

Time Series Notes

Liam Flaherty

August 24, 2024

Contents

1	Introductory Notes	3
1.1	Preliminaries	3
1.2	Stationarity, Autocorrelation, Partial Autocorrelation	4
1.3	White Noise, Backstep Operator, Invertibility	6
1.4	Autoregressive Models	8
1.5	Moving Average Models	10
1.6	ARMA Models	11
2	Problems	12
2.1	Stationarity, Invertibility, ACF/PACF Calculations	12
2.2	Simulation, Theoretical vs. Empirical ACF/PACF, Reduced Forms	18
2.3	Reduced Forms, MA and AR Conversions	26
2.4	Analyzing Datasets, White-Noise Tests, Forecasting	31
2.5	Model Diagnostics, ARIMA Models, Unit Root Tests	39
3	Appendix	47
3.1	Problem Set 1	47
3.2	Problem Set 2	48
3.3	Problem Set 4	51
3.4	Problem Set 5	54

1 Introductory Notes

1.1 Preliminaries

1. **Time Series** is a sequence of random variables X_t indexed by time (can be either discrete or continuous).
 - A. Observations in a time series likely aren't independent. The dependence between prior terms is called **autocorrelation**.
 - B. Goals might be three-fold
 - a. **Forecasting**: decompose into seasonal, trend, and noise
 - b. **transfer models**: output in 1 series effects the inputs in another (e.g. marketing and sales)
 - c. **intervention analysis**: try and determine the effect of an external event on a series

1.2 Stationality, Autocorrelation, Partial Autocorrelation

1. A time-series $\{Z_t\}_{t \in \mathbb{R}}$ is **strictly-stationary** if the joint distribution of $Z_{t_1}, Z_{t_2}, \dots, Z_{t_n}$ is the joint distribution of $Z_{t_1+k}, Z_{t_2+k}, \dots, Z_{t_n+k}$ for all choices of t_i and k .
 - A) A strictly stationary process must be identically distributed
 - B) The joint distribution of $Z_{t_1}, Z_{t_2}, \dots$ depends only on the **lag**
2. A time-series $\{Z_t\}_{t \in \mathbb{R}}$ is **weakly-stationary** if $\mathbb{E}(Z_t) = \mu$ for all t (constant mean assumption) and $\text{Cov}(Z_t, Z_{t+\tau})$ depends only on the lag τ (ergo constant variance assumption, $\mathbb{V}(Z_t) = \sigma^2$ for all t).
3. Denote $\tilde{Z}_t = Z_t - \mu$. The **Autocovariance function** of a (weakly) stationary series for some lag τ is:

$$\begin{aligned}
 \gamma_\tau &= \text{Cov}(Z_t, Z_{t+\tau}) \\
 &= \mathbb{E}(Z_t Z_{t+\tau}) - \mathbb{E}(Z_t) \mathbb{E}(Z_{t+\tau}) \\
 &= \mathbb{E}(Z_t Z_{t+\tau}) - \mu^2 \\
 &= \mathbb{E}(\tilde{Z}_t \tilde{Z}_{t+\tau})
 \end{aligned}$$

Note that $\gamma_0 = \mathbb{V}(\tilde{Z}_t) = \mathbb{E}(\tilde{Z}_t^2) - \mathbb{E}(\tilde{Z}_t)^2 = \mathbb{E}(\tilde{Z}_t^2) = \sigma^2$.

4. The **autocorrelation function (ACF)** of a (weakly) stationary series for some lag τ is:

$$\rho_\tau = \frac{\text{Cov}(Z_t, Z_{t+\tau})}{\sqrt{\mathbb{V}(Z_t) \mathbb{V}(Z_{t+\tau})}} = \frac{\gamma_\tau}{\sqrt{\sigma^2 \sigma^2}} = \frac{\gamma_\tau}{\sigma^2} = \frac{\gamma_\tau}{\gamma_0}$$

- A) A property of autocorrelation/covariance functions is symmetry ($\gamma_\tau = \gamma_{-\tau}$). This follows from the constant variance assumption of weak-stationarity
- B) We can write the two functions in matrix form as follows. The **covariance matrix**, denoted Γ_n is:

$$\Gamma_n = \begin{bmatrix} \gamma_0 & \gamma_1 & \cdots & \gamma_{n-1} \\ \gamma_1 & \gamma_0 & \cdots & \gamma_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ \gamma_{n-1} & \gamma_{n-2} & \cdots & \gamma_0 \end{bmatrix} = \gamma_0 \begin{bmatrix} 1 & \rho_1 & \cdots & \rho_{n-1} \\ \rho_1 & 1 & \cdots & \rho_{n-2} \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{n-1} & \rho_{n-2} & \cdots & 1 \end{bmatrix} = \gamma_0 P_n$$

5. The **Partial Autocorrelation Function (PACF)** is the correlation at a given lag after removing mutual linear dependence, i.e. $\text{Corr}(Z_t, Z_{t+k} \mid Z_{t+1}, Z_{t+2}, \dots, Z_{t+k-1})$.
 - A) If $Z_{t+k} = \phi_{k1} Z_{t+k-1} + \phi_{k2} Z_{t+k-2} + \cdots + \phi_{kk} Z_t + a_{t+k}$ is a linear regression, then ϕ_{kk} is the partial autocorrelation.
 - a. For example, if $y = \beta_0 + \beta_1 x^2$, β_1 is the linear dependence between x^2 and y , whereas if $y = \beta_0 + \beta_1 x + \beta_2 x^2$, β_2 is the linear dependence between x^2 and y *after already accounting for the dependence between x and y*

B) In general, we can compute the partial autocorrelations as $\phi_{k,k} = \frac{\left| \begin{bmatrix} 1 & \rho_1 & \cdots & \rho_{k-2} & \rho_1 \\ \rho_1 & 1 & \cdots & \rho_{k-3} & \rho_2 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \cdots & \rho_1 & \rho_k \end{bmatrix} \right|}{\left| \begin{bmatrix} 1 & \rho_1 & \cdots & \rho_{k-2} & \rho_{k-1} \\ \rho_1 & 1 & \cdots & \rho_{k-3} & \rho_{k-2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \cdots & \rho_1 & 1 \end{bmatrix} \right|}$

6. Non-stationarity can arise in seasonality (periodic and regular changes of the mean), long-term trend, expanding variance, etc.
7. **ARIMA** models difference terms to turn a non-stationary series into stationary residuals.

1.3 White Noise, Backstep Operator, Invertibility

1. **White Noise** is a purely random process $\{a_t\}_{t \in \mathbb{R}}$ such that the a_t are identically distributed with mean 0, finite variance, and no covariance between terms. As such, for all k , we must have:

$$\gamma_k = \text{Cov}(a_t, a_{t+k}) = \mathbb{E}(a_t a_{t+k}) = \begin{cases} \sigma_a^2, & k=0 \\ 0, & \text{else} \end{cases} \quad \rho_k = \begin{cases} 1, & k=0 \\ 0, & \text{else} \end{cases} \quad PACF_k = 0$$

Also, since every process can be written as a linear combination of white noise:

$$\text{Cov}(a_t, \tilde{Z}_{t-k}) = \mathbb{E}(a_t \tilde{Z}_{t-k}) = \begin{cases} \sigma_a^2, & k \leq 0 \\ 0, & \text{else} \end{cases}$$

This follows since:

$$\begin{aligned} \mathbb{E}(a_t \tilde{Z}_{t-k}) &= \mathbb{E}(a_t (a_{t-k} + \psi_{k+1} a_{t-k-1} + \dots)) && \text{Substituting} \\ &= \mathbb{E}(a_t a_{t-k} + \psi_{k+1} a_t a_{t-k-1} + \dots) && \text{Expanding} \\ &= \mathbb{E}(a_t a_{t-k}) + \psi_{k+1} \mathbb{E}(a_t a_{t-k-1}) + \dots && \text{Expectations are linear} \\ &= \mathbb{E}(a_t a_{t-k}) && \text{Properties of white noise} \end{aligned}$$

2. For ease, use **Backstep Operator** B . $BX_t = X_{t-1}$, $B^j X_t = X_{t-j}$, $\psi(B) = \sum_{j=0}^{\infty} \psi_j B^j$
3. **MA representation** Any non-deterministic stationary process can be written as a linear combination of white noise terms.

A) Form is $\tilde{Z}_t = a_t + \psi_1 a_{t-1} + \psi_2 a_{t-2} + \dots = \sum_{j=0}^{\infty} \psi_j a_{t-j} = \psi(B) a_t$ where $\sum_{j=0}^{\infty} |\psi_j| < \infty$

a. If you can write with $\sum_{j=0}^{\infty} |\psi_j| < \infty$, the series is **stationary**.

4. **AR Representation** May be able to represent a stationary process as a linear combination of its past values.

A) Form is $\tilde{Z}_t = a_t + \pi_1 \tilde{Z}_{t-1} + \pi_2 \tilde{Z}_{t-2} + \dots = a_t + \sum_{j=1}^{\infty} \pi_j \tilde{Z}_{t-j}$. So, $\pi(B) \tilde{Z}_t = a_t$.

a. Process is called **Invertible** if the above decomposition is possible (i.e. when a_t can be written as a linear combo of past Z_t 's). This only happens when $\sum_{j=1}^{\infty} |\pi_j| < \infty$.

5. There must be a connection between the AR and MA representation

A) $\tilde{Z}_t = \psi(B) a_t$ (where $\psi(B) = \sum_{j=0}^{\infty} \psi_j B^j = (1 + \psi_1 B + \psi_2 B^2 + \dots)$) and

$a_t = \pi(B) \tilde{Z}_t$ (where $\pi(B) = 1 - \sum_{j=1}^{\infty} \pi_j B^j = (1 - \pi_1 B - \pi_2 B^2 - \dots)$), so

left multiplying by $\psi(B)$ we have $\tilde{Z}_t = \psi(B) a_t = \psi(B) \pi(B) \tilde{Z}_t \implies \psi(B) \pi(B) = 1$

a. An example: $\tilde{Z}_t = a_t - 0.6a_{t-1} = (1 - 0.6B)a_t$.

Then since $\psi(B)\pi(B) = 1$, we have $(1 - 0.6B)(1 - \pi_1 B - \pi_2 B^2 - \dots) = 1$. Equating the coefficients for each B term (they must all be zero on the left to match the zeros on the right), we see:

$$\begin{aligned} -\pi_1 B - 0.6B &= 0 \implies \pi_1 = -0.6 \\ -\pi_2 B^2 + 0.6\pi_1 B^2 &= 0 \implies -\pi_2 B^2 + 0.6(-0.6)B^2 \implies \pi_2 = -0.36 \\ -\pi_3 B^3 + 0.6\pi_2 B^3 &= 0 \implies -\pi_3 B^3 + 0.6(-0.36)B^3 \implies \pi_3 = -0.216 \end{aligned}$$

From the perspective of the AR representation $a_t = (1 - \pi_1 B - \pi_2 B^2 - \dots)\tilde{Z}_t$, by continuing in the above fashion, we see $a_t = \tilde{Z}_t + 0.6\tilde{Z}_{t-1} + 0.36\tilde{Z}_{t-2} + \dots = \left(\sum_{j=0}^{\infty} (0.6B)^j\right)\tilde{Z}_t$ and have thus shown how to switch from a finite MA representation to an infinite AR representation.

Alternatively, since $\psi(B) = 1 - 0.6B$ and $\pi(B) = \psi(B)^{-1}$, we have $\pi(B) = \frac{1}{1-0.6B}$, which is $\sum_{j=0}^{\infty} (0.6B)^j$ by the geometric series $\frac{a}{1-r} = \sum_{k=0}^{\infty} ar^k$ (since $0.6 < 1$). This agrees with the above reasoning.

1.4 Autoregressive Models

1. AR(P) models are of the form $\tilde{Z}_t = \pi_1 \tilde{Z}_{t-1} + \cdots + \pi_p \tilde{Z}_{t-p} + a_t$.
 - A) Since there are only finitely many terms, every AR(P) process is automatically invertible (recall invertible if the absolute sum of AR coefficients is finite; $\sum_{j=1}^{\infty} |\pi_j| < \infty$)
 - B) Since $\psi(B) = \pi^{-1}(B)$, AR(P) processes are stationary if the roots of $\pi(B) = 0$ lie outside the unit circle (recall stationary if the absolute sum of MA coefficients is finite; $\sum_{j=1}^{\infty} |\psi_j| < \infty$).
2. Can compute autocovariance and autocorrelation functions of AR(P) processes easily (if it's stationary!).

$$\gamma_k = \mathbb{E}(\tilde{Z}_t \tilde{Z}_{t-k})$$

Definition

$$= \mathbb{E}\left((\pi_1 \tilde{Z}_{t-1} + \cdots + \pi_p \tilde{Z}_{t-p} + a_t) \tilde{Z}_{t-k}\right)$$

 $\tilde{Z}_t$ as an AR(P) process

$$= \pi_1 \mathbb{E}(\tilde{Z}_{t-1} \tilde{Z}_{t-k}) + \cdots + \pi_p \mathbb{E}(\tilde{Z}_{t-p} \tilde{Z}_{t-k}) + \mathbb{E}(a_t \tilde{Z}_{t-k})$$

Expectations are linear

$$= \pi_1 \gamma_{k-1} + \cdots + \pi_p \gamma_{k-p} + \mathbb{E}(a_t \tilde{Z}_{t-k})$$

Definition

$$\rho_k = \pi_1 \rho_{k-1} + \cdots + \pi_p \rho_{k-p} + \frac{\mathbb{E}(a_t \tilde{Z}_{t-k})}{\rho_0}$$

Dividing above by γ_0

$$\mathbb{V}(\tilde{Z}_t) = \pi_1 \gamma_1 + \cdots + \pi_p \gamma_p + \mathbb{E}(a_t \tilde{Z}_t)$$

From derivation of γ_k

$$= \pi_1 \gamma_1 + \cdots + \pi_p \gamma_p + \sigma_a^2$$

Properties of white noise

- A) the system of equations for ρ_k with $k = 1, 2, \dots$ (and hence $\mathbb{E}(a_t \tilde{Z}_{t-k}) = 0$) are called the **Yule-Walker Equations**
 - B) If the roots of $\pi(B)\rho_k = 0$ are real, get a damped exponential for the ACF. If roots are complex, get a damped sinusoidal.
3. Can convert between AR(P) and infinite MA representations.

- A) Since $1 = \pi(B)\psi(B)$, in the AR(1) case, $1 = (1 - \pi_1 B)(1 + \psi_1 B + \psi_2 B^2 + \cdots)$. Both sides of this equation are functions in B ; the order on either side must be the same. Expanding and grouping $1 = 1 + (\psi_1 - \pi_1)B + (\psi_2 - \pi_1\psi_1)B^2 + (\psi_3 - \pi_1\psi_2)B^3 + \cdots$. From the second term, $\psi_1 = \pi_1$. From the third term, $\psi_2 = \pi_1\psi_1 = \pi_1^2$, and, in general, $\psi_k = \pi_1^k$.
- B) Example: $\tilde{Z}_t = 0.6\tilde{Z}_{t-1} - 0.08\tilde{Z}_{t-2} + a_t$. Is this stationary?

$$\begin{aligned}
a_t &= \tilde{Z}_t - 0.6\tilde{Z}_{t-1} + 0.08\tilde{Z}_{t-2} + a_t && \text{Rearranging} \\
&= (1 - 0.6B + 0.08B^2)\tilde{Z}_t && \text{Write with backshift operator} \\
&= (1 - 0.4B)(1 - 0.2B)\tilde{Z}_t && \text{Factoring}
\end{aligned}$$

Since the roots of the polynomial are $\frac{5}{2}, 5$ and both are outside the unit circle, the process is stationary.

