select_acvf)
    }
  }

  # Solve Yule Walker equation
  return(c(solve(Corr)%*%matrix(rho_p[2:(1+AR_order)])))
}
```

```
n=100 # n= 10000
sigma = 1
ar_coeff = c(.5,.4)
main_exp = paste("AR sim: ", paste(ar_coeff,collapse=", "), paste(" (n=",n,")", sep=""))

(roots = polyroot(c(1,-1*ar_coeff)) )
```

```
## [1]  1.075184-0i -2.325184+0i
```

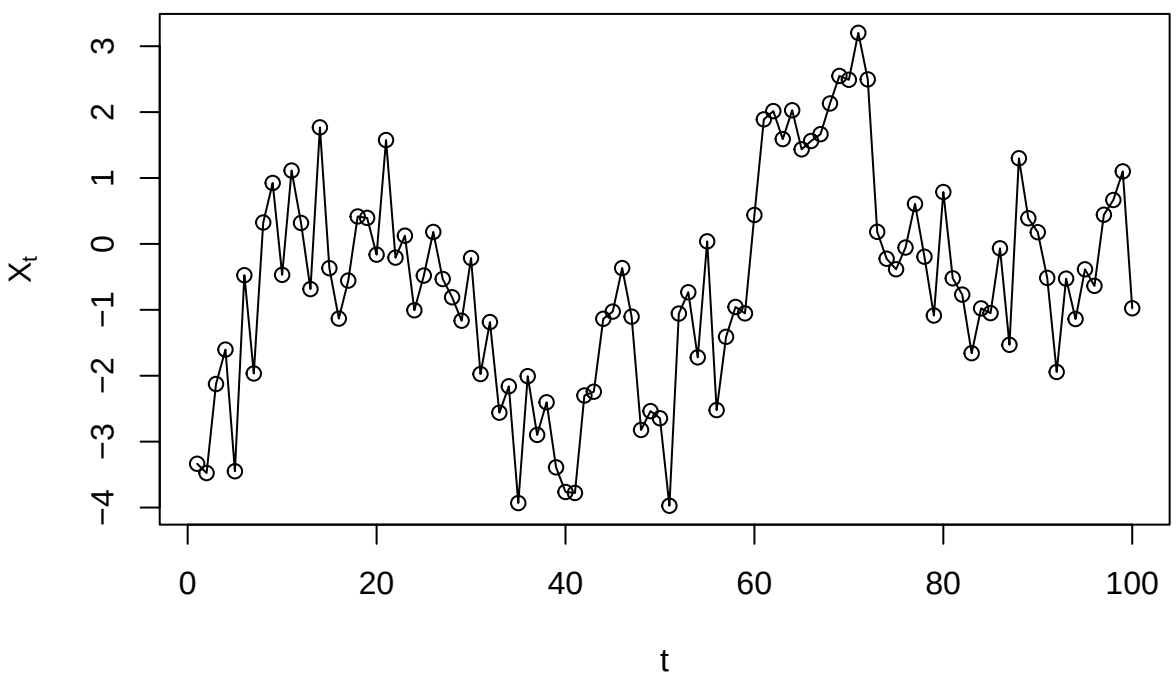
```
Mod(roots)
```

```
## [1] 1.075184 2.325184
```

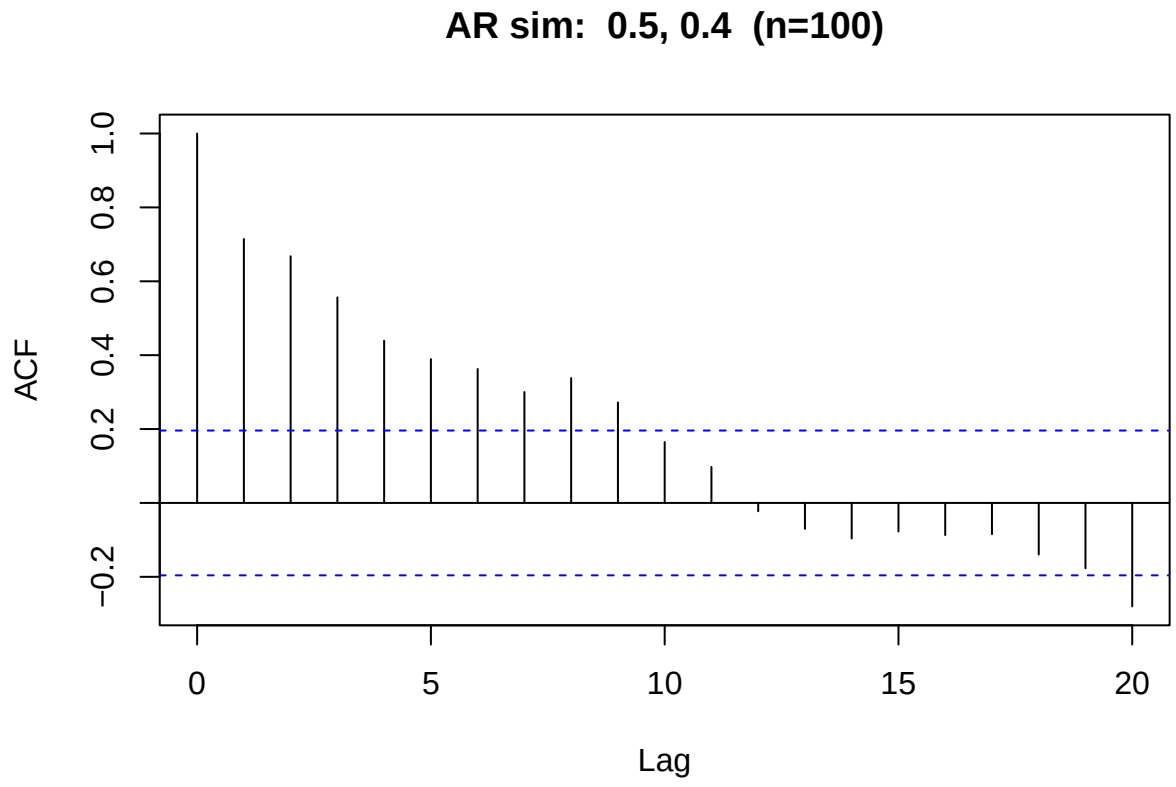
```
ar_sim = my_AR_sim(n=n, ar_coeff=ar_coeff, sigma=sigma)
length(ar_sim)
```

```
## [1] 100
```

```
plot(ar_sim,type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
```



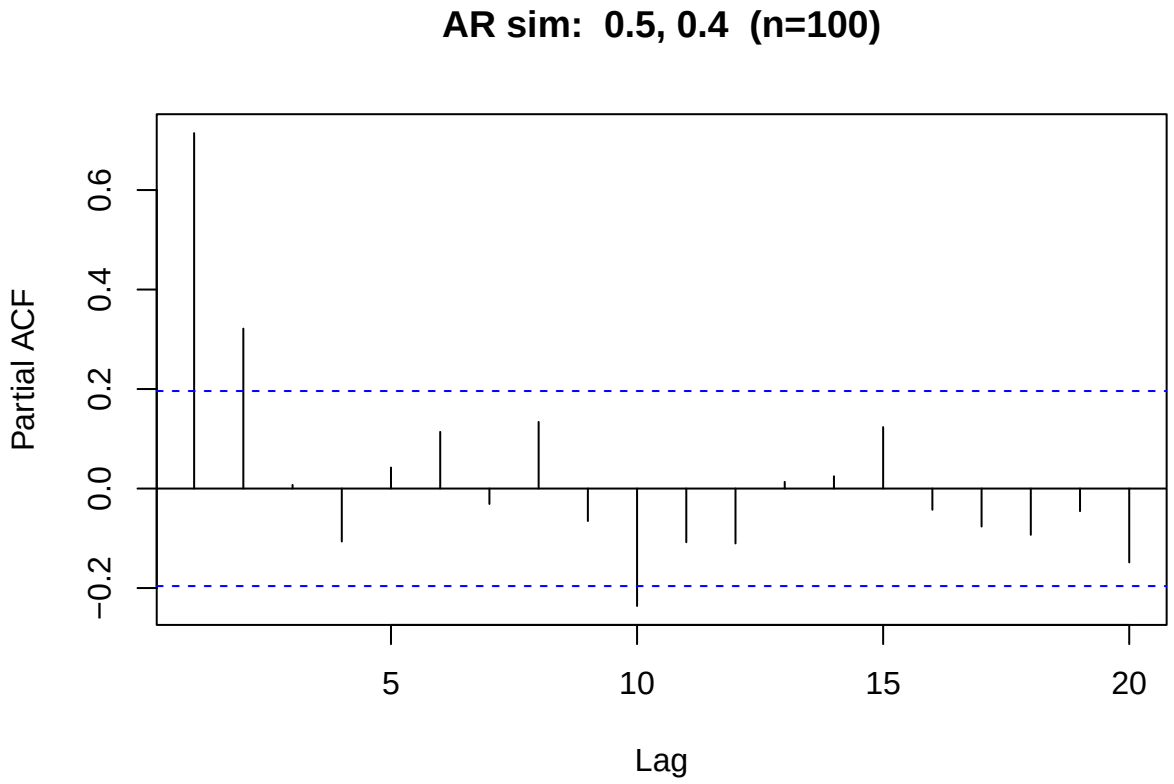

```
# decreases quickly for pure AR process - hard to tell
acf1 = acf(ar_sim, main=main_exp)
```



```
names(acf1)
```

```
## [1] "acf"    "type"   "n.used" "lag"    "series" "snames"
```

```
# extract autocorrelations
rho_ar2 = c(acf1$acf)

# indicates that we have an AR(2)
pacf(ar_sim, main=main_exp)
```




```
## Yule Walker estimates
yw = my_yule_walker(ts_data=ar_sim,AR_order=2)

# truth
ar = ar_coeff
data.frame(yw =yw ,ar=ar)
```

```
##          yw  ar
## 1 0.4847907 0.5
## 2 0.3214006 0.4
```

```
n=100 # n= 10000
sigma = 1
ar_coeff = c(.5,.4)
main_exp = paste("AR sim: ",paste(ar_coeff, collapse=", "), paste(" (n=",n,")", sep=""))

(roots <- polyroot(c(1,-1*ar_coeff)) )
```

```
## [1]  1.075184-0i -2.325184+0i
```

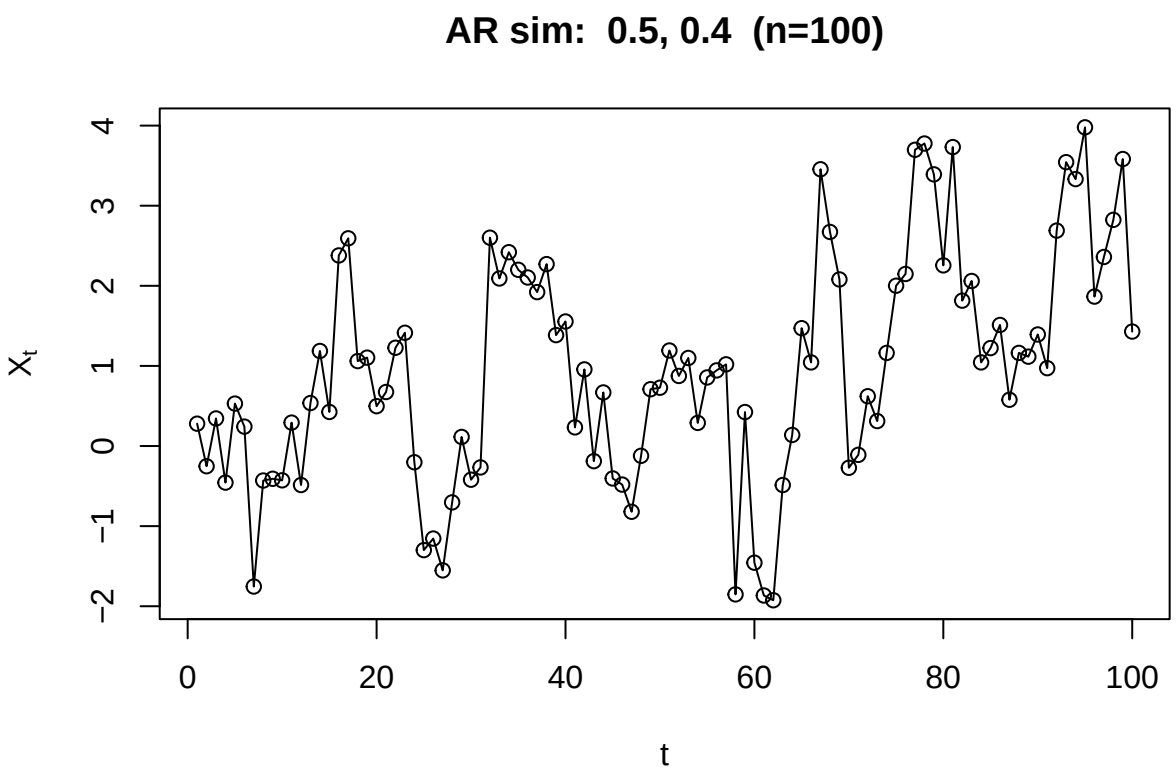
```
Mod(roots)
```

```
## [1] 1.075184 2.325184
```

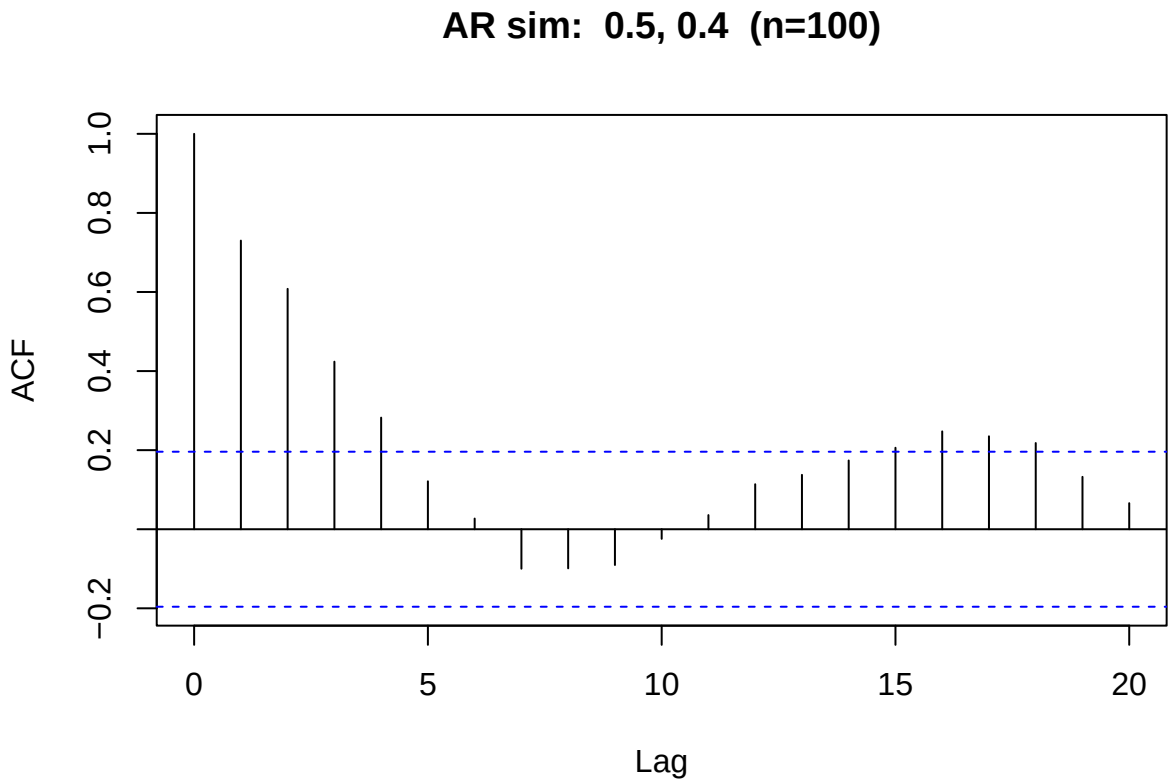
```
# we know the true order is 2 (but in practice only god knows that)
# we guess order 3

# EDA
ar_sim = my_AR_sim(n=n, ar_coeff=ar_coeff, sigma=sigma)
length(ar_sim)
```

```
## [1] 100
```

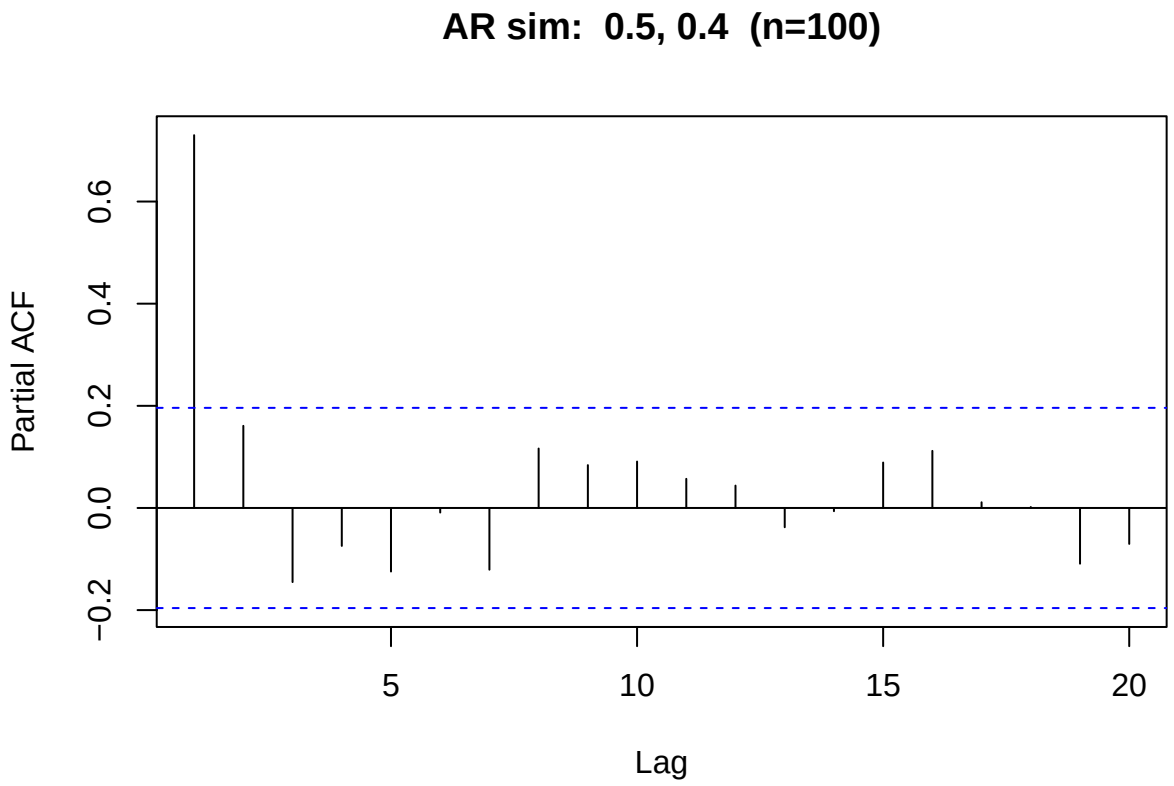
```
plot(ar_sim,type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
```




```
acf(ar_sim, main=main_exp)
```




```
pacf(ar_sim, main=main_exp)
```



```r
# Yule Walker estimates (Suppose we guess order 3 based on PACT)
yw = my_yule_walker(ts_data=ar_sim,AR_order=3)

# truth
ar = ar_coeff
# compare
yw
```

```
## [1]  0.6358921  0.2497064 -0.1451025
```

```r
ar
```

```
## [1] 0.5 0.4
```

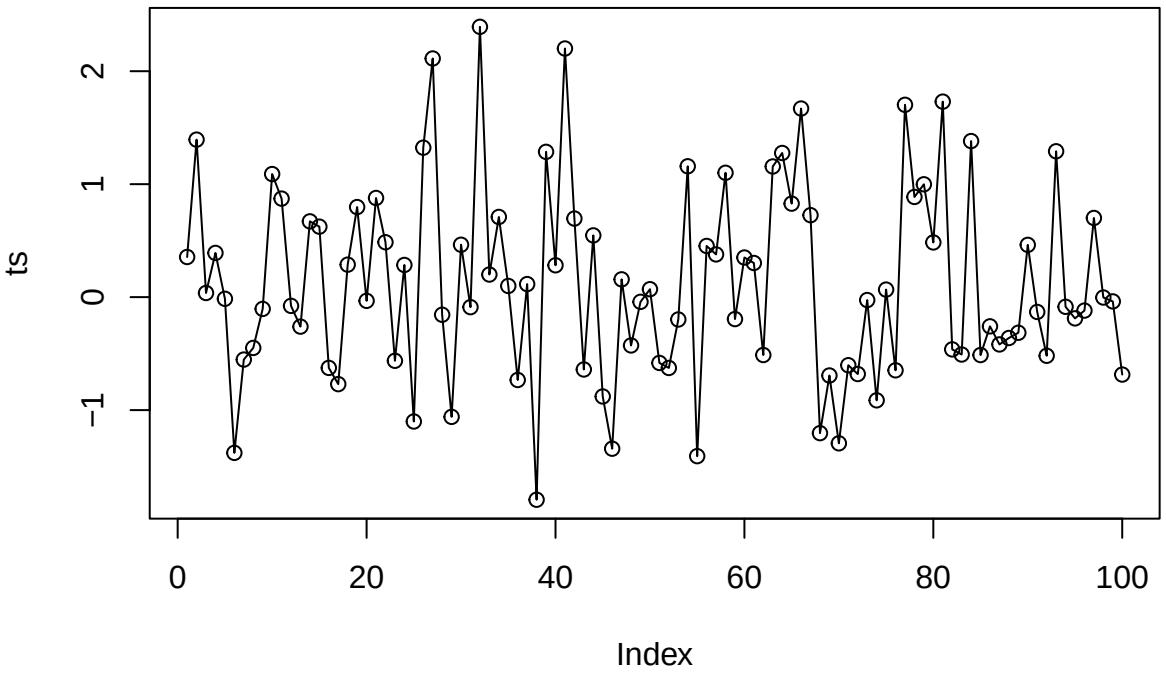
```r
# 3rd phi is close to 0 because true is p=2
# We could alternatively do a hypothesis test
# Increase sample size - more consistent estimator

# What about the estimated roots? Are they causal/stationary?
(roots = polyroot(c(1,-1*yw)) )
```

```
## [1]  1.439553+0i -2.051853-0i  2.333197+0i
```

Example: MLE - Simulation

```r
# AR Simulation function - 20% burn in period
my_AR_sim = function(n=189, ar_coeff=c(.5), sigma=1) {
  n.start = n + floor(.2/(1-.2) * n)
  burnin = floor(.20 * n.start)
  model_params = list(ar = ar_coeff)
  ts_sim = arima.sim(model = model_params, n = n.start, sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
  return(ts_sim)
}

# AR(1) Covariance Matrix from ACVF
covariance_matrix_AR1 = function(phi,sigma,n) {
  cov_mat = matrix(0,
                   nrow = n,
                   ncol = n)

  # Fill matrix with AR(1) lag terms
  for (i in 1:n) {
    for (j in 1:n) {
      cov_mat[i,j] = phi^abs(i-j)
    }
  }

  # Multiply by constant in AR(1) ACVF expression
  cov_mat = sigma^2/(1-phi^2)*cov_mat

  return(cov_mat)
}

# AR(1) Negative Log-Likelihood using Multivariate Normal Distribution
neg_log_like_MN_AR1 = function(params,x) {
  phi = params[1]
  sigma =  params[2]
```

```
  n = length(x)
  x = as.matrix(x)

  # covariance matrix Omega
  Omega = covariance_matrix_AR1(phi=phi,
                                sigma=sigma,
                                n=n)

  # quadratic form in exponential as vector
  quad_form = c(t(x)%*%solve(Omega)%*%x)

  # calculate negative log-likelihood assuming MN
  neg_log_like = (n/2)*log(2*pi)-(1/2)*log(det(solve(Omega)))+(1/2)*quad_form

  return(neg_log_like)
}

# AR(1) Negative Log-Likelihood Using Exact Likelihood
neg_log_like_exact_AR1 = function(params,x) {
  phi = params[1]
  sigma =  params[2]
  n = length(x)

  # calculate exact negative log-likelihood
  x0_part = (1/2)*log(2*pi)+(1/2)*log(sigma^2/(1-phi^2))+(1/(2*sigma^2))*(1-phi^2)*x[1]^2
  x_part = (1/2)*log(2*pi)+(n/2)*log(sigma^2)+(1/(2*sigma^2))*sum((x[2:n]-phi*x[1:(n-1)])^2)
  neg_log_like = x0_part+x_part

  return(neg_log_like)
}

# Test AR(1) negative log-likelihood minimization
set.seed(1)
ts = my_AR_sim(n=100,
               ar_coeff=c(.1),
               sigma=1)
plot(ts,
     type="o")
```




```
covariance_matrix_AR1(0.1,1,4)
```

```
##             [,1]       [,2]       [,3]        [,4]
## [1,] 1.010101010 0.10101010 0.01010101 0.001010101
## [2,] 0.101010101 1.01010101 0.10101010 0.010101010
## [3,] 0.010101010 0.10101010 1.01010101 0.101010101
## [4,] 0.001010101 0.01010101 0.10101010 1.010101010
```

```
# Compare estimates using multivariate normal distribution and exact log-likelihoods
# p is starting point
nlm(neg_log_like_MN_AR1,p=c(.1,1),x=ts)$estimate
```

```
## [1] 0.1082824 0.8496418
```

```
nlm(neg_log_like_exact_AR1,p=c(.1,1),x=ts)$estimate
```

```
## [1] 0.1082930 0.8454253
```

```
# MA Simulation function - 20% burn in period
my_MA_sim = function(n=189,ma_coeff=c(.5),sigma=1) {
  n.start = n + floor(.2/(1-.2)*n)
  burnin = floor(.20*n.start)
  model_params = list(ma = ma_coeff)
  ts_sim = arima.sim(model = model_params, n = n.start,sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
  return(ts_sim)
}

# MA(1) ACVF
```

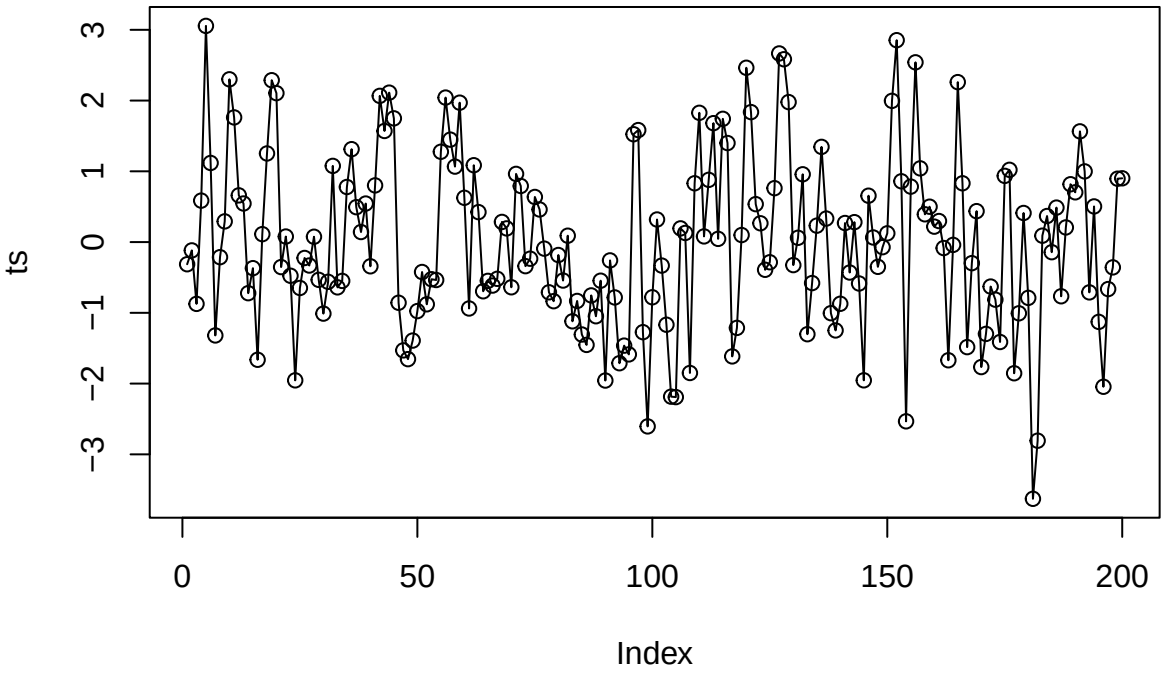
```r
MA_gamma = function(h,theta,sigma) {
  acvf = ifelse(abs(h)==0,sigma^2*(1+theta^2),
         ifelse(abs(h)==1,sigma^2*theta,0))
  return(acvf)
}

# MA(1) Covariance Matrix from ACVF
covariance_matrix_MA = function(theta,sigma,n) {
  cov_mat = matrix(0,
                   nrow = n,
                   ncol = n)

  for (i in 1:n) {
    for (j in 1:n) {

      cov_mat[i,j] <- MA_gamma(h=i-j,theta=theta,sigma=sigma)
    }
  }

  return(cov_mat)
}

# AR(1) Negative Log-Likelihood assuming Multivariate Normal Distribution
neg_log_like_MN_MA = function(params,x) {
  theta = params[1]
  sigma =  params[2]
  n = length(x)
  x = as.matrix(x)

  # covariance matrix Omega
  Omega = covariance_matrix_MA(theta=theta,sigma=sigma,n=n)

  # quadratic form in exponential as vector
  quad_form = c(t(x)%*%solve(Omega)%*%x)

  # calculate negative log-likelihood assuming MN distribution
  neg_log_like = (n/2)*log(2*pi)-(1/2)*log(det(solve(Omega)))+(1/2)*quad_form

  return(neg_log_like)
}

MA_gamma(h = 0,
         theta = .5,
         sigma = 1)
```

```
## [1] 1.25
```

```r
covariance_matrix_MA(theta = .5,
                     sigma = 1,
                     n = 4)
```

```
##      [,1] [,2] [,3] [,4]
## [1,] 1.25 0.50 0.00 0.00
## [2,] 0.50 1.25 0.50 0.00
## [3,] 0.00 0.50 1.25 0.50
## [4,] 0.00 0.00 0.50 1.25
```

```r
# Test MA(1) negative log-likelihood minimization
set.seed(1)
ts = my_MA_sim(n = 200,
               ma_coeff = c(.75),
               sigma = 1)
plot(ts,
     type="o")
```




```r
neg_log_like_MN_MA(c(0.2,1), x=ts)
```

```
## [1] 307.9527
```

```r
# p is starting point
nlm(neg_log_like_MN_MA,p=c(.2,1),x=ts)$estimate
```

```
## [1] 0.7821838 0.9868396
```

Example: Innovations Algorithm - MA(q) Simulation

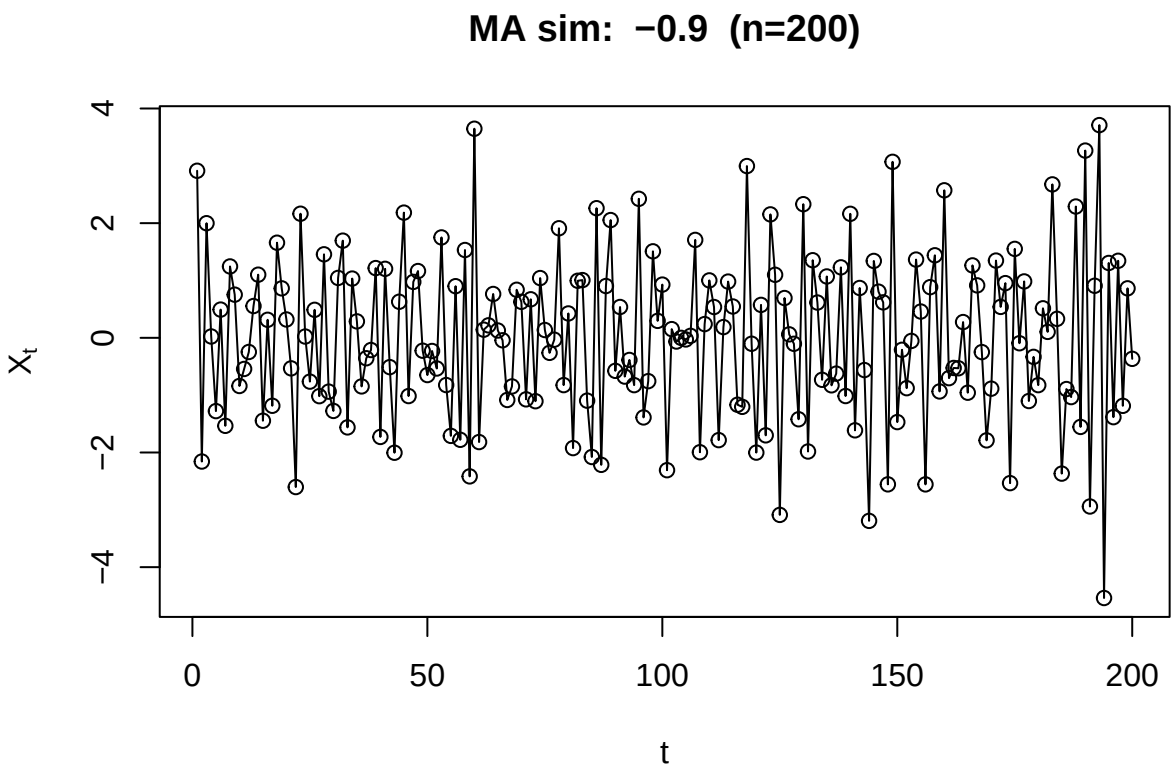
We will simulate $X_t = Z_t - 0.9Z_{t-1}$ and attempt to estimate $\theta = -0.9$ using the innovations algorithm equations.