- C) Example: $\tilde{Z}_t = 0.2\tilde{Z}_{t-1} - 0.6\tilde{Z}_{t-2} + a_t$. Is this stationary? We can write $a_t = (1 - 0.2B + 0.6B^2)\tilde{Z}_t$. Using the quadratic formula, we see our roots are $\frac{0.2 \pm \sqrt{0.4 - 2.4}}{1.2}$. These are complex roots with a squared modulus greater than 1, so the un-squared modulus is also greater than 1— the process is also stationary.
4. One can identify an AR(p) process when the ACF is infinite in extent, and the PACF cuts off after lag p

1.5 Moving Average Models

1. We can write our autocovariances and autocorrelations as follows:

$\gamma_k = \mathbb{E}(\tilde{Z}_t \tilde{Z}_{t+k})$	Definition
$= \mathbb{E} \left(\sum_{i=0}^{\infty} \psi_i a_{t-i} \sum_{j=0}^{\infty} \psi_j a_{t+k-j} \right)$	MA representation
$= \mathbb{E} \left(\sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \psi_i \psi_j a_{t-i} a_{t+k-j} \right)$	Grouping
$= \sum_{i=0}^{\infty} \sum_{j=0}^{\infty} \psi_i \psi_j \mathbb{E}(a_{t-i} a_{t+k-j})$	Expectations are linear
$= \sum_{j=0}^{\infty} \psi_{j-k} \psi_j \mathbb{E}(a_{t+k-j} a_{t+k-j})$	$i \neq j - k \implies \mathbb{E}(a_{t-i} a_{t+k-j}) = 0$
$= \sigma_a^2 \sum_{j=0}^{\infty} \psi_{j-k} \psi_j$	Properties of White Noise

2. MA(q) processes have an ACF function that cuts off after lag q with an infinite PACF

1.6 ARMA Models

1. Use when both ACF and PACF appear infinite
2. Form of ARMA(p,q) is $\tilde{Z}_t = \pi_1 \tilde{Z}_{t-1} + \dots + \pi_p \tilde{Z}_{t-p} + \psi_1 a_{t-1} + \dots + \psi_q a_{t-q} + a_t$ and then $\pi(B)\tilde{Z}_t = \psi(B)a_t$
 - A) Can think of it as a p^{th} order AR process with a q^{th} order error term
 - B) Stationarity occurs if the roots of $\pi(B)$ lie outside the unit circle, invertibility occurs if the roots of $\psi(B)$ lie outside the unit circle. This is because we can write $\tilde{Z}_t = \frac{\psi(B)}{\pi(B)}a_t$ to put the model in terms of an AR process, and do the opposite to put the model in terms of an MA process. The AR polynomial will have roots outside the unit circle only when $\pi(B)$ has roots outside the unit circle, and the opposite is true for the MA polynomial.
3. If we assume stationarity, then multiplying both sides of the equation for $\tilde{Z}_t$ by $\tilde{Z}_{t-k}$ and taking expectations yields the covariance function:

$$\gamma_k = \pi_1 \gamma_{k-1} + \dots + \pi_p \gamma_{k+p} + \psi_1 \mathbb{E}(\tilde{Z}_{t-k} a_{t-1}) + \dots + \psi_q \mathbb{E}(\tilde{Z}_{t-k} a_{t-q}) + \mathbb{E}(\tilde{Z}_{t-k} a_t)$$
 - A) If our model is $\tilde{Z}_t = \pi_1 \tilde{Z}_{t-1} + a_t - \psi_1 a_{t-1}$, then we can compute the autocovariance functions as:

$$\begin{aligned}
 \gamma_k &= \mathbb{E}(\tilde{Z}_t \tilde{Z}_{t-k}) && \text{Multiply left by } \tilde{Z}_{t-k} \\
 &= \mathbb{E}(\pi_1 \tilde{Z}_{t-1} \tilde{Z}_{t-k} + a_t \tilde{Z}_{t-k} - \psi_1 a_{t-1} \tilde{Z}_{t-k}) && \text{Multiply right by } \tilde{Z}_{t-k} \\
 &= \pi_1 \gamma_{k-1} + \mathbb{E}(a_t \tilde{Z}_{t-k}) - \psi_1 \mathbb{E}(a_{t-1} \tilde{Z}_{t-k}) && \text{Expectations are linear}
 \end{aligned}$$

Using this function, we compute the autocovariance for the first two lags as:

$$\begin{aligned}
 \gamma_0 &= \pi_1 \gamma_1 + \sigma_a^2 - \psi_1 \psi_2 \sigma_a^2 && \text{Properties of white noise, } \psi_1 \mathbb{E}(a_t \tilde{Z}_{t+1}) = \psi_1 \psi_2 \sigma_a^2 \\
 \gamma_1 &= \pi_1 \gamma_0 - \psi_1 \sigma_a^2 && \text{Properties of white noise, } \mathbb{E}(a_t \tilde{Z}_{t-1}) = 0
 \end{aligned}$$

Where ψ_2 is the coefficient to the MA expansion of $\tilde{Z}_{t+1}$. For instance, if $\mathbb{E}(a_t \tilde{Z}_{t+1}) = \mathbb{E}(a_t b_0 \tilde{Z}_t) + \mathbb{E}(a_t b_1 \tilde{Z}_{t-1}) + \mathbb{E}(a_t b_2 \tilde{Z}_{t-2}) + \dots = b_0 \sigma_a^2 + 0 + 0 + \dots$, $\psi_2 = b_0$.

Substituting γ_1 into γ_0 , we have:

$$\begin{aligned}
 \gamma_0 &= \pi_1 [\pi_1 \gamma_0 - \psi_1 \sigma_a^2] + \sigma_a^2 - \psi_1 \psi_2 \sigma_a^2 \\
 \gamma_0 - \pi_1^2 \gamma_0 &= -\pi_1 \psi_1 \sigma_a^2 + \sigma_a^2 - \psi_1 \psi_2 \sigma_a^2 \\
 \gamma_0 &= \frac{\sigma_a^2 (1 - \pi_1 \psi_1 - \psi_1 \psi_2)}{1 - \pi_1^2}
 \end{aligned}$$

Substituting this γ_0 into γ_1 , we get γ_1 , and then have the recursion $\gamma_k = \pi_1 \gamma_{k-1}$ for $k \geq 3$.

2 Problems

2.1 Stationarity, Invertibility, ACF/PACF Calculations

1) Let $Z_t = 2 + 0.5t + a_t$ where $\{a_t\}$ is a white noise sequence with mean zero and variance σ_a^2 . Determine the mean and variance functions for the process $\{Z_t\}$. Is the process stationary?

We can compute the expectation as:

$$\begin{aligned}\mathbb{E}(Z_t) &= \mathbb{E}(2 + 0.5t + a_t) \\ &= \mathbb{E}(2) + \mathbb{E}(0.5t) + \mathbb{E}(a_t) \\ &= 2 + 0.5t\end{aligned}$$

Expectations are linear
Only non-constant is a_t , and $\mathbb{E}(a_t) = 0$

We can compute the variance as:

$$\begin{aligned}\mathbb{V}(Z_t) &= \mathbb{V}(2 + 0.5t + a_t) \\ &= \mathbb{V}(a_t) \\ &= \sigma_a^2\end{aligned}$$

Since $\mathbb{V}(c + Y) = \mathbb{V}(Y)$ for constants c

The mean of Z_t changes over time, so by definition, the process can not be stationary.

2) Henceforth let $\tilde{Z}_t = Z_t - \mu$ be a mean zero process with $\{a_t\}$ a white noise sequence with variance σ_a^2 . For each of the following, determine if the process is stationary and if it is invertible. If it is stationary, compute the ACF and PACF for lags of $k = 0, 1, 2, 3, 4$.

An AR process $\tilde{Z}_t = a_t + \sum_{j=1}^{\infty} \pi_j \tilde{Z}_{t-j}$ (equivalently, $a_t = \pi(B)\tilde{Z}_t$) is invertible if $\sum_{j=1}^{\infty} |\pi_j| < \infty$.

Since all the below processes have finitely many non-zero coefficients π_i , all the processes are automatically invertible (a finite sum of finite values is finite).

The Yule-Walker equations give us an easy way to compute the autocovariances (and thus autocorrelations) of an AR(p) process. We have $\gamma_k = \pi_1 \gamma_{k-1} + \dots + \pi_p \gamma_{k-p}$ for all lags $k = 1, 2, 3, \dots$. The autocorrelation of a value with itself is always 1.

$$\text{In general, we can compute the partial autocorrelations as } \phi_{k,k} = \frac{\begin{vmatrix} 1 & \rho_1 & \dots & \rho_{k-2} & \rho_1 \\ \rho_1 & 1 & \dots & \rho_{k-3} & \rho_2 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \dots & \rho_1 & \rho_k \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \dots & \rho_{k-2} & \rho_{k-1} \\ \rho_1 & 1 & \dots & \rho_{k-3} & \rho_{k-2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \dots & \rho_1 & 1 \end{vmatrix}}.$$

Like the autocorrelations, the partial autocorrelation of a value against itself (i.e. lag zero) is one. The partial autocorrelation of a lag 1 is the same as the autocorrelation. For an AR(P) process, the partial autocorrelation of a lag $k > P$ is zero. Of course, if we had a way to write our model as $\tilde{Z}_{t+k} = \phi_{k,1}\tilde{Z}_{t+k-1} + \phi_{k,2}\tilde{Z}_{t+k-2} + \dots + \phi_{k,k}\tilde{Z}_t + a_{t+k}$, then we could simply pick off the autocorrelation as the coefficient to the last term.

a. $\tilde{Z}_t - 0.5\tilde{Z}_{t-1} = a_t$

We can write the process as $(1 - 0.5B)\tilde{Z}_t = a_t$. Since the root of $(1 - 0.5B)$ is 2, which lies outside the complex unit circle, the process is stationary.

The process is an AR(1) and so the autocorrelation function from the Yule-Walker equation has just one term— $\rho_k = \pi_1 \rho_{k-1} = 0.5 \rho_{k-1}$. So $\rho_1 = 0.5$, $\rho_2 = 0.25$, $\rho_3 = 0.125$, and $\rho_4 = 0.06125$.

The partial autocorrelation is 0.5 at a lag of 1 (it is an AR(1) process, so is the same as the autocorrelation), and 0 at every other non-zero k .

b. $\tilde{Z}_t - 0.9\tilde{Z}_{t-1} = a_t$

We can write the process as $(1 - 0.9B)\tilde{Z}_t = a_t$. Since the root of $(1 - 0.9B)$ is $\frac{10}{9}$, which lies outside the complex unit circle, the process is stationary.

For the same reasoning as part a, the autocorrelations are $\rho_1 = 0.9$, $\rho_2 = 0.81$, $\rho_3 = 0.729$, and $\rho_4 = 0.6561$. The partial autocorrelation is 0.9 at a lag of 1, and 0 at other non-zero k .

c. $\tilde{Z}_t - 1.3\tilde{Z}_{t-1} + 0.4\tilde{Z}_{t-2} = a_t$

We can write the process as $(1 - 1.3B + 0.4B^2)\tilde{Z}_t = a_t$. The roots of $(0.4B^2 - 1.3B + 1)$ are: $\frac{-(-1.3) \pm \sqrt{(-1.3)^2 - 4(0.4)(1)}}{2(0.4)} = \frac{1.3 \pm \sqrt{1.69 - 1.6}}{0.8} = \frac{1.3 \pm \sqrt{0.09}}{0.8} = \frac{-1.3 \pm 0.3}{0.8} = 1.25, 2$. These are real roots which both lie outside the complex unit circle, so the process is stationary.

The autocorrelations can be computed from the Yule-Walker equations with $p = 2$. Recalling $\rho_0 = 1$ and the symmetry of the correlations, we compute:

$$\begin{aligned}\rho_1 &= \pi_1\rho_{1-1} + \pi_2\rho_{1-2} = (1.3)(1) + (-0.4)(\rho_1) \implies \rho_1 = \frac{1.3}{1 + 0.4} \approx 0.93 \\ \rho_2 &= \pi_1\rho_{2-1} + \pi_2\rho_{2-2} = (1.3)(\rho_1) + (-0.4)(1) \implies \rho_2 = \frac{1.3 \cdot 1.3}{1.4} - 0.4 \approx 0.81 \\ \rho_3 &= \pi_1\rho_{3-1} + \pi_2\rho_{3-2} = (1.3)(\rho_2) + (-0.4)(\rho_1) \implies \rho_3 \approx 0.68 \\ \rho_4 &= \pi_1\rho_{4-1} + \pi_2\rho_{4-2} = (1.3)(\rho_3) + (-0.4)(\rho_2) \implies \rho_4 \approx 0.56\end{aligned}$$

For the partial autocorrelations, we see:

$$\begin{aligned}\phi_{1,1} &= \rho_1 \approx 0.93 \\ \phi_{2,2} &= \frac{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & \rho_2 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} \approx \frac{0.81 - 0.93^2}{1 - 0.93^2} \approx -0.4\end{aligned}$$

Double-checking our work:

```
> ar_coeff=c(1.3, -.4)
> round(ARMAacf(ar=ar_coeff, lag.max=5),3)
      0      1      2      3      4      5
1.000 0.929 0.807 0.678 0.558 0.455
> round(ARMAacf(ar=ar_coeff, lag.max=5, pacf=TRUE),3)
[1] 0.929 -0.400 0.000 0.000 0.000
```

d. $\tilde{Z}_t - 1.2\tilde{Z}_{t-1} + 0.8\tilde{Z}_{t-2} = a_t$

We can write the process as $(1 - 1.2B + 0.8B^2)\tilde{Z}_t = a_t$. The roots of $(0.8B^2 - 1.2B + 1)$ are: $\frac{-(1.2) \pm \sqrt{(-1.2)^2 - 4(0.8)(1)}}{2(0.8)} = \frac{-1.2 \pm \sqrt{1.44 - 3.2}}{1.6} = \frac{-1.2 \pm \sqrt{-1.76}}{1.6}$. We see there are complex roots, and with the help of software we see that the modulus of these roots lie outside the complex unit circle, so the process is stationary.

```
> coef=c(1,-1.2,0.8)
> Mod(polyroot(coef))
[1] 1.118034 1.118034
```

The autocorrelations can be computed from the Yule-Walker equations with $p = 2$. Recalling $\rho_0 = 1$ and the symmetry of the correlations, we compute:

$$\begin{aligned}\rho_1 &= \pi_1\rho_{1-1} + \pi_2\rho_{1-2} = (1.2)(1) + (-0.8)(\rho_1) \implies \rho_1 = \frac{1.2}{1+0.8} = \frac{2}{3} \\ \rho_2 &= \pi_1\rho_{2-1} + \pi_2\rho_{2-2} = (1.2)(\rho_1) + (-0.8)(1) \implies \rho_2 = \frac{1.2 \cdot 2}{3} - 0.8 = 0 \\ \rho_3 &= \pi_1\rho_{3-1} + \pi_2\rho_{3-2} = (1.2)(\rho_2) + (-0.8)(\rho_1) \implies \rho_3 = -0.8\frac{2}{3} \approx -0.53 \\ \rho_4 &= \pi_1\rho_{4-1} + \pi_2\rho_{4-2} = (1.2)(\rho_3) + (-0.8)(\rho_2) \implies \rho_4 \approx -0.64\end{aligned}$$

For the partial autocorrelations, we see:

$$\begin{aligned}\phi_{1,1} &= \rho_1 = \frac{2}{3} \\ \phi_{2,2} &= \frac{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & \rho_2 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}} = \frac{0 - \frac{2^2}{3}}{1 - \frac{2^2}{3}} = \frac{-\frac{4}{9}}{\frac{5}{9}} = \frac{-4}{5}\end{aligned}$$

Double-checking our work:

```
> ar_coeff=c(1.2, -.8)
> round(ARMAacf(ar=ar_coeff, lag.max=5),3)
      0      1      2      3      4      5
1.000 0.667 0.000 -0.533 -0.640 -0.341
> round(ARMAacf(ar=ar_coeff, lag.max=5, pacf=TRUE),3)
[1] 0.667 -0.800 0.000 0.000 0.000
```

3) Compute the AR and MA processes for the below representations.

a. Find the AR representation of the MA(1) process $\tilde{Z}_t = a_t - 0.4a_{t-1}$.

In general, the MA representation is $\tilde{Z}_t = \psi(B)a_t$ and AR representation is $\pi(B)\tilde{Z}_t = a_t$ for some functions ψ and π of B . Left multiplying the AR representation by $\psi(B)$, we see that $\psi(B)\pi(B)\tilde{Z}_t = \psi(B)a_t = \tilde{Z}_t$, or that $\psi(B) = \pi^{-1}(B)$.

Applying this to the problem at hand, $\psi(B) = (1 - 0.4B)$ and so, by the formula for a geometric series (since $0.4 < 1$), $\pi(B) = (1 - 0.4B)^{-1} = \sum_{n=0}^{\infty} 0.4B^n$. Our AR representation is then $\tilde{Z}_t + 0.4\tilde{Z}_{t-1} + 0.16\tilde{Z}_{t-2} + \cdots = a_t$, or more compactly $a_t = \sum_{n=0}^{\infty} 0.4^n \tilde{Z}_{t-n}$.

b. Find the MA representation of the AR(2) process $\tilde{Z}_t = 0.2\tilde{Z}_{t-1} + 0.4\tilde{Z}_{t-2} + a_t$.

For notational ease we can rewrite the above as $\tilde{Z}_t - 0.2\tilde{Z}_{t-1} - 0.4\tilde{Z}_{t-2} = a_t$ and then read off $\pi(B)$ as $(1 - 0.2B - 0.4B^2)$. Here we hit a dead-end since we can't factor the polynomial into the form $(1 - xB)(1 - yB)$ and use the same geometric series trick in part a. Instead, we try substitution.

First, we substitute for $\tilde{Z}_{t-1}$ to eliminate the $\tilde{Z}_{t-1}$ term and add a a_{t-1} term in our representation:

$$\begin{aligned}\tilde{Z}_t &= 0.2[0.2\tilde{Z}_{t-2} + 0.4\tilde{Z}_{t-3} + a_{t-1}] + 0.4\tilde{Z}_{t-2} + a_t \\ &= 0.04\tilde{Z}_{t-2} + 0.08\tilde{Z}_{t-3} + 0.2a_{t-1} + 0.4\tilde{Z}_{t-2} + a_t \\ &= 0.44\tilde{Z}_{t-2} + 0.08\tilde{Z}_{t-3} + a_t + 0.2a_{t-1}\end{aligned}$$

Next, we substitute for $\tilde{Z}_{t-2}$ for the same reason:

$$\begin{aligned}\tilde{Z}_t &= 0.44[0.2\tilde{Z}_{t-3} + 0.4\tilde{Z}_{t-4} + a_{t-2}] + 0.08\tilde{Z}_{t-3} + a_t + 0.2a_{t-1} \\ &= 0.088\tilde{Z}_{t-3} + 0.176\tilde{Z}_{t-4} + 0.44a_{t-2} + 0.08\tilde{Z}_{t-3} + a_t + 0.2a_{t-1} \\ &= 0.168\tilde{Z}_{t-3} + 0.176\tilde{Z}_{t-4} + a_t + 0.2a_{t-1} + 0.44a_{t-2}\end{aligned}$$

To see where we are going, we can substitute one more time:

$$\begin{aligned}\tilde{Z}_t &= 0.168[0.2\tilde{Z}_{t-4} + 0.4\tilde{Z}_{t-5} + a_{t-3}] + 0.176\tilde{Z}_{t-4} + a_t + 0.2a_{t-1} + 0.44a_{t-2} \\ &= 0.0336\tilde{Z}_{t-4} + 0.0672\tilde{Z}_{t-5} + 0.168a_{t-3} + 0.176\tilde{Z}_{t-4} + a_t + 0.2a_{t-1} + 0.44a_{t-2} \\ &= 0.2096\tilde{Z}_{t-4} + 0.0672\tilde{Z}_{t-5} + a_t + 0.2a_{t-1} + 0.44a_{t-2} + 0.168a_{t-3}\end{aligned}$$

Continuing in this fashion, we can continually add an a_{t-n} term. Carefully backtracking the calculation of the a_{t-3} term, we see it is the coefficient of the $\tilde{Z}_{t-3}$ term, which itself is the sum of the product of 0.44 (the a_{t-2} term) and 0.2 with 0.08 (the product of 0.2– the a_{t-1} term– and 0.4). So generically, for $n \geq 3$, we have $\psi_{t-n} = 0.2\psi_{t-n+1} + 0.4\psi_{t-n+2}$. Compactly, we have $\tilde{Z}_t = a_t + 0.2a_{t-1} + 0.44a_{t-2} + \sum_{j=3}^{\infty} (0.2\psi_{t-j+1} + 0.4\psi_{t-j+2})$.

4) Consider the AR(3) process $(1 - 0.4B)(1 - 0.2B + 0.6B^2)\tilde{Z}_t = a_t$. Let $\sigma_a^2 = 1$. Determine the roots of $\phi(B) = 0$ and then answer the following: Is the process $\tilde{Z}_t$ stationary? Invertible? Why or why not? If the process is stationary, determine its autocorrelation function for integer values of k (you may give recursive equations for ρ_k for $k > 3$).

The roots of $\phi(B)$ are $\frac{5}{2}$ (read off from the first factor) and about $0.16 \pm 1.28i$ from the quadratic factor (as calculated by software, see below). Since all these roots have a modulus outside the complex unit circle, the process is stationary. Any AR(P) process with finite P is invertible as explained in question 2.

```
> coef=c(1,-.2,0.6)
> polyroot(coef)
[1] 0.1666667+1.280191i 0.1666667-1.280191i
> Mod(polyroot(coef))
[1] 1.290994 1.290994
```

To compute the autocorrelations, we can expand out the product, write our process in full, and then use the Yule-Walker equations to compute the correlations. We have $\pi(B) = 1 - 0.2B + 0.6B^2 - 0.4B + 0.08B^2 - 0.24B^3$ or $1 - 0.6B + 0.68B^2 - 0.24B^3$. In full, our process is $\tilde{Z}_t = 0.6\tilde{Z}_{t-1} - 0.68\tilde{Z}_{t-2} + 0.24\tilde{Z}_{t-3} + a_t$. We calculate as follows:

$$\begin{aligned}\rho_1 &= \pi_1\rho_{1-1} + \pi_2\rho_{1-2} + \pi_3\rho_{1-3} = (0.6)\rho_0 + (-0.68)\rho_1 + (0.24)\rho_2 = (0.6) + (-0.68)\rho_1 + (0.24)\rho_2 \\ \rho_2 &= \pi_1\rho_{2-1} + \pi_2\rho_{2-2} + \pi_3\rho_{2-3} = (0.6)\rho_1 + (-0.68)\rho_0 + (0.24)\rho_1 = (0.84)\rho_1 + (-0.68) \\ \rho_3 &= \pi_1\rho_{3-1} + \pi_2\rho_{3-2} + \pi_3\rho_{3-3} = (0.6)\rho_2 + (-0.68)\rho_1 + (0.24)\rho_0 = (0.6)\rho_2 + (-0.68)\rho_1 + (0.24)\end{aligned}$$

Substituting the second equation into the first, we see:

$$\rho_1 = (0.6) + (-0.68)\rho_1 + (0.24)((0.84)\rho_1 + (-0.68)) = 0.4368 - 0.4784\rho_1$$

So $\rho_1 = \frac{0.4368}{1+0.4784} \approx 0.295$, and then $\rho_2 = 0.84(\frac{0.4368}{1.4784}) - 0.68 \approx -0.432$, and finally $\rho_3 = 0.6(0.84(\frac{0.4368}{1.4784}) - 0.68) + (-0.68)(\frac{0.4368}{1.4784}) + 0.24 \approx -0.22$. For $k > 3$, we get the recursion $\rho_k = 0.6\rho_{k-1} - 0.68\rho_{k-2} + 0.24\rho_{k-3}$. As always, we get the negative “lags” by appealing to symmetry; $\rho_k = \rho_{-k}$.

Double checking our work:

```
> ar_coeffs=c(0.6,-.68,.24)
> round(ARMAacf(ar=ar_coeffs),3)
      0      1      2      3
1.000 0.295 -0.432 -0.220
```

2.2 Simulation, Theoretical vs. Empirical ACF/PACF, Reduced Forms

1) Consider the time series

$$\text{Model 1: } Z_t = a_t + 0.5a_{t-1} + 0.24a_{t-2}$$

$$\text{Model 2: } Z_t = 0.8Z_{t-1} + a_t - 0.3a_{t-1}$$

a. Simulate data of lengths 50 and 1000 for the models. Use a burn-in period of length 101 ($t = -100$ to 0) before outputting data from the model.

We can use the `arima.sim()` command from R. The below script gives us our simulation:

```

2 ▾ #####1. Load Required Packages#####
3   library(stats)
4
5
6 ▾ #####2. Specify Parameters#####
7   length1=50
8   length2=1000
9   burnin=101
10
11  #m1=a_t+0.5a_{t-1}+0.24a_{t-2}#
12  m1coefma=c(0.5, 0.24)          #ARIMA(0,0,2) process#
13
14  #m2=0.8Z_{t-1}+a_t-0.3a_{t-1}#
15  m2coefma=-0.3
16  m2coefar=0.8                  #ARIMA(1,0,1) process#
17
18
19 ▾ #####3. Specify Models#####
20  set.seed(534)                  #To make reproducible#
21  m1_short=arima.sim(model=list(ma=m1coefma),
22                      n=length1,
23                      n.start=burnin)
24
25  m1_long=arima.sim(model=list(ma=m1coef),
26                    n=length2,
27                    n.start=burnin)
28
29  m2_short=arima.sim(model=list(ar=m2coefar, ma=m2coefma),
30                    n=length1,
31                    n.start=burnin)
32
33  m2_long=arima.sim(model=list(ar=m2coefar, ma=m2coefma),
34                    n=length2,
35                    n.start=burnin)
36

```

And this code results in output like the below:

```

> m1_short
Time Series:
Start = 1
End = 50
Frequency = 1
[1] 0.47869805 0.21060373 -1.18648917 -0.90270106 -1.70181202 1.37558369 -0.43062264 -1.53659288 -1.58523395 -1.05513791 0.08008187
[12] -1.15878363 -0.21750940 -1.05108398 0.53885936 -0.74126957 -2.24613275 -2.11626422 -1.96868241 -0.26191589 1.21242265 0.14918433
[23] 0.89292468 0.10197492 -0.11404904 -1.40130243 -1.76749607 -0.68300524 -0.05418151 1.11855449 1.50300725 2.33500825 1.46505142
[34] 0.60403725 -0.32464004 -0.28893189 -1.35669185 -0.27787101 0.68879004 -0.23163217 0.20071319 1.47539302 -0.45619082 0.38454498
[45] -0.55748997 0.60319607 0.93271552 0.02528023 -0.65376996 -1.54489792

```

b. Use software to produce plots of the simulated time series, and the estimated autocorrelation (ACF) and partial autocorrelation (PACF) functions.

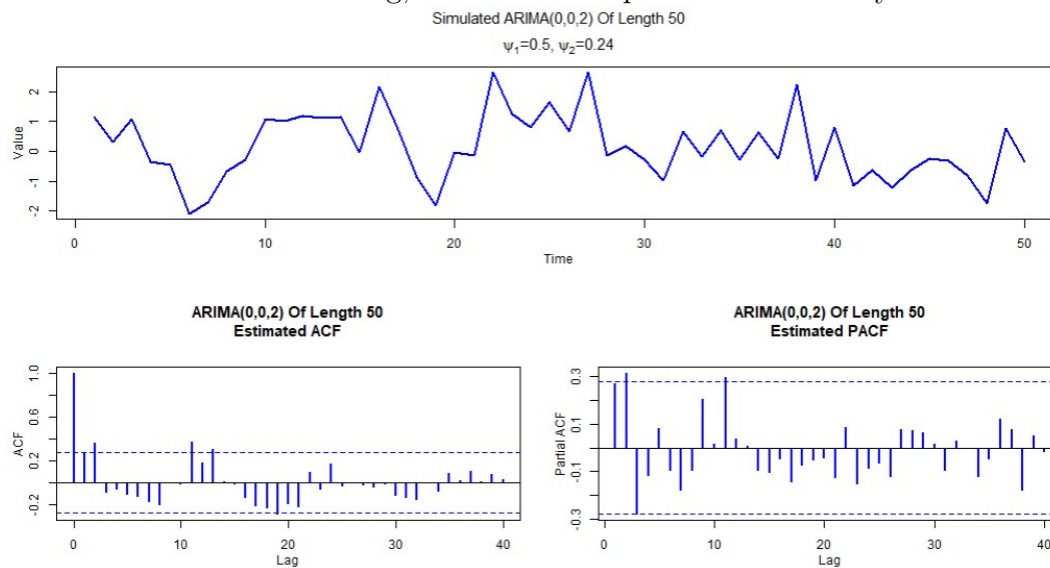
We can plot the simulated data and estimated ACF and PACF functions. The full script is in the appendix, but the gist is below.

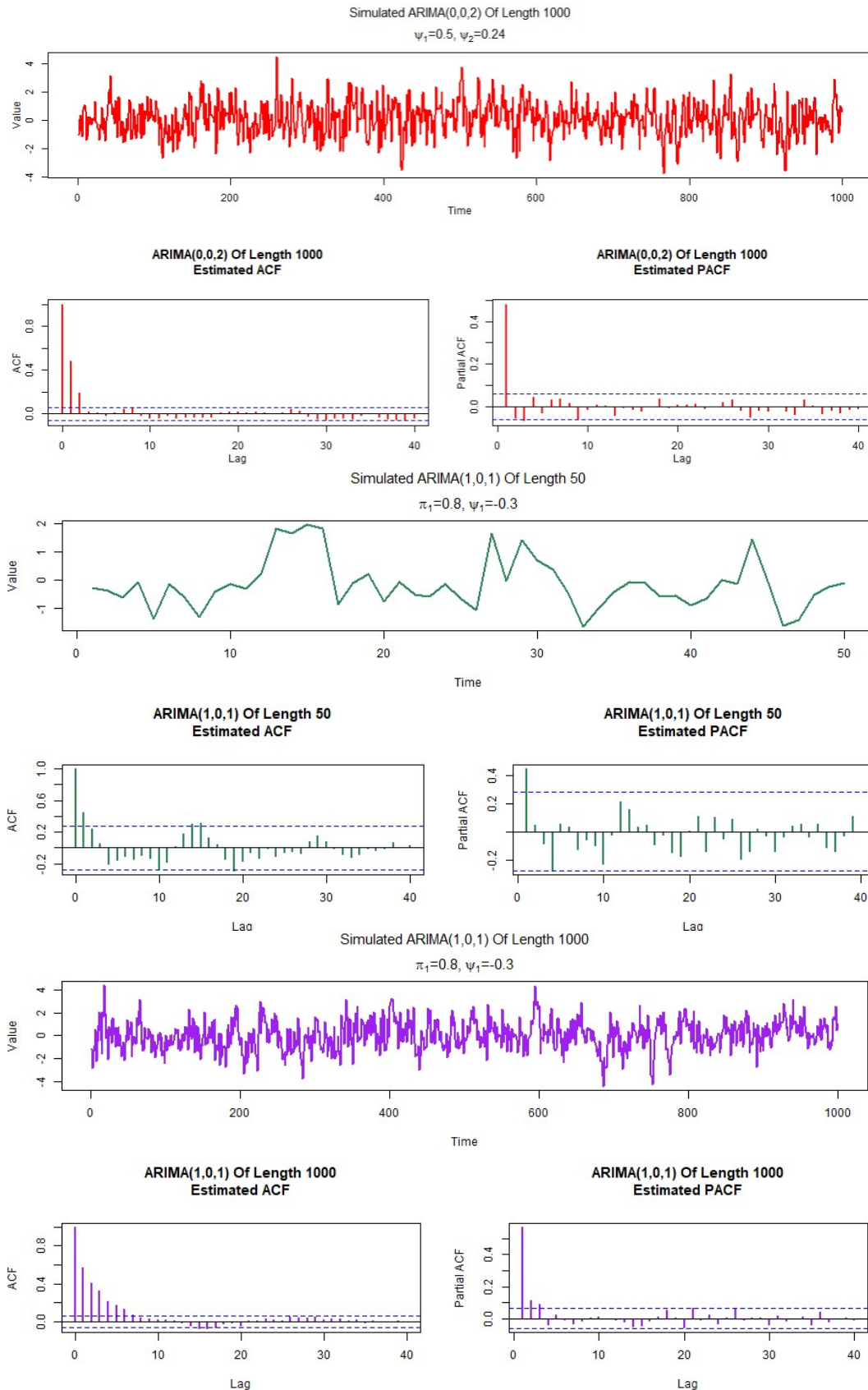
```

48- #####4. Plot Results Along with ACF and PACF#####
49- ###4a. ARIMA(0,0,2) short###
50- layout(matrix(c(1,1,2,3), nrow=2, ncol=2, byrow=TRUE))
51- plot(m1_short,
52-      main=bquote(atop(
53-        paste("Simulated ARIMA(0,0,2) of Length ", .(length1)),
54-        paste(psi[1], "=", .(m1coefma[1]), ", ", psi[2], "=", .(m1coefma[2])))),
55-      xlab="Time",
56-      ylab="value",
57-      lwd=2,
58-      lty=1,
59-      col="blue")
60-
61- acf(m1_short, lag.max=40,
62-     main=paste0("ARIMA(0,0,2) of Length ", length1, "\n", "Estimated ACF"),
63-     ci.col="blue",
64-     col="blue",
65-     lwd=2)
66- pacf(m1_short, lag.max=40,
67-      main=paste0("ARIMA(0,0,2) of Length ", length1, "\n", "Estimated PACF"),
68-      ci.col="blue",
69-      col="blue",
70-      lwd=2)

```

The output for the four models are shown below. Note that with smaller sample sizes, the estimated ACF and PACF is not as reliable. For example, the ACF of the MA(2) model spikes at lags out to 10 when there is only 50 data points, whereas with more data, the ACF seems to cut off after the second lag, as would be expected theoretically.





c. Show whether or not the model is stationary, and whether or not the model is invertible. If the model is stationary, do the following:

The MA representation of a model, say $\tilde{Z}_t = \sum_{j=0}^{\infty} \psi_j a_{t-j} = \psi(B)a_t$, is stationary when $\sum_{j=0}^{\infty} |\psi_j| < \infty$. It is invertible when the roots of $\psi(B)$ lie outside the unit circle. Similarly, the AR representation of a model, say $\tilde{Z}_t = a_t + \sum_{j=1}^{\infty} \pi_j Z_{t-j} \implies \pi(B)\tilde{Z}_t = a_t$, is invertible when $\sum_{j=1}^{\infty} |\pi_j| < \infty$. It is stationary when the roots of $\pi(B)$ lie outside the unit circle. Logically, an ARMA model $\pi(B)\tilde{Z}_t = \psi(B)a_t$ is stationary when the roots of $\psi(B)$ lie outside the unit circle, and invertible when the roots of $\pi(B)$ lie outside the unit circle.

The first model, $\tilde{Z}_t = a_t + 0.5a_{t-1} + 0.24a_{t-2} = (1 + 0.5B + 0.24B^2)a_t$, is an MA(2) and so is automatically stationary. It is also invertible, since its roots are $\frac{-0.5 \pm \sqrt{0.5^2 - 4(0.24)(1)}}{2(0.24)}$ or $\frac{-0.5 \pm i\sqrt{0.71}}{0.48} = \frac{-0.5}{0.48} \pm \frac{\sqrt{0.71}}{0.48}i$, and the complex modulus of this is $\sqrt{\left(\frac{-0.5}{0.48}\right)^2 + \left(\frac{\sqrt{0.71}}{0.48}\right)^2}$ or better yet $\sqrt{\frac{0.25}{0.48^2} + \frac{0.71}{0.48^2}}$. The numerator of the second term in the square root is larger than its denominator, so the entire value in the square root is greater than one and thus the entire expression is greater than one.

The second model, $\tilde{Z}_t = 0.8Z_{t-1} + a_t - 0.3a_{t-1} \implies (1 - 0.8B)\tilde{Z}_t = (1 - 0.3B)a_t$, is both stationary and invertible since the roots of $\pi(B)$ and $\psi(B)$ both lie inside the unit circle. To see this, see that the constant in both functions is 1, and the coefficient to the B terms being subtracted are both below 1 (so the root must be above 1).

i. Determine theoretical autocorrelations (the true model values) ρ_1, ρ_2 , and ρ_3 as well as partial autocorrelations ϕ_{11}, ϕ_{22} , and ϕ_{33} for each of the two models.

The covariance function for the MA(2) is given by:

$\gamma_1 = \mathbb{E}(\tilde{Z}_t \tilde{Z}_{t-1})$	Definition
$= \mathbb{E}((a_t + 0.5a_{t-1} + 0.24a_{t-2})(a_{t-1} + 0.5a_{t-2} + 0.24a_{t-3}))$	Substitution
$= \mathbb{E}(0.5a_{t-1}^2 + (0.24)(0.5)a_{t-2}^2)$	Ignore off diagonal terms
$= 0.5\sigma_a^2 + 0.12\sigma_a^2$	Expectations are linear

and $\gamma_2 = \mathbb{E}((a_t + 0.5a_{t-1} + 0.24a_{t-2})(a_{t-2} + 0.5a_{t-3} + 0.24a_{t-4})) = \mathbb{E}(0.24a_{t-2}^2) = 0.24\sigma_a^2$. Note that we can ignore the off-diagonal terms since $\mathbb{E}(a_t a_{t+k}) = 0$ when $k \neq 0$ by the definition of white-noise. The variance of model is:

$\gamma_0 = \mathbb{V}(\tilde{Z}_t) = \mathbb{V}(a_t + 0.5a_{t-1} + 0.24a_{t-2})$	Substitution
$= \mathbb{V}(a_t) + 0.5^2\mathbb{V}(a_{t-1}) + 0.24^2\mathbb{V}(a_{t-2})$	No covariance between terms
$= \sigma_a^2(1 + 0.5^2 + 0.24^2)$	Constant variance assumption

In general, the autocorrelations at lag k are $\rho_k = \frac{\gamma_k}{\gamma_0}$ and thus $\rho_1 = \frac{0.5\sigma_a^2 + 0.12\sigma_a^2}{\sigma_a^2(1 + 0.5^2 + 0.24^2)} \approx 0.47$, $\rho_2 = \frac{0.24\sigma_a^2}{\sigma_a^2(1 + 0.5^2 + 0.24^2)} \approx 0.18$, and, since the process is an MA(2), $\rho_3 = 0$.