```r
library(itsmr)

# MA Simulation function - 20% burn in period
my_MA_sim = function(n=189,ma_coeff=c(.5),sigma=1) {
  n.start = n + floor(.2/(1-.2)*n)
  burnin = floor(.20*n.start)
  model_params = list(ma = ma_coeff)
  ts_sim = arima.sim(model = model_params, n = n.start,sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
```

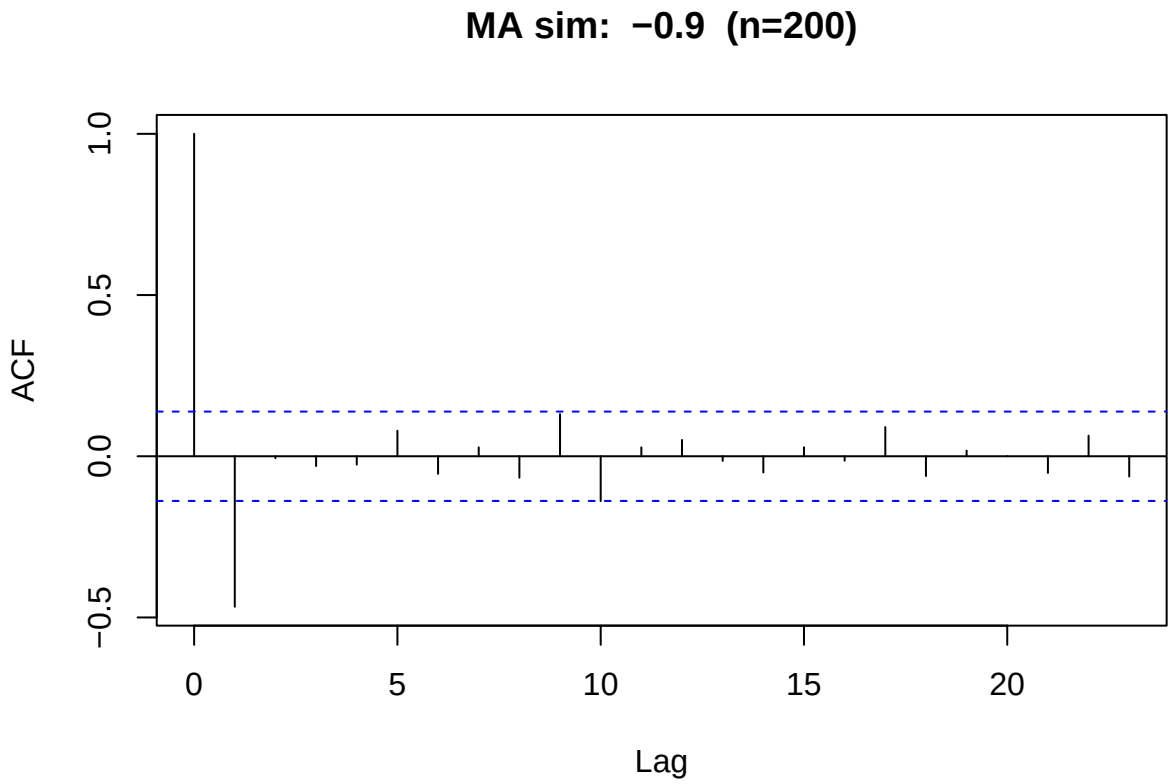
```
  return(ts_sim)
}

# Set sample size and true parameters
n = 200
ma_coeff = c(-.9)
sigma = 1

# Simulate MA(q)
ma_sim = my_MA_sim(n=n,ma_coeff=ma_coeff,sigma=sigma)
length(ma_sim)
```

```
## [1] 200
```

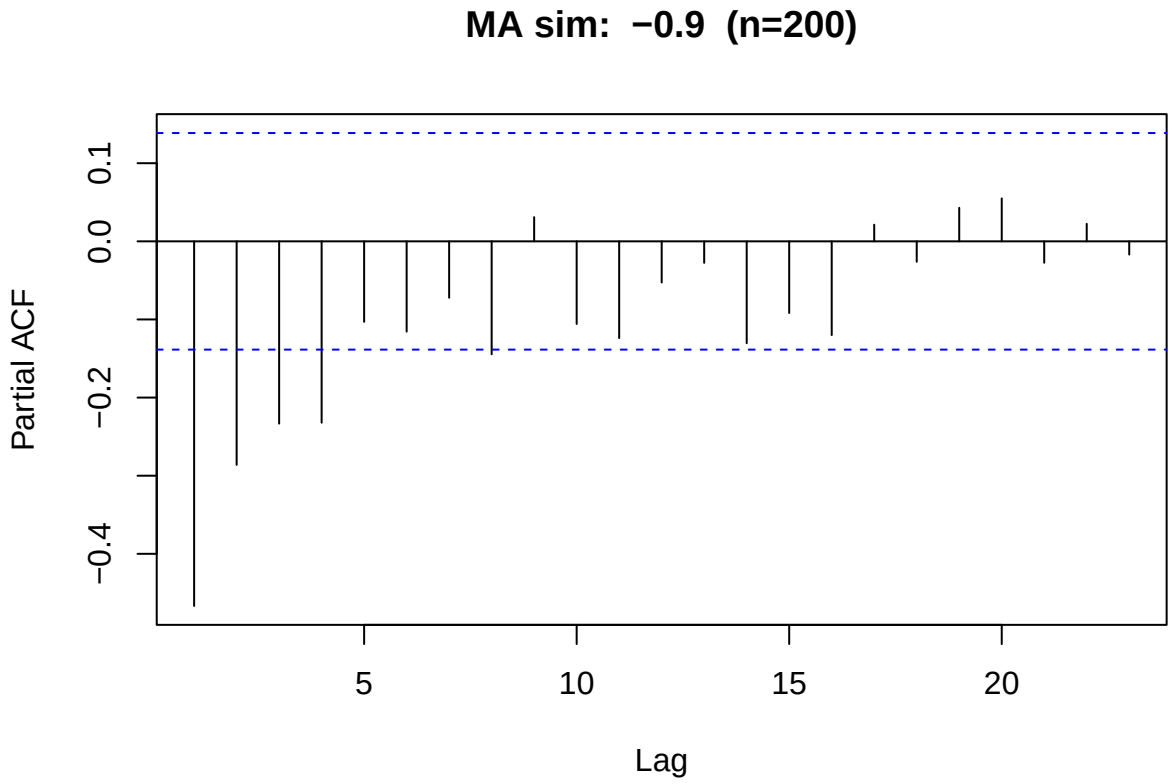
```
# Plots
main_exp = paste("MA sim: ",
                 paste(ma_coeff,collapse=", "),
                 paste(" (n=",n,")",sep=""))
plot(ma_sim,
     type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
```




```
# Sample acf plot - only lag 1 exceeds bands
acf(ma_sim, main=main_exp)
```




```
# Sample pacf plot - goes to 0, hard to read for MA(q) model
pacf(ma_sim, main=main_exp)
```



```
# Innovations MA(q) with theta and sigma^2 unknown
# Use the innovations algorithm "ia" from itsm package on the simulated dataset
# The algorithm does not know the true order - note that the first coefficients theta_n1 converge to -0.9

# Note that beyond the true model order q=1, the innovations algorithm yields
# the hat(theta) terms of the associated linear process
(innovations_q1 = ia(x=ma_sim,q=1)$theta)
```

```
## [1] -0.8131927
```

```
(innovations_q2 = ia(x=ma_sim,q=2)$theta)
```

```
## [1] -0.8131927 -0.0460879
```

```
(innovations_q3 = ia(x=ma_sim,q=3)$theta)
```

```
## [1] -0.81319273 -0.04608790 -0.03825317
```

```
(innovations_q4 = ia(x=ma_sim,q=4)$theta)
```

```
## [1] -0.813192729 -0.046087899 -0.038253174 -0.005054091
```

```
(innovations_q5 = ia(x=ma_sim,q=5)$theta)
```

```
## [1] -0.813192729 -0.046087899 -0.038253174 -0.005054091  0.089321945
```

```
innovations_q5[1]
```

```
## [1] -0.8131927
```

```
ma_coeff
```

```
## [1] -0.9
```

```
# Won't work because m=17 parameter limits recursion level
# Have to adjust m to obtain higher q

# innovations_q20 <- ia(x=ma_sim,q=20)$theta - won't work
# innovations_q20 <- ia(x=ma_sim,q=20,m=20)$theta - will work
```

```
# MA(3) Innovations Algorithm Estimation
# Note that beyond the true model order q=3, the innovations algorithm yields
# the hat(theta) terms of the associated linear process
n = 500
ma_coeff = c(-.5,.3,.1)
ma_sim = my_MA_sim(n=n,ma_coeff=ma_coeff,sigma=sigma)
(innovations_q5_new = ia(x=ma_sim,q=5)$theta)
```

```
## [1] -0.54385560  0.27570484  0.09886484 -0.06030122  0.06201564
```

```
ma_coeff
```

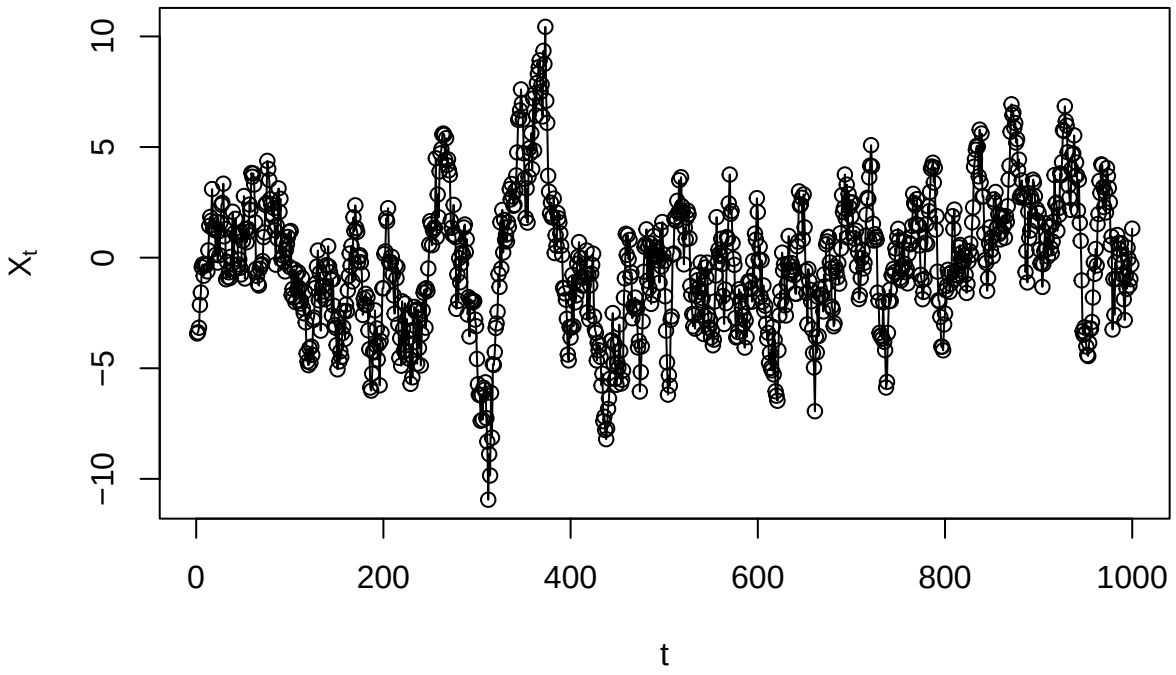
```
## [1] -0.5  0.3  0.1
```

Example: Innovations Algorithm - ARMA(p,q) Simulation

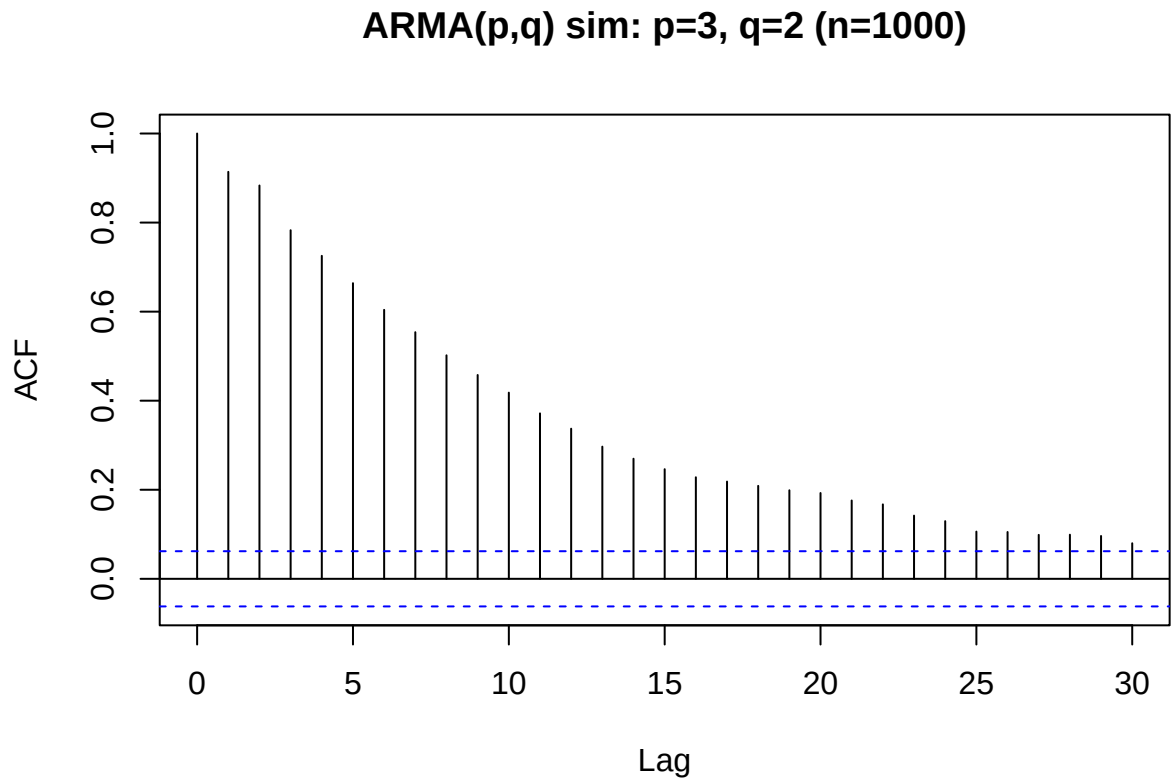
```r
library(itsmr)

# ARMA Simulation function - 20% burn in period
my_ARMA_sim = function(n=189,ma_coeff=c(.5),ar_coeff=c(.5),sigma=1) {
  n.start = n + floor(.2/(1-.2)*n)
  burnin = floor(.20*n.start)
  model_params = list(ma = ma_coeff,ar=ar_coeff)
  ts_sim = arima.sim(model = model_params, n = n.start,sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
  return(ts_sim)
}

# Set sample size and true parameters
n = 1000
ar_coeff = c(.4,.4,.1)
ma_coeff= c(.4,.5)
sigma = 1

# Simulate ARMA(p=3,q=2)
ARMA_sim = my_ARMA_sim(n=n,
                       ma_coeff = ma_coeff,
                       ar_coeff = ar_coeff,
                       sigma = sigma)
length(ARMA_sim)
```

```
## [1] 1000
```

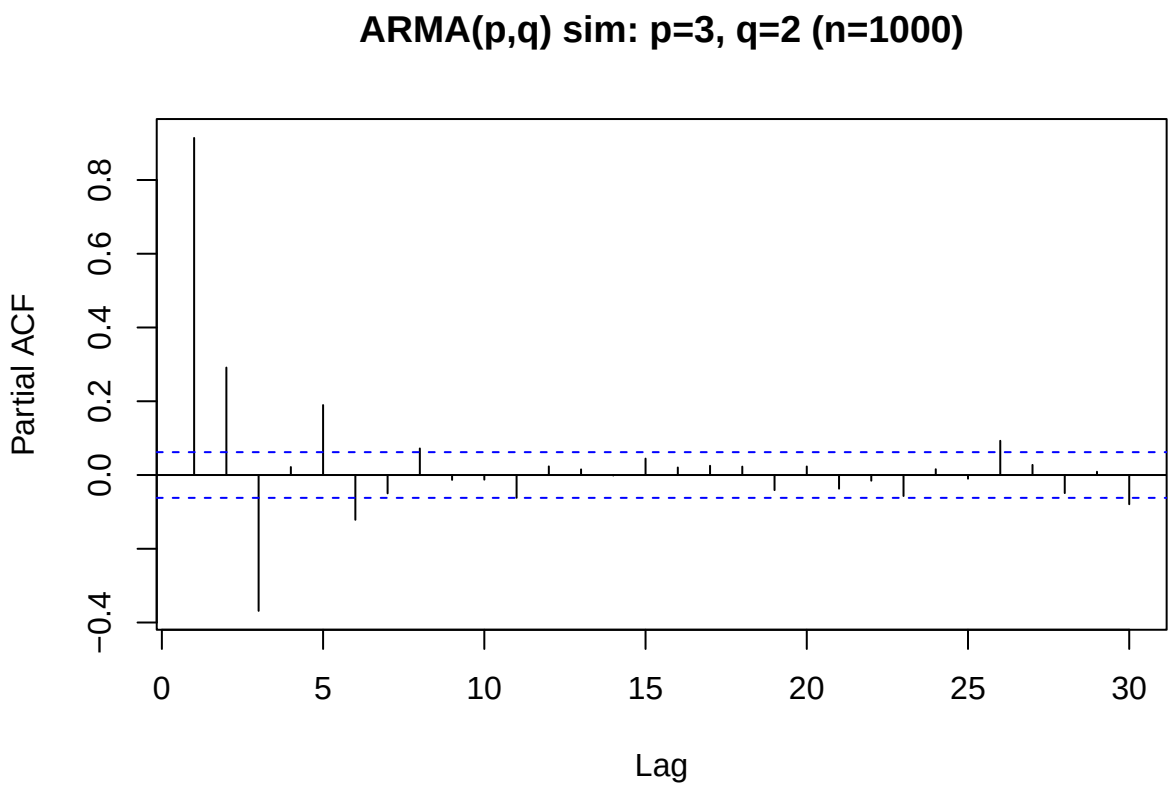
```r
# Plots
main_exp = paste("ARMA(p,q) sim:",
                 paste("p=",length(ar_coeff),", ",
                       "q=",length(ma_coeff),sep=""),
                 paste("(n=",n,")",sep=""))
plot(ARMA_sim,
     type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
```




```
# Sample acf plot - order q=2 not visible for ARMA
acf(ARMA_sim, main=main_exp)
```



```r
# Sample pacf plot - order p=3
pacf(ARMA_sim, main=main_exp)
```




```r
# Recall ARMA orders
(p = length(ar_coeff))
```

```
## [1] 3
```

```r
(q = length(ma_coeff))
```

```
## [1] 2
```

```r
# Run innovations algorithm to estimate linear process coefficients
psi_hats = ia(x=ARMA_sim,q=5)$theta

# Solve for phi_hat's
p:(p+q)
```

```
## [1] 3 4 5
```

```r
l_process_vec = psi_hats[p:(p+q)]
l_process_mat = matrix(c(psi_hats[2:4],psi_hats[1:3],1,psi_hats[1:2]),nrow=3)
phi_hat = solve(l_process_mat,l_process_vec)
phi_hat
```

```
## [1] 0.3304549 0.3935351 0.1353577
```

```r
# Solve for theta_hat's
theta_hat_1 = psi_hats[1]-phi_hat[1]
theta_hat_2 = psi_hats[2]-(phi_hat[1]*psi_hats[1]+phi_hat[2])
c(theta_hat_1,theta_hat_2)
```

```
## [1] 0.4487689 0.5520829
```

```r
# Combine final results
ARMA32_out = cbind(c(phi_hat,theta_hat_1,theta_hat_2),c(ar_coeff,ma_coeff))
rownames(ARMA32_out) = c("phi1","phi2","phi3","theta1","theta2")
colnames(ARMA32_out) = c("Estimate","Parameter")
ARMA32_out
```

```
##         Estimate Parameter
## phi1   0.3304549       0.4
## phi2   0.3935351       0.4
## phi3   0.1353577       0.1
## theta1 0.4487689       0.4
## theta2 0.5520829       0.5
```

Example: HW 3 #2 - Construct Yule Walker Function

Goal: Estimate autoregressive parameters ϕ_j and noise variance σ^2 based on the sample Yule-Walker equations (MOM).

Assume a causal AR(p) model with zero mean.

a) Build your own sample Yule-Walker estimator function using `R`.
Test your estimator on datasets `HW3_AR_Data1.csv` and `HW3_AR_Data2.csv`.
The model orders $p_1 > 0$ and $p_2 > 0$ should be consistent with problem (1c).

The sample Yule-Walker equations are given by
i. $\hat{\phi}_p = \hat{\mathbf{\Gamma}}_p^{-1}\hat{\gamma}_p$
ii. $\hat{\sigma}^2 = \hat{\gamma}(0) - \hat{\phi}_p^T\hat{\gamma}_p$

```r
# sample Yule-Walker function, specify AR order p
sampleYW = function(data, AR_order) {

  # extract sample autocorrelations hat{rho}(h) from acf function
  rho_p = c(acf(data, plot=F)$acf)
  len = length(rho_p)

  # setup correlation matrix of hat{rho}(h) values
  Corr = matrix(NA, nrow = len, ncol = len)

  # construct correlation matrix of hat{rho}(h) values
  for (i in 1:len) {
    for (j in 1:len) {

        index = abs(i-j)
        Corr[i,j] = c(rho_p)[index+1]
    }
  }

    # Yule-Walker hat{phi} vector for order i
    # rho vector should start with hat{rho}(1) not hat{rho}(0) (so start with index 2)
    # pacf functions takes ith value of Yule-Walker hat{phi} vector
    vecPhi = c(round(solve(Corr[1:AR_order,1:AR_order])%*%rho_p[2:(AR_order+1)],4))
    sigmaSq = round(rho_p[1] - t(vecPhi)%*%rho_p[2:(AR_order+1)],4)

    vecParam = c(vecPhi, sigmaSq)
```

```
    names(vecParam) = c(paste("hat(phi)_", 1:AR_order, sep=""), "hat(sigma)^2")

  return(vecParam)
}

sampleYW(ARdat1[,2], 1)
```

```
##   hat(phi)_1 hat(sigma)^2
##      -0.7225       0.4780
```

```
sampleYW(ARdat2[,2], 3)
```

```
##   hat(phi)_1   hat(phi)_2   hat(phi)_3 hat(sigma)^2
##       0.3993       0.2143       0.1645       0.5415
```

b) Run your Yule-Walker estimates on the dataset `HW3_AR_Data2.csv` using AR(p) orders `p=1,2,3,4,5`.
Display all estimated parameters in a tabular format and interpret the final result.

```
matPhi = matrix(data=NA, nrow=5, ncol=6)
rownames(matPhi) = paste("p2=", 1:5, sep="")
colnames(matPhi) = c(paste("hat(phi)_", 1:5, sep=""), "hat(sigma)^2")

for (i in 1:5) {

  vec = sampleYW(ARdat2[,2], i)
  matPhi[i,] = c(vec[1:length(vec)-1], rep(NA,5-i), vec[length(vec)])
}

matPhi
```

```
##      hat(phi)_1 hat(phi)_2 hat(phi)_3 hat(phi)_4 hat(phi)_5 hat(sigma)^2
## p2=1     0.6271         NA         NA         NA         NA       0.6068
## p2=2     0.4466     0.2877         NA         NA         NA       0.5566
## p2=3     0.3993     0.2143     0.1645         NA         NA       0.5415
## p2=4     0.4140     0.2334     0.2001    -0.0892         NA       0.5372
## p2=5     0.4140     0.2332     0.1999    -0.0896      0.001       0.5372
```

As the model order increases to the determined AR(p) model order $p_2 = 3$, the values of $\hat{\phi}_1$, $\hat{\phi}_2$, and $\hat{\phi}_3$ converge to their true values.
The estimates $\hat{\phi}_4$ and $\hat{\phi}_5$ are approximately 0, because they exceed the determined model order $p_2 = 3$ and thus have no meaningful contribution to the model.

$\hat{\sigma}^2$ continues to decrease as the model order increases, but note that beyond the determined model order $p_2 = 3$, the reduction in variance slows dramatically.

c) Construct a 95% confidence interval for ϕ_1 based on the dataset `HW3_AR_Data1.csv`.
Using the 95% confidence interval, test the null/alternative pair:

$$H_0 : \phi_1 = -1$$
$$H_1 : \phi_1 \neq -1$$

Interpret your testing procedure and its results in a few sentences.

The 95% confidence interval for ϕ_1 is given by

$$\hat{\phi}_1 \pm z_{\alpha/2} \frac{1}{\sqrt{n}} \sqrt{\hat{v}_{11}}$$

Where $\hat{\nu}_{11}$ is the 1st diagonal element of $\hat{\nu}_p \hat{\mathbf{\Gamma}}_p^{-1}$.

```r
# AR(p) order of ARdat1 is p1=1
AR_order = 1

# extract sample autocovariances hat{gamma}(h) from acf function
gamma_p = c(acf(ARdat1[,2], plot=F, type="covariance")$acf)

# extract hat{gamma}(0)
gamma_0_hat = gamma_p[1]

# extract sample autocorrelations hat{rho}(h) from acf function
rho = c(acf(ARdat1[,2],plot=F)$acf)
len = length(rho)

# limit hat{rho} vector to dimension p1 starting with hat{rho}(1)
vec_rho_p_hat = rho[2:(AR_order+1)]

# setup correlation matrix of hat{rho}(h) values
Corr = matrix(NA, nrow = len, ncol = len)

# construct correlation matrix of hat{rho}(h) values
for (i in 1:len) {
  for (j in 1:len) {

      index = abs(i-j)
      Corr[i,j] = c(rho)[index+1]
  }
}

# Limit correlation matrix to dimension p1=1
Corr_p_hat = Corr[1:AR_order,1:AR_order]

# find hat{nu}_p
nu_p_hat = gamma_0_hat*(1-t(vec_rho_p_hat)%*%solve(Corr_p_hat)%*%vec_rho_p_hat)

# extract 1st diagonal element hat{nu}_11
nu_11_hat = nu_p_hat%*%solve(Corr_p_hat)[1,1]

# use sample YW function to find hat{phi}_1
phi_1_hat = sampleYW(ARdat1[,2], 1)[1]

# extract dimension of sample data
n = dim(ARdat1)[1]

# By symmetry of the normal distribution use 1-alpha/2 = 0.975 quantile of standard normal
quant = qnorm(0.975)

# construct confidence interval
phi_1_hat_interval = round(c(phi_1_hat - quant*(1/sqrt(n))*sqrt(nu_11_hat),
                             phi_1_hat + quant*(1/sqrt(n))*sqrt(nu_11_hat)),4)
names(phi_1_hat_interval) = c("2.5%", "97.5%")
phi_1_hat_interval
```

```
##    2.5%   97.5%
## -0.8557 -0.5893
```

Thus the 95% confidence interval for ϕ_1 is $(-0.8857, -0.5893)$.

The 95% confidence interval for ϕ_1 does not contain the null value of the parameter $\phi_{1_0} = -1$.
Therefore, for a 5% significance level (95% confidence level), we reject the null hypothesis $H_0 : \phi_1 = -1$, that the AR(1) process is a random walk.

Note that this is analogous to a test for stationarity, but is not what is used in practice.
Nonetheless, the testing procedure suggests that the process is stationary.

Example: HW 3 #3 - Asymptotics of Sample Yule-Walker Estimator

Consider the sample $X_1, X_2, \ldots, X_n$ where each $\{X_t\}$ is generated by the AR(1) process

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t \overset{iid}{\sim} N(0, \sigma^2)$$

For which parameter σ^2 is known.

Assuming that the realized time series is centered ($x_t := x_t - \bar{x}_t$), define the sample autocovariance function as

$$\hat{\gamma}(h) = \frac{1}{n} \sum_{i=1}^{n-|h|} x_{t+|h|} x_t, \quad -n < h < n$$

a) Show that the sample Yule-Walker estimator of a causal AR(1) is

$$\hat{\phi} = \frac{\sum_{i=1}^{n-1} X_{i+1} X_i}{\sum_{i=1}^{n} X_i^2}$$

The sample Yule-Walker equations for a causal AR(1) process, with known parameter σ^2 and model order $p = 1$ are given by

$$\hat{\phi}_1 = \hat{\mathbf{\Gamma}}_1^{-1} \hat{\gamma}_1$$

$$\hat{\mathbf{\Gamma}}_1 = \left[\ \hat{\gamma}(0) \ \right] = \frac{1}{n} \sum_{i=1}^{n} X_i^2$$

$$\hat{\mathbf{\Gamma}}_1^{-1} = \left(\frac{1}{n} \sum_{i=1}^{n} X_i^2 \right)^{-1} = \frac{1}{\frac{1}{n} \sum_{i=1}^{n} X_i^2}$$

$$\hat{\gamma}_1 = \left[\ \hat{\gamma}(1) \ \right] = \frac{1}{n} \sum_{i=1}^{n-1} X_{i+1} X_i$$

Therefore,

$$\hat{\phi}_1 = \hat{\mathbf{\Gamma}}_1^{-1} \hat{\gamma}_1 = \frac{\frac{1}{n} \sum_{i=1}^{n-1} X_{i+1} X_i}{\frac{1}{n} \sum_{i=1}^{n} X_i^2} = \frac{\sum_{i=1}^{n-1} X_{i+1} X_i}{\sum_{i=1}^{n} X_i^2}$$

b) Now consider the sample $X_0, X_1, \ldots, X_n$ where each $\{X_t\}$ is generated by the AR(1) process and we allow for the non-stationary solution $\phi = 1$.
Also assume that σ^2 is unknown.

One method of testing the null/alternative pair:

$$H_0 : \phi = 1$$
$$H_1 : \phi \neq 1$$

Is based on the test statistic

$$T_n = \frac{\hat{\phi}_n - 1}{\sqrt{\frac{S_n^2}{\sum_{i=1}^{n} X_{i-1}^2}}}$$

Where

$$\hat{\phi}_n = \frac{\sum_{i=1}^{n} X_{i-1} X_i}{X_{i-1}^2}$$

$$S_n^2 = \frac{1}{n-1} \sum_{i=1}^{n} (X_i - \hat{\phi}_n X_{i-1})^2$$

Rejecting the null hypothesis indicates that the series exhibits stationary behavior against the random walk null.
It can be shown, under $H_0 : \phi = 1$, that the limiting distribution of T_n is

$$T_n \xrightarrow{d} \frac{[W(1)]^2 - 1}{2\sqrt{\int_0^1 [W(r)]^2 \, dr}}$$

Where $W(r)$ is standard Brownian motion.

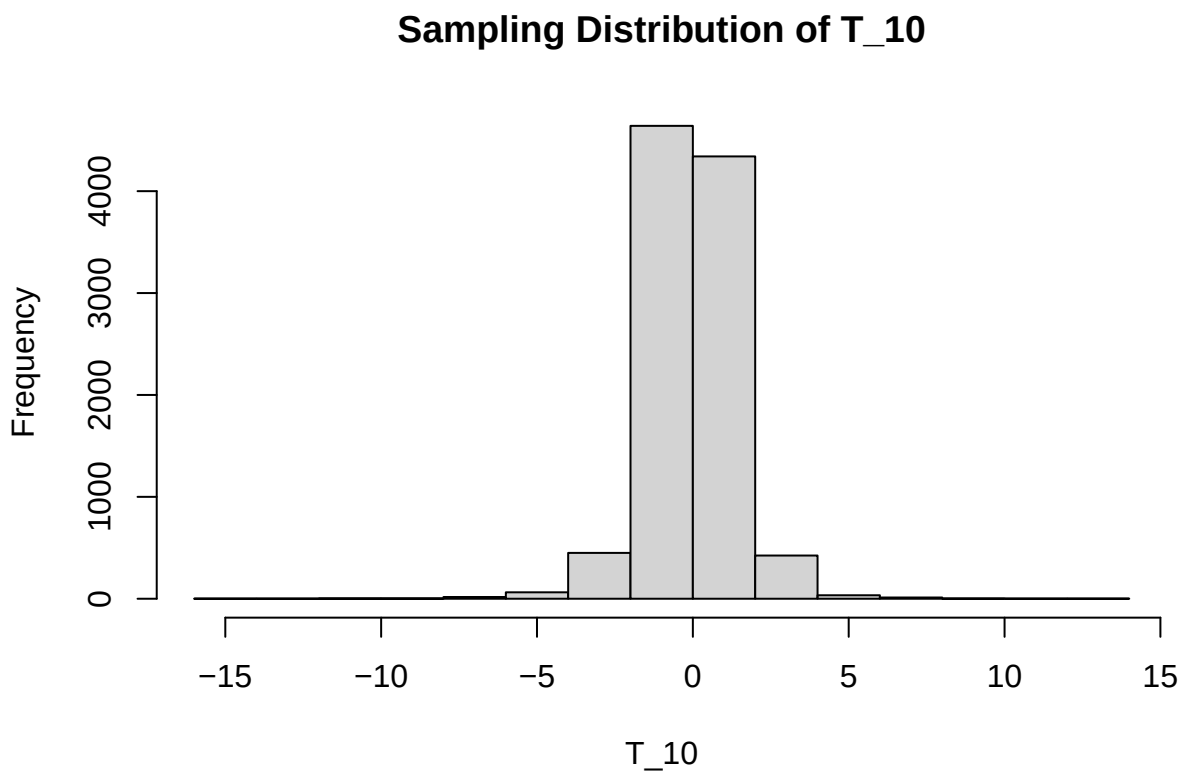
Assuming that the data generating process is a random walk, simulate the limiting distribution of the test statistic T_n using $n = 10, 30, 100, 1000$.
In particular, you will simulate the limiting distribution by generating $R = 10000$ random walk processes and then computing the resulting estimators $\hat{\phi}_n^{(1)}, \ldots, \hat{\phi}_n^{10000}$ and test statistics $T_n^{(1)}, \cdots T_n^{(10000)}$.
Then construct histograms of the T_n's, one for each sample size $n = 10, 30, 100, 1000$.

In other words, to examine the asymptotics of the Yule-Walker estimator, we will produce empirical large-sample sampling distributions of the test statistic T_n (using $R = 10,000$ random walk processes), because it is difficult to simulate the limiting distribution (consisting of a function of Brownian bridges and Wiener processes) directly.

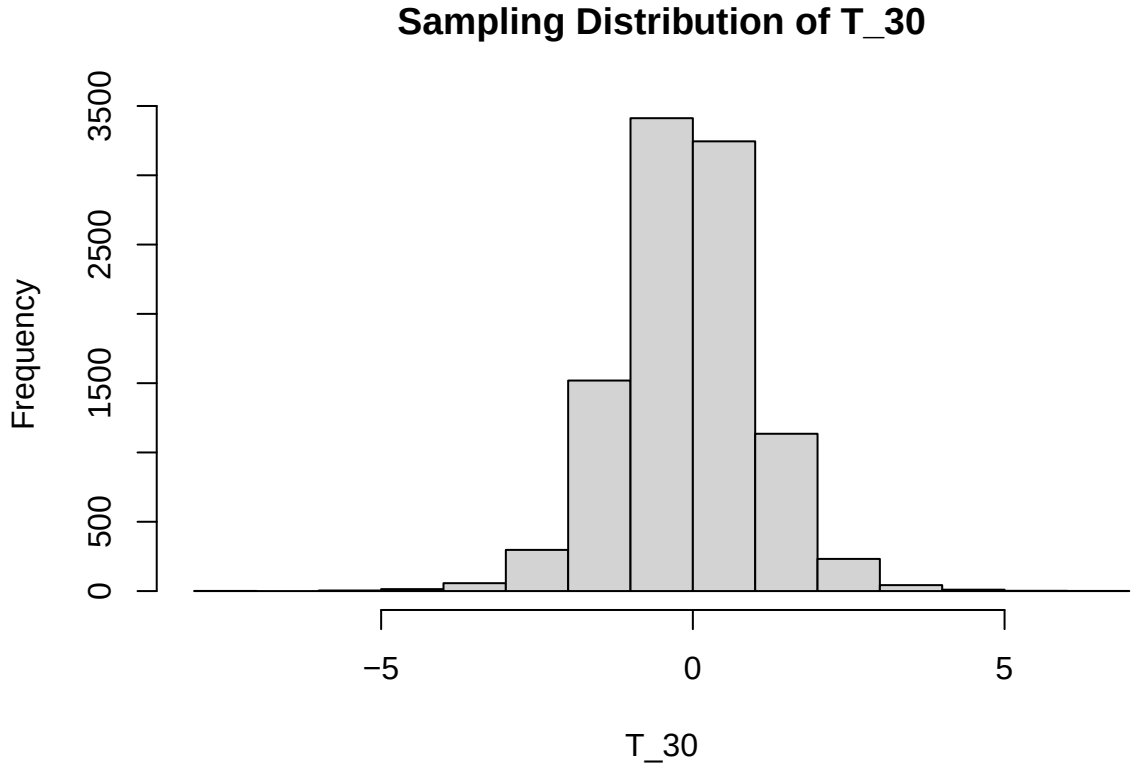
```
# Under the null, we assume that the data generating process is a random walk