In general, we can compute the partial autocorrelations as $\phi_{k,k} = \frac{\begin{vmatrix} 1 & \rho_1 & \cdots & \rho_{k-2} & \rho_1 \\ \rho_1 & 1 & \cdots & \rho_{k-3} & \rho_2 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \cdots & \rho_1 & \rho_k \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \cdots & \rho_{k-2} & \rho_{k-1} \\ \rho_1 & 1 & \cdots & \rho_{k-3} & \rho_{k-2} \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \rho_{k-1} & \rho_{k-2} & \cdots & \rho_1 & 1 \end{vmatrix}}.$

Here, we have:

$$\begin{aligned} \phi_{1,1} &= \rho_1 \approx 0.47 \\ \phi_{2,2} &= \frac{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{vmatrix}} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} \approx \frac{0.18 - (0.47^2)}{1 - 0.47^2} \approx -0.05 \\ \phi_{3,3} &= \frac{\begin{vmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_2 \\ \rho_2 & \rho_1 & 1 \end{vmatrix}}{\begin{vmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_1 \\ \rho_2 & \rho_1 & 1 \end{vmatrix}} = \frac{(1)(\rho_3 - \rho_1\rho_2) - (\rho_1)(\rho_1\rho_3 - \rho_2^2) + (\rho_1)(\rho_1^2 - \rho_2)}{(1)(1 - \rho_1^2) - (\rho_1)(\rho_1 - \rho_1\rho_2) + (\rho_2)(\rho_1^2 - \rho_2)} \approx -0.09 \end{aligned}$$

We can double check our calculations with the below:

```
> macoef=c(.5,.24)
> round(ARMAacf(ma=macoef, lag.max=5),2)
  0  1  2  3  4  5
1.00 0.47 0.18 0.00 0.00 0.00
> round(ARMAacf(ma=macoef, lag.max=5, pacf=TRUE),2)
[1] 0.47 -0.05 -0.09 0.06 -0.01
```

We compute the theoretical autocovariance of the ARMA(1,1) as follows:

$$\begin{aligned}
 \gamma_k &= \mathbb{E}(\tilde{Z}_t \tilde{Z}_{t-k}) && \text{Definition} \\
 &= \mathbb{E}\left(0.8\tilde{Z}_{t-1}\tilde{Z}_{t-k} + \tilde{Z}_{t-k}a_t - 0.3\tilde{Z}_{t-k}a_{t-1}\right) && \text{Multiplying through by } \tilde{Z}_{t-k} \\
 &= 0.8\mathbb{E}(\tilde{Z}_{t-1}\tilde{Z}_{t-k}) + \mathbb{E}(\tilde{Z}_{t-k}a_t) - 0.3\mathbb{E}(\tilde{Z}_{t-k}a_{t-1}) && \text{Expectations are linear} \\
 &= \begin{cases} 0.8\gamma_0 - 0.3\sigma_a^2, & k = 1 \\ 0.8\gamma_1, & k = 2 \\ 0.8\gamma_2, & k = 3 \end{cases} && \text{Properties of white noise}
 \end{aligned}$$

We can compute the variance as follows:

$$\begin{aligned}
 \mathbb{V}(\tilde{Z}_t) &= \mathbb{V}\left(0.8\tilde{Z}_{t-1} + a_t - 0.3a_{t-1}\right) && \text{Substitution} \\
 &= 0.8^2\mathbb{V}(\tilde{Z}_{t-1}) + \sigma_a^2 + 0.3^2\sigma_a^2 + 2((0.8)(-0.3)\sigma_a^2) \\
 &= \frac{(1 + 0.3^2 - 0.48)\sigma_a^2}{1 - 0.8^2} = \frac{0.61\sigma_a^2}{0.36} && \text{Weak Stationarity}
 \end{aligned}$$

On the second line, we use the general formula for the variance of a linear combination, $\mathbb{V}\left(\sum_{i=1}^n c_i X_i\right) = \sum_{i=1}^n c_i^2 \mathbb{V}(X_i) + 2 \sum_{i=1}^n \sum_{j:j>i}^n c_i c_j \text{Cov}(X_i, X_j)$, and the observation that the covariance between the other two random variables in the double sum are zero by the properties of white-noise.

The autocorrelations are $\rho_1 = \frac{\gamma_1}{\gamma_0}$, so $\rho_1 = \frac{0.8\left(\frac{0.61\sigma_a^2}{0.36}\right) - 0.3\sigma_a^2}{\frac{0.61\sigma_a^2}{0.36}} = 0.8 - \frac{0.3 \cdot 0.36}{.61} \approx 0.62$, $\rho_2 \approx 0.8(0.62) \approx 0.50$, and $\rho_3 \approx 0.8(0.5) \approx 0.4$.

To find the partial autocorrelations, we use the same process as before:

$$\begin{aligned}
 \phi_{1,1} &= \rho_1 \approx 0.62 \\
 \phi_{2,2} &= \frac{\left| \begin{bmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{bmatrix} \right|}{\left| \begin{bmatrix} 1 & \rho_1 \\ \rho_1 & 1 \end{bmatrix} \right|} = \frac{\rho_2 - \rho_1^2}{1 - \rho_1^2} \approx \frac{0.5 - (0.62^2)}{1 - 0.62^2} \approx 0.18 \\
 \phi_{3,3} &= \frac{\left| \begin{bmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_2 \\ \rho_2 & \rho_1 & 1 \end{bmatrix} \right|}{\left| \begin{bmatrix} 1 & \rho_1 & \rho_2 \\ \rho_1 & 1 & \rho_2 \\ \rho_2 & \rho_1 & 1 \end{bmatrix} \right|} = \frac{(1)(\rho_3 - \rho_1\rho_2) - (\rho_1)(\rho_1\rho_3 - \rho_2^2) + (\rho_1)(\rho_1^2 - \rho_2)}{(1)(1 - \rho_1^2) - (\rho_1)(\rho_1 - \rho_1\rho_2) + (\rho_2)(\rho_1^2 - \rho_2)} \approx 0.05
 \end{aligned}$$

We verify our calculations with the below:

```

> ARmacoef=0.8
> arMAcoef=-0.3
> armallacf=round(ARMAacf(ar=ARmacoef, ma=arMAcoef, lag.max=10),2)
> armallpacf=round(ARMAacf(ar=ARmacoef, ma=arMAcoef, lag.max=10, pacf=TRUE),2)
> armallacf
  0    1    2    3    4    5    6    7    8    9   10
1.00 0.62 0.50 0.40 0.32 0.26 0.20 0.16 0.13 0.10 0.08
> armallpacf
[1] 0.62 0.18 0.05 0.02 0.00 0.00 0.00 0.00 0.00 0.00 0.00

```

ii. Discuss what you observe as far as the proximity of the estimated autocorrelations values to the true values.

We see that with an increased sample, the estimated ACF moves closer to the theoretical ACF. The comparison between the two models at the two different sample sizes are shown below.

```
> #####6. Comparison####
> acf_comparison=data.frame(
+   mylag=1:10,
+   Estimated_MA_50=as.numeric(m1_shortacf[1:10]$acf),
+   Estimated_MA_1000=as.numeric(m1_longacf[1:10]$acf),
+   Theoretical_MA=as.numeric(ma2acf[2:11]),
+   Estimated_ARMA_50=as.numeric(m2_shortacf[1:10]$acf),
+   Estimated_ARMA_1000=as.numeric(m2_longacf[1:10]$acf),
+   Theoretical_ARMA=as.numeric(arma11acf[2:11])
+ )
>
> round(acf_comparison,2)
```

	mylag	Estimated_MA_50	Estimated_MA_1000	Theoretical_MA	Estimated_ARMA_50	Estimated_ARMA_1000	Theoretical_ARMA
1	1	0.27	0.48	0.47	0.45	0.57	0.62
2	2	0.36	0.19	0.18	0.23	0.40	0.50
3	3	-0.08	0.02	0.00	0.05	0.33	0.40
4	4	-0.05	0.01	0.00	-0.21	0.22	0.32
5	5	-0.11	-0.01	0.00	-0.15	0.17	0.26
6	6	-0.12	0.02	0.00	-0.10	0.12	0.20
7	7	-0.17	0.04	0.00	-0.15	0.07	0.16
8	8	-0.20	0.05	0.00	-0.09	0.04	0.13
9	9	0.00	-0.01	0.00	-0.13	0.02	0.10
10	10	0.00	-0.04	0.00	-0.27	0.02	0.08

2) Let $\tilde{Z}_t = 0.4\tilde{Z}_{t-1} + 0.21\tilde{Z}_{t-2} + a_t + 0.7a_{t-1} + 0.12a_{t-2}$. Is the model in its reduced form? If not, write the model in reduced form.

The above model can be written as $\tilde{Z}_t - 0.4\tilde{Z}_{t-1} - 0.21\tilde{Z}_{t-2} = a_t + 0.7a_{t-1} + 0.12a_{t-2}$ or equivalently $(1 - 0.4B - 0.21B^2)\tilde{Z}_t = (1 + 0.7B + 0.12B^2)a_t$.

We can compute the roots of $\pi(B) = (1 - 0.4B - 0.21B^2)$ as:

$$\frac{0.4 \pm \sqrt{(0.4^2) - 4(-0.21)(1)}}{2(-0.21)} = \frac{0.4 \pm \sqrt{1}}{-0.42} = \left(\frac{-0.6}{-0.42}, \frac{1.4}{-0.42} \right) = \left(\frac{10}{7}, -\frac{10}{3} \right)$$

And the roots of $\psi(B) = (1 + 0.7B + 0.12B^2)$ as:

$$\frac{-0.7 \pm \sqrt{(0.7^2) - 4(0.12)(1)}}{2(0.12)} = \frac{-0.7 \pm \sqrt{0.01}}{0.24} = \frac{-\frac{7}{10} \pm \frac{1}{10}}{0.24} = \left(-\frac{10}{3}, -\frac{5}{2} \right)$$

In general, we can tell if a model is in reduced form if there are no common roots in the $\psi(B)$ and $\pi(B)$ polynomials. Here, there is one common root, $-\frac{10}{3}$. So we can write:

$$\begin{aligned} (1 - 0.4B - 0.21B^2)\tilde{Z}_t &= (1 + 0.7B + 0.12B^2)a_t \\ -0.21\left(B - \frac{10}{7}\right)\left(B + \frac{10}{3}\right)\tilde{Z}_t &= 0.12\left(B + \frac{10}{3}\right)\left(B + \frac{5}{2}\right)a_t \\ -0.21\left(B - \frac{10}{7}\right)\tilde{Z}_t &= 0.12\left(B + \frac{5}{2}\right)a_t \\ -0.21\tilde{Z}_{t-1} + 0.3\tilde{Z}_t &= 0.12a_{t-1} + 0.3a_t \\ -0.7\tilde{Z}_{t-1} + \tilde{Z}_t &= 0.4a_{t-1} + 1a_t \end{aligned}$$

And we see that the reduced model is $\tilde{Z}_t = 0.7\tilde{Z}_{t-1} + a_t + 0.4a_{t-1}$

3) For the following two time series models, determine if $W_t = (1 - B)\tilde{Z}_t$ is stationary and if it is invertible. $(1 - B)\tilde{Z}_t = a_t - a_{t-1}$ and $(1 - B)^2\tilde{Z}_t = a_t - 0.81a_{t-1} + 0.38a_{t-2}$.

The first model is $W_t = (1 - B)\tilde{Z}_t = a_t - a_{t-1}$, which can be written $W_t = \psi(B)a_t = (1 - B)a_t$. This is an MA(2), and since all finite MA processes are stationary, W_t is stationary. The process is invertible if the roots of $\psi(B)$ lie outside the unit circle. Since the root is 1, the process is not invertible.

The second model is $W_t = (1 - B)\tilde{Z}_t = (a_t - 0.81a_{t-1} + 0.38a_{t-2})(1 - B)^{-1}$ which can be written $W_t = \psi(B)a_t = \frac{(1 - 0.81B + 0.38B^2)}{(1 - B)}a_t$. The denominator has a root of 1, so $\psi(B)$ is infinite in extent and fails to be absolutely summable; W_t is non-stationary. The function $\psi(B)$ has a root at $B = x$ if and only if the numerator of the function has a root at the same $B = x$. So we can use the quadratic formula and identify the roots as $\frac{0.81 \pm \sqrt{0.81^2 - 4(0.38)(1)}}{2(0.38)} = \frac{0.81 \pm i\sqrt{0.8639}}{0.76}$. The complex modulus is then $\sqrt{\left(\frac{0.81}{0.76}\right)^2 + \left(\frac{0.8639}{0.76^2}\right)} > \sqrt{\frac{0.8639}{0.76^2}} > 1$, and so the process is invertible.

2.3 Reduced Forms, MA and AR Conversions

1) Consider the following models:

$$A : (1 - B)\tilde{Z}_t = (1 - 1.5B)a_t$$

$$B : (1 - 0.8B)\tilde{Z}_t = (1 - 0.5B)a_t$$

$$C : (1 - 1.1B + 0.8B^2)\tilde{Z}_t = (1 - 1.7B + 0.72B^2)a_t$$

$$D : (1 - 0.6B)\tilde{Z}_t = (1 - 1.2B + 0.2B^2)a_t$$

a. Verify whether or not the model for Z_t is stationary and/or invertible.

An ARMA model is stationary when all the roots of it's AR polynomial lie outside the complex unit circle. From inspection, that means models B and D are stationary. We can find the roots of the AR polynomial for model C as $\frac{1.1 \pm \sqrt{1.1^2 - 4(0.8)(1)}}{2(0.8)} = \frac{1.1 \pm \sqrt{-1.99}}{1.6} = \frac{11}{16} \pm \frac{i\sqrt{1.99}}{1.6}$ and so the complex modulus is $\sqrt{\frac{121}{256} + \frac{199}{256}} > \sqrt{1} = 1$, which means model C is also stationary. All told, models B, C, and D are stationary.

An ARMA model is invertible if all the roots of it's MA polynomial lie outside the complex unit circle. From inspection, that means model B is invertible. We can find the roots of the MA polynomial for model C as $\frac{1.7 \pm \sqrt{1.7^2 - 4(0.72)(1)}}{2(0.72)} = \frac{1.7 \pm \sqrt{2.89 - 2.88}}{1.44} = \frac{1.7 \pm 0.1}{1.44} > 1$ and so model C is also invertible. We can find the roots of the MA polynomial for model D as $\frac{1.2 \pm \sqrt{1.2^2 - 4(0.2)(1)}}{2(0.2)} = \frac{1.2 \pm 0.8}{0.4}$; one of the roots lies on the unit circle and so model D is not invertible. All told, models B and C are invertible.

b. Express the model as an infinite MA if the process is stationary.

We equate the coefficients of the backshift operator to get the expression. In the case of Model B, we have:

$$(1 - 0.8B)\tilde{Z}_t = (1 - 0.5B)a_t$$

$$(1 - 0.8B)(1 + \psi_1 B + \psi_2 B^2 + \dots) a_t = (1 - 0.5B)a_t$$

And so:

$$\begin{aligned}\psi_1 B - 0.8B &= -0.5B \implies \psi_1 = 0.3 \\ \psi_2 B^2 - 0.8\psi_1 B^2 &= 0B^2 \implies \psi_2 = 0.24 \\ \psi_3 B^3 - 0.8\psi_2 B^3 &= 0B^3 \implies \psi_3 = 0.8(0.24)\end{aligned}$$

Continuing in this fashion, we have: $\psi_{B_k} = \begin{cases} 0.3, & k = 1 \\ 0.8(0.3)^{k-1}, & k > 1 \end{cases}$

In the case of model C, we have:

$$(1 - 1.1B + 0.8B^2)\tilde{Z}_t = (1 - 1.7B + 0.72B^2)a_t$$

$$(1 - 1.1B + 0.8B^2)(1 + \psi_1 B + \psi_2 B^2 + \dots) = (1 - 1.7B + 0.72B^2)a_t$$

And so:

$$\begin{aligned}\psi_1 B - 1.1B &= -1.7B \implies \psi_1 = -0.6 \\ \psi_2 B^2 - 1.1\psi_1 B^2 + 0.8B^2 &= 0.72B^2 \implies \psi_2 = 0.72 + (1.1 \cdot -0.6) - 0.8 = -0.74 \\ \psi_3 B^3 - 1.1\psi_2 B^3 + 0.8\psi_1 B^3 &= 0B^3 \implies \psi_3 = 1.1(-0.74) - 0.8(-0.6)\end{aligned}$$

Continuing in this way, we achieve the recursion $\psi_{C_k} = 1.1(\psi_{C_{k-1}}) - 0.8(\psi_{C_{k-2}})$ for $k \geq 3$ where $\psi_{C_{k_1}} = -0.6$ and $\psi_{C_{k_2}} = -0.74$.

In the case of model D, we have:

$$(1 - 0.6B)\tilde{Z}_t = (1 - 1.2B + 0.2B^2)a_t$$

$$(1 - 0.6B)(1 + \psi_1 B + \psi_2 B^2 + \dots) = (1 - 1.2B + 0.2B^2)a_t$$

And so:

$$\begin{aligned}\psi_1 B - 0.6B &= -1.2B \implies \psi_1 = -0.6 \\ \psi_2 B^2 - 0.6\psi_1 B^2 &= 0.2B^2 \implies \psi_2 = 0.2 + 0.6(-0.6) = -0.16 \\ \psi_3 B^3 - 0.6\psi_2 B^3 &= 0 \implies \psi_3 = 0.6(-0.16)\end{aligned}$$

Continuing in this way, we see $\psi_{D_k} = 0.6^{k-2}(-0.16)$ for $k \geq 3$ while $\psi_{D_1} = -0.6$ and $\psi_{D_2} = -0.16$.

c. Express the model as an infinite AR representation if the process is invertible.