R = 10000
vec_phi_hat = rep(NA, R)
vec_t_n = rep(NA, R)

TnSampDistSim = function(n) {

  for (i in 1:R) {

    # Simulate a random walk
    x0 = 0
    rw = x0 + cumsum(rnorm(R))

    # 20% burn in period, plus random start point
    ind = sample(2000:R, 1)

    # size n sample from random walk starting at index ind, x_{i-1} and x_1 needed
    x_i_1 = rw[(ind):(ind + n)]
    x_i = rw[(ind + 1):(ind + (n+1))]

    phi_hat = sum(x_i_1*x_i)/sum(x_i_1*x_i_1)
    vec_phi_hat[i] = phi_hat

    s_n = sum((x_i - (phi_hat*x_i_1))^2)/(n-1)

    t_n_den = sqrt((s_n^2)/sum(x_i_1*x_i_1))
    t_n = (phi_hat-1)/t_n_den

    vec_t_n[i] = t_n
  }

  return(vec_t_n)
}
```

```
sampDist_10 = TnSampDistSim(10)
sampDist_30 = TnSampDistSim(30)
sampDist_100 = TnSampDistSim(100)
sampDist_1000 = TnSampDistSim(1000)
```

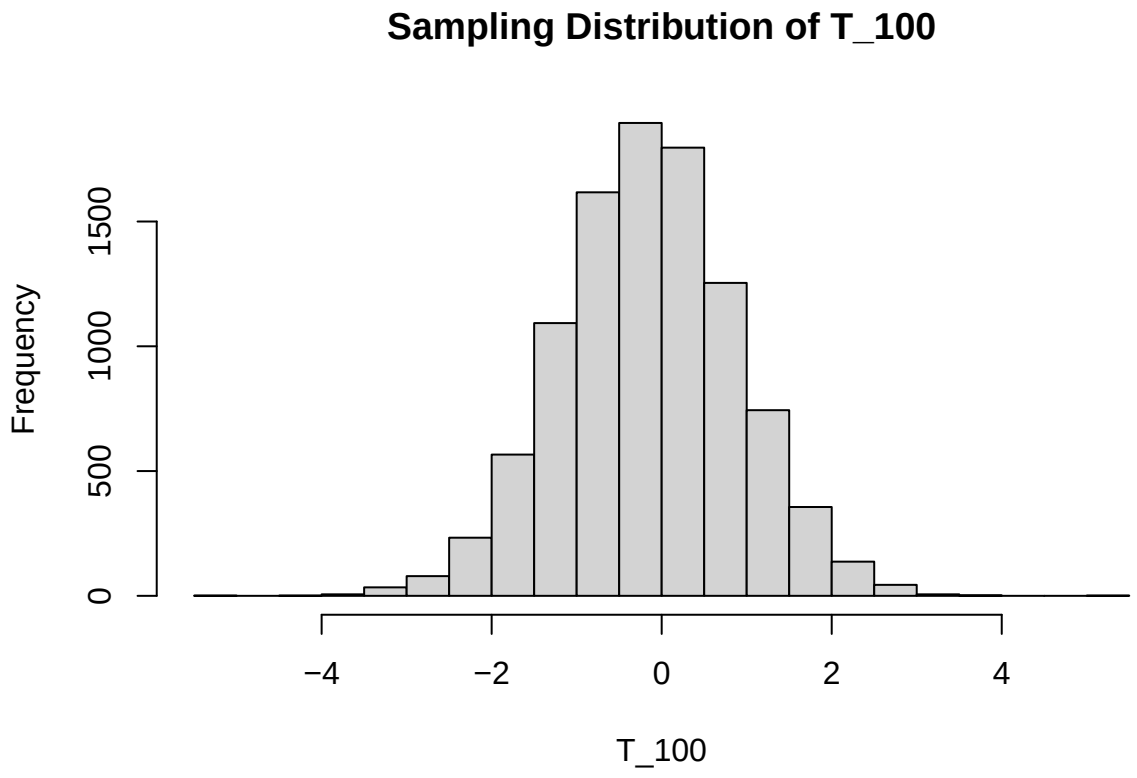
```
hist(sampDist_10, xlab = "T_10", main = "Sampling Distribution of T_10")
```



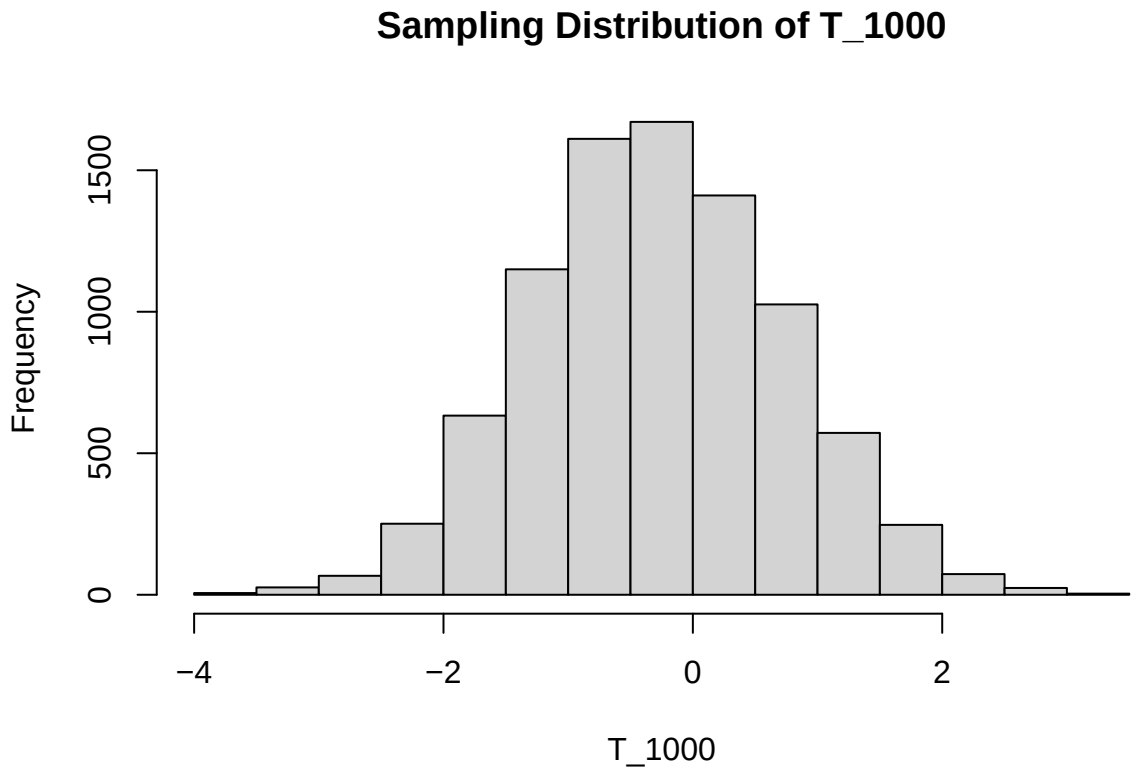

```
hist(sampDist_30, xlab = "T_30", main = "Sampling Distribution of T_30")
```



```
hist(sampDist_100, xlab = "T_100", main = "Sampling Distribution of T_100")
```




```
hist(sampDist_1000, xlab = "T_1000", main = "Sampling Distribution of T_1000")
```



c) If $H_0 : \phi = 1$ is true, argue the below limit.

$$T_n \xrightarrow{d} \frac{[W(1)]^2 - 1}{2\sqrt{\int_0^1 [W(r)]^2 \, dr}}$$

Where $W(r)$ is standard Brownian motion.

You can directly use the following:
Under the null,

$$\frac{1}{n}\sum_{i=1}^{n} X_{i-1}Z_i \xrightarrow{d} \frac{1}{2}\sigma^2([W(1)]^2 - 1)$$
$$\frac{1}{n^2}\sum_{i=1}^{n} X_{i-1}^2 \xrightarrow{d} \sigma^2 \int_0^1 [W(r)]^2 \, dr$$

Suppose $\{X_t\}$ is a random walk under the null $H_0 : \phi = 1$.

$$X_t = X_{t-1} + Z_t$$

We have

$$\hat{\phi} = \hat{\phi}_{MLE,YW} = \frac{\sum_{i=0}^{n-1} X_i X_{i+1}}{\sum_{i=0}^{n-1} X_i^2}$$

Note that $X_t = X_{t-1} + Z_t$ implies

$$X_i X_{i+1} - X_i^2 = X_i(X_{i+1} - X_i) = X_i Z_{i+1}$$

Therefore,

$$\hat{\phi} - 1 = \frac{\sum_{i=0}^{n-1} X_i X_{i+1} - \sum_{i=0}^{n-1} X_i^2}{\sum_{i=0}^{n-1} X_i^2} = \frac{\sum_{i=0}^{n-1} X_i Z_{i+1}}{\sum_{i=0}^{n-1} X_i^2}$$

Then,

$$n(\hat{\phi} - 1) = \frac{\frac{1}{n}\sum_{i=0}^{n-1} X_i Z_{i+1}}{\frac{1}{n^2}\sum_{i=0}^{n-1} X_i^2}$$

It is given that under the null hypothesis,

$$\frac{1}{n}\sum_{i=1}^{n} X_{i-1}Z_i \xrightarrow{d} \frac{1}{2}\sigma^2([W(1)]^2 - 1)$$
$$\frac{1}{n^2}\sum_{i=1}^{n} X_{i-1}^2 \xrightarrow{d} \sigma^2 \int_0^1 [W(r)]^2 \, dr$$

Therefore,

$$n(\hat{\phi} - 1) = \frac{\frac{1}{n}\sum_{i=0}^{n-1} X_i Z_{i+1}}{\frac{1}{n^2}\sum_{i=0}^{n-1} X_i^2} \xrightarrow{d} \frac{\frac{1}{2}\sigma^2((W(1))^2 - 1)}{\sigma^2 \int_0^1 [W(r)]^2 \, dr} = \frac{((W(1))^2 - 1)}{2\int_0^1 [W(r)]^2 \, dr}$$

Consider the test statistic T_n.

$$T_n = \frac{\hat{\phi}_n - 1}{\sqrt{\frac{S_n^2}{\sum_{i=1}^{n} X_{i-1}^2}}}$$

Where

$$\hat{\phi}_n = \frac{\sum_{i=1}^{n} X_{i-1}X_i}{X_{i-1}^2}$$

$$S_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \hat{\phi}_n X_{i-1})^2$$

Introducing an $(1/n)^2/(1/n)^2$ term in the denominator, we have

$$= \frac{\hat{\phi}_n - 1}{\sqrt{\frac{S_n^2}{\sum_{i=1}^n X_{i-1}^2}\left(\frac{(1/n)^2}{(1/n)^2}\right)}} = \frac{\hat{\phi}_n - 1}{\frac{1}{n}\sqrt{\frac{S_n^2}{\frac{1}{n^2}\sum_{i=1}^n X_{i-1}^2}}} = \frac{n(\hat{\phi}_n - 1)}{\sqrt{\frac{S_n^2}{\frac{1}{n^2}\sum_{i=1}^n X_{i-1}^2}}}$$

By the Law of Large Numbers,

$$S_n^2 = \frac{1}{n-1}\sum_{i=1}^{n}(X_i - \hat{\phi}_n X_{i-1})^2 \xrightarrow{P} \sigma^2$$

It is given that under the null hypothesis,

$$\frac{1}{n^2}\sum_{i=1}^{n} X_{i-1}^2 \xrightarrow{d} \sigma^2 \int_0^1 [W(r)]^2 \, dr$$

By the continuous mapping theorem, if $Y_n \xrightarrow{d} Y$, and $g(\cdot)$ is a continuous function, then $g(Y_n) \xrightarrow{d} g(Y)$.
For the continuous function $g(x) = \frac{1}{x}$, by the continuous mapping theorem, we have

$$\frac{1}{\frac{1}{n^2}\sum_{i=1}^n X_{i-1}^2} \xrightarrow{d} \frac{1}{\sigma^2 \int_0^1 [W(r)]^2 \, dr}$$

By Slutsky's theorem, if $X_n \xrightarrow{d} X$, for a random variable X, and $Y_n \xrightarrow{P} c$, for some constant c, then
i. $X_n + Y_n \xrightarrow{d} X + c$
ii. $X_n Y_n \xrightarrow{d} Xc$
iii. $\frac{X_n}{Y_n} \xrightarrow{d} \frac{X}{c}$, provided that c is invertible
Therefore, by Slutsky's theorem,

$$\frac{S_n^2}{\frac{1}{n^2}\sum_{i=1}^n X_{i-1}^2} \xrightarrow{d} \frac{\sigma^2}{\sigma^2 \int_0^1 [W(r)]^2 \, dr} = \frac{1}{\int_0^1 [W(r)]^2 \, dr}$$

For the continuous function $g(x) = \frac{1}{\sqrt{x}}$ (which is continuous because it is a composition of continuous functions $g_1(x) = \frac{1}{x}$, $g_2(x) = \sqrt{x}$), by the continuous mapping theorem, we have

$$\frac{1}{\sqrt{\frac{S_n^2}{\frac{1}{n^2}\sum_{i=1}^n X_{i-1}^2}}} \xrightarrow{d} \frac{1}{\sqrt{\frac{1}{\int_0^1 [W(r)]^2 \, dr}}}$$

Thus, combining the above results, we can conclude

$$T_n = \frac{\hat{\phi}_n - 1}{\sqrt{\frac{S_n^2}{\sum_{i=1}^n X_{i-1}^2}}} = \frac{n(\hat{\phi}_n - 1)}{\sqrt{\frac{S_n^2}{\frac{1}{n^2}\sum_{i=1}^n X_{i-1}^2}}} \xrightarrow{d} \frac{\frac{((W(1))^2 - 1)}{2\int_0^1 [W(r)]^2 \, dr}}{\sqrt{\frac{1}{\int_0^1 [W(r)]^2 \, dr}}} = \frac{[W(1)]^2 - 1}{2\sqrt{\frac{1}{\int_0^1 [W(r)]^2 \, dr}}\sqrt{\left(\int_0^1 [W(r)]^2 \, dr\right)^2}} = \frac{[W(1)]^2 - 1}{2\sqrt{\int_0^1 [W(r)]^2 \, dr}}$$

Example: HW 3 #4 - Innovations Algorithm

Consider the time series process $\{X_t\}$ with mean $E[X_i] = 0$ and covariance function $\gamma(i,j) = E[X_i X_j]$.
Define the nth innovation as $U_n = X_n - \hat{X}_n$, where

$$\hat{X}_n = \begin{cases} 0 & for \;\; n = 1 \\ P(X_n | X_{n-1}, \ldots, X_1) & for \;\; n = 2, 3, \ldots \end{cases}$$

a) In the innovations algorithm,
i) Show that for each $n \geq 2$, the innovation $U_n = X_n - \hat{X}_n$ is uncorrelated with $X_1, \ldots, X_{n-1}$.
ii) Conclude that $U_n = X_n - \hat{X}_n$ is uncorrelated with the innovations $U_1 = X_1 - \hat{X}_1, \ldots, U_{n-1} = X_{n-1} - \hat{X}_{n-1}$.

From the properties of the BLP, we know that the innovations ("residuals"), $U_{n+h} = X_{n+h} - \hat{X}_{n+h}$, are orthogonal to the predictors X_j, $j = 1, 2, \ldots, n$.
In particular, $E[U_{n+h}X_j] = E[(X_{n+h} - \hat{X}_{n+h})X_j] = 0, \quad j = 1, 2, \ldots, n$.
Therefore, for each $n \geq 2$, the innovation $U_n = X_n - \hat{X}_n$ is uncorrelated with $X_1, \ldots, X_{n-1}$.

We would like to show that the innovation $U_n = X_n - \hat{X}_n$ is uncorrelated with the innovations $U_1 = X_1 - \hat{X}_1, \ldots, U_{n-1} = X_{n-1} - \hat{X}_{n-1}$,

$$E[U_n U_k] = 0, \quad k = 1, \ldots, n-1$$

For $k = 1, \ldots, n-1$

$$E[U_n X_k] = E[(X_n - \hat{X}_n)(X_k - \hat{X}_k]$$
$$= E[(X_n - \hat{X}_n)X_k] - E[(X_n - \hat{X}_n)\hat{X}_k]$$

Noting that for $n \geq 2$, $\hat{X}_k = P_{k-1}X_k = a_1 X_{k-1} + \cdots + a_k X_1$, which is a linear combination of X's, we have

$$= E[(X_n - \hat{X}_n)X_k] - (E[(X_n - \hat{X}_n)a_1 X_{k-1}] + E[(X_n - \hat{X}_n)a_2 X_{k-2}] + \cdots + E[(X_n - \hat{X}_n)a_n X_1])$$
$$= E[(X_n - \hat{X}_n)X_k] - (a_1 E[(X_n - \hat{X}_n)X_{k-1}] + a_2 E[(X_n - \hat{X}_n)X_{k-2}] + \cdots + a_n E[(X_n - \hat{X}_n)X_1])$$

For $k = 1, \ldots, n-1$, we note that $E[U_n X_k] = 0$ by the property of the BLP above.
Therefore, the above expression reduces to 0.

$$= 0$$

This implies that the innovation $U_n = X_n - \hat{X}_n$ is uncorrelated with the innovations $U_1 = X_1 - \hat{X}_1, \ldots, U_{n-1} = X_{n-1} - \hat{X}_{n-1}$.

b) Derive the update step for θ in the innovations algorithm.
That is, derive the expression

$$\theta_{n,n-k} = v_k^{-1}\left(\gamma(n+1, k+1) - \sum_{j=0}^{k-1} \theta_{k,k-j}\theta_{n,n-j}v_j\right), \quad 0 \leq k < n$$

You may assume that the update step for v is

$$v_n = \gamma(n+1, n+1) - \sum_{j=0}^{n-1} \theta_{n,n-j}^2 v_j$$

The goal is to derive the update step above.

We start with the expression,

$$X_{n+1} = \sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})$$

Multiplying each side by the innovation $U_{k+1} = (X_{k+1} - \hat{X}_{k+1})$, we have

$$U_{k+1}X_{n+1} = U_{k+1}\sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})$$

$$(X_{k+1} - \hat{X}_{k+1})X_{n+1} = (X_{k+1} - \hat{X}_{k+1})\sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})$$

Taking the expectation of both sides, we have

$$E[(X_{k+1} - \hat{X}_{k+1})X_{n+1}] = E\left[\sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})(X_{k+1} - \hat{X}_{k+1})\right]$$

$$E[X_{k+1}X_{n+1}] - E\left[\hat{X}_{k+1}X_{n+1}\right] = \sum_{j=0}^{n} \theta_{nj}E[(X_{n+1-j} - \hat{X}_{n+1-j})(X_{k+1} - \hat{X}_{k+1})]$$

We first expand the RHS.

$$\sum_{j=0}^{n} \theta_{nj} E[(X_{n+1-j} - \hat{X}_{n+1-j})(X_{k+1} - \hat{X}_{k+1})]$$

Note the $j = 0, \ldots, k+1 \equiv n-k, \ldots, n$.
Noting that innovations U_j, $j = 1, \ldots, n-1$ are uncorrelated with U_n, the only terms that remain when performing the double sum occur when the indices are the same, which occurs only for the index $k+1 \equiv n-k$.
Recall that $v_n = E[(X_{n+1} - \hat{X}_{n+1})^2]$.
Therefore, the RHS simplifies to

$$= \theta_{n,n-k} E\left[(X_{k+1} - \hat{X}_{k+1})^2\right] = \theta_{n,n-k} v_k$$

We now expand the LHS.

Recall that

$$\hat{X}_{n+1} = \begin{cases} 0 & for\ n = 0 \\ \sum_{j=1}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j}) & for\ n = 1, 2, \ldots \end{cases}$$

$$X_{n+1} = \sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})$$

Therefore,

$$E\left[\hat{X}_{k+1} X_{n+1}\right] = E\left[\sum_{i=1}^{k} \theta_{ki}(X_{k+1-i} - \hat{X}_{k+1-i}) \sum_{j=0}^{n} \theta_{nj}(X_{n+1-j} - \hat{X}_{n+1-j})\right]$$

Pulling out the $j = 0$ term from the second sum, and noting that $\theta_{n0} = 1$, we obtain

$$= \sum_{i=1}^{k} \theta_{ki} \sum_{j=1}^{n} \theta_{nj} E\left[(X_{k+1-i} - \hat{X}_{k+1-i})(X_{n+1-j} - \hat{X}_{n+1-j})\right]$$

We note that the terms in the expectation are in fact innovations, $U_n = (X_n - \hat{X}_n)$.

$$= \sum_{i=1}^{k} \theta_{ki} \sum_{j=1}^{n} \theta_{nj} E\left[U_{k+1-i} U_{n+1-j}\right]$$

Noting that innovations U_j, $j = 1, \ldots, n-1$ are uncorrelated with U_n, the only terms that remain when performing the double sum occur when the indices are the same.
For $0 \leq k < n$ indicates innovation indices 1 to k $(U_1, \ldots, U_k)$.
In terms of the index i, the innovations $U_k, \ldots, U_1$ correspond to the sequence $i = 1, \ldots, k$.
In terms of the index j, the innovations $U_k, \ldots, U_1$ correspond to the sequence $j = n-k+1, \ldots, n$.
Recall that $v_n = E[(X_{n+1} - \hat{X}_{n+1})^2] = E[U_{n+1}^2]$.
The innovations $U_1, \ldots, U_k$ correspond to $v_0, \ldots, v_{k-1}$.
Re-indexing the sums to a common sum with indexes j corresponding to $v_0, \ldots, v_{k-1}$, ie $j = 0, \ldots, k-1$, we obtain

$$= \sum_{j=0}^{k-1} \theta_{k,k-j} \theta_{n,n-j} v_j$$

Furthermore, note that $E[X_{k+1} X_{n+1}] = \gamma(n+1, k+1)$.
Therefore, the LHS simplifies to

$$\gamma(n+1, k+1) - \sum_{j=0}^{k-1} \theta_{k,k-j} \theta_{n,n-j} v_j$$

Setting up the full equation, we have

$$\theta_{n,n-k} v_k = \gamma(n+1, k+1) - \sum_{j=0}^{k-1} \theta_{k,k-j} \theta_{n,n-j} v_j$$

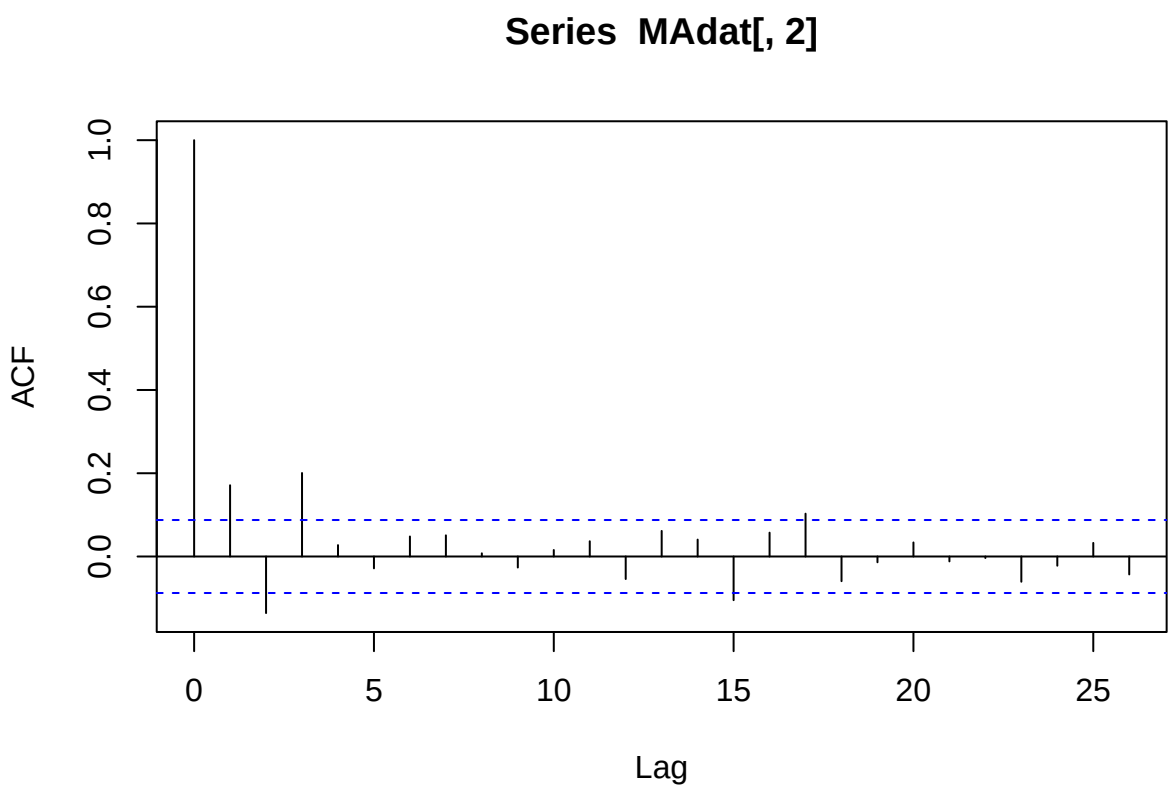
We thus conclude,

$$\theta_{n,n-k} = v_k^{-1}\left(\gamma(n+1,k+1) - \sum_{j=0}^{k-1}\theta_{k,k-j}\theta_{n,n-j}v_j\right)$$

c) Consider the observed data `HW3_MA_Data.csv`.
For this exercise, use the `R` function `ia()` from the `itsmr` library to estimate the pure MA(q) model.
You will also have to choose the MA order $q > 0$ based on tools discussed in class.
Report how you chose the model order and your estimated θ's, i.e., reported $\hat{\theta}$'s.

```
# read in data
MAdat = read.csv("HW3_MA_Data.csv")
```

```
acf(MAdat[,2])
```




```
library(itsmr)

innovations_q3 = ia(x=MAdat[,2],q=3)$theta
names(innovations_q3) = paste("hat(theta)_", 1:3, sep = "")
innovations_q3
```

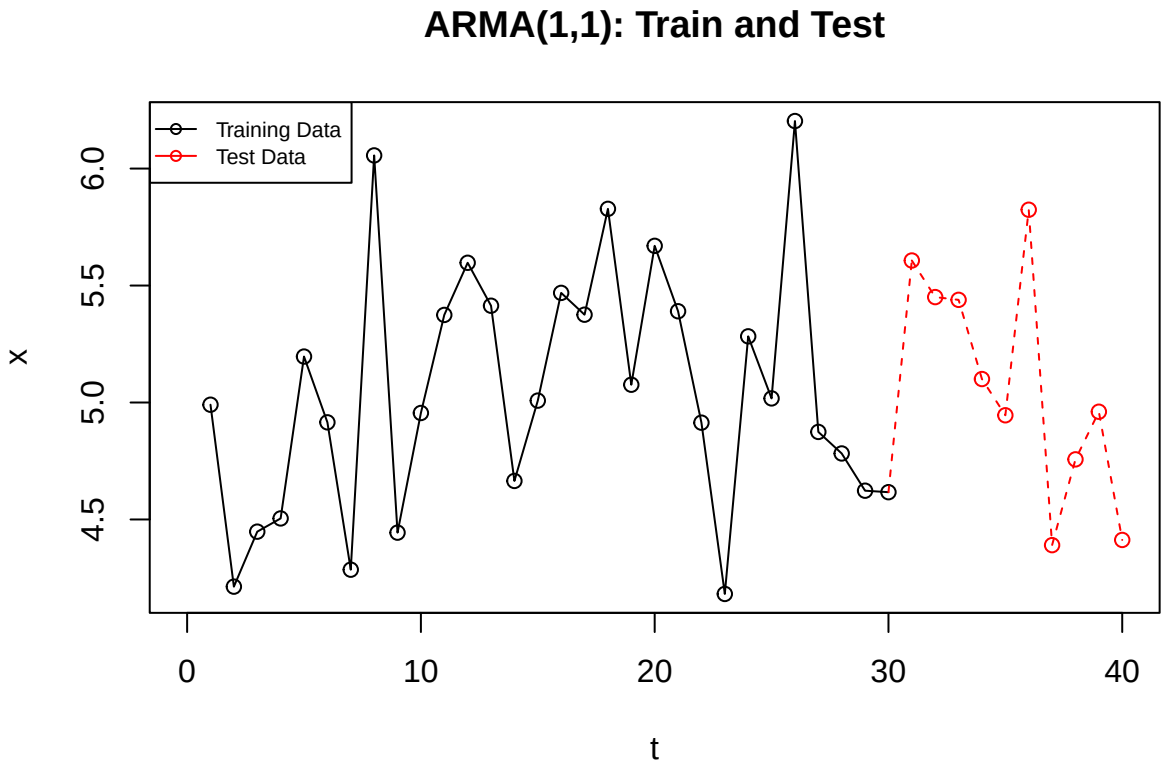
```
## hat(theta)_1 hat(theta)_2 hat(theta)_3
##    0.3093623   -0.1948741    0.1961668
```

The ACF of a pure MA(q) model will exceed the confidence bands for lag h values up to the model order q.
For the dataset `HW3_MA_Data.csv`, the MA order is $q = 3$.

The estimated $\hat{\theta}$'s are $\hat{\theta} = (0.3094, -0.1985, 0.1962)$.

Example: HW 3 #5 - ARMA MLE Estimation, ARMA Prediction

The following figure shows the training data `ARMA_11_train.csv` and testing data `ARMA_11_test.csv` used to solve problem 5.



Consider the training sample $X_1, X_2, \ldots, X_{30}$, where each $\{X_t\}$ is generated by the causal and invertible ARMA(1,1) process,

$$X_t = \mu + \phi(X_{t-1} - \mu) + Z_t + \theta Z_{t-1}, \quad Z_t \overset{iid}{\sim} N(0, \sigma^2)$$

Based on the Gaussian error structure Z_t, the ARMA(1,1) process can be described by a multivariate normal distribution,

$$f(x_1, \ldots, x_n | \mu_n, \mathbf{\Sigma}_{n\times n}) = (2\pi)^{-n/2} \det\left(\mathbf{\Sigma}_{n\times n}^{-1}\right)^{1/2} \exp\left(-\frac{1}{2}(\mathbf{x}_n - \mu_n)^T \mathbf{\Sigma}_{n\times n}^{-1}(\mathbf{x}_n - \mu_n)\right), \quad -\infty < x_1, \ldots, x_n < \infty$$

Hence the likelihood can be expressed as,

$$L(\mu_n, \mathbf{\Sigma}_{n\times n} | x_1, \ldots, x_n) = (2\pi)^{-n/2} \det\left(\mathbf{\Sigma}_{n\times n}^{-1}\right)^{1/2} \exp\left(-\frac{1}{2}(\mathbf{x}_n - \mu_n)^T \mathbf{\Sigma}_{n\times n}^{-1}(\mathbf{x}_n - \mu_n)\right), \quad -\infty < x_1, \ldots, x_n < \infty$$

a) Write down the exact likelihood for the ARMA(1,1) in terms of the parameters $\mu, \theta, \phi, \sigma$.
Note that you must express the mean vector, μ_n, and the covariance matrix $\mathbf{\Sigma}_{n\times n}$ in terms of $\mu, \phi, \theta, \sigma$.

$$\mu_n = \begin{bmatrix} \mu & \mu & \cdots & \mu & \mu \end{bmatrix}^T$$

The ACVF of the causal ARMA(1,1) process is

$$\gamma_X(h) = \begin{cases} \sigma^2\left[1 + \frac{(\phi+\theta)^2}{1-\phi^2}\right] & for\ h = 0 \\ \sigma^2\left[(\phi+\theta) + \frac{(\phi+\theta)^2\phi}{1-\phi^2}\right] & for\ |h| = 1 \\ \phi^{|h|-1}\gamma_X(1) & for\ |h| > 1 \end{cases}$$

Therefore,

$$\mathbf{\Sigma}_{n\times n} = \begin{bmatrix} \sigma^2\left[1+\frac{(\phi+\theta)^2}{1-\phi^2}\right] & \sigma^2\left[(\phi+\theta)+\frac{(\phi+\theta)^2\phi}{1-\phi^2}\right] & \cdots & \phi^{|h|-1}\gamma_X(1) & \phi^{|h|-1}\gamma_X(1) \\ \sigma^2\left[(\phi+\theta)+\frac{(\phi+\theta)^2\phi}{1-\phi^2}\right] & \sigma^2\left[1+\frac{(\phi+\theta)^2}{1-\phi^2}\right] & \cdots & \phi^{|h|-1}\gamma_X(1) & \phi^{|h|-1}\gamma_X(1) \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ \phi^{|h|-1}\gamma_X(1) & \phi^{|h|-1}\gamma_X(1) & \cdots & \sigma^2\left[1+\frac{(\phi+\theta)^2}{1-\phi^2}\right] & \sigma^2\left[(\phi+\theta)+\frac{(\phi+\theta)^2\phi}{1-\phi^2}\right] \\ \phi^{|h|-1}\gamma_X(1) & \phi^{|h|-1}\gamma_X(1) & \cdots & \sigma^2\left[(\phi+\theta)+\frac{(\phi+\theta)^2\phi}{1-\phi^2}\right] & \sigma^2\left[1+\frac{(\phi+\theta)^2}{1-\phi^2}\right] \end{bmatrix}$$

Therefore, given the above mean vector (μ_n) and covariance matrix ($\mathbf{\Sigma}_{n\times n}$) above, we can express the likelihood in terms of $\mu, \phi, \theta, \sigma$.

b) Consider the dataset `ARMA_11_train.csv`.
Write the negative log-likelihood as a `R` function.
Test your function at the point $(\mu = 0, \phi = 0.9, \theta = 0.9, \sigma = 1)$.

The likelihood of the ARMA(1,1) process is given by

$$L(\mu_n, \mathbf{\Sigma}_{n\times n}|x_1,\ldots,x_n) = (2\pi)^{-n/2}\det\left(\mathbf{\Sigma}_{n\times n}^{-1}\right)^{1/2}\exp\left(-\frac{1}{2}(\mathbf{x}_n-\mu_n)^T\mathbf{\Sigma}_{n\times n}^{-1}(\mathbf{x}_n-\mu_n)\right), \quad -\infty < x_1,\ldots,x_n < \infty$$

Therefore, the negative log-likelihood of the ARMA(1,1) process is given by

$$-l(\mu_n, \mathbf{\Sigma}_{n\times n}|x_1,\ldots,x_n) = \left(\frac{n}{2}\right)\log(2\pi) - \frac{1}{2}\log\left(\det\left(\mathbf{\Sigma}_{n\times n}^{-1}\right)\right) + \frac{1}{2}(\mathbf{x}_n-\mu_n)^T\mathbf{\Sigma}_{n\times n}^{-1}(\mathbf{x}_n-\mu_n), \quad -\infty < x_1,\ldots,x_n < \infty$$