We use the same strategy of equating coefficients. In the case of model B, we have:

$$\begin{aligned}(1 - 0.8B)\tilde{Z}_t &= (1 - 0.5B)a_t \\ (1 - 0.8B)\tilde{Z}_t &= (1 - 0.5B)(1 + \pi_1 B + \pi_2 B^2 + \cdots)\tilde{Z}_t\end{aligned}$$

And so:

$$\begin{aligned}-0.8B &= \pi_1 B - 0.5B \implies \pi_1 B = -0.3B \implies \pi_1 = -0.3 \\ 0 &= \pi_2 B^2 - 0.5\pi_1 B^2 \implies \pi_2 = (0.5)(-0.3) \\ 0 &= \pi_3 B^2 - 0.5\pi_2 B^2 \implies \pi_3 = 0.5\pi_2 \implies \pi_3 = 0.5(0.5)(-0.3)\end{aligned}$$

Continuing in this way, we have $\pi_k = 0.5^{k-1}(-0.3)$.

In the case of model C, we have:

$$\begin{aligned}(1 - 1.1B + 0.8B^2)\tilde{Z}_t &= (1 - 1.7B + 0.72B^2)a_t \\ (1 - 1.1B + 0.8B^2)\tilde{Z}_t &= (1 - 1.7B + 0.72B^2)(1 + \pi_1 B + \pi_2 B^2 + \cdots)\tilde{Z}_t\end{aligned}$$

And so:

$$\begin{aligned}-1.1B &= \pi_1 B - 1.7B \implies \pi_1 = 0.6 \\ 0.8B^2 &= \pi_2 B^2 - 1.7\pi_1 B^2 + 0.72B^2 \implies \pi_2 = 0.8 - 0.72 + 1.7(0.6) = 1.1 \\ 0 &= \pi_3 B^3 - 1.7\pi_2 B^3 + 0.72\pi_1 B^3 \implies \pi_3 = 1.7(1.1) + 0.72(0.6) = 2.302\end{aligned}$$

Continuing in this fashion, we see $\pi_{C_k} = \begin{cases} 0.6, & k = 1 \\ 1.1, & k = 2 \\ 1.7\pi_{C_{k-1}} - 0.72\pi_{C_{k-2}}, & k \geq 3 \end{cases}$

2) Consider the model $\tilde{Z}_t = 0.3\tilde{Z}_{t-1} + 0.34\tilde{Z}_{t-2} - 0.12\tilde{Z}_{t-3} + a_t - 0.7a_{t-1} + 0.12a_{t-2}$. Determine whether or not the model is in reduced form, and if it is not, find the reduced form.

The model is in reduced form if the AR and MA polynomials share no common factors. We go directly for the definition. The model is $(1 - 0.3B - 0.34B^2 + 0.12B^3)\tilde{Z}_t = (1 - 0.7B + 0.12B^2)a_t$. The quadratic does not divide directly into the cubic, so we try to find shared linear factors. The quadratic factors as $(1 - 0.4B)(1 - 0.3B)$, so we try to find if either of these divides into the cubic.

Upon conducting polynomial long-division with the $(-0.4B + 1)$ term, the quotient starts with $-0.3B^2$. The product of $-0.3B^2$ and $(-0.4B + 1)$ is $0.12B^3 - 0.3B^2$, which, upon being subtracted from the dividend $0.12B^3 + 0.34B^2 - 0.3B + 1$ is $-0.04B^2 - 0.3B + 1$. The next term in the quotient is then $0.1B$, which leaves a remainder of $-0.4B + 1$; we can factor the cubic as $(1 - 0.4B)(1 + 0.1B - 0.3B^2)$.

So we have reduced the model to $(1 + 0.1B - 0.3B^2)\tilde{Z}_t = (1 - 0.3B)a_t$. A quick check reveals that the remaining linear term does not factor into the quadratic. So the total reduced model is $\tilde{Z}_t = 0.1\tilde{Z}_{t-1} + 0.3\tilde{Z}_{t-2} + a_t - 0.3a_{t-1}$.

3) Consider the model $(1 - B)^2 Z_t = (1 - 0.3B - 0.5B^2)a_t$.

a. Is the model for Z_t a stationary model? Why or why not?

Z_t is not stationary since its AR model has two roots that lie *on* the complex unit circle.

b. Is the model for $W_t = (1 - B)^2 Z_t$ a stationary model? Why or why not?

W_t is stationary since it is a finite MA model, and finite MA's are always stationary.

c. Determine the autocorrelation function for W_t .

The autocovariance function is $\mathbb{E}(W_t W_{t-k})$:

$$\begin{aligned}\mathbb{E}(W_t W_{t-k}) &= \mathbb{E}((a_t - 0.3a_{t-1} - 0.5a_{t-2})(a_{t-k} - 0.3a_{t-k-1} - 0.5a_{t-k-2})) \\ &= \mathbb{E}(a_t a_{t-k}) - 0.3\mathbb{E}(a_t a_{t-k-1}) - 0.5\mathbb{E}(a_t a_{t-k-2}) \\ &\quad - 0.3\mathbb{E}(a_{t-1} a_{t-k}) + 0.09\mathbb{E}(a_{t-1} a_{t-k-1}) + 0.15\mathbb{E}(a_{t-1} a_{t-k-2}) \\ &\quad - 0.5\mathbb{E}(a_{t-2} a_{t-k}) + 0.15\mathbb{E}(a_{t-2} a_{t-k-1}) + 0.25\mathbb{E}(a_{t-2} a_{t-k-2})\end{aligned}$$

By the properties of white noise, $\mathbb{E}(a_t a_{t-k}) = \begin{cases} \sigma_a^2, & k = 0 \\ 0, & k > 0 \end{cases}$. So:

$$\begin{aligned}\gamma_0 &= \sigma_a^2(1 + 0.09 + 0.25) = 1.34\sigma_a^2 \\ \gamma_1 &= \sigma_a^2(-0.3 + 0.15) = -0.15\sigma_a^2 \\ \gamma_2 &= \sigma_a^2(-0.5) = -0.5\sigma_a^2 \\ \gamma_3 &= \sigma_a^2() \\ \gamma_4 &= 0\end{aligned}$$

. Dividing by γ_0 gives us the correlation function (here $s > 2$):

$$\begin{aligned}\rho_0 &= 1 \\ \rho_1 &= \frac{-0.15}{1.34} \approx -0.11 \\ \rho_2 &= \frac{-0.5}{1.34} \approx -0.37 \\ \rho_s &= 0\end{aligned}$$

2.4 Analyzing Datasets, White-Noise Tests, Forecasting

1) Analyze the third data set on Moodle.

a. Determine possible models for the data set using diagnostics such as the ACF, PACF, and white noise test. Include a unit root test and discuss those results as well. Include relevant plots and tables with your submission.

Our first step is to plot the data. We show it in Figure 2.1 below.

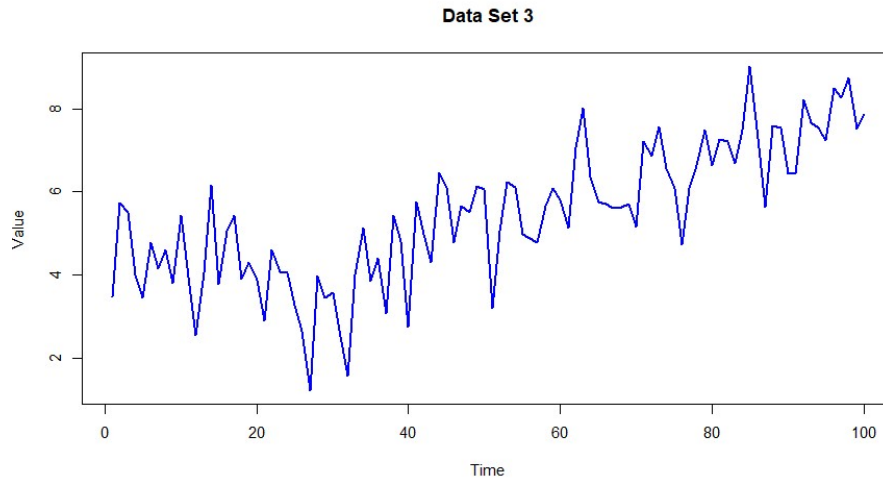


Figure 2.1: Posted Time Series Data

There is no obvious change in variance. The data may have some trending (e.g. from time point 30 to 100), but it is not abundantly obvious either way. Additionally, there might be some seasonality (the gaps between local peaks and valleys is approximately equal).

Before proceeding further, we should test if a model is needed in the first place (i.e. if the series is just a random walk). We use the Ljung-Box Q Test in Figure 2.2 below at lags of 6 and 12 to determine if a fit is needed. With p-values near machine-epsilon, we can comfortably reject white-noise at any reasonable significance level α .

```
> whitenoise6=Box.test(ts_data3,          #Do we need to fit model?#
+                       lag=6,
+                       type="Ljung-Box")
> whitenoise6                               #p small \implies yes#

Box-Ljung test

data:  ts_data3
X-squared = 254.02, df = 6, p-value < 2.2e-16

> whitenoise12=Box.test(ts_data3,
+                       lag=12,
+                       type="Ljung-Box")
> whitenoise12

Box-Ljung test

data:  ts_data3
X-squared = 422.81, df = 12, p-value < 2.2e-16
```

Figure 2.2: Ljung-Box Q Test For White Noise

Now that we know we need to fit a model, we use R's built in `ACF()` and `PACF()` functions to get an idea of which models we want to try fitting. The code is shown in Figure 2.3 below.

```

20 ###1b. Plot ACF and PACF###
21 ts_data3=ts(data3$val)           #Convert from DF to Time Series Object#
22 length2=nrow(data3)
23
24 par(mfrow=c(1,2))
25 data3_acf=acf(ts_data3, lag.max=40,
26               main=paste0("Time Series Data Of Length ", length2, "\n", "Estimated ACF"),
27               ci.col="blue",
28               col="red",
29               lwd=4)
30 data3_acf
31
32
33 data3_pacf=pacf(ts_data3, lag.max=40,
34                main=paste0("Time Series Data Of Length ", length2, "\n", "Estimated PACF"),
35                ci.col="blue",
36                col="red",
37                lwd=4)
38
39 data3_pacf

```

Figure 2.3: R Code For ACF And PACF Functions

The plots in Figure 2.4 are the result. Notice how the ACF dies out slowly, while the PACF seems to significantly cut off at the second lag; a natural choice for our model is an $AR(2)$. While that second lag has the largest partial autocorrelation, the PACF does not completely die out. To account for the PACF's reluctance to cut off, some type of ARIMA model may be necessary.

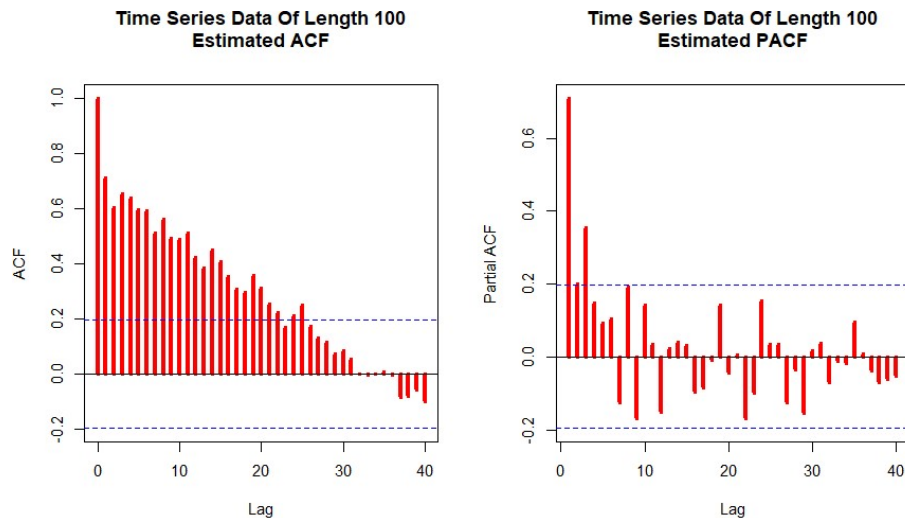


Figure 2.4: Autocorrelation And Partial Autocorrelation Of Data

To test if a difference is needed, we use the Augmented Dickey-Fuller Test in Figure 2.5 below. Under a significance level of $\alpha = 0.05$, we fail to reject the null hypothesis of “there is a unit root” ($p = 0.21$). As such, we will fit ARIMA models in addition to our $AR(2)$.

```

> ###1c. Augmented Dickey-Fuller (Unit Root Test)###
> adf_result=adf.test(ts_data3)           #Null is that there is a root#
> adf_result                               #since p is 0.2, don't reject null; assume non-stationary#

Augmented Dickey-Fuller Test

data: ts_data3
Dickey-Fuller = -2.8748, Lag order = 4, p-value = 0.2142
alternative hypothesis: stationary

```

Figure 2.5: R Code For Augmented Dickey-Fuller Test

b. Fit the models that you identified as good possibilities and compare their fits using output diagnostics such as the residual test for white noise, AIC, SBC, etc.

After differencing the time series like indicated in the above, we see the following series.

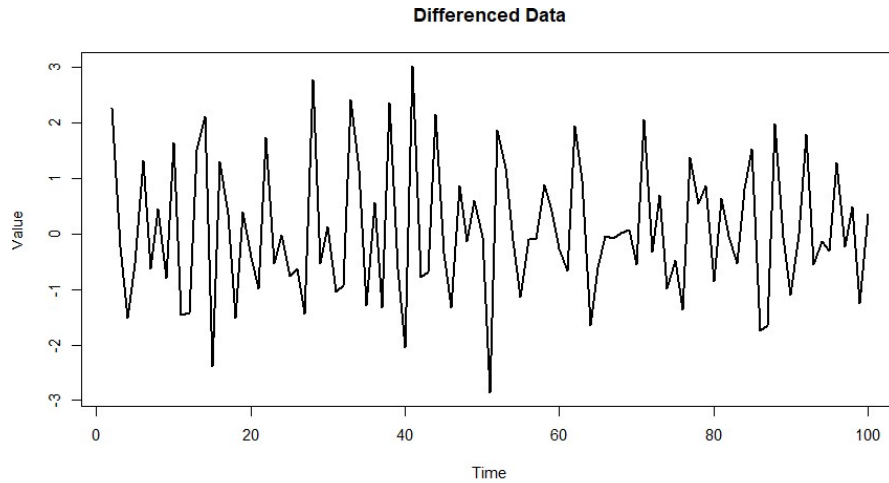


Figure 2.6: Plot Of Differenced Time Series

At least visually, this series shows better signs of weak stationarity than our first plot. Nevertheless we will proceed with our fitting of an AR(2) for comparison purposes.

The ACF and PACF for the residuals of the AR(2) model, plotted in Figure 2.7 still show signs of a signal.

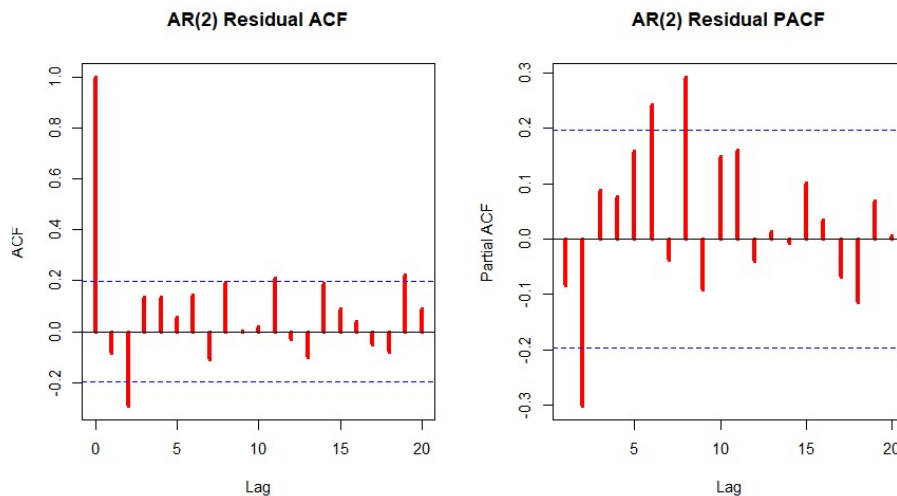


Figure 2.7: AR(2) Residual ACF and PACF

Our model diagnostics show that one cannot reject the presence of a signal at any significance level greater than $\alpha = 0.01$. The AIC for the model is about 314 while the BIC is about 325. These results are shown in Figure 2.8 below.

```

> AIC(ar2)
[1] 314.3033
> BIC(ar2)
[1] 324.7239
> result_ar2=Box.test(resid_ar2, lag=20, type="Ljung-Box")
> result_ar2                                     #p-value still low \implies need to fit more#

Box-Ljung test

data: resid_ar2
X-squared = 41.797, df = 20, p-value = 0.002939

```

Figure 2.8: AR(2) Model Diagnostics

We can now try a variety of ARIMA models to see which one gives us the best fit. The model diagnostics are shown in Figure 2.9. The best model, in terms of all three of AIC (285.56), BIC (295.94), and Ljung-Box Q Test ($p = 0.48$), is an ARIMA(2,1,1) model.

```

> #ARIMA Model Diagnostics#
> ARIMA_model=vector()
> aic=vector()
> bic=vector()
> LBtest=vector()
>
> for (p in 1:3) {
+   for (q in 1:3) {
+     model=arima(ts_data3, order=c(p-1,1,q-1))
+     resid=residuals(model)
+
+     ARIMA_model[3*(p-1)+q]=paste0("ARIMA(", p-1, ",1,", q-1, ")")
+     aic[3*(p-1)+q]=round(AIC(model),2)
+     bic[3*(p-1)+q]=round(BIC(model),2)
+     LBtest[3*(p-1)+q]=round(Box.test(resid, lag=21, type="Ljung-Box")$p.value,2)
+   }
+ }
>
> df=data.frame(ARIMA_model, aic, bic, LBtest)
> df
  ARIMA_model  aic    bic LBtest
1 ARIMA(0,1,0) 322.44 325.03  0.00
2 ARIMA(0,1,1) 289.13 294.32  0.29
3 ARIMA(0,1,2) 288.72 296.50  0.24
4 ARIMA(1,1,0) 314.78 319.97  0.00
5 ARIMA(1,1,1) 289.99 297.78  0.27
6 ARIMA(1,1,2) 288.82 299.20  0.39
7 ARIMA(2,1,0) 294.31 302.10  0.02
8 ARIMA(2,1,1) 285.56 295.94  0.48
9 ARIMA(2,1,2) 287.51 300.49  0.48

```

Figure 2.9: ARIMA Model Diagnostics

We see the ACF and PACF of the residuals for our ARIMA(2,1,1) model in Figure 2.10.