```
ARMAtrain = read.csv("ARMA_11_train.csv")
ARMAtest = read.csv("ARMA_11_test.csv")
```

```
# Function to construct covariance matrix
matCov_ARMA11 = function(params, vecX) {
  mu = params[1]
  phi = params[2]
  theta = params[3]
  sigma = params[4]
  n = length(vecX)

  # setup nxn matrix
  matCov = matrix(NA,
                  nrow = n,
                  ncol = n)

  # iterating over the matrix
  for (i in 1:n) {
    for (j in 1:n) {

    # calculate ACVF lag0, lag1, lag>1 terms
    lag0 = sigma^2 * ( 1 + ( ((phi+theta)^2) / (1-(phi^2)) ) )
    lag1 = sigma^2 * ( (phi+theta) + ( (((phi+theta)^2)*phi)/(1-(phi^2)) ) )
    lag2plus = phi^(abs(i-j)-1) * lag1

    # lag in covariance matrix depends on absolute value of lag
    if (abs(i-j) == 0) { matCov[i,j] = lag0 }
    if (abs(i-j) == 1) { matCov[i,j] = lag1 }
    if (abs(i-j) > 1) { matCov[i,j] = lag2plus }

    }
  }

  return(matCov)
}
```

```r
# Function to calculate negative log-likelihood of ARMA(1,1)
negLogLike_ARMA11 = function(params, vecX) {
  mu = params[1]
  phi = params[2]
  theta = params[3]
  sigma = params[4]
  n = length(vecX)

  # construct mu vector and covariance matrix
  vecMu = rep(mu, n)
  matCov = matCov_ARMA11(params, vecX)

  # calculate likelihood (for testing purposes)
  likelihood = c( (2*pi)^(-n/2) * det(solve(matCov))^(1/2) *
                    exp( -(1/2) * (t(vecX - vecMu) %*% matCov %*% (vecX - vecMu)) ) )

  # calculate negativel log-likelihood
  negLogLike = c( (n/2)*log(2*pi) - (1/2)*log(det(solve(matCov))) +
                    (1/2)*(t(vecX - vecMu) %*% matCov %*% (vecX - vecMu)) )

  return(negLogLike)
}

# Test point
params = c(0, 0.9, 0.9, 1)
negLogLike_ARMA11(params, ARMAtrain[,2])
```

```
## [1] 97257.89
```

c) Use a built-in minimizer to optimize the negative log-likelihood.
In `R`, if you are using `nlm()`, try the starting point: `p=c(mean(train), 0, 0, sd(train))`

`nlm()` is not a constrained optimization algorithm.
Therefore, we provide a starting point for the `nlm()` algorithm that allows the algorithm to proceed within reasonable bounds of our constraints (for a causal, invertible ARMA(1,1) with $Z_t \sim WN(0, \sigma^2)$):
i. $-1 < \phi < 1$
ii. $-1 < \theta < 1$
iii. $\sigma > 0$

```r
# MLE using nlm()
params = c(mean(ARMAtrain[,2]), 0.5, -0.5, sd(ARMAtrain[,2]))
vecMLE = nlm(negLogLike_ARMA11, params, ARMAtrain[,2])$estimate
names(vecMLE) = c("hat(mu)", "hat(theta)", "hat(phi)", "hat(sigma)")
vecMLE
```

```
##       hat(mu)    hat(theta)      hat(phi)    hat(sigma)
##  5.045613e+00  4.986105e-01 -5.013895e-01  7.878381e-06
```

```r
# Check answer with automated procedure
arma11_R = arima(ARMAtrain[,2], order=c(1,0,1), method="ML", include.mean=T)

vecMLE_2 = c(coef(arma11_R)[3], coef(arma11_R)[1:2], sqrt(arma11_R$sigma2))
names(vecMLE_2) = c("hat(mu)", "hat(phi)", "hat(theta)", "hat(sigma)")
vecMLE_2
```

```
##    hat(mu)   hat(phi) hat(theta) hat(sigma)
##  5.0372523  0.6915469 -0.6432522  0.5179652
```

d) Using the data `ARMA_11_train.csv`, forecast $h = 1, 2, 3, \ldots, 10$ steps ahead.
i.e. Compute $\hat{X}_{n+1}, \hat{X}_{n+2}, \hat{X}_{n+3}, \cdots, \hat{X}_{n+10}$.
Also compute the corresponding mean square prediction errors.

Let
i. $\mathbf{a}_n = (a_1, \ldots, a_n)^T$
ii. $\mathbf{\Gamma}_n = [\gamma(i-j)]_{i,j=1}^n$, the covariance matrix
iii. $\gamma_n(h) = [\gamma(h), \gamma(h+1), \ldots, \gamma(h+n-1)]^T$

The best linear predictor is given by

$$\hat{X}_{n+h} = P_n X_{n+h} = a_0 + a_1 X_n + \cdots + a_n X_1$$

For a_0 and $\mathbf{a}_n$ that satisfy the forecasting equations

$$a_0 = \mu \left(1 - \sum_{i=1}^{n} a_i\right)$$

$$\mathbf{a}_n = \mathbf{\Gamma}_n^{-1} \gamma_n(h)$$

Note that a_0 relates to the mean, and $\mathbf{a}_n$ relates to the covariances.

Mean square prediction error (MSPE) is given by

$$MSPE = E\left[\left(X_{n+h} - \hat{X}_{n+h}\right)^2\right] = \gamma_X(0) - \mathbf{a}_n^T \gamma_n(h)$$

Note that we will substitute sample values into the above expressions.

i) Assuming the ARMA(1,1) structure, replace θ, ϕ, σ with their respective MLE's $\hat{\theta}, \hat{\phi}, \hat{\sigma}$, and solve the forecasting equations.

For this part, we will use $\hat{\mu}_{MLE}$ as an estimator of μ to calculate a_0, because this problem assumes the ARMA(1,1) structure.

```
# h-step ahead prediction based on ARMA(1,1) ACVF using MLE
hStepPred_mleACVF_ARMA11 = function(h, vecX) {

  # setup vector of desired h-step ahead predicted values
  vecH = 1:h

  # extract MLE values
  mu = vecMLE[1]
  phi = vecMLE[2]
  theta = vecMLE[3]
  sigma = vecMLE[4]
  n = length(ARMAtrain[,2])

  # calculate ACVF values based on lag
  lag1 = sigma^2 * ( (phi+theta) + ( (((phi+theta)^2)*phi)/(1-(phi^2)) ) )
  lagVec = 2:(n+length(vecH)-1)
  vecLag2Plus = phi^(lagVec-1) * lag1

  # construct extended gamma(h) vector that includes all possible lags that will be used (starting with lag 1)
  extendedVecGamma_h = c(lag1,vecLag2Plus)

  # construct covariance matrix
  matGamma = matCov_ARMA11(vecMLE, ARMAtrain[,2])

  # setup vectors for predicted values and MSPE
  vecPred = rep(NA, 10)
  vecMSPE = rep(NA, 10)

  # loop to construct gamma(h) vector of appropriate size and h-step ahead prediction
```

```
  for (i in vecH) {

    # construct gamma(h) vector from h to h+n-1 (length n)
    vecGamma_h = extendedVecGamma_h[i:(n+i-1)]

    # find prediction values of forecasting equations
    vecAn = solve(matGamma) %*% vecGamma_h
    a0 = mu*(1-sum(vecAn))

    # vector of prediction values
    vecA0_n = c(a0, vecAn)

    # vector of x-values with 1 term in the linear combination is used to estimate the mean mu
    vec1X = c(1,vecX[n:1])

    # calculate prediction, store in vecPred
    vecPred[i] = sum(vecA0_n %*% vec1X)

    # calculate MSPE, store in vecMSPE
    vecMSPE[i] = matGamma[1,1]-(t(vecAn) %*% vecGamma_h)
  }

  outList = list(vecPred,vecMSPE)
  names(outList) = c("h-Step Predictions", "h-Step MSPE")

  return(outList)
}

hStepPred_mleACVF_ARMA11(10, ARMAtrain[,2])
```

```
## $`h-Step Predictions`
##  [1] 5.047444 5.046526 5.046068 5.045840 5.045727 5.045670 5.045642 5.045628
##  [9] 5.045620 5.045617
##
## $`h-Step MSPE`
##  [1] 6.206889e-11 6.206937e-11 6.206949e-11 6.206952e-11 6.206953e-11
##  [6] 6.206953e-11 6.206953e-11 6.206953e-11 6.206953e-11 6.206953e-11
```

```
# Check answer with automated procedure
arma11_R = arima(ARMAtrain[,2], order=c(1,0,1), method="ML", include.mean=T)
predict(arma11_R, n.ahead=10)
```

```
## $pred
## Time Series:
## Start = 31
## End = 40
## Frequency = 1
##  [1] 5.006190 5.015771 5.022397 5.026979 5.030148 5.032339 5.033855 5.034903
##  [9] 5.035627 5.036129
##
## $se
## Time Series:
## Start = 31
## End = 40
## Frequency = 1
##  [1] 0.5179652 0.5185689 0.5188573 0.5189952 0.5190612 0.5190927 0.5191078
##  [8] 0.5191150 0.5191184 0.5191201
```

ii) Don't assume an ARMA(1,1) and set up $\hat{\mathbf{\Gamma}}$ and $\hat{\gamma}$ directly using the sample ACVF values from `acf()`.

For this part, we will use $\hat{\mu} = \bar{X}_n$ as an estimator of μ to calculate a_0, because this problem assumes a naive estimation procedure using sample information.

```
# h-step ahead prediction based on sample ACVF
hStepPred_sampleACVF = function(h, vecX) {

  # setup vector of desired h-step ahead predicted values
  vecH = 1:h
  mu = mean(vecX)

  # extract all possible sample autocovariances for lags 0 to length(trainData)-1 from acf function
  # to construct extended gamma(h) vector (starts with lag 0 term)
  # Note that we only have n-1 lags, not the necessary n+h-1 lags to construct
  # complete gamma(h) vectors and Gamma covariance matrices
  n = length(vecX)
  extendedVecGamma_h = c( acf(vecX, plot=F, type="covariance", lag.max = (n-1))$acf )

  # setup vectors for predicted values and MSPE
  vecPred = rep(NA, 10)
  vecMSPE = rep(NA, 10)

  for (i in vecH) {

    # Construct gamma(h) vector from extendedVecGamma(h)
    # recall that it starts with a lag 0 term and ends at n-1 lags
    vecGamma_h = extendedVecGamma_h[(i+1):length(extendedVecGamma_h)]
    n2 = length(vecGamma_h)

    matGamma = matrix(NA,
                      nrow = n2,
                      ncol = n2)

    for (j in 1:n2) {

      if (j == 1) {

        matGamma[j,] = extendedVecGamma_h[1:n2]

      } else if (j != 1 & j != n2) {

        matGamma[j,] = c( extendedVecGamma_h[j:1], extendedVecGamma_h[2:(n2-(j-1))] )

      } else if (j == n2) {

        matGamma[j,] = extendedVecGamma_h[n2:1]
      }
    }

    # find prediction values of forecasting equations
    vecAn = solve(matGamma) %*% vecGamma_h
    a0 = mu*(1-sum(vecAn))

    # vector of prediction values
    vecA0_n = c(a0, vecAn)

    # vector of x-values with 1 term in the linear combination is used to estimate the mean mu
    vec1X = c(1,vecX[n:(n-length(vecA0_n)+2)])

    # calculate prediction, store in vecPred
    vecPred[i] = sum(vecA0_n %*% vec1X)
```

```
    # calculate MSPE, store in vecMSPE
    vecMSPE[i] = extendedVecGamma_h[1]-(t(vecAn) %*% vecGamma_h)

  }

  outList = list(vecPred,vecMSPE)
  names(outList) = c("h-Step Predictions", "h-Step MSPE")

  return(outList)
}

hStepPred_sampleACVF(10, ARMAtrain[,2])
```

```
## $`h-Step Predictions`
##  [1] 4.981108 4.948973 5.035447 5.137408 4.711076 4.934121 4.672827 5.104209
##  [9] 5.098443 5.092189
##
## $`h-Step MSPE`
##  [1] 0.1981428 0.1982658 0.1985099 0.2046977 0.2047954 0.2077927 0.2082066
##  [8] 0.2104279 0.2136010 0.2144246
```

```
# Check answer with automated procedure
arma11_R = arima(ARMAtrain[,2], order=c(1,0,1), method="ML", include.mean=T)
predict(arma11_R, n.ahead=10)
```

```
## $pred
## Time Series:
## Start = 31
## End = 40
## Frequency = 1
##  [1] 5.006190 5.015771 5.022397 5.026979 5.030148 5.032339 5.033855 5.034903
##  [9] 5.035627 5.036129
##
## $se
## Time Series:
## Start = 31
## End = 40
## Frequency = 1
##  [1] 0.5179652 0.5185689 0.5188573 0.5189952 0.5190612 0.5190927 0.5191078
##  [8] 0.5191150 0.5191184 0.5191201
```

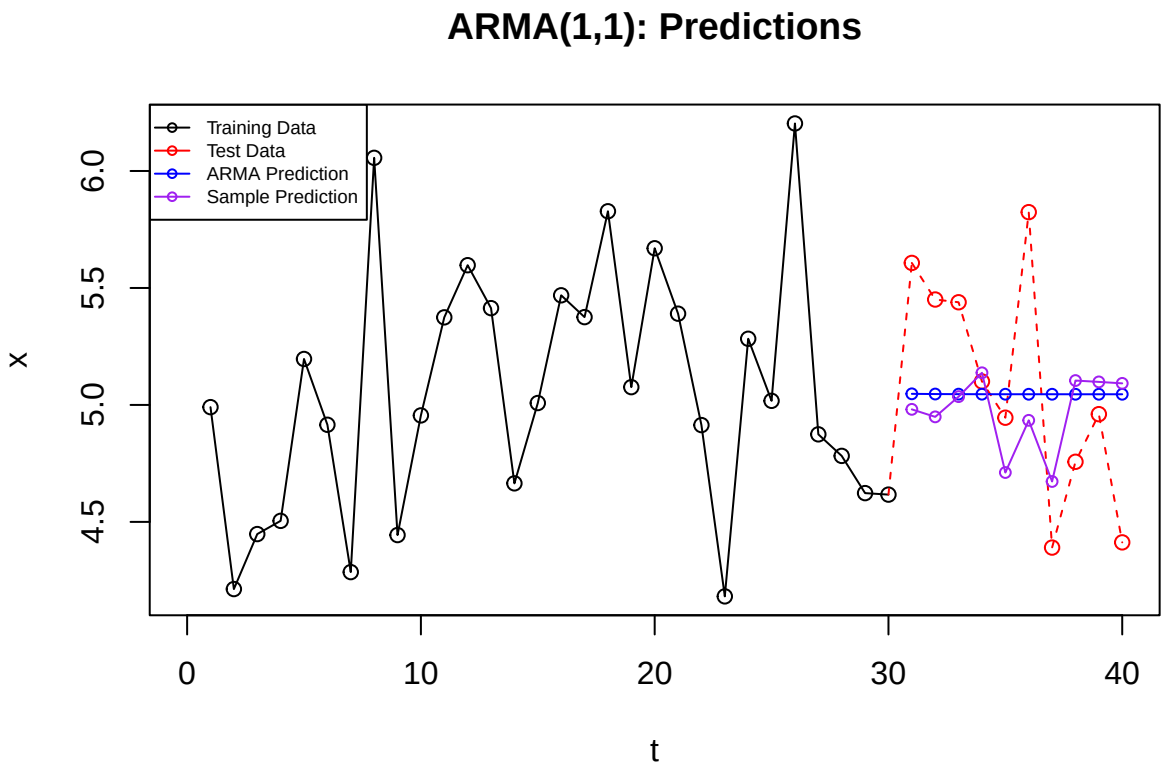
e) Using the data `ARMA_11_test.csv`, compute the test error for each method described above.

```
testData = ARMAtest[,2]

# Test error of prediction using ARMA(1,1) ACVF and MLE's
pred_ARMA11 = hStepPred_mleACVF_ARMA11(10, ARMAtrain[,2])[[1]]
testError_ARMA11 = mean((testData - pred_ARMA11)^2)
paste("Test Error ARMA(1,1) Prediction:", round(testError_ARMA11,4))
```

```
## [1] "Test Error ARMA(1,1) Prediction: 0.217"
```

```
# Test error of prediction using sample ACVF and sample mean
pred_sample = hStepPred_sampleACVF(10, ARMAtrain[,2])[[1]]
testError_sampleACVF = mean((testData - pred_sample)^2)
paste("Test Error Sample Prediction:", round(testError_sampleACVF,4))
```

```
## [1] "Test Error Sample Prediction: 0.2336"
```



Example: ARMA Order Selection - AICC

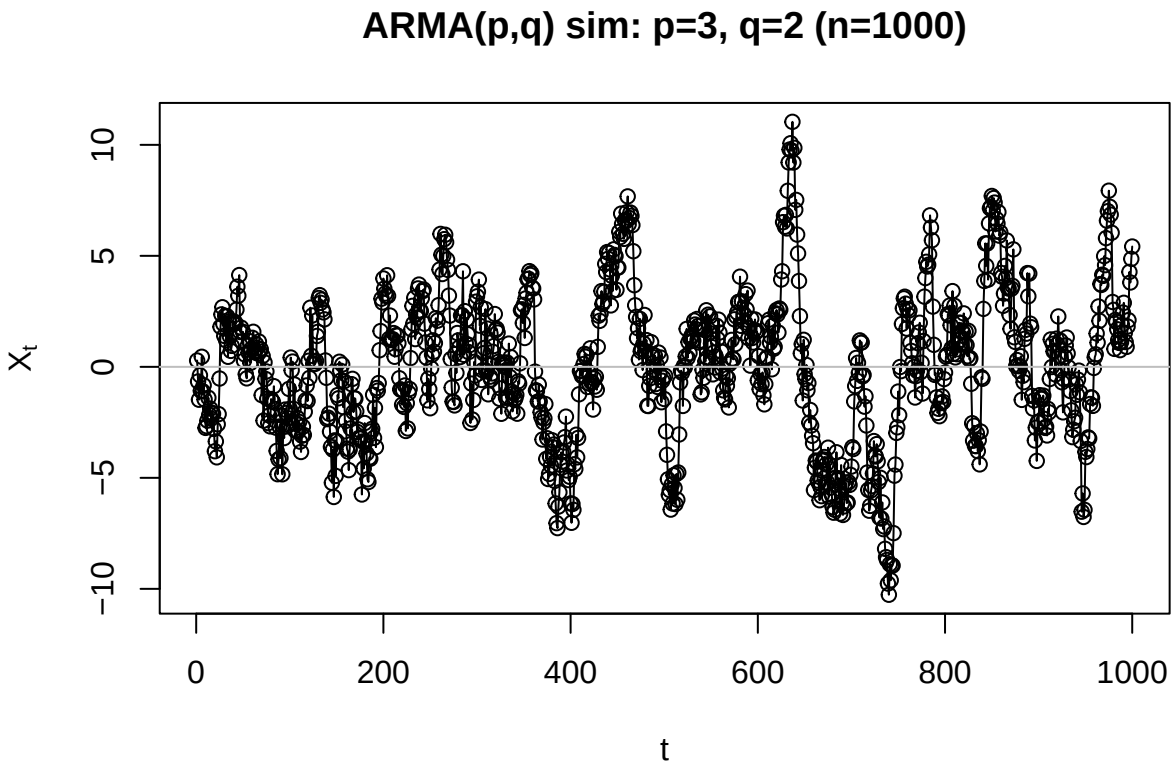
```
# ARMA Simulation Function, 20% burn in period
my_ARMA_sim = function(n=189,
                       ma_coeff=c(.5),
                       ar_coeff=c(.5),
                       sigma=1) {

  n.start = n + floor(.2/(1-.2)*n)
  burnin = floor(.20*n.start)
  model_params = list(ma = ma_coeff,ar=ar_coeff)
  ts_sim = arima.sim(model = model_params, n = n.start,sd=sigma)
  ts_sim = ts_sim[(burnin+1):n.start]
  return(ts_sim)
}

#####################################################################
# Test some ARMA(p,a) models with their sample ACF/PACF's
#####################################################################

# Causal, Invertible ARMA
n=1000 # Try other sample sizes
ar_coeff=c(.4,.4,.1)
ma_coeff=c(.4,.5)
sigma = 1

paste("p=",length(ar_coeff),", ","q=",length(ma_coeff),sep="")
```

```
## [1] "p=3, q=2"
```

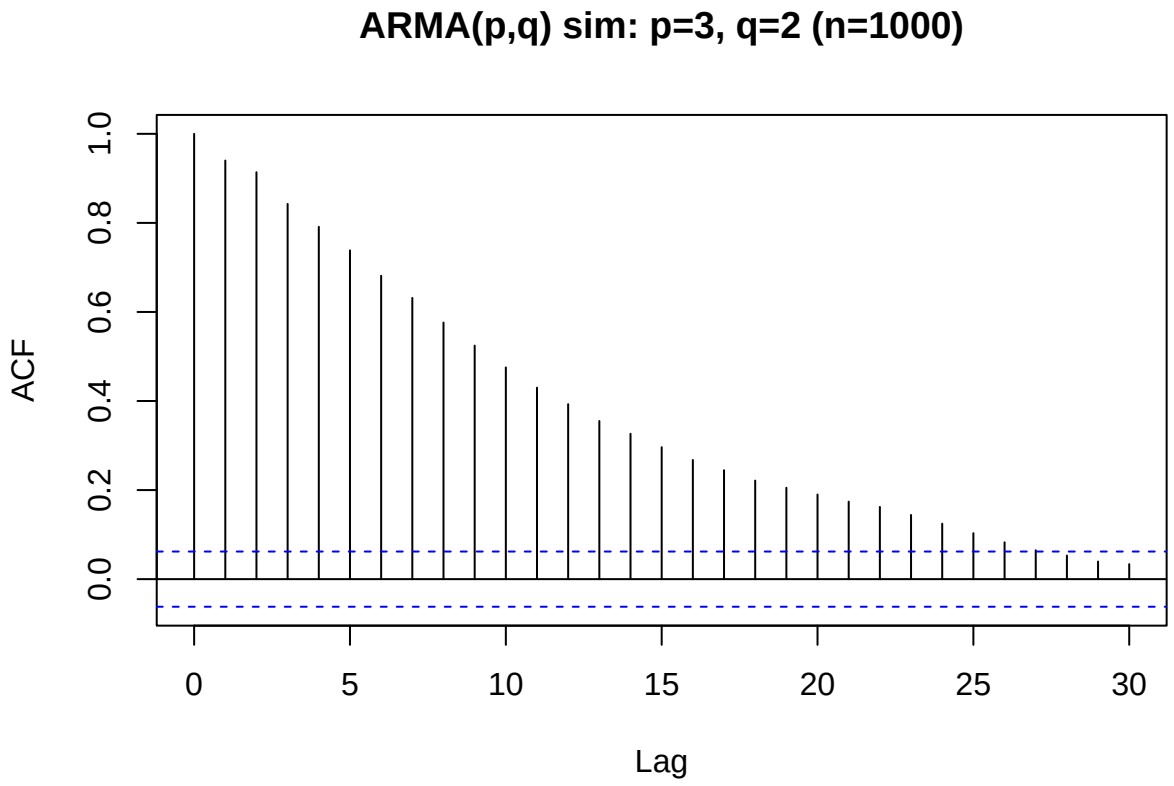
```
main_exp <- paste("ARMA(p,q) sim:",paste("p=",length(ar_coeff),", ","q=",length(ma_coeff),sep=""), paste("(n="
ARMA_sim = my_ARMA_sim(n=n,ma_coeff=ma_coeff,ar_coeff=ar_coeff,sigma=sigma)
length(ARMA_sim)
```

```
## [1] 1000
```

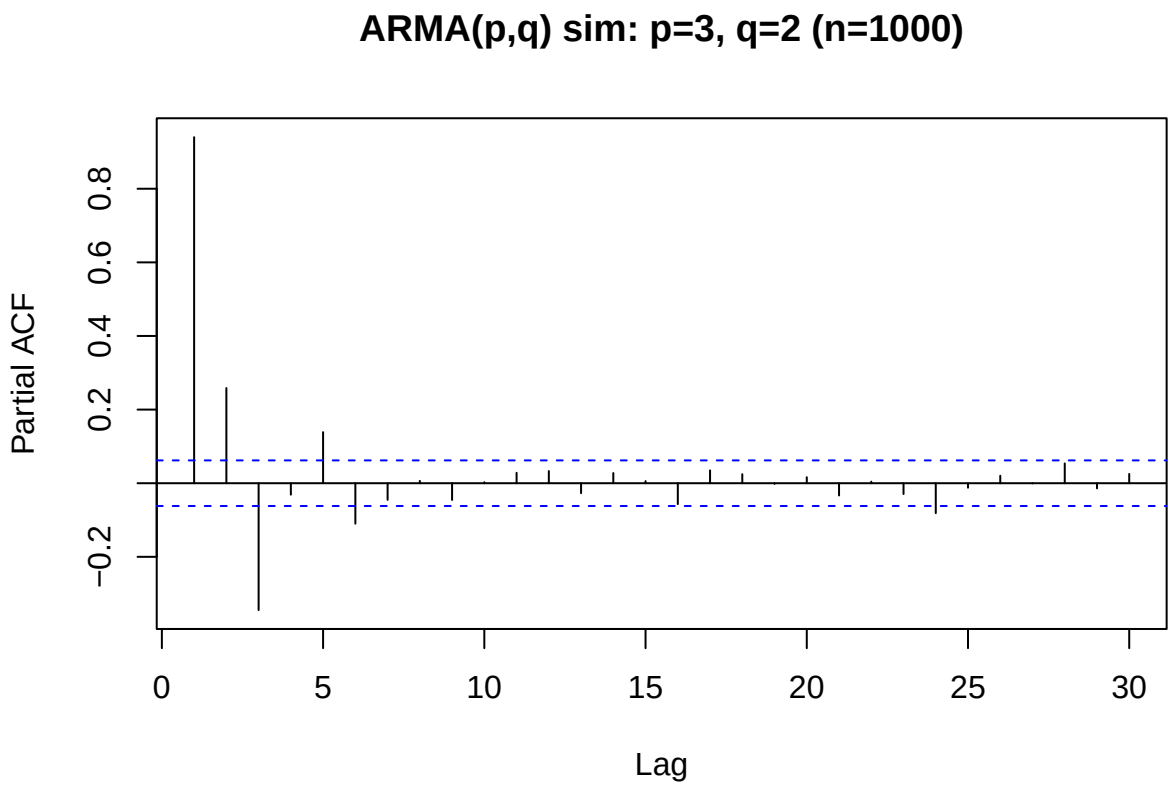
```
set.seed(2)
ARMA_sim = ARMA_sim - mean(ARMA_sim)
plot(ARMA_sim,type="o",
     main=main_exp,
     xlab="t",
     ylab=expression(X[t]))
abline(h=0,lty=1,col="grey")
```




```
# ACF and PACF do not clearly indicate order
acf(ARMA_sim, main=main_exp) # Decreasing moderately
```




```
pacf(ARMA_sim, main=main_exp)
```



```
#####################################################################
# AIC order selection
#####################################################################
library("forecast")
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
##
## Attaching package: 'forecast'
```

```
## The following object is masked from 'package:itsmr':
##
##     forecast
```

```
# Grid Search Using AICC to select ARIMA
auto.arima(y=ARMA_sim,max.P=0,max.Q=0,max.d=0,max.D=0)
```

```
## Series: ARMA_sim
## ARIMA(4,0,2) with zero mean
##
## Coefficients:
##          ar1     ar2     ar3      ar4     ma1     ma2
##       0.1913  0.6082  0.2223  -0.1313  0.6071  0.4449
## s.e.  0.0997  0.1122  0.0732   0.0602  0.0962  0.0757
##
## sigma^2 = 0.9989:  log likelihood = -1416.86
## AIC=2847.71   AICc=2847.82   BIC=2882.07
```

```
# p=4, q=3 based on AICC
# Fit with with mean (mu)
Arima(y=ARMA_sim,order=c(4,0,3),include.mean = T)
```

```
## Series: ARMA_sim
## ARIMA(4,0,3) with non-zero mean
##
## Coefficients:
##           ar1     ar2     ar3      ar4     ma1     ma2     ma3    mean
##       -0.0641  0.7029  0.3120  -0.0850  0.8630  0.5524  0.1395  0.0925
## s.e.   0.3584  0.1224  0.1577   0.1244  0.3583  0.2267  0.2071  0.5914
##
## sigma^2 = 1.002:  log likelihood = -1417.24
## AIC=2852.48   AICc=2852.66   BIC=2896.65
```

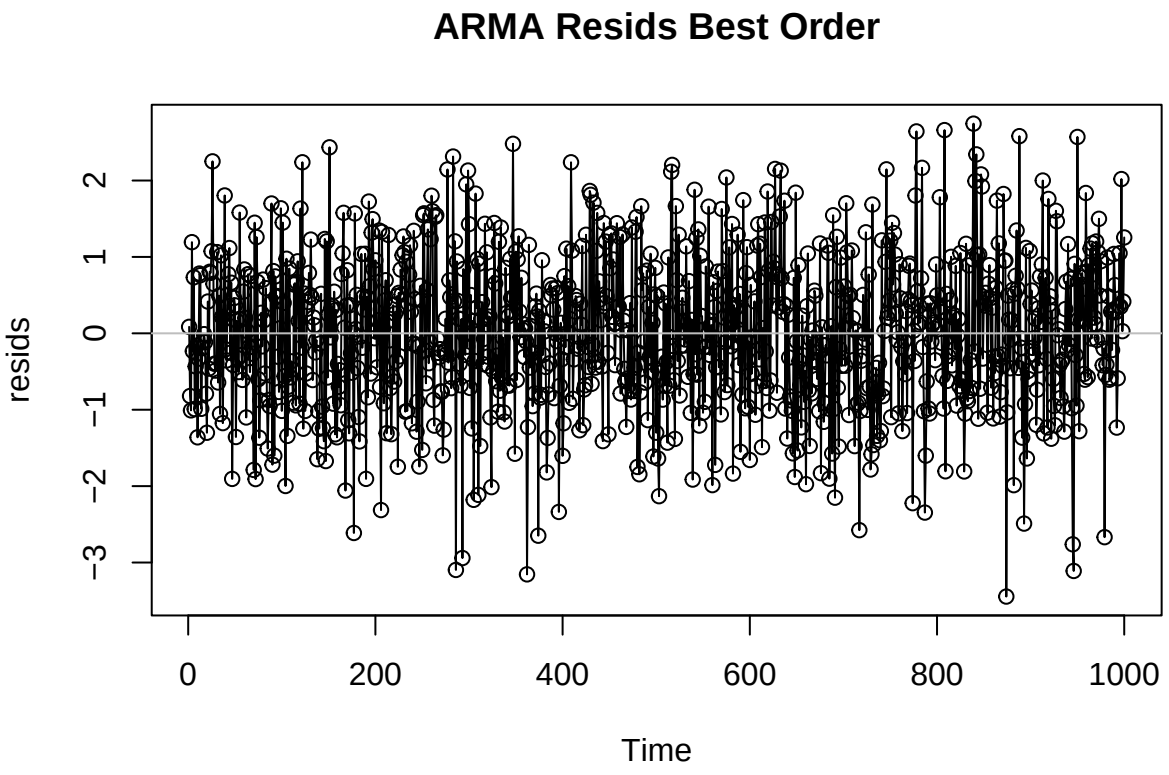
```
# Fit without with mean (mu=0)
Arima_fit = Arima(y=ARMA_sim,order=c(4,0,3),include.mean = F)
Arima_fit
```

```
## Series: ARMA_sim
## ARIMA(4,0,3) with zero mean
##
## Coefficients:
##           ar1     ar2     ar3      ar4     ma1     ma2     ma3
##       -0.0684  0.7013  0.3134  -0.0823  0.8662  0.5565  0.1433
## s.e.   0.3686  0.1255  0.1606   0.1250  0.3685  0.2291  0.2119
##
## sigma^2 = 1.001:  log likelihood = -1417.26
## AIC=2850.53   AICc=2850.67   BIC=2889.79
```

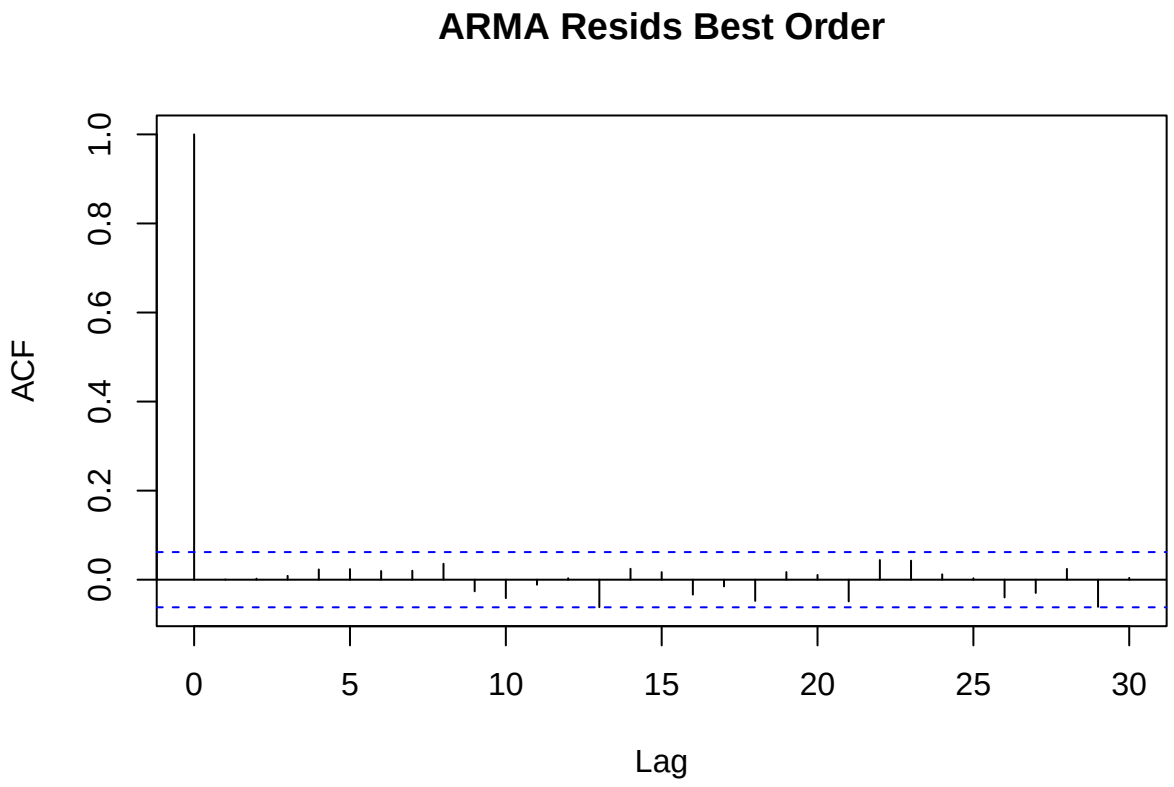
```
#################################################################
# Diagnostics
#################################################################