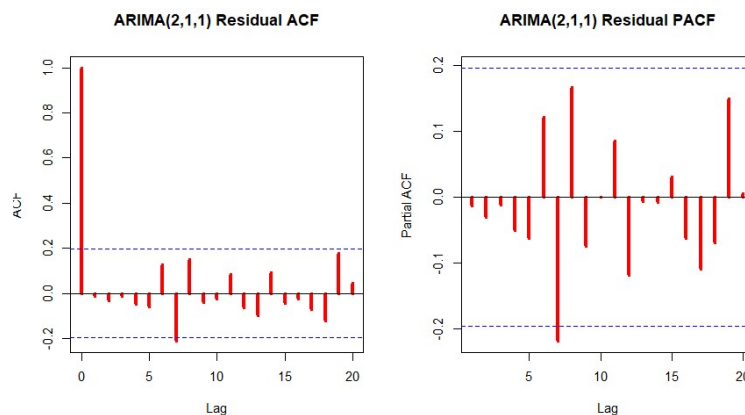


Figure 2.10: ARIMA(2,1,1) Residual ACF

c. Use your model to forecast the series 12 time units into the future.

In totality, our model is $(1 - 0.214B + 0.3056B^2)(1 - B)Z_t = (1 + 0.6546)a_t$. The results come from the code in Figure 2.11 below.

```
> #ARIMA(2,1,1)#
> arima211=arima(ts_data3, order=c(2,1,1))
> summary(arima211)

Call:
arima(x = ts_data3, order = c(2, 1, 1))

Coefficients:
      ar1      ar2      ma1
    0.0214 -0.3056 -0.6546
s.e.  0.1338  0.1138  0.1153

sigma^2 estimated as 0.9568:  log likelihood = -138.78,  aic = 285.56

Training set error measures:
      ME      RMSE      MAE      MPE      MAPE      MASE      ACF1
Training set 0.1414653 0.9732733 0.7752467 -2.498994 17.7722 0.7913862 -0.01409562
```

Figure 2.11: Coefficients Of ARIMA(2,1,1) Model

We can forecast the next 12 time units into the future with R's `predict()` function. The 101st value is predicted to be about 8.15, the 102nd value is predicted to be about 8.05, and so on until the 112th value is predicted to be about 8.00.

```
> pred=predict(arima211, n.ahead = 12)
> pred
$pred
Time Series:
Start = 101
End = 112
Frequency = 1
[1] 8.147981 8.045737 7.954332 7.983623 8.012179 8.003839 7.994934 7.997293 8.000064 7.999403 7.998542 7.998725

$se
Time Series:
Start = 101
End = 112
Frequency = 1
[1] 0.9781764 1.0418812 1.0429226 1.0678142 1.1172006 1.1505038 1.1759539 1.2041510 1.2338354 1.2618514 1.2886182 1.3151066
```

Figure 2.12: Forecast Next 12 Values From ARIMA(2,1,1)

2) Consider the AR(2) model $(1 - 1.2B + 0.6B^2)(Z_t - 65) = a_t$ where $\sigma_a^2 = 1$ and we have the observations are $Z_{76} = 60.4, Z_{77} = 58.9, Z_{78} = 64.7, Z_{79} = 70.4$, and $Z_{80} = 62.6$.

a. Forecast Z_{81}, Z_{82}, Z_{83} , and Z_{84} .

We first write out the model as $Z_t = 65 + 1.2(Z_{t-1} - 65) - 0.6(Z_{t-2} - 65) + a_t$. Our forecast for l steps in the future is: $\widehat{Z}_{80}(l) = 65 + 1.2(\widehat{Z}_{80+l-1} - 65) - 0.6(\widehat{Z}_{80+l-2} - 65)$. Explicitly:

$$\begin{aligned}\widehat{Z}_{80}(1) &= 65 + 1.2(62.6 - 65) - 0.6(70.4 - 65) = 58.88 \\ \widehat{Z}_{80}(2) &= 65 + 1.2(58.88 - 65) - 0.6(62.6 - 65) = 59.096 \\ \widehat{Z}_{80}(3) &= 65 + 1.2(59.096 - 65) - 0.6(58.88 - 65) = 61.5872 \\ \widehat{Z}_{80}(4) &= 65 + 1.2(61.5872 - 65) - 0.6(59.096 - 65) = 64.44704\end{aligned}$$

b. Determine the 95% forecast limits for the forecasts in part a.

The standard error for our forecast is given by $\sqrt{\sum_{j=0}^{l-1} \psi_j^2}$. This follows from the fact that $\mathbb{V}(e_n(l)) = \sigma_a^2 \sum_{j=0}^{l-1} \psi_j^2$ and we are given $\sigma_a^2 = \sigma_a = 1$. The critical value is $z_{\alpha/2}$ which is the value with $\frac{\alpha}{2}$ of the mass of the standard normal distribution to its right (a choice of $\alpha = 0.05$ yields about 1.96). It remains to be seen what our ψ weights are from the MA representation of the above model. We can write out our model as $(1 - 1.2B + 0.6B^2)(1 + \psi_1 B + \psi_2 B^2 + \dots)a_t = a_t$. Equating the B coefficients, we find (where $n \geq 3$):

$$\begin{aligned}\psi_1 B - 1.2B &= 0 \implies \psi_1 = 1.2 \\ \psi_2 B^2 - 1.2\psi_1 B^2 + 0.6B^2 &= 0 \implies \psi_2 = 1.2(1.2) - 0.6 = 0.86 \\ \psi_n B^n - 1.2\psi_{n-1} B^n + 0.6\psi_{n-2} B^n &\implies \psi_n = 1.2\psi_{n-1} - 0.6\psi_{n-2}\end{aligned}$$

Then the forecast limits (point-estimate plus/minus margin of error) are approximately:

$$\begin{aligned}\widehat{Z}_{80}(1) : 58.88 \pm 1.96 \left(\sum_{j=0}^0 \psi_j^2 \right)^{1/2} &\approx 58.88 \pm 1.96 (1)^{1/2} \approx (56.92, 60.84) \\ \widehat{Z}_{80}(2) : 59.10 \pm 1.96 \left(\sum_{j=0}^1 \psi_j^2 \right)^{1/2} &\approx 59.10 \pm 1.96 (1^2 + 1.2^2)^{1/2} \approx (56.03, 62.16) \\ \widehat{Z}_{80}(3) : 61.59 \pm 1.96 \left(\sum_{j=0}^2 \psi_j^2 \right)^{1/2} &\approx 61.59 \pm 1.96 (1^2 + 1.2^2 + 0.86^2)^{1/2} \approx (58.09, 65.08) \\ \widehat{Z}_{80}(4) : 64.45 \pm 1.96 \left(\sum_{j=0}^3 \psi_j^2 \right)^{1/2} &\approx 64.45 \pm 1.96 (1^2 + 1.2^2 + 0.86^2 + 0.312^2)^{1/2} \approx (60.90, 68.00)\end{aligned}$$

c. Suppose that the observations at $t = 81$ turns out to be $Z_{81} = 62.2$. Determine the updated forecasts Z_{82} , Z_{83} , and Z_{84} .

We have:

$$\begin{aligned}
 \widehat{Z}_{81}(1) &= \widehat{Z}_{80}(2) + \psi_1 [Z_{81} - \widehat{Z}_{80}(1)] \\
 &= 59.096 + 1.2 [62.2 - 58.88] \\
 &= 63.08 \\
 \widehat{Z}_{81}(2) &= \widehat{Z}_{80}(3) + \psi_2 [Z_{81} - \widehat{Z}_{80}(1)] \\
 &= 61.5872 + 0.86 [62.2 - 58.88] \\
 &= 64.4424 \\
 \widehat{Z}_{81}(3) &= \widehat{Z}_{80}(4) + \psi_3 [Z_{81} - \widehat{Z}_{80}(1)] \\
 &= 64.44704 + 0.312 [62.2 - 58.88] \\
 &= 65.48288
 \end{aligned}$$

3) A sales series was fitted by the ARIMA(2,1,0) model $(1 - 0.14B + 0.48B^2)(1 - B)Z_t = a_t$ where $\sigma_a^2 = 58000$ and the last three observations are $Z_{n-2} = 640$, $Z_{n-1} = 770$, and $Z_n = 800$.

a. Calculate the forecast of the next three observations.

We can write the model as $(1 - 0.14B + 0.48B^2 - B + 0.14B^2 - 0.48B^3)Z_t = a_t$ or equivalently $Z_t = 1.14Z_{t-1} - 0.62Z_{t-2} + 0.48Z_{t-3} + a_t$. Our forecast for l steps in the future is: $\widehat{Z}_n(l) = 1.14\widehat{Z}_{n+l-1} - 0.62\widehat{Z}_{n+l-2} + 0.48\widehat{Z}_{n+l-3}$. Explicitly:

$$\begin{aligned}\widehat{Z}_n(1) &= 1.14(800) - 0.62(770) + 0.48(640) = 741.8 \\ \widehat{Z}_n(2) &= 1.14(741.8) - 0.62(800) + 0.48(770) = 719.252 \\ \widehat{Z}_n(3) &= 1.14(719.252) - 0.62(741.8) + 0.48(800) = 744.0313\end{aligned}$$

b. Calculate the 95% forecast limits for the forecasts in part a.

We know we can write $(1 - 1.14B + 0.62B^2 - 0.48B^3)Z_t = a_t$. Writing Z_t in terms of it's AR representation, we have $(1 - 1.14B + 0.62B^2 - 0.48B^3)(1 + \psi_1B + \psi_2B^2 + \dots) = a_t$. Equating coefficients of B , we arrive at our AR coefficients. We have:

$$\begin{aligned}\psi_1B - 1.14B &= 0 \implies \psi_1 = 1.14 \\ \psi_2B^2 - 1.14\psi_1B^2 + 0.62B^2 &= 0 \implies \psi_2 = 1.14(1.14) - 0.62 = 0.6796 \\ \psi_3B^3 - 1.14\psi_2B^3 + 0.62\psi_1B^3 - 0.48B^3 &= 0 \implies \psi_3 = 1.14(0.6796) + 0.62(1.14) + 0.48 = 1.961544\end{aligned}$$

$$\text{Since } \mathbb{V}(e_n(l)) = \sigma_a^2 \sum_{j=0}^{l-1} \psi_j^2 \text{ and we are given } \sigma_a^2 = 58000, \text{ our standard error is } 240.8319 \left(\sum_{j=0}^{l-1} \psi_j^2 \right)^{\frac{1}{2}}.$$

As such, our forecast limits are approximately:

$$\begin{aligned}\widehat{Z}_n(1) : 741.8 \pm (1.96)240.8 \left(\sum_{j=0}^0 \psi_j^2 \right)^{\frac{1}{2}} &\approx 741.8 \pm 471.97 (1^2)^{\frac{1}{2}} \approx (269.83, 1213.77) \\ \widehat{Z}_n(2) : 719.25 \pm (1.96)240.8 \left(\sum_{j=0}^1 \psi_j^2 \right)^{\frac{1}{2}} &\approx 719.25 \pm 471.97 (1^2 + 1.14^2)^{\frac{1}{2}} \approx (3.54, 1434.96) \\ \widehat{Z}_n(3) : 744.03 \pm (1.96)240.8 \left(\sum_{j=0}^2 \psi_j^2 \right)^{\frac{1}{2}} &\approx 744.03 \pm 471.97 (1^2 + 1.14^2 + 0.6796^2)^{\frac{1}{2}} \approx (-40.27, 1528.33)\end{aligned}$$

2.5 Model Diagnostics, ARIMA Models, Unit Root Tests

1) Analyze the fourth data set on Moodle.

a. Determine possible models for the data set using diagnostics such as the ACF, PACF, and white noise test. Include a unit root test and discuss those results as well.

Our first step is to plot the data, which we do in Figure 2.13 below.

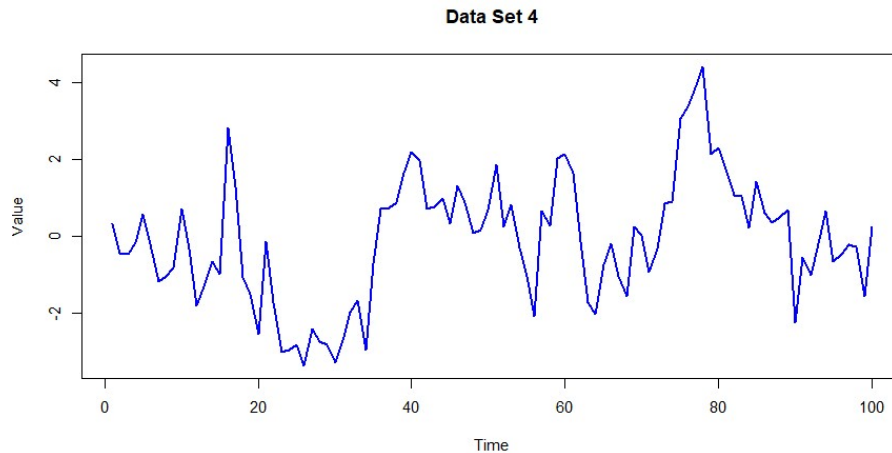


Figure 2.13: Data Set 4 Time Series

The Ljung-Box White Noise test has a p-value on the order of machine-epsilon for lags of 6 and 12 (see Figure 2.14 below)– we need to fit a model.

```
> whitenoise12=Box.test(ts_data4,
+                        lag=12,
+                        type="Ljung-Box")
> whitenoise12

Box-Ljung test

data: ts_data4
X-squared = 168.44, df = 12, p-value < 2.2e-16
```

Figure 2.14: White Noise Test For Data Set 4

The Augmented-Dickey Fuller test in Figure 2.15 suggests that we might need to take a difference.

```
> ##2b. Augmented Dickey-Fuller (Unit Root Test)##
> adf_result_data4=adf.test(ts_data4)      #Null is that there is a root#
> adf_result_data4                        #since p is 0.2, don't reject null; assume non-stationary#

Augmented Dickey-Fuller Test

data: ts_data4
Dickey-Fuller = -2.8152, Lag order = 4, p-value = 0.2389
alternative hypothesis: stationary
```

Figure 2.15: Augmented-Dickey Fuller Test (Unit Root Test)

After doing so, our ACF and PACF for the differenced data are shown in Figure 2.16. Notice that neither the ACF nor PACF seems to completely die off.

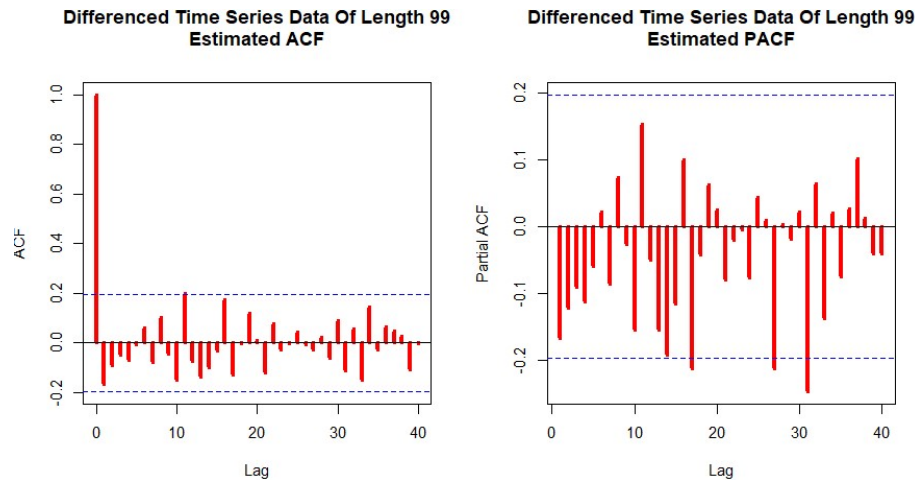


Figure 2.16: ACF And PACF Of Differenced Data

Again performing the white-noise test, this time on the differenced series, we see that the model is not distinguishable from white noise in Figure 2.17.

```
> diff_data4=diff(ts_data4, differences = 1)
> whitenoise6=Box.test(diff_data4,                #Do we need to fit model?#
+                      lag=6,
+                      type="Ljung-Box")
> whitenoise6                                     #large p \implies no#

Box-Ljung test

data: diff_data4
X-squared = 4.8704, df = 6, p-value = 0.5605

> whitenoise12=Box.test(diff_data4,                #Do we need to fit model?#
+                      lag=12,
+                      type="Ljung-Box")
> whitenoise12                                    #large p \implies no#

Box-Ljung test

data: diff_data4
X-squared = 14.568, df = 12, p-value = 0.2659
```

Figure 2.17: White Noise Test For Differenced Data

Our assessment of the model agrees with the `auto.arima()` function from R's `forecast` package; it recommends an ARIMA(0,1,0) model.

```
> auto.arima(ts_data4)
Series: ts_data4
ARIMA(0,1,0)

sigma^2 = 1.28: log likelihood = -152.69
AIC=307.38 AICC=307.42 BIC=309.97
```

Figure 2.18: `auto.arima()` For Data Set 4

b. Fit the models that you identified as good possibilities and compare their fits using output diagnostics such as the residual test for white noise, AIC, SBC, etc.