# For chosen model
Arima_fit = Arima(y=ARMA_sim,order=c(4,0,3),include.mean = F)
Arima_fit
```

```
## Series: ARMA_sim
## ARIMA(4,0,3) with zero mean
##
## Coefficients:
##           ar1     ar2     ar3      ar4     ma1     ma2     ma3
##       -0.0684  0.7013  0.3134  -0.0823  0.8662  0.5565  0.1433
## s.e.   0.3686  0.1255  0.1606   0.1250  0.3685  0.2291  0.2119
##
## sigma^2 = 1.001:  log likelihood = -1417.26
## AIC=2850.53   AICc=2850.67   BIC=2889.79
```

```
# Residual diagonistics - proxy for noise Z (resemble Z structure)
resids = Arima_fit$residuals
plot(resids,type="o",main="ARMA Resids Best Order")
abline(h=0,lty=1,col="grey")
```



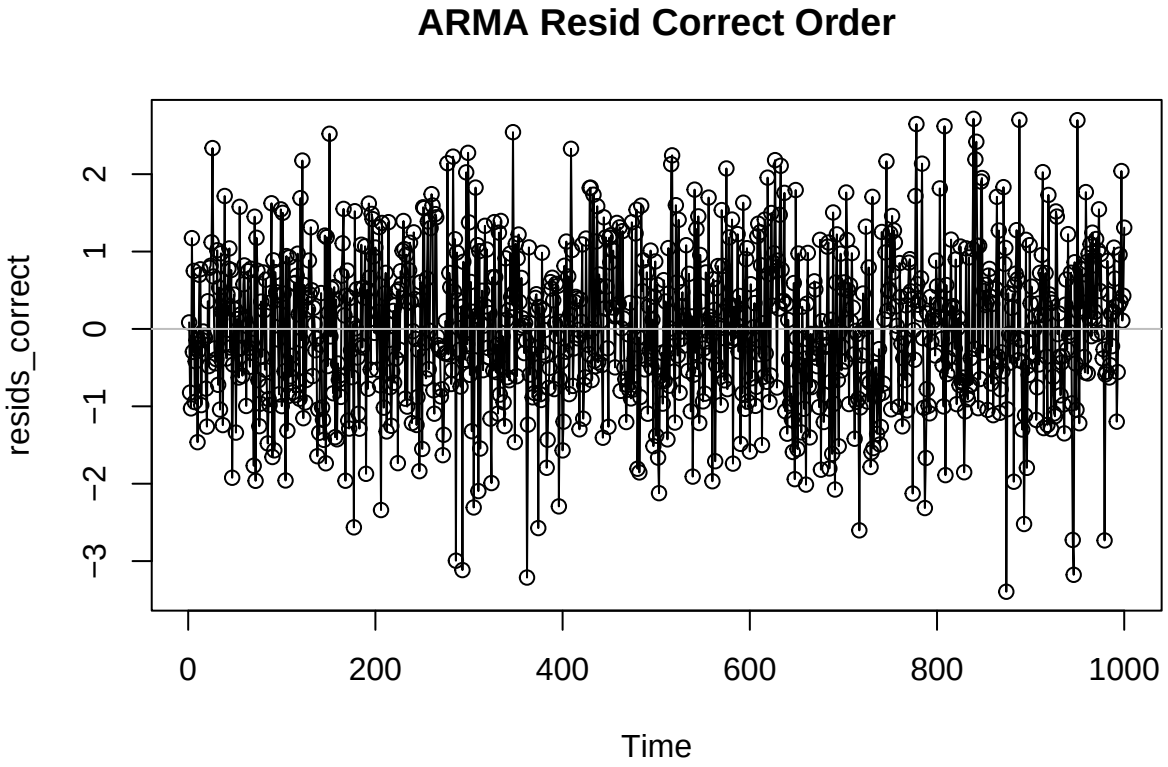
```r
acf(resids,main="ARMA Resids Best Order")
```



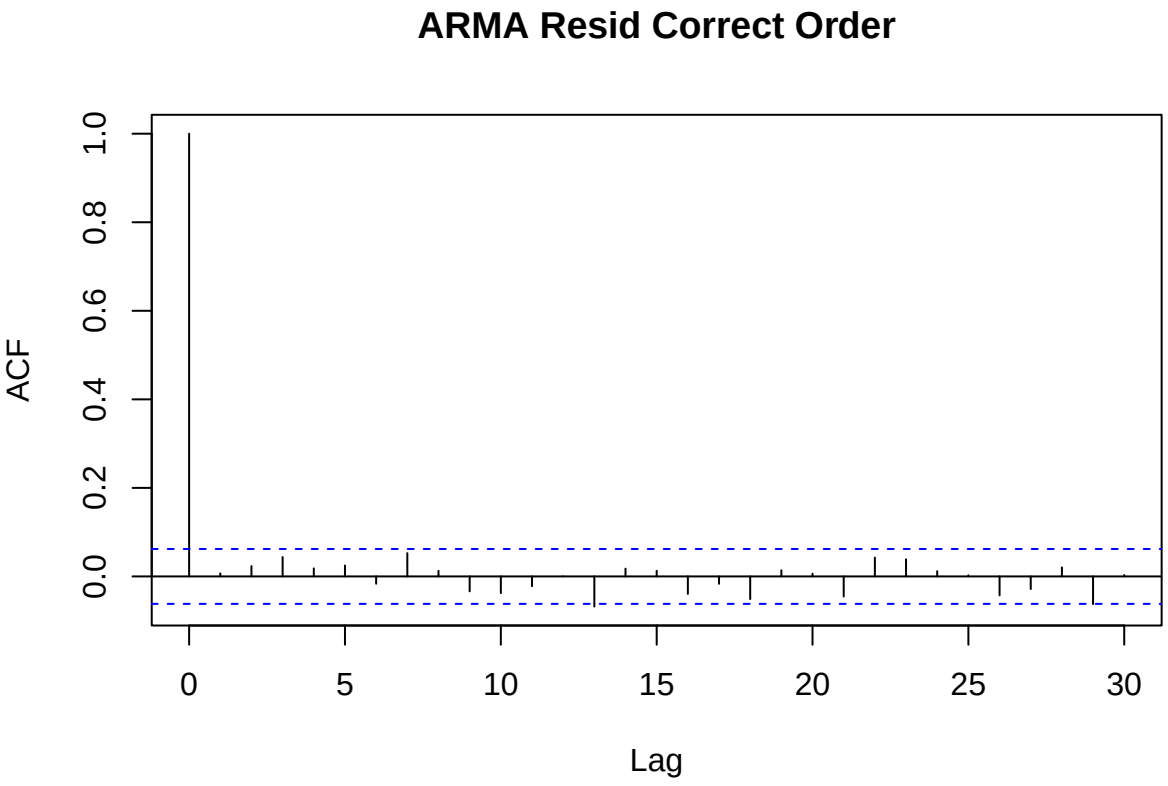

```r
# ACF plot of residuals is nice for autofit

# For correct model (which we don't know in practice)
Arima_fit_correct = Arima(y=ARMA_sim,order=c(3,0,2),include.mean = F)
Arima_fit_correct
```

```
## Series: ARMA_sim
## ARIMA(3,0,2) with zero mean
##
## Coefficients:
##          ar1     ar2     ar3     ma1     ma2
##       0.3314  0.4063  0.1642  0.4609  0.5107
## s.e.  0.1116  0.0506  0.1004  0.1040  0.0724
##
## sigma^2 = 1.003:  log likelihood = -1419.38
## AIC=2850.77   AICc=2850.85   BIC=2880.21
```

```r
resids_correct = Arima_fit_correct$residuals
plot(resids_correct,type="o",main="ARMA Resid Correct Order")
abline(h=0,lty=1,col="grey")
```




```
acf(resids_correct,main="ARMA Resid Correct Order")
```



```r
# Residuals of autofit and true model are very comparable (in terms of residual analysis)

# Compare AICC of two models
Arima_fit$aicc
```

```
## [1] 2850.673
```

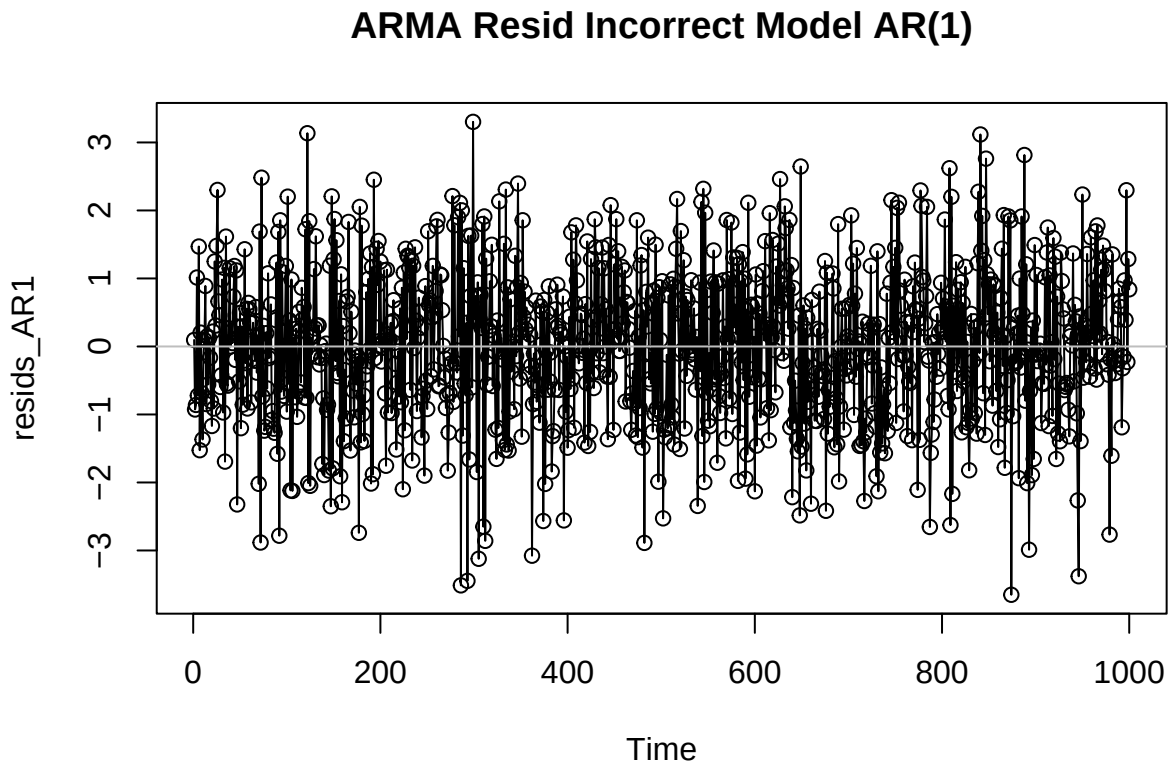
```r
Arima_fit_correct$aicc
```

```
## [1] 2850.851
```

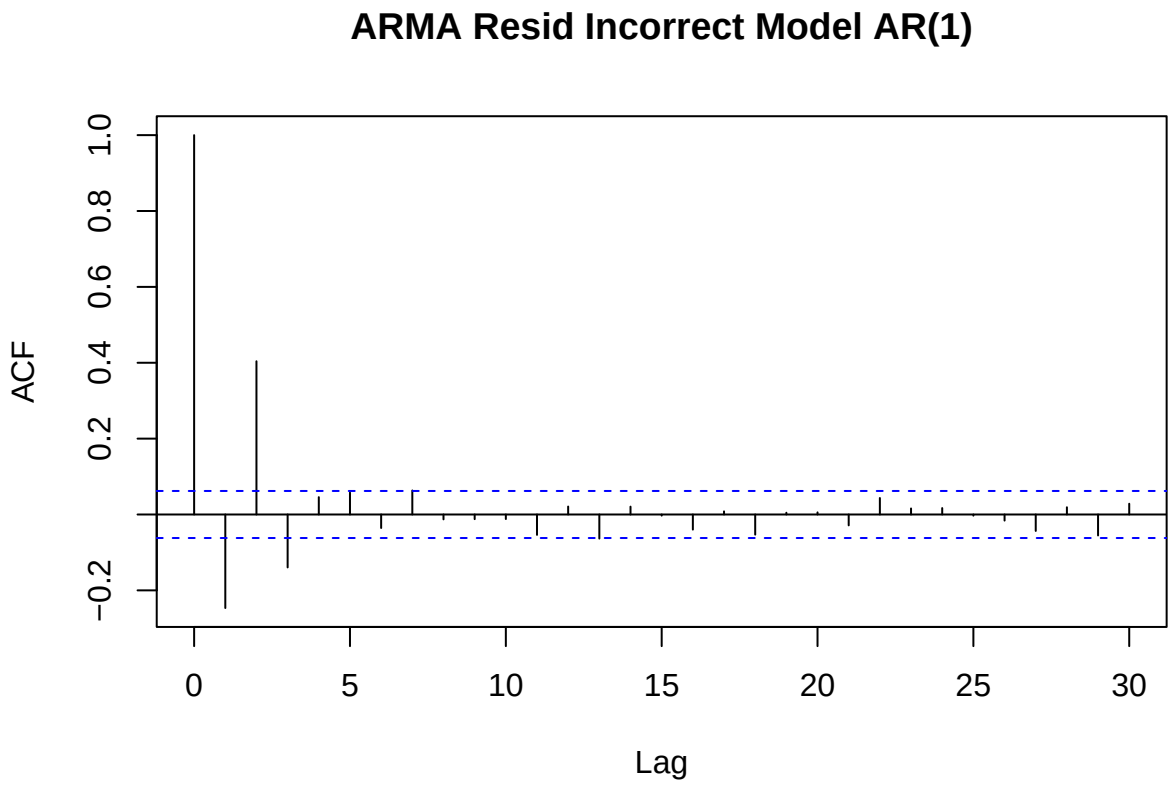
```r
# Maybe we over-fit the data using auto.arima()?

# For incorrect model AR(1)
Arima_fit_AR1 = Arima(y=ARMA_sim,order=c(1,0,0),include.mean = F)
Arima_fit_AR1
```

```
## Series: ARMA_sim
## ARIMA(1,0,0) with zero mean
##
## Coefficients:
##          ar1
##       0.9416
## s.e.  0.0106
##
## sigma^2 = 1.27:  log likelihood = -1539.01
## AIC=3082.01   AICc=3082.02   BIC=3091.83
```

```r
resids_AR1 = Arima_fit_AR1$residuals
plot(resids_AR1,type="o",main="ARMA Resid Incorrect Model AR(1)")
abline(h=0,lty=1,col="grey")
```



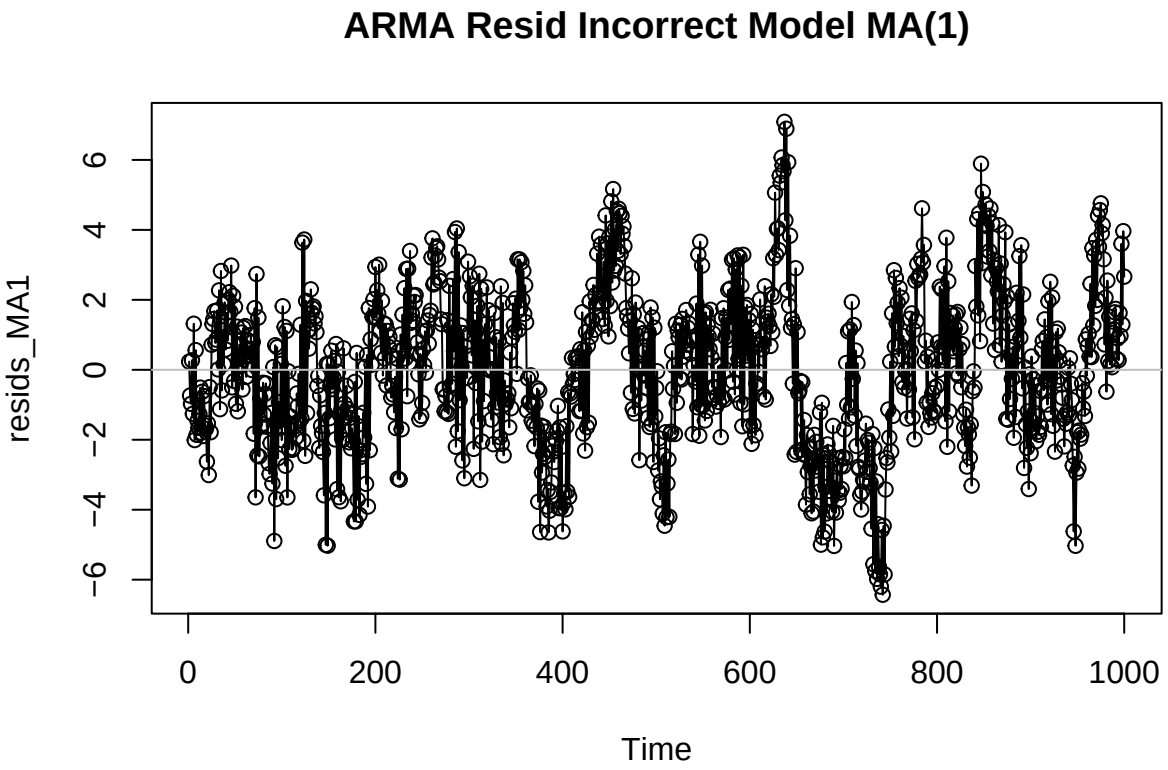
```r
acf(resids_AR1,main="ARMA Resid Incorrect Model AR(1)")
```



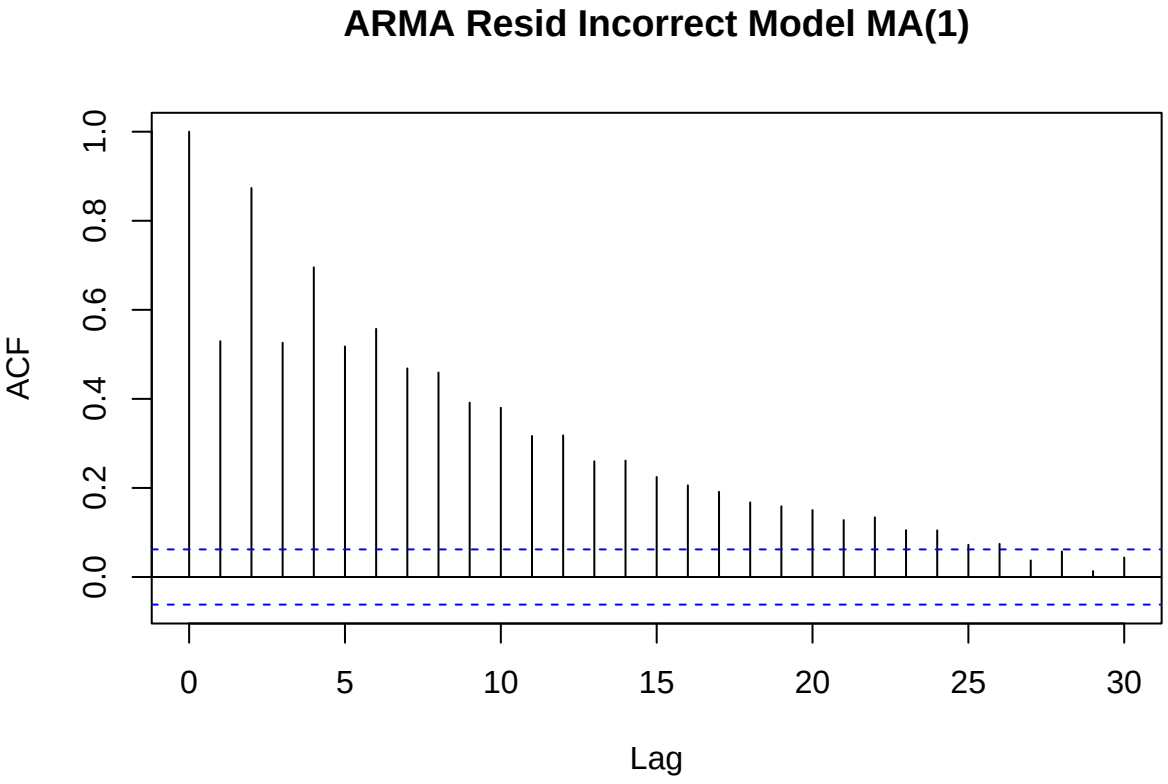

```r
# Residuals look ok even though wrong model

# For incorrect model MA(1)
Arima_fit_MA1 = Arima(y=ARMA_sim,order=c(0,0,1),include.mean = F)
Arima_fit_MA1
```

```
## Series: ARMA_sim
## ARIMA(0,0,1) with zero mean
##
## Coefficients:
##          ma1
##       0.6955
## s.e.  0.0164
##
## sigma^2 = 5.024:  log likelihood = -2225.86
## AIC=4455.73   AICc=4455.74   BIC=4465.54
```

```r
resids_MA1 = Arima_fit_MA1$residuals
plot(resids_MA1,type="o",main="ARMA Resid Incorrect Model MA(1)")
abline(h=0,lty=1,col="grey")
```




```
acf(resids_MA1,main="ARMA Resid Incorrect Model MA(1)")
```



```r
# Residuals are not good

# Compare AICC's of all models
data.frame(Model=c("auto Arima (p=4,q=3)",
                   "True model (p=3,q=2)",
                   "AR(1) (p=1,q=0)",
                   "MA(1) (p=0,q=1)"),
           AICC=c(Arima_fit$aicc,
                  Arima_fit_correct$aicc,
                  Arima_fit_AR1$aicc,
                  Arima_fit_MA1$aicc)
           )
```