Just for thoroughness, we test a few different models up to order 3 in Figure 2.19.

```
###2d. Try A Few Different Models###
ARIMA_model=vector()
aic=vector()
bic=vector()
LBtest=vector()

for (p in 1:4) {
  for (d in 1:2) {
    for (q in 1:4) {
      model=arima(ts_data4,
                  order=c(p-1,d-1,q-1))
      resid=residuals(model)

      ARIMA_model[4*2*(p-1)+4*(d-1)+q]=paste0("ARIMA(", p-1, ", ", d-1, ", ", q-1, ")")
      aic[4*2*(p-1)+4*(d-1)+q]=round(AIC(model),2)
      bic[4*2*(p-1)+4*(d-1)+q]=round(BIC(model),2)
      LBtest[4*2*(p-1)+4*(d-1)+q]=round(Box.test(resid, lag=21, type="Ljung-Box")$p.value,2)
    }
  }
}

df=data.frame(ARIMA_model, aic, bic, LBtest)
df=df[order(df$aic),]
df
```

Figure 2.19: R Code To Derive Model Diagnostics

Figure 2.20 below sorts our diagnostics from lowest to highest AIC values.

```
> df=df[order(df$aic),]
> df
```

	ARIMA_model	aic	bic	LBtest
14	ARIMA(1,1,1)	301.39	309.17	0.47
9	ARIMA(1,0,0)	301.54	309.36	0.52
15	ARIMA(1,1,2)	302.93	313.31	0.51
22	ARIMA(2,1,1)	302.97	313.35	0.50
10	ARIMA(1,0,1)	303.21	313.63	0.55
17	ARIMA(2,0,0)	303.23	313.65	0.54
27	ARIMA(3,0,2)	304.75	322.99	0.64
23	ARIMA(2,1,2)	304.77	317.75	0.54
16	ARIMA(1,1,3)	304.79	317.77	0.56
30	ARIMA(3,1,1)	304.83	317.81	0.56
18	ARIMA(2,0,1)	305.13	318.16	0.58
11	ARIMA(1,0,2)	305.14	318.16	0.59
25	ARIMA(3,0,0)	305.15	318.18	0.59
6	ARIMA(0,1,1)	305.54	310.73	0.33
7	ARIMA(0,1,2)	305.70	313.49	0.52
32	ARIMA(3,1,3)	306.38	324.55	0.56
13	ARIMA(1,1,0)	306.50	311.69	0.24
31	ARIMA(3,1,2)	306.55	322.12	0.46
24	ARIMA(2,1,3)	306.77	322.34	0.55
21	ARIMA(2,1,0)	307.05	314.83	0.46
8	ARIMA(0,1,3)	307.09	317.47	0.42
12	ARIMA(1,0,3)	307.13	322.76	0.59
19	ARIMA(2,0,2)	307.13	322.76	0.59
26	ARIMA(3,0,1)	307.13	322.76	0.59
5	ARIMA(0,1,0)	307.38	309.97	0.15
29	ARIMA(3,1,0)	308.25	318.63	0.41
20	ARIMA(2,0,3)	309.13	327.37	0.58
28	ARIMA(3,0,3)	310.65	331.49	0.57
4	ARIMA(0,0,3)	313.15	326.17	0.06
3	ARIMA(0,0,2)	320.13	330.55	0.00
2	ARIMA(0,0,1)	334.72	342.53	0.00
1	ARIMA(0,0,0)	384.68	389.89	0.00

Figure 2.20: Model Diagnostics For Data Set 4

In terms of AIC, the ARIMA(0,1,0) is actually one of the worst performing models. If one model had to be chosen, we would prefer the ARIMA(1,0,0) for the sake of parsimony. It is only slightly worse than the ARIMA(1,1,1) model (the combined difference in AIC and BIC in the two models is less than 0.5) while having a higher Ljung-Box p-value and two less parameters.

2) Analyze the quarterly beer data set on Moodle.

a. Determine possible models for the data using diagnostics such as the ACF and white noise test. Include a unit root test and discuss those results as well.

Our first step is to plot the data, which we do in Figure 2.21 below.

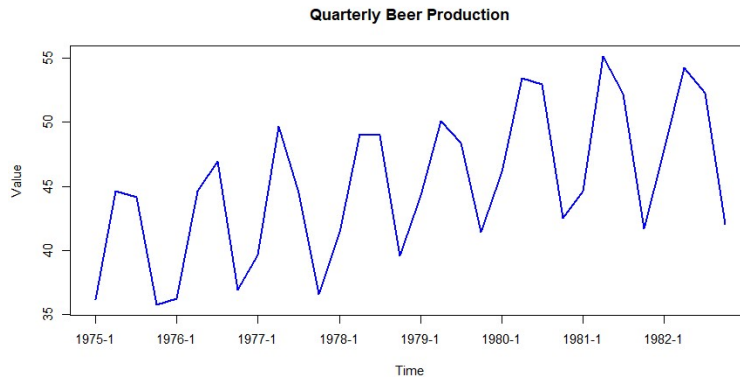


Figure 2.21: Quarterly Beer Time Series

It is clear from inspection that we have seasonal data with period 4. We are dealing with a limited amount of data (thirty-two total observations with a period of 4 means 8 seasonal observations), but at least visually, it seems that the series is trending; all but one of the seven points is larger than it's previous value. Applying the Augmented Dickey-Fuller Test at seasonal increments provides evidence for the alternative hypothesis that the seasonal lags are actually stationary. This is shown in Figure 2.22 below along with the differenced plot. We will test both when building our models.

```
> ts_beer_seasonal=ts_beer[seq(2, length(ts_beer), by=4)] #Only look at every 4th#
> ts_beer_seasonal
[1] 44.60 44.63 49.72 49.07 50.09 53.44 55.18 54.27
> sadf_result=suppresswarnings(adf.test(ts_beer_seasonal))
> sadf_result
```

Augmented Dickey-Fuller Test

```
data: ts_beer_seasonal
Dickey-Fuller = -5.5155, Lag order = 1, p-value = 0.01
alternative hypothesis: stationary
```

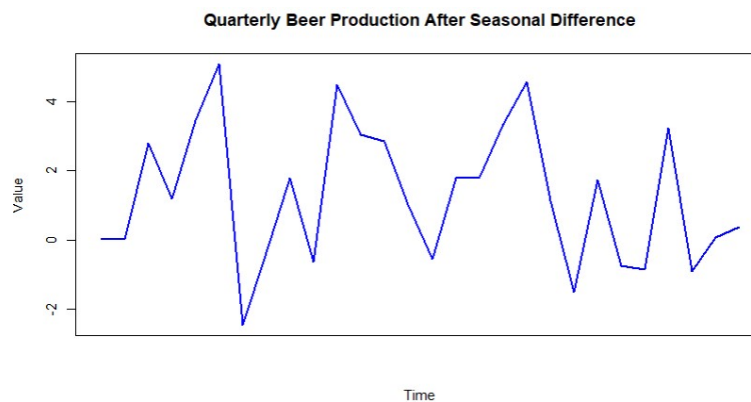


Figure 2.22: Quarterly Beer Time Series After Seasonal Difference

Our initial ACF and PACF plots are shown in Figure 2.23 below. The seasonal lags in the ACF plot seem to quickly die out while the first seasonal lag in the PACF seems pronounced and subsequently cuts off. A natural choice for the seasonal portion of the model is an MA(1).

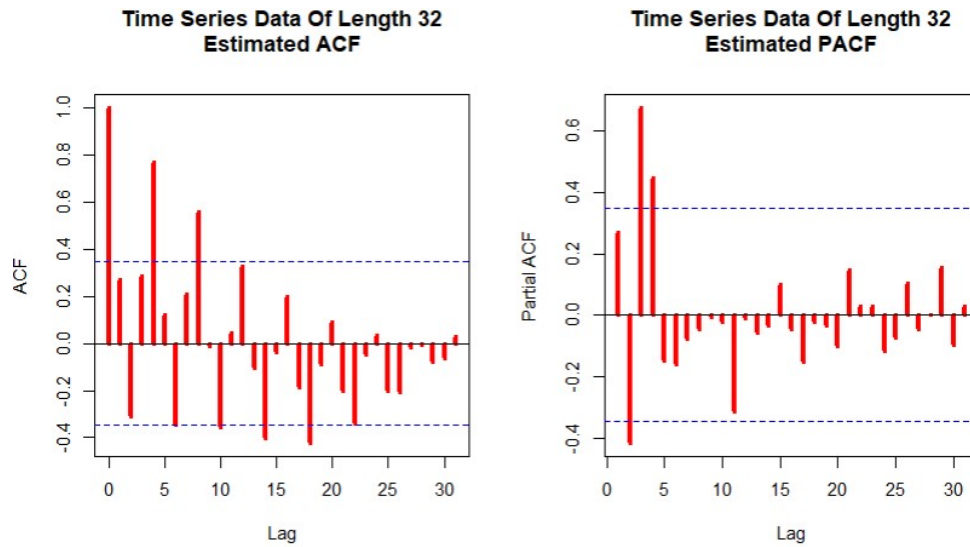


Figure 2.23: ACF And PACF Of Quarterly Beer

We can also look at the ACF and PACF after taking a seasonal difference. This is shown in Figure 2.24. Notice that the ACF has a semi-gradual sinusoidal decay, and the PACF cuts off after the first seasonal lag. An argument could be made that neither really cuts off but instead gradually decays. From that perspective, some choices for the seasonal portion could be an ARIMA(1,1,0), and ARIMA(0,1,1), or an ARIMA(1,1,1)

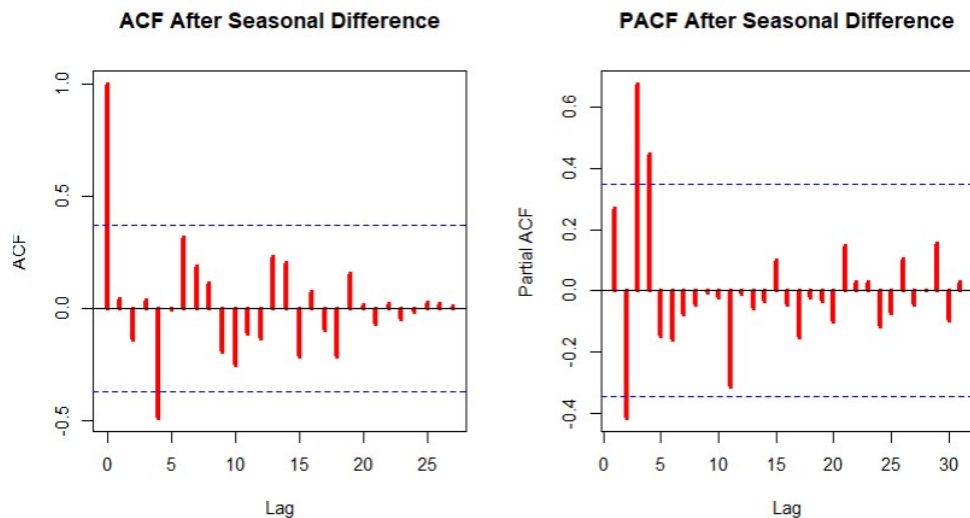


Figure 2.24: ACF And PACF After Seasonal Difference

We now try to determine the non-seasonal part of the model. We can test if a difference is needed by using the Augmented Dickey-Fuller test. Under any reasonable alpha level, we fail to reject the null hypothesis of "there is a unit root". The results are shown in Figure 2.25 below.

```
> adf_result_beer=adf.test(ts_beer)
> adf_result_beer

Augmented Dickey-Fuller Test

data: ts_beer
Dickey-Fuller = -1.4835, Lag order = 3, p-value = 0.7721
alternative hypothesis: stationary
```

Figure 2.25: ADF For Beer Data

The ACF and PACF, after taking a difference, are shown in Figure 2.26 below. Notice that the ACF dies out immediately, while the PACF cuts off after the third lag.

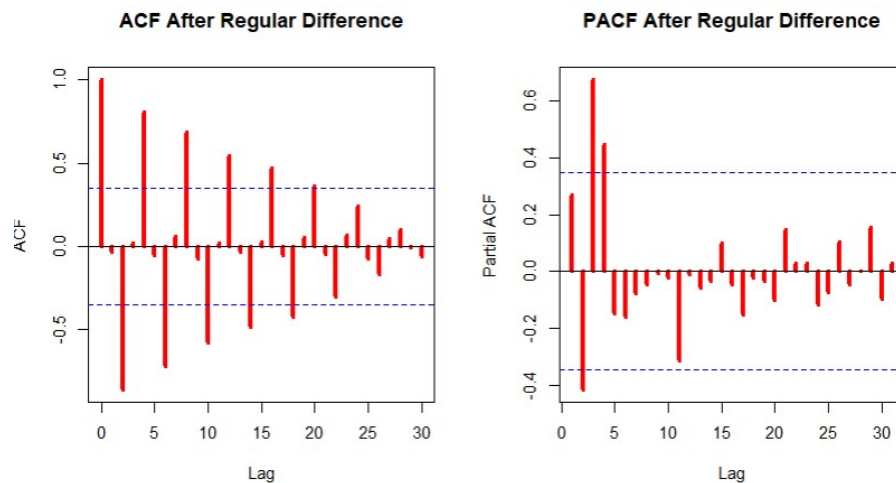


Figure 2.26: ACF And PACF After Regular Difference

Taken together, we have a couple of models that might be good fits: an $\text{ARIMA}(0,1,3)(0,0,1)$, an $\text{ARIMA}(0,1,3)(1,1,0)$, an $\text{ARIMA}(0,1,3)(0,1,1)$, and an $\text{ARIMA}(0,1,3)(1,1,1)$ all seem reasonable.

b. Fit the models that you identified as good possibilities and compare their fits using output diagnostics such as the residual test for white noise, AIC, SBC, etc.

For thoroughness, we test all models with p, q, P , and Q terms less than 4 with d and D terms less than 2. The script to run this is shown in Figure 2.27 below.

```
###3f. Try A Few Different Models###
ARIMAS_model=vector()
aic=vector()
bic=vector()
LBtest=vector()
s=4                                #clear from data#

for (p in 1:4) {
  for (d in 1:2) {
    for (q in 1:4) {
      for (P in 1:4) {
        for (D in 1:2) {
          for (Q in 1:4) {
            mycount=(((((p-1)*2 + (d-1))*4 + (q-1))*4 + (P-1))*2 + (D-1))*4 + Q
            mymodel=paste0("ARIMA(", p-1, ",", d-1, ",", q-1, ")(", P-1, ",", D-1, ",", Q-1, ")",s)

            tryCatch({
              model=arima(ts_beer,
                          order=c(p-1,d-1,q-1),
                          seasonal=list(order=c(P-1,D-1,Q-1), period=s))
              resid=residuals(model)

              ARIMAS_model[mycount]=mymodel
              aic[mycount]=round(AIC(model),2)
              bic[mycount]=round(BIC(model),2)
              LBtest[mycount]=round(Box.test(resid, lag=21, type="Ljung-Box")$p.value,2)
            }, error=function(e) {
              ARIMAS_model[mycount]=mymodel
              aic[mycount]=999
              bic[mycount]=999
              LBtest[mycount]=999
            })
          }
        }
      }
    }
  }
}

df=data.frame(ARIMAS_model, aic, bic, LBtest)
df=df[order(df$aic),]
df
```

Figure 2.27: Script For Fitting Models

The top ten models in terms of AIC are shown in Figure 2.28 below. Of the models we planned to test in part 2a, the ARIMA(0,1,3)(1,1,1) actually was the second best model overall. Right behind was our ARIMA(0,1,3)(0,1,1) model. Since the data was so short, the marginally better AIC and BIC from the model with a second seasonal MA term is not as convincing as the ARIMA(0,1,3)(1,1,1) we proposed. For that reason, we would prefer that model best of all.

```
> df[1:10,]
      ARIMAS_model    aic    bic LBtest
231 ARIMA(0,1,3)(0,1,2)4 111.78 119.55  0.82
238 ARIMA(0,1,3)(1,1,1)4 111.97 119.74  0.80
230 ARIMA(0,1,3)(0,1,1)4 112.28 118.75  0.35
245 ARIMA(0,1,3)(2,1,0)4 112.51 120.28  0.82
487 ARIMA(1,1,3)(0,1,2)4 112.80 121.87  0.86
486 ARIMA(1,1,3)(0,1,1)4 113.14 120.92  0.51
494 ARIMA(1,1,3)(1,1,1)4 113.20 122.27  0.81
237 ARIMA(0,1,3)(1,1,0)4 113.24 119.71  0.52
647 ARIMA(2,1,0)(0,1,2)4 113.63 120.11  0.74
166 ARIMA(0,1,1)(0,1,1)4 113.70 117.59  0.31
```

Figure 2.28: Model Selection Criteria For Beer Data

3 Appendix

3.1 Problem Set 1

```
#####Problem Set 1#####
ar_coefs=c(1.2, -.8)                                #Z_t=1.2Z_{t-1}-0.8Z_{t-2}+a_t#
round(ARMAacf(ar=ar_coefs, lag.max=5),3)

ar_coefs=c(0.6,-.68, .24)                            #Z_t=0.6Z_{t-1}-0.68Z_{t-2}+0.24Z_{t-3}+a_t#
round(ARMAacf(ar=ar_coefs, lag.max=5),3)

ar_coeff=.5                                           #Z_t=0.5Z_{t-1}+a_t#
round(ARMAacf(ar=ar_coeff),3)
ARMAacf(ar=ar_coeff, lag.max=3, pacf=TRUE)

ar_coeff=c(1.3, -.4)
round(ARMAacf(ar=ar_coeff, lag.max=5),3)
round(ARMAacf(ar=ar_coeff, lag.max=5, pacf=TRUE),3)

ar_coeff=c(1.2, -.8)
round(ARMAacf(ar=ar_coeff, lag.max=5),3)
round(ARMAacf(ar=ar_coeff, lag.max=5, pacf=TRUE),3)

coef=c(1,-.2,-.4)
polyroot(coef)
```