```
##                  Model     AICC
## 1 auto Arima (p=4,q=3) 2850.673
## 2 True model (p=3,q=2) 2850.851
## 3      AR(1) (p=1,q=0) 3082.024
## 4      MA(1) (p=0,q=1) 4455.739
```

NON-STATIONARY AND SEASONAL TIME SERIES MODELS

ARIMA(p,d,q) Models:
If d is a non-negative integer, then the process $\{X_t\}$ is an ARIMA(p,d,q) (auto-regressive iterative moving average) if

$$Y_t := (1-B)^d X_t$$

Is a causal ARMA(p,q) process.

Thus $\{X_t\}$ satisfies

$$\Phi^*(B)X_t = \Phi(B)(1-B)^d X_t = \Phi(B)Y_t = \Theta(B)Z_t, \quad Z_t \sim WN(0,\sigma^2)$$

Where
i. $\Phi^*(B) = \Phi(B)(1-B)^d$
ii. $\Phi(u)$ and $\Theta(u)$ are polynomials of degree p and q respectively,
iii. $\Phi(u) \neq 0$ for all $|u| \leq 1$.

In other words, $\{X_t\}$ is an ARIMA(p,d,q) process if d applications of a difference operator to a non-stationary time series $\{X_t\}$ results in a stationary ARMA(p,q) process $\{Y_t\}$.
Recall that differencing a random walk yields WN, and that d applications of the difference operator removes polynomial trend of order d.

i. Linear Trend ($d=1$)

$$\nabla(\beta_0 + \beta_1 t) = (\beta_0 + \beta_1 t) - (\beta_0 + \beta_1(t-1)) = \beta_1$$

ii. Quadratic Trend ($d=2$)

$$\nabla(\beta_0 + \beta_1 t + \beta_2 t^2) = (\beta_0 + \beta_1 t + \beta_2 t^2) - (\beta_0 + \beta_1(t-1) + \beta_2(t-1)^2) = (\beta_1 - \beta_2) + 2t\beta_2$$
$$\nabla^2(\beta_0 + \beta_1 t + \beta_2 t^2) = (\beta_1 - \beta_2 + 2t\beta_2) - (\beta_1 - \beta_2 + 2(t-1)\beta_2) = 2\beta_2$$

Forecasting ARIMA(p,d,q) Models - Formulating a BLP of the ARIMA(p,d,q) Model:

Forecasting ARIMA(p,1,q) Models, d=1:
For example, let $\{X_t\}$ be an ARIMA(1,1,0) process for some $-1 < \phi < 1$,

$$(1-\phi B)(1-B)X_t = Z_t, \quad Z_t \sim WN(0,\sigma^2)$$

Notice that we can write X_t as a telescoping sum of the differenced time series for $d=1$,

$$X_t = X_0 + (X_1 - X_0) + (X_2 - X_1) + \cdots + (X_t - X_{t-1})$$
$$= X_0 + \sum_{j=1}^{t}(1-B)X_j, \quad t \geq 1$$

For $X_t - X_{t-1} = (1-B)X_t = Y_t$, we have

$$= X_0 + \sum_{j=1}^{t} Y_j, \quad t \geq 1$$

Where Y_t is a causal AR(1) process,

$$Y_t = (1-B)X_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}$$

Consider the future case X_{n+1}, e.g. from a data set of size n.
We can express X_{n+1} as

$$X_{n+1} = X_n + (X_{n+1} - X_n) \equiv X_n + Y_{n+1}$$

Applying the BLP to both sides of the equation and applying properties of the BLP, we can formulate a one-step ahead BLP for the ARIMA(p,1,q) model.

$$P_n(X_{n+1}|X_n,\ldots,X_0) = P(X_n + Y_{n+1}|X_n,\ldots,X_0) \overset{filtration}{=} X_n + P(Y_{n+1}|X_n,\ldots,X_0)$$

Where Y_{n+1} comes from Y_t, a causal AR(1) process.

To forecast an ARIMA(p,1,q) model, the one-step ahead BLP of an ARIMA(p,1,q) model is given by
i. Using the BLP, forecast the differenced causal ARMA(p,q) series $Y_t = (1-B)X_t$ (i.e. find $P_n Y_{n+1}$).

Note that $P_n Y_{n+1}$ represents the BLP of a causal ARMA(p,q) process, for which we can use the BLP formula with MLE parameter estimators, innovations, etc.

ii. Add the most recent data point, X_n.

Forecasting ARIMA(p,d,q) Models, d>1:
Recall that the backshift operator B behaves similar to a polynomial, e.g.,

$$(1-B)^2 = 1 - 2B + B^2$$

We would like to expand a binomial expression of backshift operators, $(1-B)^d$.

Recall that by the binomial theorem,

$$(a+b)^d = \sum_{k=0}^{d} \binom{d}{k} a^d b^{d-k}$$

Substituting $a = -B$, $b = 1$, we obtain

$$(1-B)^d = (-B+1)^d = \sum_{k=0}^{d} \binom{d}{k} (-B)^k (1)^{d-k} = 1 + \sum_{k=1}^{d} \binom{d}{k} (-1)^k (B)^k$$

Incorporating this result with an ARIMA(p,d,q) model, we obtain

$$Y_t = (1-B)^d X_t = X_t \left(1 + \sum_{j=1}^{d} \binom{d}{j} (-1)^j (B)^j \right) = X_t + \sum_{j=1}^{d} \binom{d}{j} (-1)^j X_{t-j}$$

Consider the future case X_{n+1}, e.g. from a data set of size n.
Using our above expression for an ARIMA(p,d,q) model, we can express X_{n+1} as

$$X_{n+1} = Y_{n+1} - \sum_{j=1}^{d} \binom{d}{j} (-1)^j X_{n+1-j}$$

Applying the BLP to both sides of the equation and applying properties of the BLP, we can formulate a one-step ahead BLP for the ARIMA(p,d,q) model.

$$P_n X_{n+1} = P_n Y_{n+1} - \sum_{j=1}^{d} \binom{d}{j} (-1)^j P_n X_{n+1-j}$$

Note that for $j = 1, \ldots, d$, $d > 1$ (in the binomial expansion), $\min(n+1-j) = n+1-d$, $d > 1$, meaning that the index will always be less than n, resulting in

$$P_n X_{n+1-j} = P(X_{n+1-j} | X_n, \ldots, X_0) \overset{filtration}{=} X_{n+1-j}$$

To forecast an ARIMA(p,1,q) model, the one-step ahead BLP of an ARIMA(p,1,q) model is given by

$$P_n X_{n+1} = P_n Y_{n+1} - \sum_{j=1}^{d} \binom{d}{j} (-1)^j P_n X_{n+1-j}$$

$$= P_n Y_{n+1} - \sum_{j=1}^{d} \binom{d}{j} (-1)^j X_{n+1-j}$$

For $d > 1$, to forecast an ARIMA(p,d,q) model, the one-step ahead BLP of an ARIMA(p,d,q) model is given by
i. Using the BLP, forecast the differenced causal ARMA(p,q) series $Y_t = (1-B)^d X_t$ (i.e. find $P_n Y_{n+1}$).

Note that $P_n Y_{n+1}$ represents the BLP of a causal ARMA(p,q) process, for which we can use the BLP formula with MLE parameter estimators, innovations, etc.

ii. Add the term $-\sum_{j=1}^{d}\binom{d}{j}(-1)^j X_{n+1-j}$.

MSPE of ARIMA(p,d,q) Models:
Note that the book uses innovations results to specify the MSPE of ARIMA models.

Unit Root Tests - Stationarity Tests for ARMA(p,q) Models:
The unit root problem in time series arises when either the AR(p) or MA(q) polynomials of an ARMA(p,q) process has a root on the unit circle (indicating non-stationarity) or near the unit circle (indicating instability).
This is analogous to issues of collinearity in regression.

Dickey-Fuller Test - Stationarity of AR(p) Processes:
For an AR(p) process, the Dickey-Fuller test examines the hypotheses:

$$H_0 : \phi = 1 \ (non - stationary)$$
$$H_1 : \phi < 1 \ (stationary)$$

Recall that the asymptotics of the Yule-Walker estimator of $\hat{\phi}_p$ are given by

$$\hat{\phi}_{p,YW} \overset{d}{\approx} MN\left(\phi_p, \frac{1}{n}\sigma^2 \mathbf{\Gamma}_p^{-1}\right)$$

Note that the scalar result of $\mathbf{\Gamma}_p^{-1}$ is given by $\Gamma^{-1} = \gamma_X^{-1}(0) = \frac{1-\phi^2}{\sigma^2}$ for $\gamma(h) = \frac{\sigma^2}{1-\phi^2}\phi^{|h|}$.
Therefore,

$$Var(\phi_i) = \frac{1}{n}\sigma^2\Gamma^{-1} = \frac{1}{n}\sigma^2\left(\frac{1-\phi_i^2}{\sigma^2}\right) = \frac{1}{n}(1-\phi_i^2)$$

Consider the AR(1) process with mean μ and IID noise given by,

$$X_t - \mu = \phi_1(X_{t-1} - \mu) + Z_t, \quad Z_t \sim IID(0, \sigma^2)$$

For $|\phi_1| < 1$ (causal) and $E[X_t] = \mu$.

Then for large n,

$$\hat{\phi}_1 \overset{d}{\approx} N\left(\phi_1, \frac{1}{n}(1-\phi_1^2)\right)$$

Where $\hat{\phi}_1$ is either the Yule-Walker or MLE estimate for ϕ_1.

In practice, note that we cannot use this asymptotic result to specify the sampling distribution of the estimator because under the null hypothesis $H_0 : \phi_i = 1$, $Var(\phi_i) = 0$, which is degenerate.

Thus we will rewrite the AR(1) model using the difference operator yielding,

$$\begin{aligned} Y_t = \nabla(X_t) = X_t - X_{t-1} &\overset{substitute\ X_t}{=} (\mu + \phi_1(X_{t-1} - \mu) + Z_t) - X_{t-1} \\ &= (1-\phi_1)\mu + (\phi_1 - 1)X_{t-1} + Z_t, \quad Z_t \sim IID(0, \sigma^2) \end{aligned}$$

Thus, we can write the model as

$$Y_t = \nabla(X_t) = \phi_0^* + \phi_1^* X_{t-1} + Z_t, \quad Z_t \sim IID(0, \sigma^2)$$

For $\phi_0^* = \mu(1-\phi_1)$ and $\phi_1^* = (\phi_1 - 1)$.

Thus, we can equivalently use SLR techniques to test the hypotheses:

$$H_0 : \phi_1^* = (\phi_1 - 1) = 0 \ \ (Non - Stationary)$$
$$H_1 : \phi_1^* = (\phi_1 - 1) < 0 \ \ (Stationary)$$

Note that this is a one-sided lower-bound test.

The SLR test statistic is given by the 't-ratio',

$$\hat{\tau}_\mu := \frac{\hat{B}_1 - 0}{\sqrt{\frac{MSE}{S_{xx}}}} = \frac{\hat{\phi}_1^*}{\hat{SE}\left(\hat{\phi}_1^*\right)}$$

However, because the data is coming from a random walk, the test statistic does not have a t-distribution - its distribution will resemble the asymptotic distribution discussed in HW 3 #3.
Dickey and Fuller derived the limiting distribution as $n \to \infty$ of the 't-ratio',

$$\hat{\tau}_\mu = \frac{\hat{\phi}_1^*}{\hat{SE}\left(\hat{\phi}_1^*\right)}$$

The relevant quantiles of $\hat{\tau}_\mu$ are $q_{0.01}(\hat{\tau}_\mu) = -3.43$, $q_{0.05}(\hat{\tau}_\mu) = -2.86$, and $q_{0.10}(\hat{\tau}_\mu) = -2.57$.
Notice that the critical values for this test statistic are much smaller than the standard normal approximation ot the t-distribution - the unit root hypothesis is less likely to be rejected using the correct limiting distribution.
Note that different quantiles are used when the mean term is excluded (i.e. for regression through the origin, without an intercept) - see the book for details.

To compute this result, we
i. Difference X_t to produce a stationary time series, $Y_t = (1 - B)X_t$.
ii. Perform regression on the differenced time series, $Y_t = \nabla X_t \sim \{1, X_{t-1}\}$.
iii. Extract the slope $\hat{\phi}_1^*$ and $\hat{SE}\left(\hat{\phi}_1^*\right)$.
iv. Use the desired critical value from the $\hat{\tau}_\mu$ distribution (not the provided t-test) to assess stationarity.

Note that for the Dickey-Fuller test,
i. Rejecting H_0 indicates stationarity (i.e., that we do not have a unit root).
ii. Failing to reject H_0 indicates non-stationarity (i.e., the existence of a unit root).

For an AR(p) process, ϕ_1^* can be estimated as the coefficient of X_{t-1}, in the OLS regression of $Y_t = \nabla X_t$ onto $\{1, X_{t-1}, \nabla X_{t-1}, \ldots, \nabla X_{t-p+1}\}$.
The same procedure, limiting distribution, critical values, etc. above will be used to test the statistic ϕ_1.

KPSS Test:
Recall that when performing inference tests, we typically construct the test (and test statistic) under the assumption of the null and assess rejection of the null - constructing a test in which we would like to retain the null hypothesis (like the Dickey-Fuller test) generally results in lower power.

The KPSS test examines the hypotheses:

$$H_0 : Stationary$$
$$H_1 : Non - Stationary$$

Define

$$\hat{\eta} := \frac{1}{n^2} \sum_{t=1}^{n} \frac{S_t^2}{\hat{\sigma}_{LR}^2(k)}$$

Where
1. $S_t = \sum_{i=1}^{t} e_i = \sum_{i=1}^{t} (X_i - \bar{X})$

   Note that this is a sequence of partial sums, and that $S_n = 0$.

2. $\hat{\sigma}_{lr}^2(k)$ is an estimator of the long-run variance ν.

   Recall that the long-run variance is given by

$$\nu = \sum_{h=-\infty}^{\infty} \gamma(h)$$

   Estimating the long-run variance is a complex topic of research in time series, because, in principle, it is necessary to estimate $\gamma(h)$ for infinite values.
   i. Using the estimator $2\sum_{i=1}^{n} \hat{\gamma}(i) + \hat{\gamma}(0)$ results in an over-estimate of ν.
   ii. As a correction, we place less weight on quantities associated with large values of n using a kernel smoother with an adjustable bandwidth,

$$\hat{\sigma}_{lr}^2 = \hat{\gamma}(0) + 2\sum_{i=1}^{n} K_i(*)\hat{\gamma}(i)$$

iii. As a result, we yield

$$\hat{\sigma}_{lr}^2 \xrightarrow{p} \sigma_{lr}^2$$

Under the null,

$$\hat{\eta} \xrightarrow{d} \int_0^1 (W(r) - rW(1))^2$$

Where $W(r) - rW(1)$ is a standard Brownian bridge.

Note that for the KPSS test,
i. Rejecting H_0 indicates non-stationarity (i.e., the existence of a unit root).
ii. Failing to reject H_0 indicates stationarity (i.e., that we do not have a unit root).

Seasonal ARIMA (SARIMA) Models:
SARIMA models (seasonal autoregressive iterative moving average models) constitute a more general set of ARIMA models which include a seasonality component.

Recall that the lag-d difference operator ∇_d is defined as

$$\nabla_d X_t = (1 - B^d)X_t = X_t - X_{t-s}$$

We can difference a time series $\{X_t\}$ at lag s to eliminate a seasonal component of period s (e.g. a period of a day, week, month, etc.).

$$\nabla_s X_t = (1 - B^s)X_t = X_t - X_{t-s}$$

Note that $\nabla_s = (1 - B^s) \neq (1 - B)^s$.

If d and D are non-negative integers, then $\{X_t\}$ is a seasonal ARIMA (SARIMA), denoted $ARIMA(p, d, q) \times (P, D, Q)_s$ with period s if the differenced series

$$Y_t = (1 - B)^d (1 - B^s)^D X_t$$

Is a causal ARMA(p,q) process defined by

$$\Phi(B)\Phi_S(B^s)Y_t = \Theta(B)\Theta_S(B^s)Z_t, \quad Z_t \sim WN(0, \sigma^2)$$

Where
i. $\Phi(u) = 1 - \phi_1 u - \cdots - \phi_p u^p$
ii. $\Phi_S(u) = 1 - \Phi_1(u) - \cdots - \Phi_P u^P$
iii. $\Theta(u) = 1 + \theta_1 u + \cdots + \theta_q u^q$
iv. $\Theta_S(u) = 1 + \Theta_1 u + \cdots + \Theta_Q u^Q$

The model is causal iff $\Phi(u) \neq 0$ and $\Phi(u) \neq 0$ for $|u| \leq 1$.
Note that
i. The $(1 - B)^d$ term de-trends and facilitates stationarity.
ii. The $(1 - B^s)^D$ term de-seasonalizes and facilitates stationarity.

Regression with ARMA Errors:
Define the scalar model of regression with ARMA errors as

$$Y_t = \mathbf{X}^T \beta + W_t, \quad W_t \sim ARMA(p, q), \quad t = 1, 2, \ldots, n$$

Where $\mathbf{X}$ and β are vectors.

Define the matrix model of regression with ARMA errors as

$$\mathbf{Y} = \mathbf{X}_{n \times n} \beta + \mathbf{W}$$

Where $\mathbf{X}_{n \times n}$ is a design matrix, and all other terms are vectors.

Recall the classical additive time series decomposition.

$$Y_t = (m_t + s_t) + W_t = \mathbf{X}^T \beta + W_t, \quad W_t \sim ARMA(p, q), \quad t = 1, 2, \ldots, n$$

The trend and seasonality can be represented by $\mathbf{X}^T \beta$, with errors distributed ARMA(p,q).

Thus regression offers an alternative to controlling for trend and seasonality as opposed to ARIMA or SARIMA. Regression gives you the option to include additional exogenous variables (like SARIMAX), or simply fit a trend and estimate parameters all at once.

Note that
i. Exogenous structure assumes that X_t is independent of the noise.
ii. Endogenous structures assumes that X_t is not independent of the noise.

Ordinary Least Squares:
The ordinary least squares objective is given by

$$Q(\beta) = (\mathbf{Y} - \mathbf{X}_{n\times n}\beta)^T(\mathbf{Y} - \mathbf{X}_{n\times n}\beta) = \sum_{t=1}^{n}(Y_t - \mathbf{x}_t^T\beta)^2$$

The estimator $\hat{\beta}_{OLS}$ is given by

$$\hat{\beta}_{OLS} = (\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\mathbf{Y}$$

i. $E\left[\hat{\beta}_{OLS}\right] = \beta$, assuming X_t's are fixed.
ii. $Cov\left(\hat{\beta}_{OLS}\right) = (\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\mathbf{\Gamma}\mathbf{X}_{n\times n}(\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}$

$$Cov\left(\hat{\beta}_{OLS}\right) = Cov((\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\mathbf{Y})$$
$$= (\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\mathbf{\Gamma}\left[(\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\right]^T$$
$$= (\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\mathbf{\Gamma}\mathbf{X}_{n\times n}(\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}$$

Recall that in the IID case, this reduces to

$$(\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\sigma^2\mathbf{I}_{n\times n}\mathbf{X}_{n\times n}(\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1} = \sigma^2(\mathbf{X}_{n\times n}^T\mathbf{X}_{n\times n})^{-1}$$

However, given serial dependency, we cannot reduce our expression any further.

Generalized (Weighted) Least Squares:
Note that the lack of an IID structure suggests that we should be using generalized (weighted) least squares as opposed to ordinary least squares, in order to incorporate the autoregressive covariance structure into our optimization procedure.
Recall that in generalized least squares each observation has a different covariance structure, which is preferred for non-IID errors.

The generalized (weighted) least squares relationship is given by

$$Q_W(\beta) = (\mathbf{Y} - \mathbf{X}_{n\times n}\beta)^T\mathbf{\Gamma}^{-1}(\mathbf{Y} - \mathbf{X}_{n\times n}\beta)$$

Recall that when minimizing the negative log-likelihood of a multivariate normal distribution, an inverse covariance matrix appears in the procedure - this resembles minimizing the weighted least squares.
The solution is given by

$$\hat{\beta}_{GLS} = (\mathbf{X}_{n\times n}^T\mathbf{\Gamma}^{-1}\mathbf{X}_{n\times n})^{-1}\mathbf{X}_{n\times n}^T\mathbf{\Gamma}^{-1}\mathbf{Y}$$

Proof:

Derive the weighted least squares equation.

$$\hat{\beta}_{GLS} = \left(\mathbf{X}_{p\times n}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_{n\times p}\right)^{-1}\mathbf{X}_{p\times n}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{Y}_{n\times 1}$$

If we define the vector,

$$\mathbf{b}_{p\times 1} = \begin{bmatrix} b_0 & b_1 & b_2 & \cdots & b_{p-1}\end{bmatrix}^T$$

The objective function can be expressed in quadratic form as

$$Q_W\left(\mathbf{b}_{p\times 1}\right)_{1\times 1} = \sum_{i=1}^{n} w_i(y_i - (b_0 + b_1x_{i1} + b_2x_{i2} + \cdots + b_{p-1}x_{i,p-1}))^2 = \left(\mathbf{Y}_{n\times 1} - \mathbf{X}_{n\times p}\mathbf{b}_{p\times 1}\right)^T\mathbf{\Gamma}_{n\times n}^{-1}\left(\mathbf{Y}_{n\times 1} - \mathbf{X}_{n\times p}\mathbf{b}_{p\times 1}\right)$$

The p-dimensional optimization problem is

$$\min_{\mathbf{b}} Q_W\left(\mathbf{b}_{p\times 1}\right)$$

We proceed with the standard minimization procedure.
Recall that matrices are not commutative under multiplication, so maintaining the order of terms in matrix products is critical.
Recall that the transpose of a diagonal matrix is the same diagonal matrix.

$$\begin{aligned} Q_W\left(\mathbf{b}_{p\times 1}\right)_{1\times 1} &= \left(\mathbf{Y}_{n\times 1} - \mathbf{X}_{n\times p}\mathbf{b}_{p\times 1}\right)^T \mathbf{\Gamma}_{n\times n}^{-1}\left(\mathbf{Y}_{n\times 1} - \mathbf{X}_{n\times p}\mathbf{b}_{p\times 1}\right) \\ = \mathbf{Y}_{1\times n}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{Y}_{n\times 1} - \mathbf{Y}_{1\times n}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_{n\times p}\mathbf{b}_{p\times 1} - \left(\mathbf{X}_{n\times p}\mathbf{b}_{p\times 1}\right)^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{Y}_{n\times 1} + \left(\mathbf{X}_{n\times p}\mathbf{b}_{p\times 1}\right)^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_{n\times p}\mathbf{b}_{p\times 1} \\ = \mathbf{Y}_{1\times n}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{Y}_{n\times 1} - \mathbf{Y}_{1\times n}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_{n\times p}\mathbf{b}_{p\times 1} - \mathbf{b}_{p\times 1}^T\mathbf{X}_{n\times p}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{Y}_{n\times 1} + \mathbf{b}_{p\times 1}^T\mathbf{X}_{n\times p}^T\mathbf{\Gamma}_{n\times n}^{-1}\mathbf{X}_{n\times p}\mathbf{b}_{p\times 1} \end{aligned}$$

Note the following matrix derivatives.

$$\frac{\partial\ \mathbf{Y}^T\mathbf{\Gamma}^{-1}\mathbf{X}\mathbf{b}}{\partial\ \mathbf{b}} = \mathbf{Y}^T\mathbf{\Gamma}^{-1}\mathbf{X}$$

$$\frac{\partial\ \mathbf{b}^T\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{Y}}{\partial\ \mathbf{b}} = \left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{Y}\right)^T = \mathbf{Y}^T\mathbf{\Gamma}^{-1}\mathbf{X}$$

$$\frac{\partial\ \mathbf{b}^T\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\mathbf{b}}{\partial\ \mathbf{b}} = \mathbf{b}^T\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right) + \mathbf{b}^T\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)^T = 2\mathbf{b}^T\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)^T$$

$$\nabla Q_W\left(\mathbf{b}\right) = \frac{\partial\ Q_W\left(\mathbf{b}\right)}{\partial\ \mathbf{b}} = 0 - 2\mathbf{Y}^T\mathbf{\Gamma}^{-1}\mathbf{X} + 2\mathbf{b}^T\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)^T \overset{set}{=} 0$$

Solving for $\mathbf{b}$, we have

$$\mathbf{b}^T\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)^T = \mathbf{Y}^T\mathbf{\Gamma}^{-1}\mathbf{X}$$

Taking the transpose of both sides of the equation, we have the normal equations.

$$\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)\mathbf{b} = \mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{Y}$$

$$\therefore \mathbf{b} = \left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{Y}$$

We proceed with the second derivative test.

$$\frac{\partial^2\ Q_W\left(\mathbf{b}\right)}{\partial\ \mathbf{b}^2} = \frac{\partial}{\partial\mathbf{b}}\left[-2\mathbf{Y}^T\mathbf{\Gamma}^{-1}\mathbf{X} + 2\mathbf{b}^T\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)^T\right] = 0 + 2\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right) = 2\left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)$$

Note that $\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}$ is positive definite.
Therefore, $Q_W\left(\mathbf{b}\right)$ achieves its minimum at $\hat{\beta}_{GLS} = \left(\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{X}\right)^{-1}\mathbf{X}^T\mathbf{\Gamma}^{-1}\mathbf{Y}$.

"Gauss-Markov Theorem" for Regression with ARMA Errors:
Note that for a non-zero vector $\mathbf{c} \in \mathbb{R}^d$,

$$Var\left(\mathbf{c}^T\hat{\beta}_{GLS}\right) \leq Var\left(\mathbf{c}^T\hat{\beta}_{OLS}\right)$$

This resembles the Gauss-Markov theorem for stationary (ARMA) errors (as opposed to iid errors).

Estimation of $\beta's, \phi's, \theta's, \sigma^2$:
Note that the generalized least squares result is true only if θ's and ϕ's are known (because of the inclusion of the $\mathbf{\Gamma}^{-1}$ matrix), which is not realistic because only the X's and Y's are observable.

A method for estimating the parameters is as follows.
i. Compute $\hat{\beta}_{OLS}$ and estimate the residuals

$$Y_t - \mathbf{X}^T\hat{\beta}_{OLS}$$

ii. Fit the ARMA(p,q) model on the residuals by MLE, which will yield $\hat{\phi}$'s and $\hat{\theta}$'s.
iii. For the fitted ARMA(p,q), compute $\hat{\mathbf{\Gamma}}$, and use this to compute $\hat{\beta}_{GLS}$.

iv. Fit residuals $Y_t - \mathbf{X}_t^T \hat{\beta}_{GLS}$ and return to step (ii).
v. Terminate algorithm when the estimates stabilize.

Examples:

Example: ARIMA(p,d,q)

$\{X_t\}$ be an ARIMA(1,1,0) process for some $-1 < \phi < 1$,

$$(1 - \phi B)(1 - B)X_t = Z_t, \quad Z_t \sim WN(0, \sigma^2)$$

Notice that we can write X_t as a telescoping sum of the differenced time series for $d = 1$,

$$\begin{aligned} X_t &= X_0 + (X_1 - X_0) + (X_2 - X_1) + \cdots + (X_t - X_{t-1}) \\ &= X_0 + \sum_{j=1}^{t} (1 - B)X_j, \quad t \geq 1 \\ &= X_0 + \sum_{j=1}^{t} Y_j, \quad t \geq 1 \end{aligned}$$

Where Y_t is a causal AR(1) process,

$$Y_t = (1 - B)X_t = \sum_{j=0}^{\infty} \phi^j Z_{t-j}$$

Thus, applying the difference operator yielded a stationary and causal time series which can be expressed as a linear process.

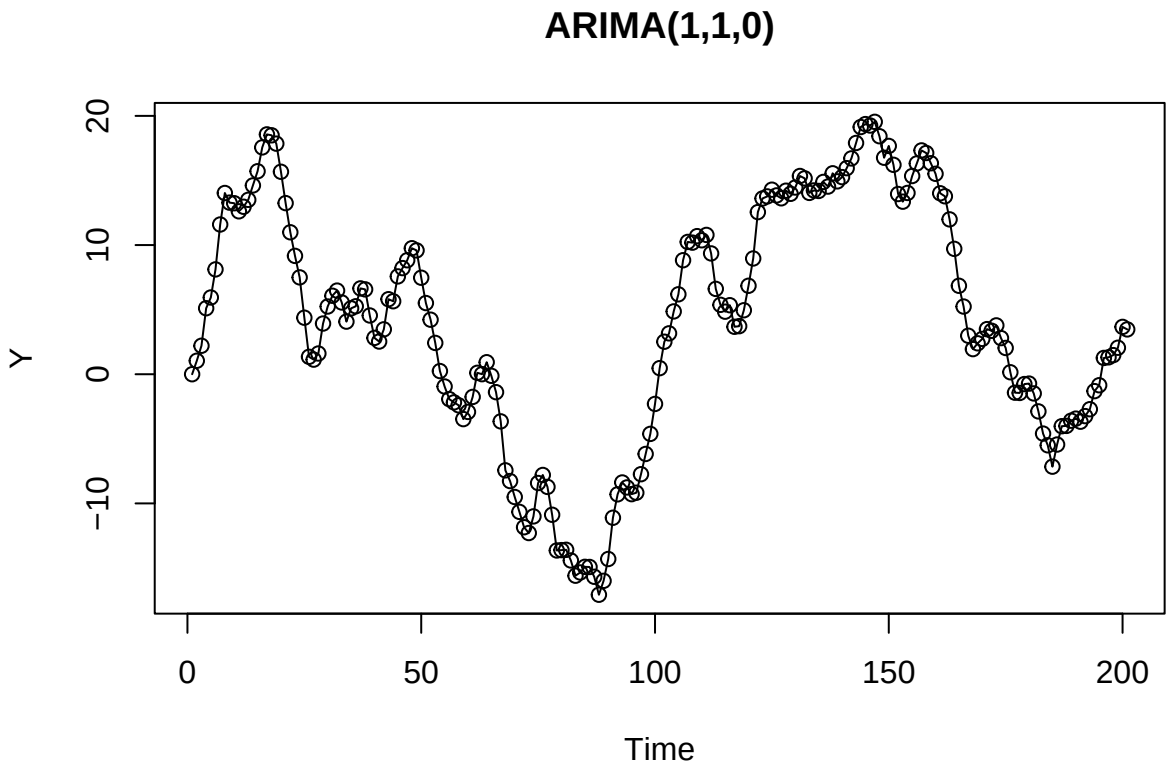
Example: Unit Root Tests on Simulations

```
library(tseries)
```

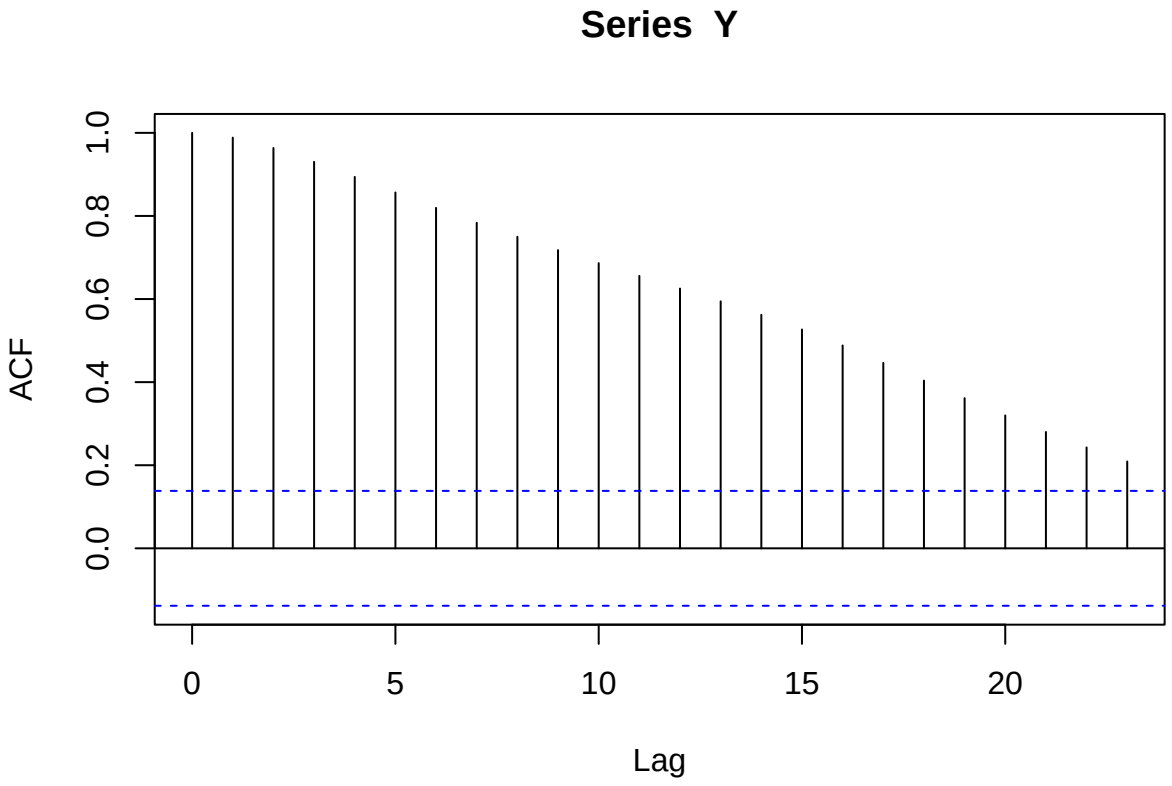
```
##
## Attaching package: 'tseries'
```

```
## The following object is masked from 'package:itsmr':
##
##     arma
```

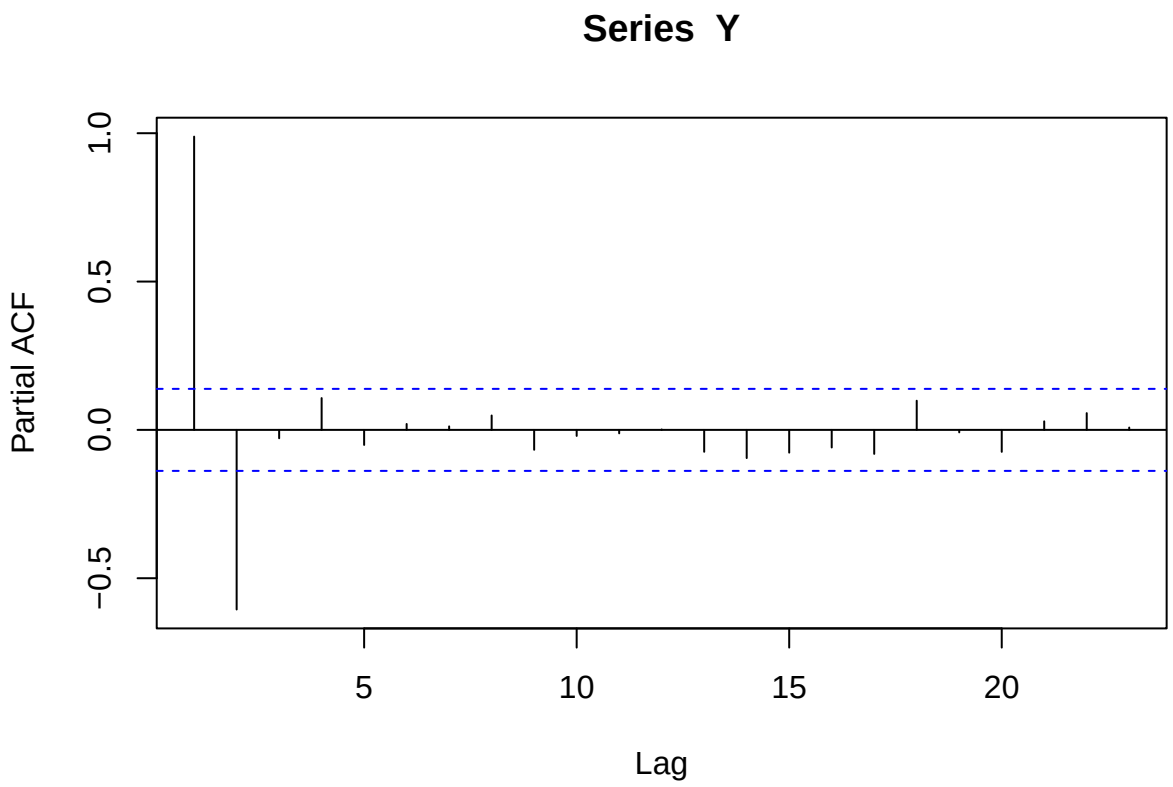
```
# Non-stationary AR(1) model
Y = arima.sim(model=list(order = c(1,1,0), ar = 0.7), n = 200)
plot(Y,type="o",main="ARIMA(1,1,0)")
```



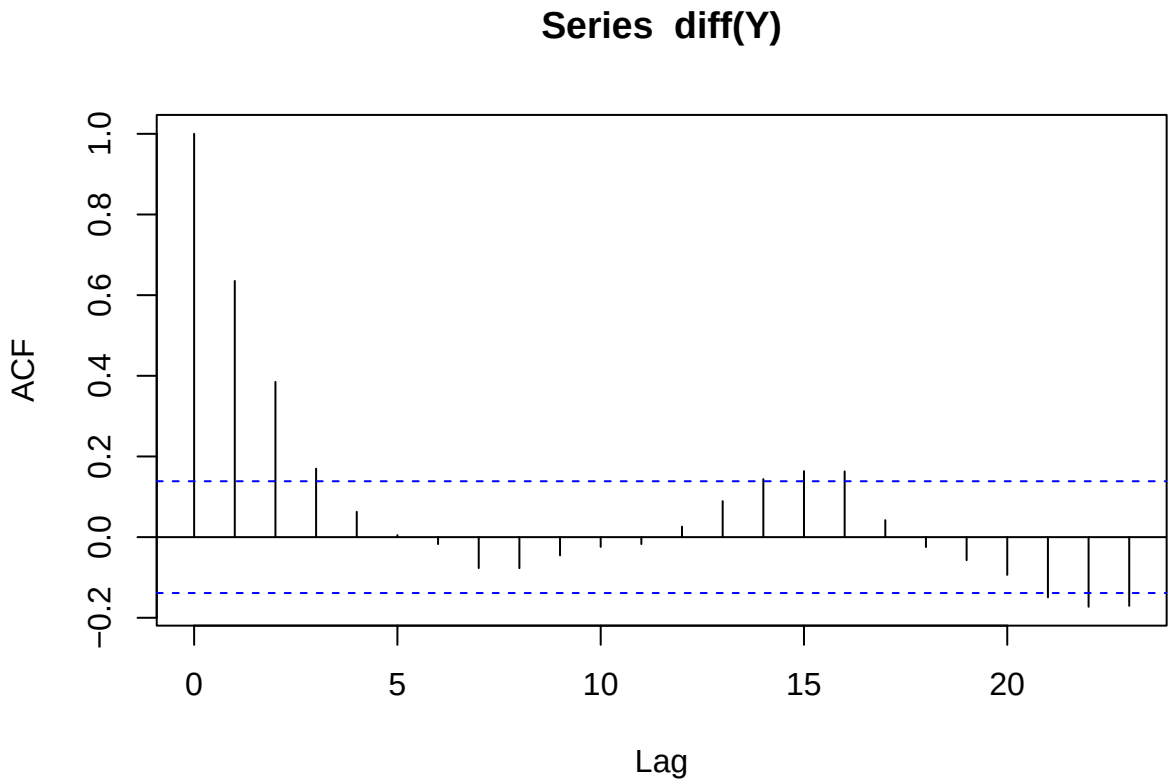

```
# The model is not stationary so it is difficult to interpret these plots
# ACF shows a heavy dependence structure on previous days (serial correlation) - ARIMA appropriate
# PACF looks like WN because the model is not stationary (and no MA component)
acf(Y)
```



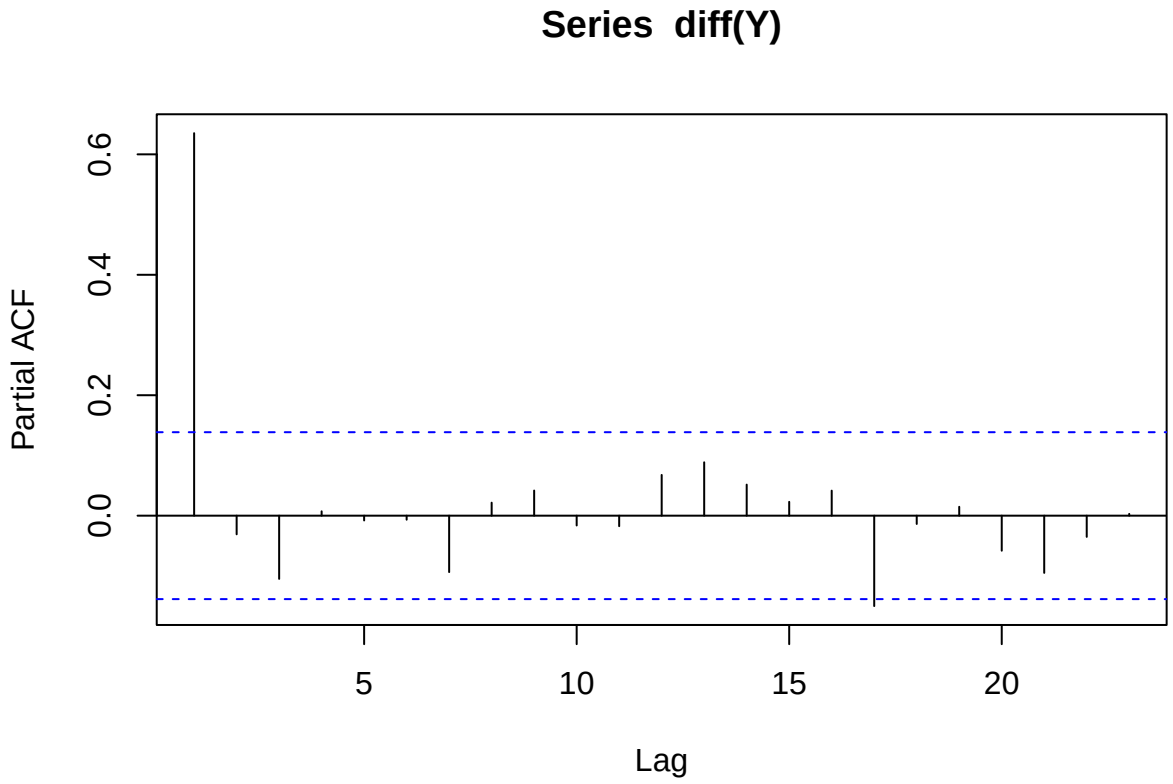
```
pacf(Y)
```




```
# diff() - lose a data point
# ACF - more manageable - AR ACF's decrease exponentially
# PACF - looks like AR(1) - spike at p=1
acf(diff(Y))
```


Series diff(Y)
ACF
1.0
0.8
0.6
0.4
0.2
0.0
−0.2
0
5
10
15
20
Lag

```
pacf(diff(Y))
```


Series diff(Y)
Partial ACF
0.6
0.4
0.2
0.0
5
10
15
20
Lag

```r
# Dickey-Fuller Test
adf.test(Y) # non-stationary
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  Y
## Dickey-Fuller = -1.9583, Lag order = 5, p-value = 0.5935
## alternative hypothesis: stationary
```

```r
adf.test(diff(Y)) # stationary
```

```
## Warning in adf.test(diff(Y)): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(Y)
## Dickey-Fuller = -5.2887, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```r
# Ho: Not Stationary, HA: Stationary

# KPSS Test
kpss.test(Y) # non-stationary
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  Y
## KPSS Level = 0.41871, Truncation lag parameter = 4, p-value = 0.06909
```

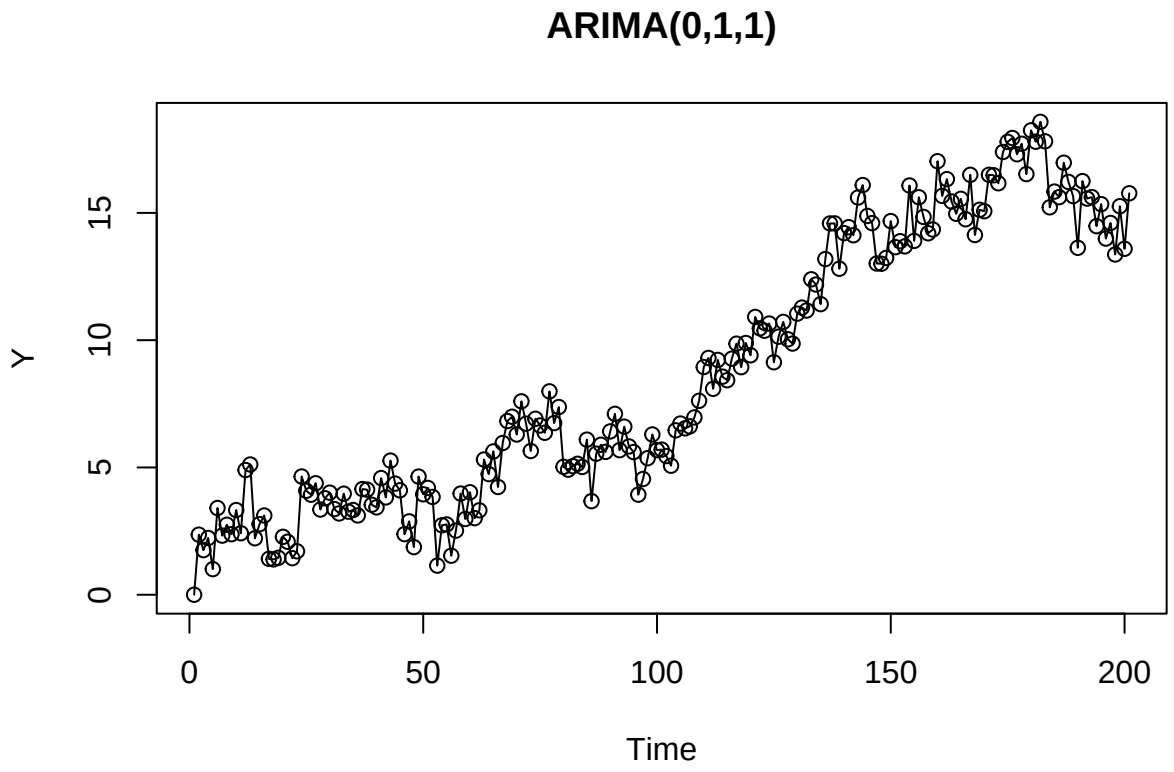
```r
kpss.test(diff(Y)) # stationary
```

```
## Warning in kpss.test(diff(Y)): p-value greater than printed p-value
```

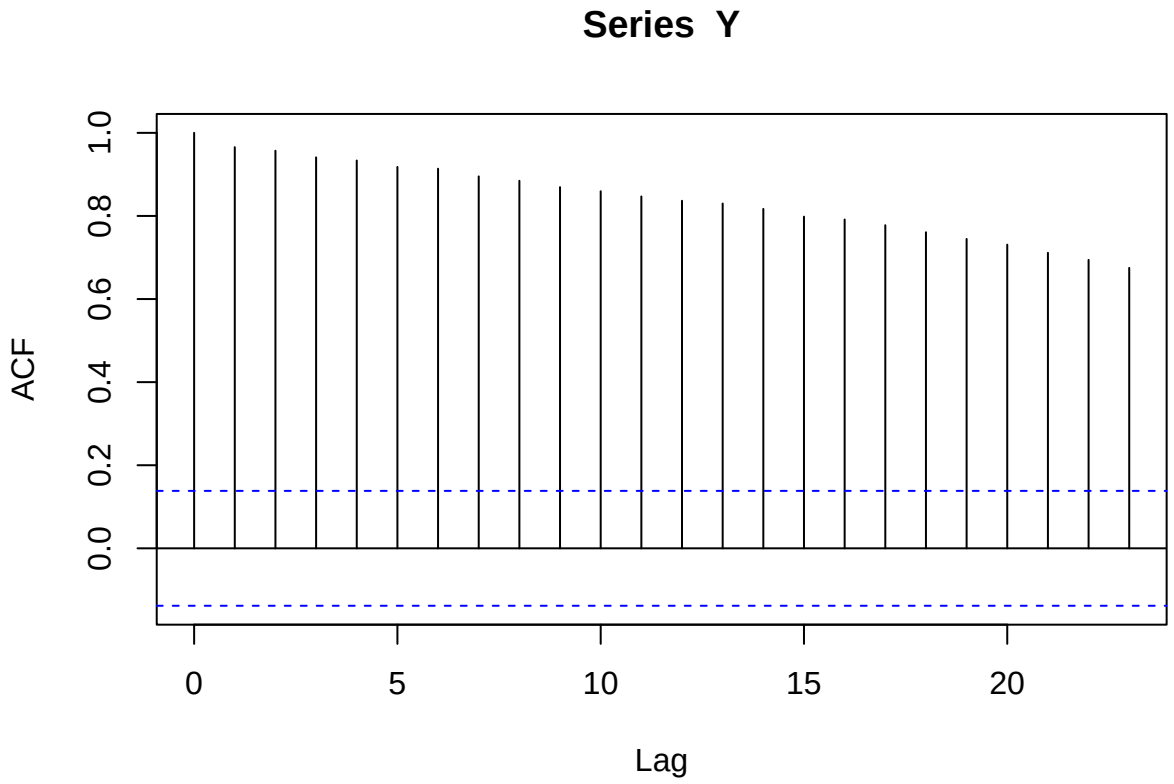
```
##
##  KPSS Test for Level Stationarity
##
## data:  diff(Y)
## KPSS Level = 0.093065, Truncation lag parameter = 4, p-value = 0.1
```