3.2 Problem Set 2

```
#####Problem Set 2#####
####1. Load Required Packages####
library(stats)

####2. Specify Parameters####
length1=50
length2=1000
burnin=101

#m1=a_t+0.5a_{t-1}+0.24a_{t-2}#
m1coefma=c(0.5, 0.24)      #ARIMA(0,0,2) process#

#m2=0.8Z_{t-1}+a_t-0.3a_{t-1}#
m2coefma=-0.3
m2coefar=0.8               #ARIMA(1,0,1) process#

####3. Specify Models####
set.seed(534)              #To make reproducible#
m1_short=arima.sim(model=list(ma=m1coefma),
n=length1,
n.start=burnin)

m1_long=arima.sim(model=list(ma=m1coef),
n=length2,
n.start=burnin)

m2_short=arima.sim(model=list(ar=m2coefar, ma=m2coefma),
n=length1,
n.start=burnin)

m2_long=arima.sim(model=list(ar=m2coefar, ma=m2coefma),
n=length2,
n.start=burnin)

####4. Plot Results Along With ACF and PACF####
####4a. ARIMA(0,0,2) short###
layout(matrix(c(1,1,2,3), nrow=2, ncol=2, byrow=TRUE))
plot(m1_short,
main=bquote(atop(
paste("Simulated ARIMA(0,0,2) Of Length ", .(length1)),
paste(psi[1], "=", .(m1coefma[1]), ", ", psi[2], "=", .(m1coefma[2])))),
xlab="Time",
ylab="Value",
lwd=2,
lty=1,
col="blue")

m1_shortacf=acf(m1_short, lag.max=40,
main=paste0("ARIMA(0,0,2) Of Length ", length1, "\n", "Estimated ACF"),
ci.col="blue",
col="blue",
lwd=2)
m1_shortacf

m1_shortpacf=pacf(m1_short, lag.max=40,
main=paste0("ARIMA(0,0,2) Of Length ", length1, "\n", "Estimated PACF"),
ci.col="blue",
col="blue",
lwd=2)
m1_shortpacf
```

```

###4b. ARIMA(0,0,2) long###
layout(matrix(c(1,1,2,3), nrow=2, ncol=2, byrow=TRUE))
plot(m1_long,
main=bquote(atop(
paste("Simulated ARIMA(0,0,2) Of Length ", .(length2)),
paste(psi[1], "=", .(m1coefma[1]), ", ", psi[2], "=", .(m1coefma[2])))),
xlab="Time",
ylab="Value",
lwd=2,
lty=1,
col="red")

m1_longacf=acf(m1_long, lag.max=40,
main=paste0("ARIMA(0,0,2) Of Length ", length2, "\n", "Estimated ACF"),
ci.col="blue",
col="red",
lwd=2)
m1_longacf

m1_longpacf=pacf(m1_long, lag.max=40,
main=paste0("ARIMA(0,0,2) Of Length ", length2, "\n", "Estimated PACF"),
ci.col="blue",
col="red",
lwd=2)
m1_longpacf

###4c. ARIMA(1,0,1) Short###
layout(matrix(c(1,1,2,3), nrow=2, ncol=2, byrow=TRUE))
plot(m2_short,
main=bquote(atop(
paste("Simulated ARIMA(1,0,1) Of Length ", .(length1)),
paste(pi[1], "=", .(m2coefar[1]), ", ", psi[1], "=", .(m2coefma[1])))),
xlab="Time",
ylab="Value",
lwd=2,
lty=1,
col="seagreen4")

m2_shortacf=acf(m2_short, lag.max=40,
main=paste0("ARIMA(1,0,1) Of Length ", length1, "\n", "Estimated ACF"),
ci.col="blue",
col="seagreen4",
lwd=2)
m2_shortacf

m2_shortpacf=pacf(m2_short, lag.max=40,
main=paste0("ARIMA(1,0,1) Of Length ", length1, "\n", "Estimated PACF"),
ci.col="blue",
col="seagreen4",
lwd=2)
m2_shortpacf

###4d. ARIMA(1,0,1) long###
plot(m2_long,
main=bquote(atop(
paste("Simulated ARIMA(1,0,1) Of Length ", .(length2)),
paste(pi[1], "=", .(m2coefar[1]), ", ", psi[1], "=", .(m2coefma[1])))),
xlab="Time",
ylab="Value",
lwd=2,

```

```

lty=1,
col="purple")

m2_longacf=acf(m2_long, lag.max=40,
main=paste0("ARIMA(1,0,1) Of Length ", length2, "\n", "Estimated ACF"),
ci.col="blue",
col="purple",
lwd=2)
m2_longacf

m2_longpacf=pacf(m2_long, lag.max=40,
main=paste0("ARIMA(1,0,1) Of Length ", length2, "\n", "Estimated PACF"),
ci.col="blue",
col="purple",
lwd=2)
m2_longpacf

#####5. Theoretical P/ACF#####
macoef=c(.5,.24)
ma2acf=round(ARMAacf(ma=macoef, lag.max=10),2)
ma2pacf=round(ARMAacf(ma=macoef, lag.max=10, pacf=TRUE),2)
ma2acf
ma2pacf

ARmacoef=0.8
arMAcoef=-0.3
arma1acf=round(ARMAacf(ar=ARmacoef, ma=arMAcoef, lag.max=10),2)
arma1pacf=round(ARMAacf(ar=ARmacoef, ma=arMAcoef, lag.max=10, pacf=TRUE),2)
arma1acf
arma1pacf

#####6. Comparison#####
acf_comparison=data.frame(
mylag=1:10,
Estimated_MA_50=as.numeric(m1_shortacf[1:10]$acf),
Estimated_MA_1000=as.numeric(m1_longacf[1:10]$acf),
Theoretical_MA=as.numeric(ma2acf[2:11]),
Estimated_ARMA_50=as.numeric(m2_shortacf[1:10]$acf),
Estimated_ARMA_1000=as.numeric(m2_longacf[1:10]$acf),
Theoretical_ARMA=as.numeric(arma1acf[2:11])
)

round(acf_comparison,4)

```

3.3 Problem Set 4

```
#####Problem Set 4#####
####1. Load Required Packages####
library(stats)
library(tseries)
path="Academics/Class/Problem Set/Problem Set 4/Data3.csv"
data3=read.csv(path)
summary(data3)
str(data3)
ts_data3=ts(data3$Val)                                #Convert from DF to Time Series Object#

###1a. EDA###
plot(data3,
type="l",
lwd="2",
col="blue",
main="Data Set 3",
xlab="Time",
ylab="Value")

whitenoise6=Box.test(ts_data3,                        #Do we need to fit model?#
lag=6,
type="Ljung-Box")
whitenoise6                                           #p small \implies yes#

whitenoise12=Box.test(ts_data3,
lag=12,
type="Ljung-Box")
whitenoise12

###1b. Plot ACF and PACF###
length2=nrow(data3)

par(mfrow=c(1,2))
data3_acf=acf(ts_data3, lag.max=40,
main=paste0("Time Series Data Of Length ", length2, "\n", "Estimated ACF"),
ci.col="blue",
col="red",
lwd=4)
data3_acf                                           #ACF dies out slowly#

data3_pacf=pacf(ts_data3, lag.max=40,
main=paste0("Time Series Data Of Length ", length2, "\n", "Estimated PACF"),
ci.col="blue",
col="red",
lwd=4)
data3_pacf                                           #PACF cuts off at lag 2#

###1c. Augmented Dickey-Fuller (Unit Root Test)###
adf_result=adf.test(ts_data3)                      #Null is that there is a root#
adf_result                                           #since p is 0.2, don't reject null; assume non-stationary#

diff_data3=diff(ts_data3, differences = 1)

par(mfrow=c(1,1))
plot(diff_data3,
main="Differenced Data",
xlab="Time",
ylab="Value",
col="black",
```

```

lwd=2)

par(mfrow=c(1,2))
data3diff_acf=acf(diff_data3, lag.max=40,
main=paste0("Differenced Time Series Data Of Length ", length2-1, "\n", "Estimated ACF"),
ci.col="blue",
col="red",
lwd=4)
data3diff_acf                                #ACF dies out slowly#

data3diff_pacf=pacf(diff_data3, lag.max=40,
main=paste0("Differenced Time Series Data Of Length ", length2-1, "\n", "Estimated PACF"),
ci.col="blue",
col="red",
lwd=4)
data3diff_pacf

###1d. Fitting Models###
#AR(2)#
ar2=arima(ts_data3, order=c(2,0,0))
summary(ar2)

resid_ar2=residuals(ar2)
acf(resid_ar2,
main="AR(2) Residual ACF",
col="red",
lwd=4)

pacf(resid_ar2,
main="AR(2) Residual PACF",
col="red",
lwd=4)

AIC(ar2)
BIC(ar2)
resid_ar2=residuals(ar2)
result_ar2=Box.test(resid_ar2, lag=20, type="Ljung-Box")
result_ar2                                #p-value still low \implies need to fit more#

#ARIMA Model Diagnostics#
ARIMA_model=vector()
aic=vector()
bic=vector()
LBtest=vector()

for (p in 1:3) {
  for (q in 1:3) {
    model=arima(ts_data3, order=c(p-1,1,q-1))
    resid=residuals(model)

    ARIMA_model[3*(p-1)+q]=paste0("ARIMA(", p-1, ",1,", q-1, ")")
    aic[3*(p-1)+q]=round(AIC(model),2)
    bic[3*(p-1)+q]=round(BIC(model),2)
    LBtest[3*(p-1)+q]=round(Box.test(resid, lag=21, type="Ljung-Box")$p.value,2)
  }
}

df=data.frame(ARIMA_model, aic, bic, LBtest)
df

#ARIMA(2,1,1)#
arima211=arima(ts_data3, order=c(2,1,1))
summary(arima211)

```

```
resid_arima211=residuals(arima211)
acf(resid_arima211,
main="ARIMA(2,1,1) Residual ACF",
col="red",
lwd=4)
pacf(resid_arima211,
main="ARIMA(2,1,1) Residual PACF",
col="red",
lwd=4)
```

```
###1e. Forecast###
pred=predict(arima211, n.ahead = 12)
pred
```

```
#####2. Problem 2#####
alpha=0.05                                #given#
qnorm(alpha/2, lower.tail=FALSE)          #critical value#
```

3.4 Problem Set 5

```
#####Problem Set 5#####
####1. Load Required Packages####
library(tidyverse)
library(stats)
library(tseries)
library(forecast)
path="Academics/ST534 Time Series/Homework/Homework 5/"

data4=read.csv(paste0(path, "data4.csv"))
summary(data4)
str(data4)
ts_data4=ts(data4$data)          #Convert from DF to Time Series Object#
n_data4=nrow(data4)
auto.arima(ts_data4)             #to give us an idea what to search for#

beer=read.csv(paste0(path, "beer.csv")) |>
mutate(yyyyq=paste0(year, "-", quarter))|>
select(yyyyq, data)
summary(beer)
str(beer)
ts_beer=ts(beer$data)           #Convert from DF to Time Series Object#
n_beer=nrow(beer)
auto.arima(ts_beer)             #to give us an idea what to search for#
s=4                             #The seasonality#

####2. Data4####
###2a. Plot Data (Ljung-Box says fit model)###
par(mfrow=c(1,1))
plot(data4,
type="l",
lwd="2",
col="blue",
main="Data Set 4",
xlab="Time",
ylab="Value")

whitenoise6=Box.test(ts_data4,          #Do we need to fit model?#
lag=6,
type="Ljung-Box")
whitenoise6                          #p small \implies yes#

whitenoise12=Box.test(ts_data4,
lag=12,
type="Ljung-Box")
whitenoise12

###2b. Augmented Dickey-Fuller (Unit Root Test)###
adf_result_data4=adf.test(ts_data4)     #Null is that there is a root#
adf_result_data4                       #since p is 0.2, don't reject null; assume non-stationary#

diff_data4=diff(ts_data4, differences = 1)

par(mfrow=c(1,1))
plot(diff_data4,
main="Differenced Data",
xlab="Time",
ylab="Value",
```

```

col="black",
lwd=2)

par(mfrow=c(1,2))
data4diff_acf=acf(diff_data4, lag.max=40,
main=paste0("Differenced Time Series Data Of Length ", n_data4-1, "\n", "Estimated ACF"),
ci.col="blue",
col="red",
lwd=4)
data4diff_acf                                #ACF refuses to die#

data4diff_pacf=pacf(diff_data4, lag.max=40,
main=paste0("Differenced Time Series Data Of Length ", n_data4-1, "\n", "Estimated PACF"),
ci.col="blue",
col="red",
lwd=4)
data4diff_pacf                                #PACF also refuses to die#

whitenoise6=Box.test(diff_data4,                #Do we need to fit model?#
lag=6,
type="Ljung-Box")
whitenoise6                                    #large p \implies no#

whitenoise12=Box.test(diff_data4,                #Do we need to fit model?#
lag=12,
type="Ljung-Box")
whitenoise12                                    #large p \implies no#

###2c. Plot ACF and PACF###
par(mfrow=c(1,2))
data4_acf=acf(ts_data4, lag.max=40,
main=paste0("Time Series Data Of Length ", n_data4, "\n", "Estimated ACF"),
ci.col="blue",
col="red",
lwd=4)
data4_acf                                    #ACF dies out slowly#

data4_pacf=pacf(ts_data4, lag.max=40,
main=paste0("Time Series Data Of Length ", n_data4, "\n", "Estimated PACF"),
ci.col="blue",
col="red",
lwd=4)
data4_pacf                                    #PACF cuts off at lag 2#

###2d. Try A Few Different Models###
ARIMA_model=vector()
aic=vector()
bic=vector()
LBtest=vector()

for (p in 1:4) {
  for (d in 1:2) {
    for (q in 1:4) {
      model=arima(ts_data4,
order=c(p-1,d-1,q-1))
      resid=residuals(model)

      ARIMA_model[4*2*(p-1)+4*(d-1)+q]=paste0("ARIMA(", p-1, ", ", d-1, ", ", q-1, ")")
      aic[4*2*(p-1)+4*(d-1)+q]=round(AIC(model),2)
      bic[4*2*(p-1)+4*(d-1)+q]=round(BIC(model),2)
      LBtest[4*2*(p-1)+4*(d-1)+q]=round(Box.test(resid, lag=21, type="Ljung-Box")$p.value,2)
    }
  }
}

```

```

}

df=data.frame(ARIMA_model, aic, bic, LBtest)
df=df[order(df$aic),]
df

####3. Beer####
###3a. Plot Data (Ljung-Box says fit model)###
par(mfrow=c(1,1))
plot(y=beer$data,
x=1:length(beer$yyyq),
type="l",
lwd=2,
col="blue",
main="Quarterly Beer Production",
xlab="Time",
ylab="Value",
xaxt="n")
axis(1,
at=seq(1, length(beer$yyyq), 4),
labels=beer$yyyq[seq(1, length(beer$yyyq), 4)],
las=1)

whitenoise6_beer=Box.test(ts_beer,                #Do we need to fit model?#
lag=6,
type="Ljung-Box")
whitenoise6_beer                                #p small \implies yes#

whitenoise12_beer=Box.test(ts_beer,
lag=12,
type="Ljung-Box")
whitenoise12_beer

###3b. Augmented Dickey-Fuller (Unit Root Test) For Seasonal Part###
ts_beer_seasonal=ts_beer[seq(2, length(ts_beer), by=4)] #Only look at every 4th#
sadf_result=suppressWarnings(adf.test(ts_beer_seasonal))
sadf_result

#technically stationary from test, but visually, looks trending#
ts_beer_sdiff=diff(ts_beer, lag = 4)
plot(y=ts_beer_sdiff,
x=1:28,
type="l",
lwd=2,
col="blue",
main="Quarterly Beer Production After Seasonal Difference",
xlab="Time",
ylab="Value",
xaxt="n")

###3c. ACF and PACF###
par(mfrow=c(1,2))
beer_acf=acf(ts_beer, lag.max=40,
main=paste0("Time Series Data Of Length ", n_beer, "\n", "Estimated ACF"),
ci.col="blue",
col="red",

```

```

lwd=4)
beer_acf                                #ACF dies out slowly at seasonal lags#

beer_pacf=pacf(ts_beer, lag.max=40,
main=paste0("Time Series Data Of Length ", n_beer, "\n", "Estimated PACF"),
ci.col="blue",
col="red",
lwd=4)
beer_pacf                                #PACF cuts off at seasonal lag 1#

#try with the seasonal difference#
par(mfrow=c(1,2))
beer_sacf=acf(ts_beer_sdiff, lag.max=40,
main=paste0("ACF After Seasonal Difference"),
ci.col="blue",
col="red",
lwd=4)
beer_sacf                                #ACF dies out slowly#

beer_spacf=pacf(ts_beer, lag.max=40,
main=paste0("PACF After Seasonal Difference"),
ci.col="blue",
col="red",
lwd=4)
beer_spacf                                #PACF cuts off at lag 2#

###3d. Regular ARIMA Components###
adf_result_beer=adf.test(ts_beer)
adf_result_beer                          #large p implies take difference#
ts_beer1=diff(ts_beer, lag=1)

###3e. Look at ACF PACF after lag 1 difference###
beer_acf1=acf(ts_beer1, lag.max=40,
main=paste0("ACF After Regular Difference"),
ci.col="blue",
col="red",
lwd=4)
beer_acf1                                #ACF dies out slowly at seasonal lags#

beer_pacf1=pacf(ts_beer, lag.max=40,
main=paste0("PACF After Regular Difference"),
ci.col="blue",
col="red",
lwd=4)
beer_pacf1                                #PACF cuts off at seasonal lag 1#

###3f. Try A Few Different Models###
ARIMAs_model=vector()
aic=vector()
bic=vector()
LBtest=vector()
s=4                                      #clear from data#

for (p in 1:4) {
  for (d in 1:2) {
    for (q in 1:4) {
      for (P in 1:4) {

```

```

for (D in 1:2) {
  for (Q in 1:4) {
    mycount=(((p-1)*2 + (d-1))*4 + (q-1))*4 + (P-1))*2 + (D-1))*4 + Q
    mymodel=paste0("ARIMA(", p-1, ",", d-1, ",", q-1, ")(", P-1, ",", D-1, ",", Q-1, ")",s)

    tryCatch({
      model=arima(ts_beer,
        order=c(p-1,d-1,q-1),
        seasonal=list(order=c(P-1,D-1,Q-1), period=s))
      resid=residuals(model)

      ARIMAs_model[mycount]=mymodel
      aic[mycount]=round(AIC(model),2)
      bic[mycount]=round(BIC(model),2)
      LBtest[mycount]=round(Box.test(resid, lag=21, type="Ljung-Box")$p.value,2)
    }, error=function(e) {
      ARIMAs_model[mycount]=mymodel
      aic[mycount]=999
      bic[mycount]=999
      LBtest[mycount]=999
    })
  }
}
}
}
}
}

df=data.frame(ARIMAs_model, aic, bic, LBtest)
df=df[order(df$aic),]
df

```