```r
# Ho: Stationary, HA: Not Stationary
```

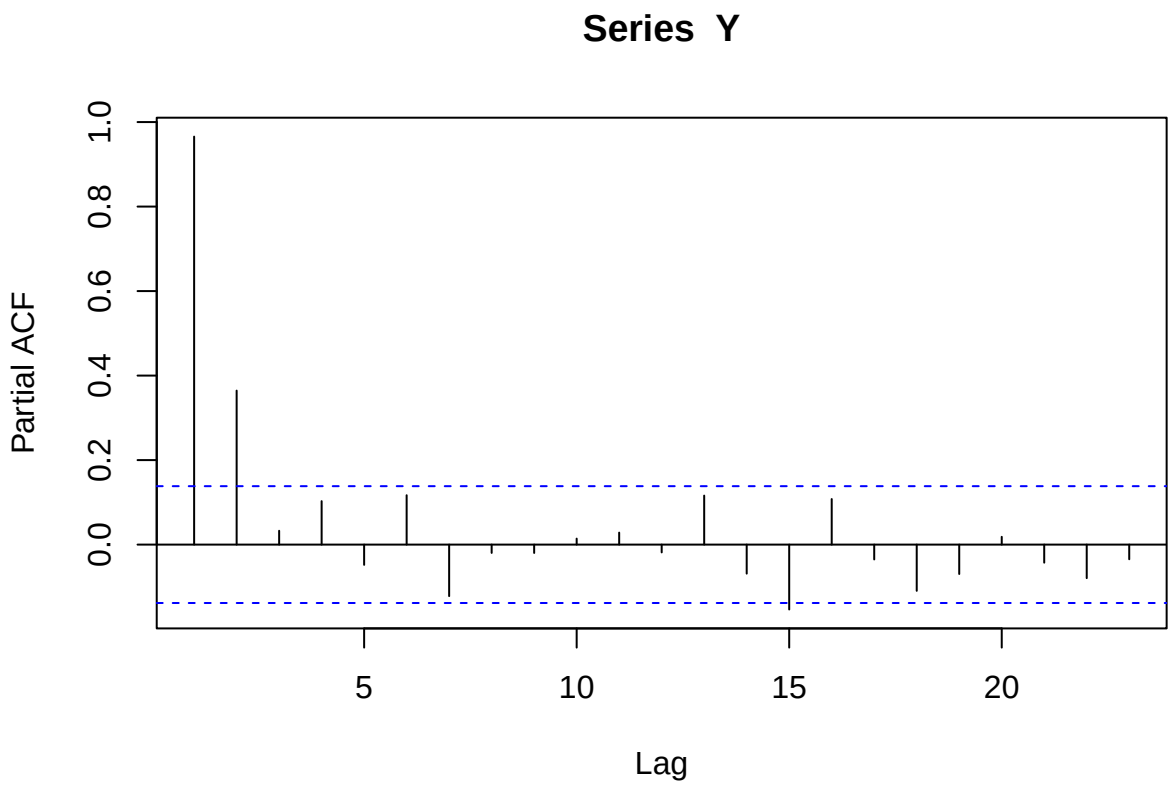
```r
library(tseries)

# Non-stationary MA(1) model
Y = arima.sim(model=list(order = c(0,1,1), ma = -.6), n = 200)
plot(Y,type="o",main="ARIMA(0,1,1)")
```



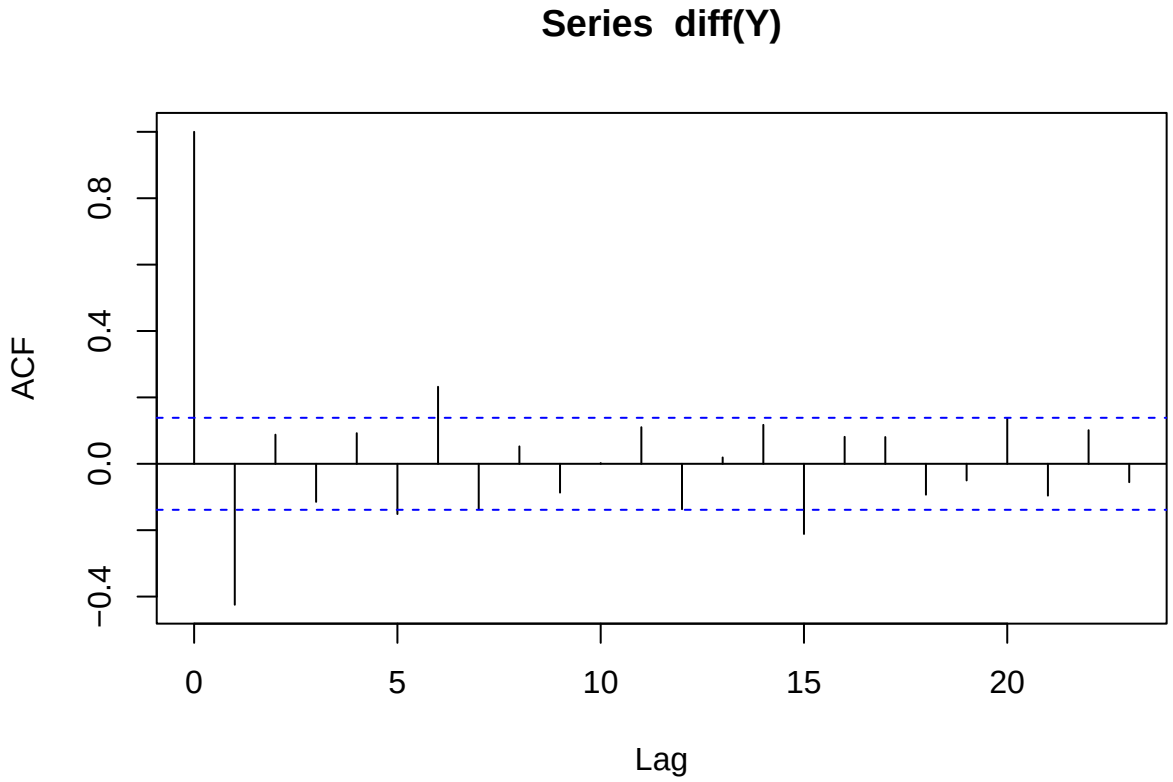

```
# The model is not stationary so it is difficult to interpret these plots
# ACF decreasing slowly
# PACF looks like AR(2)
acf(Y)
```



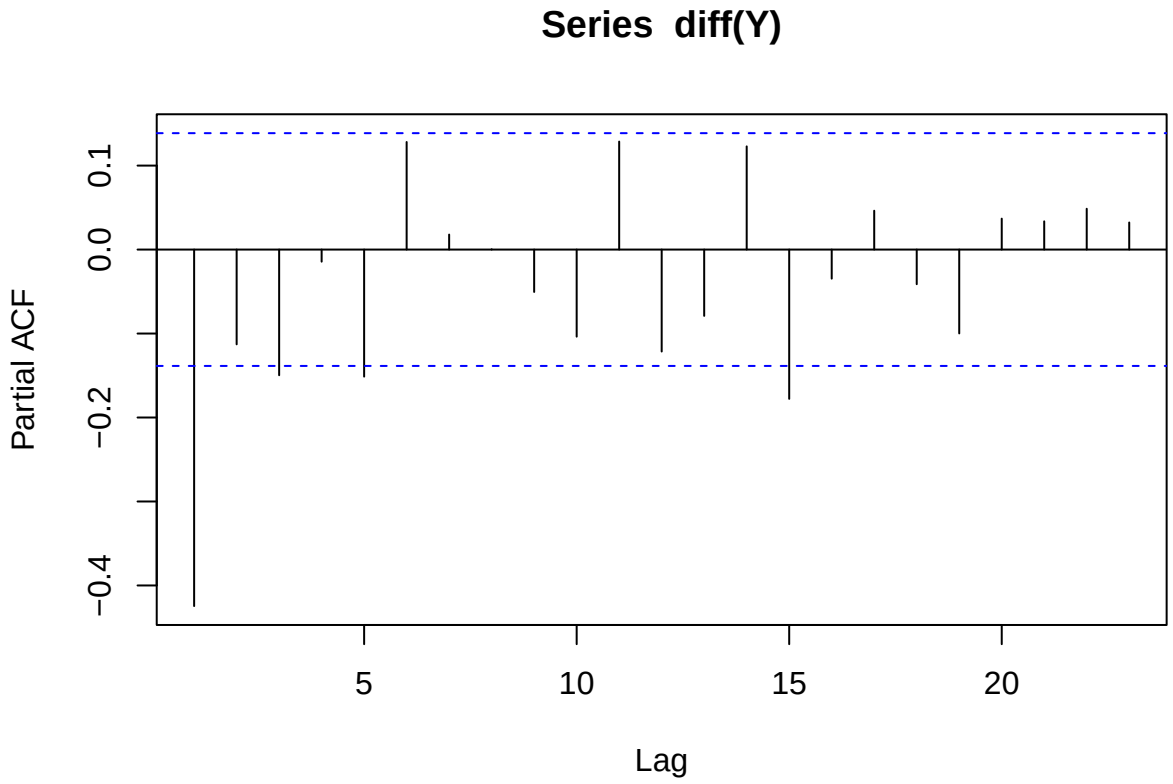
```
pacf(Y)
```




```
# diff() - lose a data point
# ACF - clearly MA(1)
# PACF - could be AR(1)
acf(diff(Y))
```


Series diff(Y)
ACF
0.8
0.4
0.0
−0.4
0
5
10
15
20
Lag

```
pacf(diff(Y))
```


Series diff(Y)
Partial ACF
0.1
0.0
−0.2
−0.4
5
10
15
20
Lag

```
# Dickey-Fuller Test
adf.test(Y) # non-stationary
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  Y
## Dickey-Fuller = -1.8843, Lag order = 5, p-value = 0.6246
## alternative hypothesis: stationary
```

```
adf.test(diff(Y)) # stationary
```

```
## Warning in adf.test(diff(Y)): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  diff(Y)
## Dickey-Fuller = -6.406, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

```
# Ho: Not Stationary, HA: Stationary

# KPSS Test
kpss.test(Y) # non-stationary
```

```
## Warning in kpss.test(Y): p-value smaller than printed p-value
```

```
##
##  KPSS Test for Level Stationarity
##
## data:  Y
## KPSS Level = 3.88, Truncation lag parameter = 4, p-value = 0.01
```

```
kpss.test(diff(Y)) # stationary
```

```
## Warning in kpss.test(diff(Y)): p-value greater than printed p-value
```

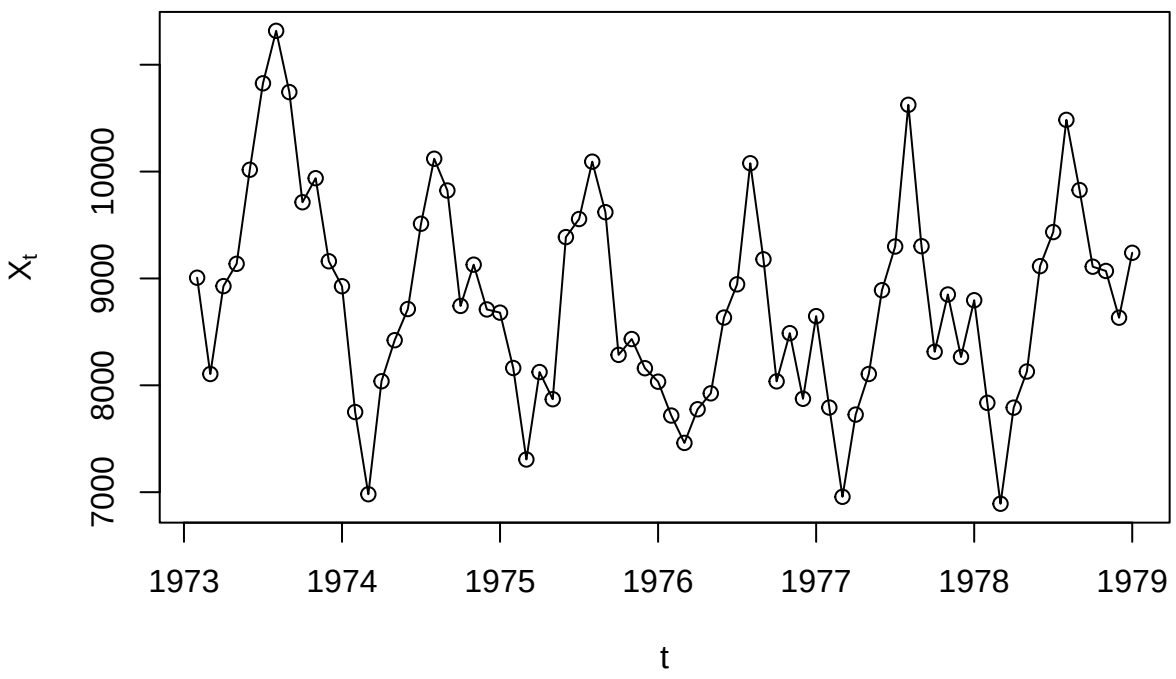
```
##
##  KPSS Test for Level Stationarity
##
## data:  diff(Y)
## KPSS Level = 0.041554, Truncation lag parameter = 4, p-value = 0.1
```

```
# Ho: Stationary, HA: Not Stationary
```

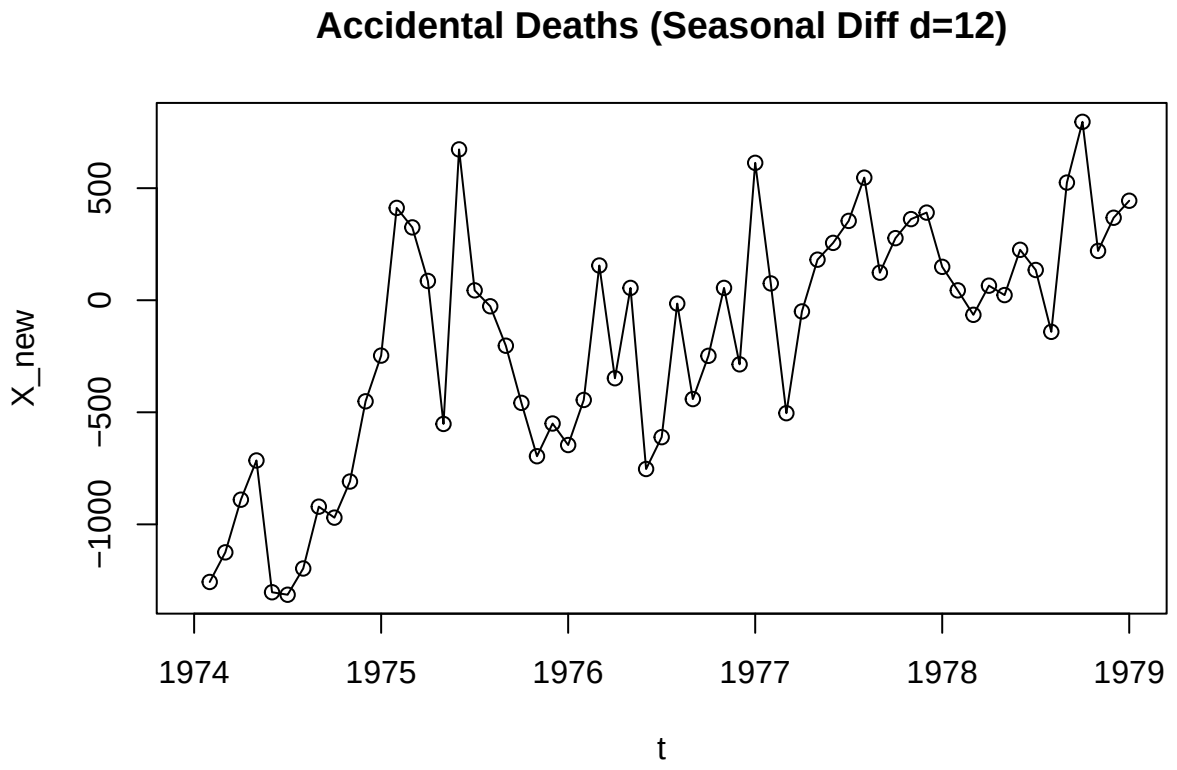
Example: SARIMA - Accidental Deaths Dataset

We will consider different SARIMA models:
i. $(1-B^{12})X_t$
ii. $(1-B)^1(1-B^{12})X_t$
Then we will model the resulting time series as an ARMA(p,q) process.

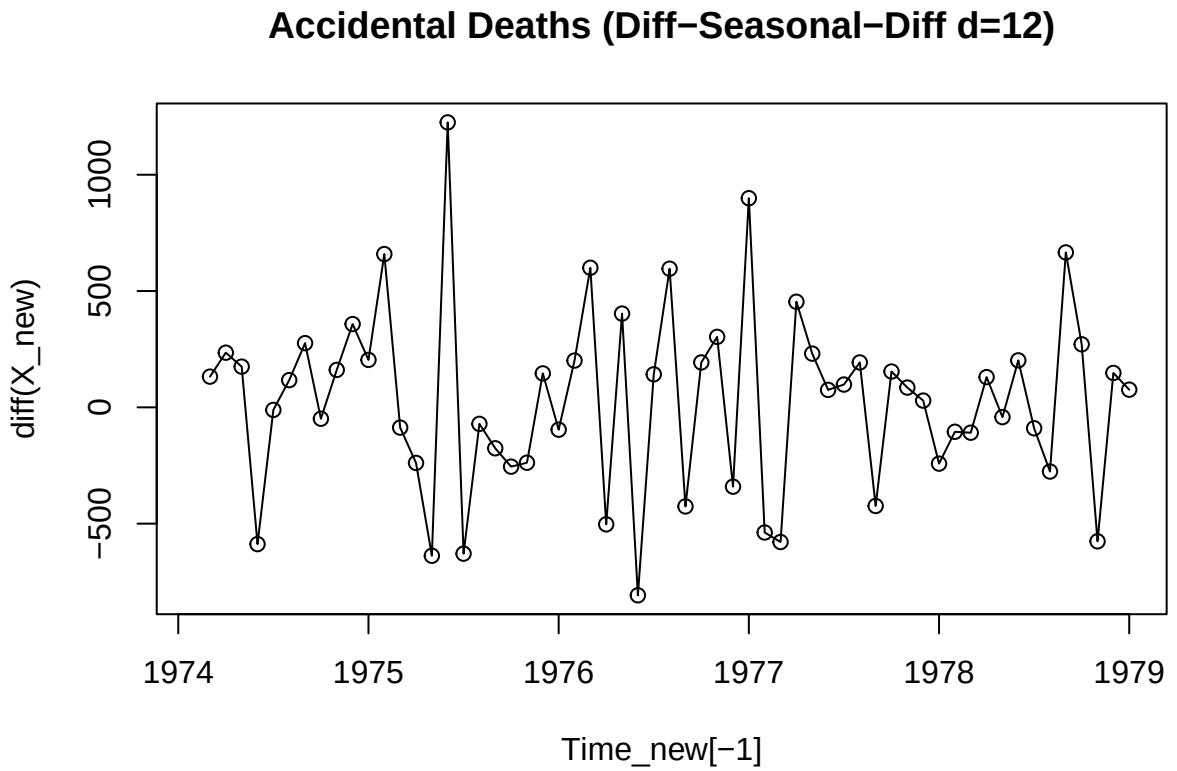
```
# read-in data
AD = read.csv("Accidental_Deaths.csv")

# plot data
# There is some cyclical (seasonal) component to this data
# Before, we fit a parametric curve to the data, now we consider SARIMA
plot(AD$time,AD$Y,type="o", main="Accidental Deaths (Raw)",xlab=expression(t),ylab=expression(X[t]))
```



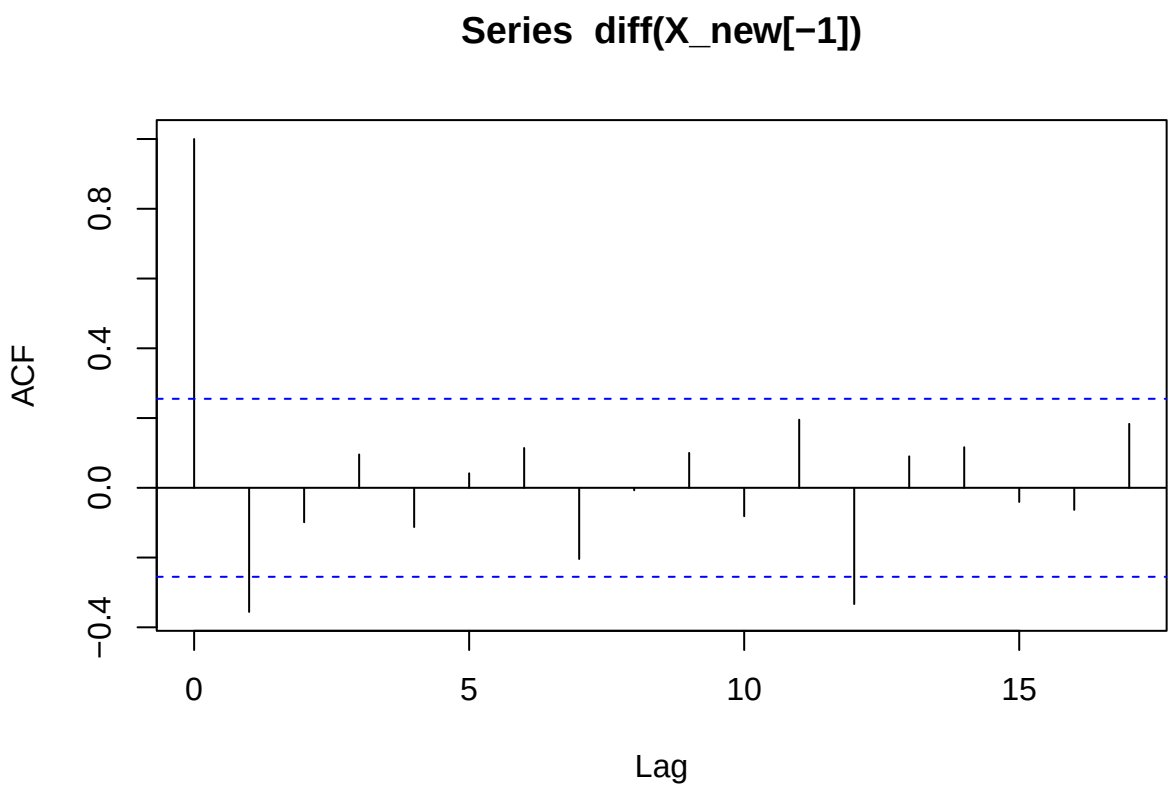

```r
# Apply seasonal difference manually
# Use diff(lag, difference) parameters in practice
n = nrow(AD)
d = 12
X = AD$Y
Time = AD$time
X_new = NULL
Time_new = NULL
j = 1

for (i in 1:n) {
  if (i-d >= 0) {
    X_new[j] = X[i] - X[i-d]
    Time_new[j] = Time[i]
    j = j + 1
  }
}

# Plot seasonal differenced time series
# Still not fully stationary ARMA(p,q)
plot(Time_new,X_new,type="o",main="Accidental Deaths (Seasonal Diff d=12)",
     xlab=expression(t))
```



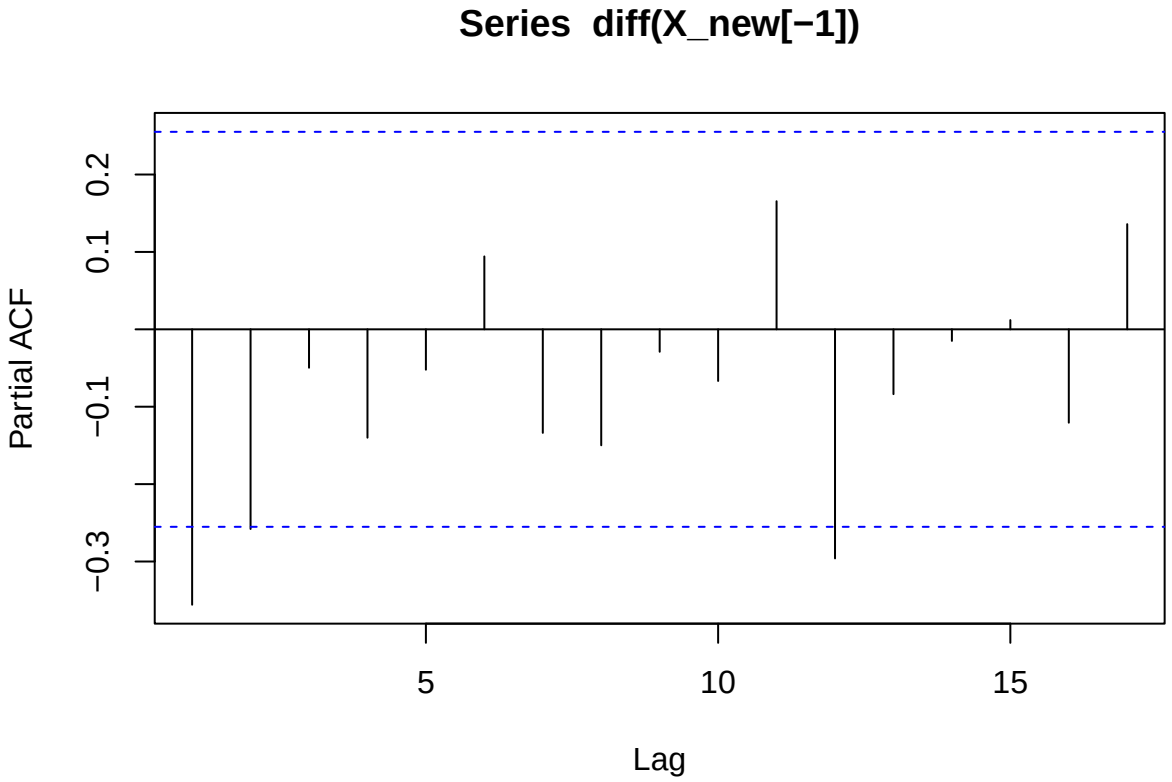

```
# Plot differenced-seasonal-differenced time series
# Now stationary ARMA(p,q)
plot(Time_new[-1],diff(X_new),type="o",main="Accidental Deaths (Diff-Seasonal-Diff d=12)")
```



```
# AR(2), MA not particularly strong
acf(diff(X_new[-1]))
```




```
pacf(diff(X_new[-1]))
```


Series diff(X_new[−1])
Partial ACF
0.2
0.1
−0.1
−0.3
5
10
15
Lag

FINANCIAL TIME SERIES MODELS - ARCH/GARCH MODELS

Financial Time Series Models:
Let P_t be the closing price of an asset on trading day t.
We note that the time series of log-prices approximate a random walk, $X_t = \log(P_t) \approx RW$.
In particular, general financial models specify that prices follow a log-normal geometric random walk.

Let the log-return Z_t for day t be defined as $Z_t = X_t - X_{t-1} = \log(P_t) - \log(P_{t-1}) = \log\left(\frac{P_t}{P_{t-1}}\right)$.
Log-returns Z_t resemble WN, but are not IID.

Recall that in earlier discussions, we have noted that a residual WN error structure indicates that the constructed model is 'complete', e.g. suitable for forecasting, etc.
While log-returns behave similarly to WN, they are not IID, meaning that there is a dependence structure that we can exploit.

ARCH(p) Models:
The ARCH(p) model is the auto-regressive conditional heteroskedasticity model - a model that has an AR (autoregressive) structure defined on the conditional variance (not the conditional mean).

We say that the (WN) error terms of a time series, $\{Z_t\}$ have an ARCH(p) structure if

$$Z_t = \sqrt{h_t} \cdot e_t, \quad e_t \overset{iid}{\sim} N(0,1)$$

Where h_t is related to past values of Z_t^2 (i.e. autoregressive) through the relation below

$$h_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i Z_{t-i}^2, \quad p \in \mathbb{Z} > 0$$

For $h_t > 0$, $\alpha_0 > 0$, $\alpha_i \geq 0$.

This model is analogous to an AR(p) process in the variance structure through h_t - the tth value of h_t depends on the p previous Z_t^2 cases.
Note that if $E[Z_t] = 0$, $Var(Z_t) = E[Z_t^2]$, meaning $E[Z_t^2]$ describes the variance.
Thus h_t specifies the variance structure of the time series.

Note that the quantities h_t, α_0, α_i, must be non-negative because this is a multiplicative model in the volatility (a positive quantity).

Because we are specifying an AR structure in the conditional variance (the variance depends on the previous variance), the standard normal error term e_t is multiplicative, not additive.
The time series Z_t can still take negative values because h_t (positive) it is multiplied by a standard normal e_t.

ARCH(1) Model:
The ARCH(1) model is given by

$$Z_t = \sqrt{h_t} \cdot e_t, \quad e_t \overset{iid}{\sim} N(0,1)$$

Where

$$h_t = \alpha_0 + \alpha_1 Z_{t-1}^2$$

For $h_t > 0$, $\alpha_0 > 0$, $\alpha_1 \geq 0$.

Note that if $\alpha_1 = 0$ there is no recursion depending on the previous case, and the error structure reduces to normal.

Deriving the Linear Process Solution of the ARCH(1) Model:
Recall that the linear process solution of the AR(1) model is given by

$$X_t - \phi X_{t-1} = Z_t$$
$$(1 - \phi B) X_t = Z_t$$
$$X_t = \sum_{j=0}^{\infty} \Phi^j B^j Z_t = \sum_{j=0}^{\infty} \Phi^j Z_{t-j}$$

Because the ARCH(1) model is also a stationary process, we would like to identify a linear process solution for the model.
Instead of using backshift operators, we can evaluate the limit (at infinity) of the partial sums of the ARCH(1) model.

The dependence structure of the ARCH(1) model indicates that Z_t^2 depends on Z_{t-1}^2,

$$Z_t^2 = (\sqrt{h_t} \cdot e_t)^2 = h_t \cdot e_t^2 = (\alpha_0 + \alpha_1 Z_{t-1}^2)e_t^2$$
$$= \alpha_0 e_t^2 + \alpha_1 Z_{t-1}^2 e_t^2$$

Substituting Z_{t-1}, we note the multiplicative structure in the e_t terms that appears in the model,

$$= \alpha_0 e_t^2 + \alpha_1(\alpha_0 + \alpha_1 Z_{t-2}^2)e_{t-1}^2 e_t^2$$

Note that the multiplicative structure grows for each term included in the model.

$$= \alpha_0 e_t^2 + \alpha_0\alpha_1 e_{t-1}^2 e_t^2 + \alpha_1^2 Z_{t-2}^2 e_{t-1}^2 e_t^2$$
$$= \cdots$$

Therefore,

$$Z_t^2 = \alpha_0 \sum_{j=0}^{n} \alpha_1^j e_t^2 e_{t-1}^2 \cdots e_{t-j}^2 + \alpha_1^{n+1} Z_{t-n-1}^2 e_t^2 e_{t-1}^2 \cdots e_{t-n}^2$$

For $\alpha_1 \in [0, 1)$, note that the latter term effectively disappears as $n \to \infty$ ($\alpha_1^{n+1} \to 0$), and we obtain the linear process solution

$$Z_t^2 = \alpha_0 \sum_{j=0}^{\infty} \alpha_1^j e_t^2 e_{t-1}^2 \cdots e_{t-j}^2$$

Thus α_1^j resembles the linear process coefficient multiplied by an infinite number of noise terms (depending on your position in the sequence $j = 0, \ldots, \infty$).

Note that

$$Z_t = \sqrt{h_t} \cdot e_t = e_t \sqrt{\alpha_0 \left(1 + \sum_{j=1}^{\infty} \alpha_1^j e_{t-1}^2 \cdots e_{t-j}^2\right)}$$

Variance of the ARCH(1) Model:
We can also identify the limit of the variance of the ARCH(1) model by evaluating the partial sums of the ARCH(1) model.

We have derived that

$$Z_t^2 = \alpha_0 \sum_{j=0}^{n} \alpha_1^j e_t^2 e_{t-1}^2 \cdots e_{t-j}^2 + \alpha_1^{n+1} Z_{t-n-1}^2 e_t^2 e_{t-1}^2 \cdots e_{t-n}^2$$

For $\alpha_1 \in [0, 1)$, the expectation of the LHS is

$$E\left[\alpha_0 \sum_{j=0}^{n} \alpha_1^j e_t^2 e_{t-1}^2 \cdots e_{t-j}^2\right] \overset{iid,\ linear}{=} \alpha_0 \sum_{j=0}^{n} \alpha_1^j E[e_t^2]E[e_{t-1}^2]E[e_{t-j}^2]$$

Note that because $e_i \sim N(0, 1)$, $e_i^2 \sim \chi^2(1)$, which implies that $E[e_i^2] = 1$.
The expectation of a χ_1^2 random variable is 1, so we obtain

$$= \alpha_0 \sum_{j=0}^{n} \alpha_1^j$$

Asymptotically, using the sum of the geometric series for $\alpha_1 \in [0, 1)$,

$$\alpha_0 \sum_{j=0}^{n} \alpha_1^j \to \frac{\alpha_0}{1 - \alpha_1} \quad as \quad n \to \infty$$

For $\alpha_1 \in [0, 1)$, the expectation of the RHS is

$$E\left[\alpha_1^{n+1} Z_{t-n-1^2} e_t^2 e_{t-1}^2 \cdots e_{t-n}^2\right]$$

Because the e_t terms are iid, we can factorize the expectation,

$$\stackrel{iid}{=} \alpha_1^{n+1} E[Z_{t-n-1}^2] E[e_t^2] E[e_{t-1}^2] \cdots E[e_{t-n}^2]$$

Note that because $e_i \sim N(0,1)$, $e_i^2 \sim \chi^2(1)$, which implies that $E[e_i^2] = 1$.

$$= \alpha_1^{n+1} E[Z_{t-n+1}^2]$$

Because $E[Z_{t-n-1}^2]$ is finite, and α_1^{n+1} goes to 0 for $\alpha_1 \in [0,1)$,

$$\alpha^{n+1} E[Z_{t-n+1}^2] \to 0 \;\; as \;\; n \to \infty$$

Therefore, for $\alpha_1 \in [0,1)$

$$Var(Z_t) = E\left[Z_t^2\right] \to \frac{\alpha_0}{1-\alpha_1} \;\; as \;\; n \to \infty$$

Linear Process Solution of the ARCH(1) Model:
If $\alpha_1 \in (0,1]$, the unique, causal, stationary solution of the ARCH(1) model is

$$Z_t = e_t \sqrt{\alpha_0 \left(1 + \sum_{j=1}^{\infty} \alpha_1^j e_{t-1}^2 \cdots e_{t-j}^2\right)}$$

The linear process solution has the following properties.
i. $E[Z_t] = E[E[Z_t | e_s, s < t]] = 0$
ii. $Var(Z_t) = \frac{\alpha_0}{1-\alpha_1}$
iii. $E[Z_{t+h} Z_t] = E[E[Z_{t+h} Z_t | e_s, s < t+h]] = 0$, for $h > 0$

The properties of the linear process solution indicate that the ARCH(1) process is fundamentally still WN.
In particular, it is a form of stationary WN that has a dependence structure in the volatility terms.
Thus, we can specify an ARCH(p) process on a time series model with a WN error structure to assess if there is a dependence structure in the volatility terms.

In particular, if $\alpha_1 \in (0,1]$, then the ARCH(1) process is stationary WN (but not IID, because of the dependence structure in the volatility terms).

Note that the conditional variance is given by

$$E[Z_t^2 | Z_{t-1}] = E[h_t e_t^2 | Z_{t-1}] = E\left[\left(\alpha_0 + \alpha_1 Z_{t-1}^2\right) e_t^2 | Z_{t-1}\right]$$

$$= \left(\alpha_0 + \alpha_1 Z_{t-1}^2\right) E[e_t^2 | Z_{t-1}] \stackrel{iid\ e_t}{=} \left(\alpha_0 + \alpha_1 Z_{t-1}^2\right) E[e_t^2] = \alpha_0 + \alpha_1 Z_{t-1}^2$$

Thus there is conditional dependence on the variance (hence they are not IID).

Properties:
1. The ARCH(p) process is symmetric: $Z_t \stackrel{d}{=} -Z_t$
2. For the ARCH(1) process: $E[Z_t^4]$ is finite iff $3\alpha_1^2 < 1$.

   Higher order moment properties exist for certain constraints, because of the normality assumption.
   This is useful in finance because we are dealing with heavy tails and may wish to characterize these moments.

Estimation of the ARCH(1) Model:
While Z_t is not Gaussian, Z_t is conditionally Gaussian.
Thus, we can formulate a conditional likelihood (i.e., an approximate likelihood) to establish an MLE estimation method of the ARCH(1) process.

If we assume the Z_t is conditionally Gaussian, and have specified its conditional mean and variance,

$$E[Z_t | Z_{t-1}] = 0$$
$$E[Z_t^2 | Z_{t-1}] = \alpha_0 + \alpha_1 Z_{t-1}^2$$

We can fully specify the conditional likelihood.

The conditional likelihood of observations $\{Z_2, \ldots, Z_n\}$ of the ARCH(1) process (provided $Z_1 = z_1$) is given by

$$L(\alpha_0, \alpha_1 | Z_1 = z_1) = \prod_{t=2}^{n} \frac{1}{\sqrt{2\pi(\alpha_0 + \alpha_1 Z_{t-1}^2)}} \exp\left(-\frac{1}{2(\alpha_0 + \alpha_1 Z_{t-1}^2)}(Z_t - 0)\right)$$

GARCH (Generalized ARCH) Model:
If the ARCH(p) model is analogous to the AR(p) model, the GARCH(p,q) model is analogous to the ARMA(p,q) model.

We say that the (WN) error terms of a time series, $\{Z_t\}$ have an GARCH(p,q) structure if

$$Z_t = \sqrt{h_t} \cdot e_t, \quad e_t \overset{iid}{\sim} N(0,1)$$

Where h_t is related to past values of Z_t^2 and h_t through the relation below

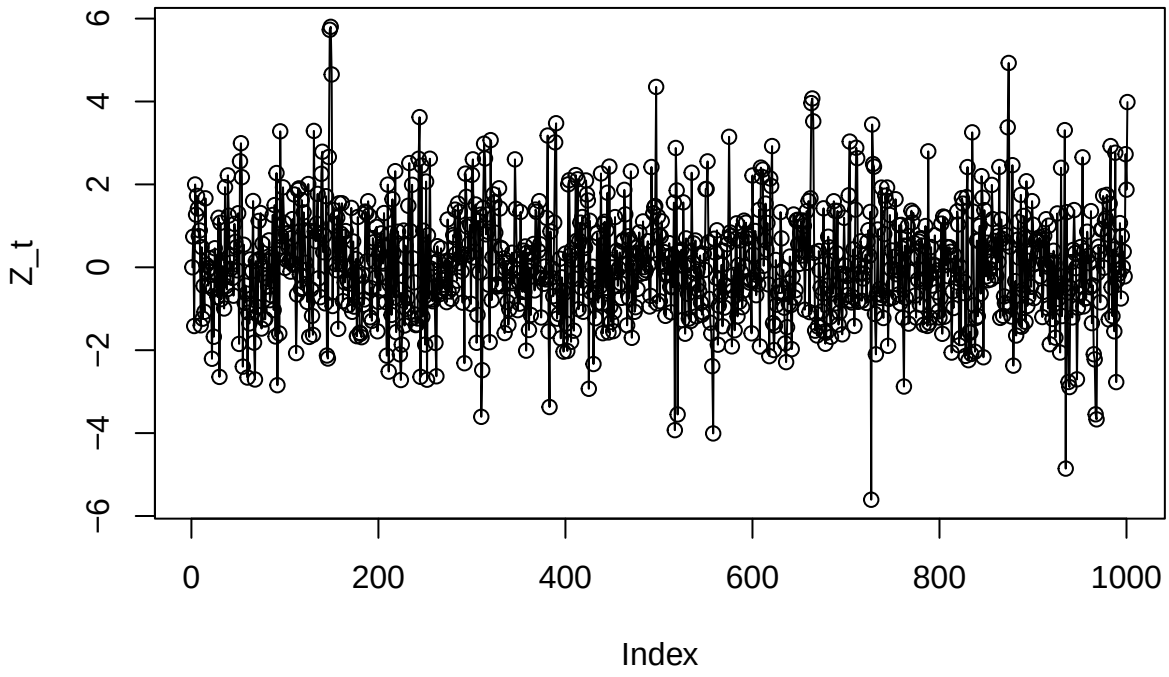
$$h_t = \alpha_0 + \sum_{i=1}^{p} \alpha_i Z_{t-i}^2 + \sum_{j=1}^{q} \beta_j h_{t-j}, \quad p, q \in \mathbb{Z} > 0$$

For $h_t > 0$, $\alpha_0 > 0$, $\alpha_i, \beta_j \geq 0$.

Examples:

Example: ARCH(1)

```
# Setup parameters
n = 1000
e_t = rnorm(n+1)
alpha0 = 1
alpha1 = .5
Z_t = rep(NA,n+1)
Z_t[1] = 0

# Recursively construct Zt
for (t in 2:(n+1)) {
  h_t = alpha0 + alpha1*(Z_t[t-1])^2
  Z_t[t] = sqrt(h_t)*e_t[t]
}

# Zt resembles WN, but there are spikes indicating ARCH structure
plot(Z_t,type="o")
```




```
mean(Z_t) # close to 0
```

```
## [1] 0.07670655
```

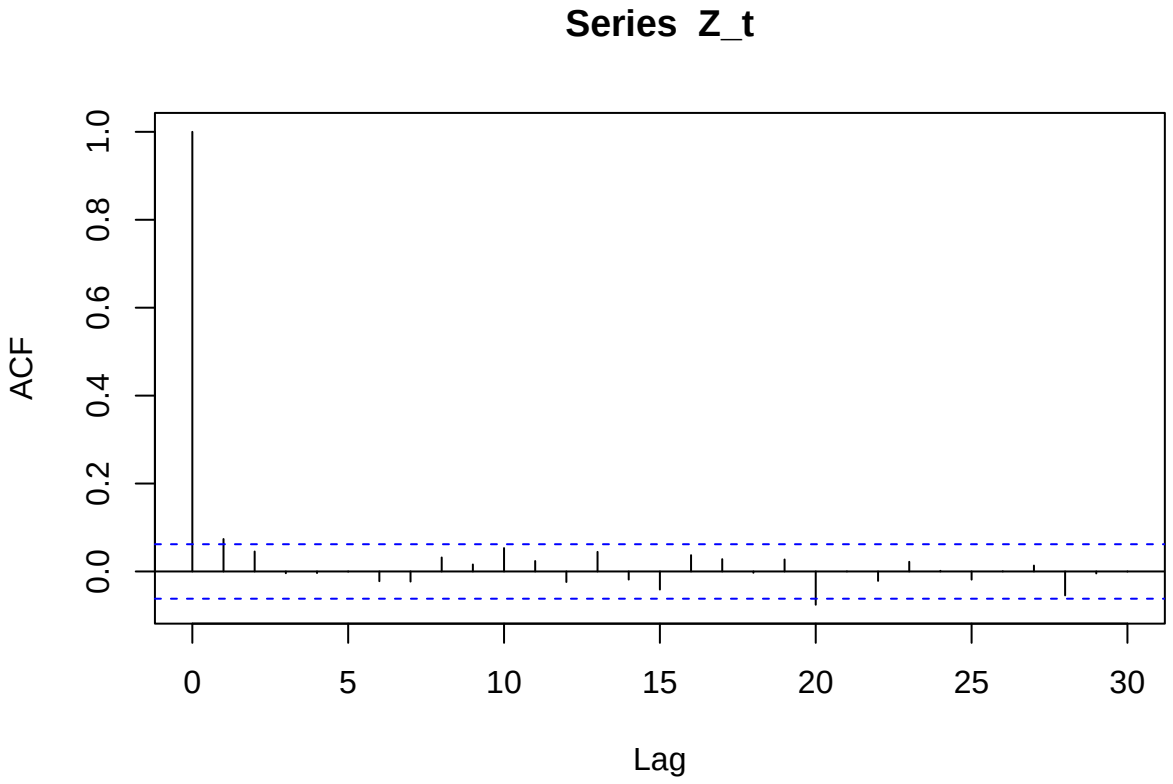
```
var(Z_t) # about 2.5
```

```
## [1] 1.831069
```

```
alpha0/(1-alpha1) # compare to long run variance
```

```
## [1] 2
```

```
# ACF resembles WN
acf(Z_t)
```



PRACTICE PROBLEMS

Example: Practice Problems - Ch 5 #1

The sunspot numbers $\{X_t, t = 1, \ldots, 100\}$, filed as `SUNSPOTS.TSM` have sample autocovariances $\hat{\gamma}(0) = 1382.2$, $\hat{\gamma}(1) = 1114.4$, $\hat{\gamma}(2) = 591.73$, and $\hat{\gamma}(3) = 96.216$.

Use these values to find the Yule-Walker estimates of ϕ_1, ϕ_2, and σ^2 in the model

$$Y_t = \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + Z_t, \quad Z_t \sim WN(0, \sigma^2)$$

For the mean-corrected series $Y_t = X_t - 46.93$, $t = 1, \ldots, 100$.

Assuming that the data really are a realization of an AR(2) process, find 95% confidence intervals for ϕ_1 and ϕ_2.

The sample Yule-Walker equations are given by

$$\hat{\phi}_p = \hat{\mathbf{\Gamma}}_p^{-1} \hat{\gamma}_p$$
$$\hat{\sigma}^2 = \hat{\gamma}(0) - \hat{\phi}_p^T \hat{\gamma}_p$$

$$\hat{\mathbf{\Gamma}} = \begin{bmatrix} \hat{\gamma}(0) & \hat{\gamma}(1) \\ \hat{\gamma}(1) & \hat{\gamma}(0) \end{bmatrix} = \begin{bmatrix} 1382.2 & 1114.4 \\ 1114.4 & 1382.2 \end{bmatrix}$$

$$\hat{\mathbf{\Gamma}}^{-1} = \begin{bmatrix} 0.00207 & -0.00167 \\ -0.00167 & 0.00207 \end{bmatrix} = \begin{bmatrix} 1382.2 & 1114.4 \\ 1114.4 & 1382.2 \end{bmatrix}$$

$$\hat{\gamma}_p = \begin{bmatrix} \hat{\gamma}(1) \\ \hat{\gamma}(2) \end{bmatrix} = \begin{bmatrix} 1114.4 \\ 591.73 \end{bmatrix}$$

$$\hat{\phi} = \begin{bmatrix} \hat{\phi}_1 \\ \hat{\phi}_2 \end{bmatrix} = \hat{\mathbf{\Gamma}}^{-1} \hat{\gamma} = \begin{bmatrix} 1.318 \\ -0.634 \end{bmatrix}$$

$$\hat{\sigma}^2 = \hat{\gamma}(0) - \hat{\phi}_p^T \hat{\gamma}_p = 1382.2 - \hat{\phi}_p^T \hat{\gamma}_p = 289.179$$

```
r1 = c(1382.2,1114.4)
r2 = c(1114.4,1382.2)
matGamma = rbind(r1,r2)
vecGamma = c(1114.4,591.73)

solve(matGamma)
```

```
##                r1           r2
## [1,]  0.002067337 -0.001666793
## [2,] -0.001666793  0.002067337
```

```
(phi = solve(matGamma) %*% vecGamma)
```

```
##             [,1]
## [1,]  1.3175495
## [2,] -0.6341682
```

```
(sigma_sq = 1382.2 - t(phi) %*% vecGamma)
```

```
##          [,1]
## [1,] 289.1791
```

The asymptotic $100(1-\alpha)\%$ confidence interval for the Yule-Walker estimator ϕ_j

$$\hat{\phi}_{p_j} \pm z_{1-\alpha/2}\frac{1}{\sqrt{n}}\sqrt{\hat{\nu}_j}$$

Where
i. $\hat{\phi}_{p_j}$ is the jth scalar in $\hat{\phi}_p$, and
ii. $\hat{\nu_j}$ is the jth diagonal element of $\hat{\sigma}^2\hat{\mathbf{\Gamma}}_p^{-1}$

The 95% confidence interval for ϕ_1 is given by

$$\hat{\phi}_1 \pm (1.96)\left(\frac{1}{\sqrt{100}}\sqrt{0.598}\right) = (1.166, 1.470)$$

The 95% confidence interval for ϕ_2 is given by

$$\hat{\phi}_2 \pm (1.96)\left(\frac{1}{\sqrt{100}}\sqrt{0.598}\right) = (-0.786, -0.482)$$

```
sigma_sq[1,1] * solve(matGamma)
```

```
##              r1         r2
## [1,]  0.5978308 -0.4820016
## [2,] -0.4820016  0.5978308
```

```
c(1.318 - qnorm(0.975)*(1/10)*sqrt(0.598),1.318 + qnorm(0.975)*(1/10)*sqrt(0.598))
```

```
## [1] 1.166435 1.469565
```

```
c(-0.634 - qnorm(0.975)*(1/10)*sqrt(0.598),-0.634 + qnorm(0.975)*(1/10)*sqrt(0.598))
```

```
## [1] -0.7855649 -0.4824351
```

Example: Practice Problems - Ch 5 #3

Consider the AR(2) process X_t satisfying

$$X_t - \phi X_{t-1} - \phi^2 X_{t-2} = Z_t, \quad Z_t \sim WN(0, \sigma^2)$$

a) For what values of ϕ is this a causal process?

$\{X_t : t \in \mathbb{Z}\}$ is causal if $\Phi(u) \neq 0$ for $|u| < 1$.

$$\Phi(u) = 1 - \phi u - \phi^2 u^2 = 0$$

$$\Phi(u) = 1 - 0.5u - 0.25u^2 = (-0.25)(u - 1.1236)(u + 3.3236) = 0$$
$$u = \{1.1236, -3.236\}$$
$$|u| = \{1.1236, 3.236\} > 1$$

This AR(2) process is stationary and causal.

Intuitively, the coefficients are decreasing at a certain rate, which seems to indicate that the AR(2) process is stationary.

```
(roots = polyroot(c(1,-0.5, -0.25)))
## [1]  1.236068+0i -3.236068-0i
Mod(roots)
## [1] 1.236068 3.236068
```

b) The following sample moments were computed after observing $X_1, \ldots, X_{200}$: $\hat{\gamma}(0) = 0.606$, $\hat{\rho}(1) = 0.687$.
Find estimates of ϕ and σ^2 by solving the Yule-Walker equations.
If you have more than one solution, select the one that is causal.

Example: Final Exam (Spring 2022) #1, Ch 7 #1

Consider the causal ARCH(1) model with $\alpha_1 \in (0, 1)$ and $\alpha_0 > 0$.

$$Z_t = \sqrt{h_t} \cdot e_t, \quad e_t \overset{iid}{\sim} N(0, 1)$$

Where

$$h_t = \alpha_0 + \alpha_1 Z_{t-1}^2$$

Note that $\alpha \in (0, 1)$ guarantees that

$$Z_t = e_t \sqrt{\alpha_0 \left(1 + \sum_{j=1}^{\infty} \alpha_1^j e_{t-1}^2 \cdots e_{t-j}^2\right)}$$

Is the unique stationary solution of the ARCH(1) process.

a) Derive $Var(Z_t)$ of the ARCH(1) process $\{Z_t\}$ defined above.
You may directly use the unique stationary solution of the ARCH(1) process if needed.

Because $E[Z_t] = 0$, $Var(Z_t) = E[Z_t^2]$.

$$Var(Z_t) = E[Z_t^2] = E\left[e_t^2 \left(\alpha_0 + \sum_{j=1}^{\infty} \alpha_1^j e_{t-1}^2 \cdots e_{t-j}^2\right)\right] = \alpha_0 E[e_t^2] + \alpha_0 \sum_{j=1}^{\infty} \alpha_1^j E[e_{t-1}^2 \cdots e_{t-j}^2]$$
$$\overset{e_i \ ind, \ E[e_i^2]=1}{=} \alpha_0 + \alpha_0 \sum_{j=1}^{\infty} \alpha_1^j = \alpha_0 \sum_{j=0}^{\infty} \alpha_1^j = \frac{\alpha_0}{1 - \alpha_1}$$

b) Evaluate $E[Z_t^4]$ for the ARCH(1) process with $\alpha_1 \in (0, 1)$ and $e_t \overset{IID}{\sim} N(0, 1)$.
Deduce that $E[X_t^4] < \infty$ if and only if $3\alpha_1^2 < 1$.

$$E[Z_t^4] = E[h_t^2 e_t^4] \overset{ind}{=} E[h_t^2]E[e_t^4]$$

The fourth moment of a $X \sim N(\mu, \sigma^2)$ random variable is $E[X^4] = 3\sigma^4$.
Because $e_t \sim N(0,1)$, $E[e_t^4] = 3$.

$$= 3E\left[\left(\alpha_0 + \alpha_1 Z_{t-1}^2\right)^2\right] = 3E\left[\alpha_0^2 + 2\alpha_0\alpha_1 Z_{t-1}^2 + \alpha_1^2 Z_{t-1}^4\right]$$

$$= 3\alpha_0^2 + 6\alpha_0\alpha_1 E[Z_{t-1}^2] + 3\alpha_1^2 E[Z_{t-1}^4]$$

Noting that $Var(Z_{t-1}) = E[Z_{t-1}^2] = \frac{\alpha_0}{1-\alpha_1}$,

$$= 3\alpha_0^2 + \frac{6\alpha_0^2\alpha_1}{1-\alpha_1} + 3\alpha_1^2 E[Z_{t-1}^4]$$

$$E[Z_{t-1}^4] = E[h_{t-1}^2 e_{t-1}^4] \overset{ind}{=} E[h_{t-1}^2]E[e_{t-1}^4]$$

The fourth moment of a $X \sim N(\mu, \sigma^2)$ random variable is $E[X^4] = 3\sigma^4$.
Because $e_{t-1} \sim N(0,1)$, $E[e_{t-1}^4] = 3$.

$$= 3E\left[\left(\alpha_0 + \alpha_1 Z_{t-2}^2\right)^2\right] = 3E\left[\alpha_0^2 + 2\alpha_0\alpha_1 Z_{t-2}^2 + \alpha_1^2 Z_{t-2}^4\right]$$

$$= 3\alpha_0^2 + 6\alpha_0\alpha_1 E[Z_{t-2}^2] + 3\alpha_1^2 E[Z_{t-2}^4]$$

Noting that $Var(Z_{t-1}) = E[Z_{t-1}^2] = \frac{\alpha_0}{1-\alpha_1}$,

$$= 3\alpha_0^2 + \frac{6\alpha_0^2\alpha_1}{1-\alpha_1} + 3\alpha_1^2 E[Z_{t-2}^4]$$

For

$$c = 3\alpha_0^2 + \frac{6\alpha_0^2\alpha_1}{1-\alpha_1}$$

We have

$$E[Z_t^4] = c + 3\alpha_1^2 E[Z_{t-1}^4] = c + 3\alpha_1^2\left(c + 3\alpha_1^2 E[Z_{t-2}^4]\right) = c + 3\alpha_1^2 c + (3\alpha_1^2)^2 E[Z_{t-2}^4]$$

We can generalize the recursive result to

$$E[Z_t^4] = c\sum_{i=0}^{n}(3\alpha^2)^i + (3\alpha^2)^{n+1} E\left[Y_{t-(n+1)}^4\right]$$

Therefore, $E[Z_t^4] < \infty$ for $3\alpha^2 < 1$ to guarantee convergence of the geometric series.

Example: Final Exam (Spring 2022) #2

Consider the time series process $\{X_t\}$ which is comprised of an $AR(1)$ with $ARCH(1)$ errors:

$$X_t = \phi X_{t-1} + Z_t, \quad Z_t = \sqrt{h_t}\cdot e^t, \quad e_t \overset{iid}{\sim} N(0,1)$$

Note that

$$h_t = \alpha_0 + \alpha_1 Z_{t-1}^2$$

And assume that $|\phi| < 1$, $\alpha \in (0,1)$, $\alpha_0 > 0$.

a) Identify the causal solution to the AR(1)-ARCH(1) model above.
You do not have to derive your solution, you can simply write it down and briefly describe how you arrived at your result.

$$X_t = \phi X_{t-1} + Z_t$$
$$(1-\phi B)X_t = Z_t$$
$$(1-\phi B)X_t = Z_t$$

Define $\Phi(B) = 1 - \phi B$.

$$\Phi(B)X_t = Z_t$$

For $|\phi| < 1$ (which guarantees absolute convergence of constants), define the inverse operator

$$\Pi(B) = \sum_{j=0}^{\infty}(\phi B)^j \stackrel{!}{=} \frac{1}{1-\phi B}$$

Although we note that division by an operator is not well-defined.

$$\Pi(B)\Phi(B) = \Phi(B)\Pi(B) = 1$$

$$\Pi(B)\Phi(B)X_t = \Pi(B)Z_t$$

$$X_t = \Pi(B)Z_t = \sum_{j=0}^{\infty}(\phi B)^j Z_t = \sum_{j=0}^{\infty}\phi^j B^j Z_t = \sum_{j=0}^{\infty}\phi^j Z_{t-j}$$

Note that $\alpha \in (0,1)$ guarantees that

$$Z_t = e_t\sqrt{\alpha_0\left(1 + \sum_{j=1}^{\infty}\alpha_1^j e_{t-1}^2 \cdots e_{t-j}^2\right)}$$

Is the unique stationary solution of the ARCH(1) process.

Substituting the unique stationary solution for Z_{t-j}, we obtain

$$X_t = \sum_{j=0}^{\infty}\phi^j Z_{t-j} = \sum_{j=0}^{\infty}\phi^j e_{t-j}\sqrt{\alpha_0\left(1 + \sum_{i=j+1}^{\infty}\alpha_1^i e_{t-j-1}^2 \cdots e_{t-i}^2\right)}$$

b) Identify the autocovariance function of the AR(1)-ARCH(1) model above.
You do not have to derive your solution, you can simply write it down and briefly describe how you arrived at your result.

$$\gamma_X(h) = \sigma^2\sum_{j=0}^{\infty}\psi_j\psi_{j+|h|}$$

We know that $\Psi_j = \phi^j$ for $j \geq 0$.

$$\gamma_X(h) = \sigma^2\sum_{j=0}^{\infty}\psi_j\psi_{j+|h|} = \sigma^2\sum_{j=0}^{\infty}\phi^j\phi^{j+|h|} = \sigma^2\phi^{|h|}\sum_{j=0}^{\infty}(\phi^2)^j$$

Because $|\phi| < 1$, this implies that $|\phi^2| < 1$.

$$= \frac{\sigma^2\phi^{|h|}}{1-\phi^2}, \quad \forall\ h \in \mathbb{Z}$$

However, note that $\sigma^2 = Var(Z) = E[Z_t^2]$ derived in problem (1).

$$\therefore \gamma_X(h) = \frac{\alpha_0}{1-\alpha_1}\frac{\phi^{|h|}}{1-\phi^2}$$

Example: Final Exam (Spring 2022) #3, Ch 5 #9

Suppose that $\{X_t\}$ is a Gaussian time series with mean zero and autocovariance function $\gamma(i,j) = E[X_iX_j]$.
Let $\mathbf{X}_n = (X_1, \ldots, X_n)^T$ and let $\hat{\mathbf{X}}_n = (\hat{X}_1, \ldots, \hat{X}_n)^T$ where $\hat{X}_1 = 0$ and $\hat{X}_j = P_{j-1}X_j$, $j \geq 2$.
Let $\mathbf{\Gamma}_n$ denote the covariance matrix $\mathbf{\Gamma}_n = E[\mathbf{X_nX_n}]^T$ and assume that $\mathbf{\Gamma}_n$ is non-singular.

Then the likelihood of $\mathbf{X}_n$ is

$$L(\mathbf{\Gamma}_n) = (2\pi)^{-n/2}(\det \mathbf{\Gamma_n})^{-1/2} \exp\left(-\frac{1}{2}\mathbf{X}_n^T \mathbf{\Gamma}_n^{-1}\mathbf{X}_n\right)$$

As described from the text and lecture the exact Gaussian likelihood of an ARMA(p,q) can be expressed in terms of its innovations and mean square prediction error

$$L(\phi, \theta, \sigma^2) = L(\mathbf{\Gamma}_n) = \frac{1}{\sqrt{(2\pi\sigma^2)r_0 \cdots r_{n-1}}} \exp\left(-\frac{1}{2\sigma^2}\sum_{j=1}^{n}\frac{(X_j - \hat{X}_j)^2}{r_{j-1}}\right)$$

Where

$$r_{j-1} = \frac{1}{\sigma^2}v_{j-1} = \frac{1}{\sigma^2}E[(X_j - \hat{X}_j)^2]$$

a) Use the expression for the exact Gaussian likelihood above to show that for $n > p$, the likelihood of the observations $\{X_1, \ldots, X_n\}$ of the causal AR(p) process defined by

$$X_t = \phi_1 X_{t-1} + \cdots + \phi_p X_{t-p} + Z_t, \quad Z_t \overset{iid}{\sim} N(0,1)$$

Is given by

$$L(\phi, \sigma^2) = (2\pi\sigma^2)^{-n/2}(\det \mathbf{G}_p)^{-1/2} \exp\left(-\frac{1}{2\sigma^2}\left(\mathbf{X}_p^T \mathbf{G}_p^{-1}\mathbf{X}_p + \sum_{j=p+1}^{n}(X_j - \phi_1 X_{j-1} - \cdots - \phi_p X_{j-p})^2\right)\right)$$

Where $\mathbf{X}_p = (X_1, \ldots, X_p)^T$ and $\mathbf{G}_p = \frac{1}{\sigma^2}\mathbf{\Gamma}_p = \frac{1}{\sigma^2}E[\mathbf{X}_p\mathbf{X}_p^T]$.

For $j \in (p+1, n)$

$$E[X_j | X_{j-1}, \ldots, X_{j-p}] = \phi_1 X_{j-1} + \phi_2 X_{j-2} + \cdots + \phi_p X_{j-p}$$

$$Var(X_j | X_{t-j}, \ldots, X_{j-p}) = 1$$

$$\implies f(x_j | x_{j-1}, \ldots, x_{j-p}) = \frac{1}{\sqrt{2\pi}} \exp\left(-\frac{1}{2\sigma^2}(X_j - \phi_1 x_{j-1} - \cdots - \phi_p x_{j-p})^2\right)$$

$$\implies \prod_{j=p+1}^{n} f(x_j | x_{j-1}, \ldots, x_{j-p}) = (2\pi)^{-(n-p)/2} \exp\left(-\frac{1}{2\sigma^2}\sum_{j=p+1}^{n}(X_j - \phi_1 x_{j-1} - \cdots - \phi_p x_{j-p})^2\right)$$

Because $\{X_1, \ldots, X_p\}$ come from an AR(p) process,

$$f(x_1, \ldots, x_p) = (2\pi)^{-p/2}(\det \mathbf{\Gamma}_p)^{-1/2} \exp\left(-\frac{1}{2}\mathbf{X}_p^T \mathbf{\Gamma}_p^{-1}\mathbf{X}_p\right)$$

For $\sigma^2 = 1$, $\mathbf{\Gamma}_p = \mathbf{G}_p$, and the result is given by the following

$$L(\phi, \sigma^2 | \mathbf{x}_n) = f(\mathbf{x}_n | \phi, \sigma^2) = f(x_1, \ldots, x_p) \prod_{j=p+1}^{n} f(x_j | x_{j-1}, \ldots, x_{j-p})$$

b) Set up the conditional log-likelihood of a causal AR(2) model.
Note that you can set up the conditional likelihood directly and not use the innovations representation, but the exact likelihood from problem (3a) does provide an easy decomposition for identifying the conditional likelihood.

The AR(2) model is given by

$$X_t - \phi_1 X_{t-1} - \phi_2 X_{t-2} = Z_t, \quad Z_t \sim N(0, \sigma^2)$$

Note that because they share the same underlying structure through the AR(2) relationship,

$$P(X_1, \ldots, X_n | X_0, X_{-1}\phi, \sigma^2) = P(Z_1, \ldots, Z_n | X_0, X_{-1}, \phi, \sigma^2)$$

Therefore,

$$L(\phi_2, \sigma^2) = \left(2\pi\sigma^2\right)^{-n/2} \exp\left(-\frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_t - \phi_1 x_{t-1} - \phi_2 x_{t-2})^2\right)$$

$$l(\phi_2, \sigma^2) = -\frac{n}{2}\log\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_t - \phi_1 x_{t-1} - \phi_2 x_{t-2})^2$$

$$-l(\phi_2, \sigma^2) = \frac{n}{2}\log\left(2\pi\sigma^2\right) + \frac{1}{2\sigma^2}\sum_{i=1}^{n}(x_t - \phi_1 x_{t-1} - \phi_2 x_{t-2})^2$$

c) Derive the maximum likelihood estimates of ϕ_1 and ϕ_2 based on the conditional log-likelihood from problem (3b).

$$\frac{\delta}{\delta\phi_1} - l(\phi_2, \sigma^2) = \frac{1}{2\sigma^2}\sum_{i=1}^{n} 2(x_t - \phi_1 x_{t-1} - \phi_2 x_{t-2})(-x_{t-1}) \stackrel{set}{=} 0$$

$$-\sum_{i=1}^{n} x_t x_{t-1} + \phi_1\sum_{i=1}^{n} x_{t-1}^2 + \phi_2\sum_{i=1}^{n} x_{t-1}x_{t-2} = 0$$

$$\phi_1\sum_{i=1}^{n} x_{t-1}^2 + \phi_2\sum_{i=1}^{n} x_{t-1}x_{t-2} = \sum_{i=1}^{n} x_t x_{t-1}$$

$$\frac{\delta}{\delta\phi_2} - l(\phi_2, \sigma^2) = \frac{1}{2\sigma^2}\sum_{i=1}^{n} 2(x_t - \phi_1 x_{t-1} - \phi_2 x_{t-2})(-x_{t-2}) \stackrel{set}{=} 0$$

$$-\sum_{i=1}^{n} x_t x_{t-2} + \phi_1\sum_{i=1}^{n} x_{t-1}x_{t-2} + \phi_2\sum_{i=1}^{n} x_{t-2}^2 = 0$$

$$\phi_1\sum_{i=1}^{n} x_{t-1}x_{t-2} + \phi_2\sum_{i=1}^{n} x_{t-2}^2 = \sum_{i=1}^{n} x_t x_{t-2}$$

Solving the above system of 2 equations for 2 unknowns, we obtain

$$\hat{\phi}_{1_{MLE}} = \frac{\sum_{i=1}^{n} x_t x_{t-1}\sum_{i=1}^{n} x_{t-2}^2 - \sum_{i=1}^{n} x_t x_{t-2}\sum_{i=1}^{n} x_{t-1}x_{t-2}}{\sum_{i=1}^{n} x_{t-1}^2\sum_{i=1}^{n} x_{t-2}^2 - (\sum_{i=1}^{n} x_{t-1}x_{t-2})^2}$$

$$\hat{\phi}_{2_{MLE}} = \frac{\sum_{i=1}^{n} x_t x_{t-1}\sum_{i=1}^{n} x_{t-1}^2 - \sum_{i=1}^{n} x_t x_{t-2}\sum_{i=1}^{n} x_{t-1}x_{t-2}}{\sum_{i=1}^{n} x_{t-1}^2\sum_{i=1}^{n} x_{t-2}^2 - (\sum_{i=1}^{n} x_{t-1}x_{t-2})^2}$$

$$\frac{\delta}{\delta\sigma^2} - l(\phi_2, \sigma^2) = \frac{n}{2\sigma^2} - \frac{1}{2\sigma^4}\sum_{i=1}^{n}(x_t - \phi_1 x_{t-1} - \phi_2 x_{t-2})^2 \stackrel{set}{=} 0$$

$$\frac{n}{2\sigma^2}\cdot 2\sigma^4 = \sum_{i=1}^{n}(x_t - \phi_1 x_{t-1} - \phi_2 x_{t-2})^2$$

$$\hat{\sigma}^2_{MLE} = \frac{1}{n}\sum_{i=1}^{n}(x_t - \hat{\phi}_{1_{MLE}} x_{t-1} - \hat{\phi}_{2_{MLE}} x_{t-2})^2$$

d) For extra credit, derive the MLE's of ϕ_1 and ϕ_2.

The ACVF of the AR(2) model is given by

$$\gamma(h) = \begin{cases} \left(\frac{1-\phi_2}{1+\phi_2}\right)\frac{\sigma^2}{(1-\phi_2)^2-\phi_1^2} & for\ h = 0 \\ \frac{\phi_1}{1-\phi_2}\gamma(0) & for\ h = 1 \\ \phi_1\gamma(h-1) + \phi_2\gamma(h-2) & for\ h = 2, 3, \ldots \end{cases}$$

Using this, the $\mathbf{G}_2$ matrix can be constructed.

Example: Ch 7 #3

Suppose that Z_t is a causal stationary GARCH(p,q) process $Z_t = \sqrt{h_t}e_t$, where $e_t \sim IID(0,1)$, $\sum_{i=1}^{p} a_i + \sum_{j=1}^{q} B_j < 1$ and

$$h_t = \alpha_0 + \alpha_1 Z_{t-1}^2 + \cdots + \alpha_p Z_{t-p}^2 + \beta_1 h_{t-1} + \cdots + \beta_q h_{t-q}$$

a) Show that $E[Z_t^2 | Z_{t-1}^2, Z_{t-2}^2, \dots) = h_t$.

$$E[Z_t^2 | Z_{t-1}^2, Z_{t-2}^2, \dots] = E[h_t e_t^2 | Z_{t-1}^2, Z_{t-2}^2, \dots] = E[(\alpha_0 + \alpha_1 Z_{t-1}^2 + \cdots + \alpha_p Z_{t-p}^2 + \beta_1 h_{t-1} + \cdots + \beta_q h_{t-q}) e_t^2 | Z_{t-1}^2, Z_{t-2}^2, \dots]$$