UNIVERSITY OF THE PHILIPPINES MINDANAO
College of Science and Mathematics
Department of Mathematics, Physics, and Computer Science

AMAT 132 (Introductory Forecasting) — Exercise 5: AR, MA, ARMA, ARIMA

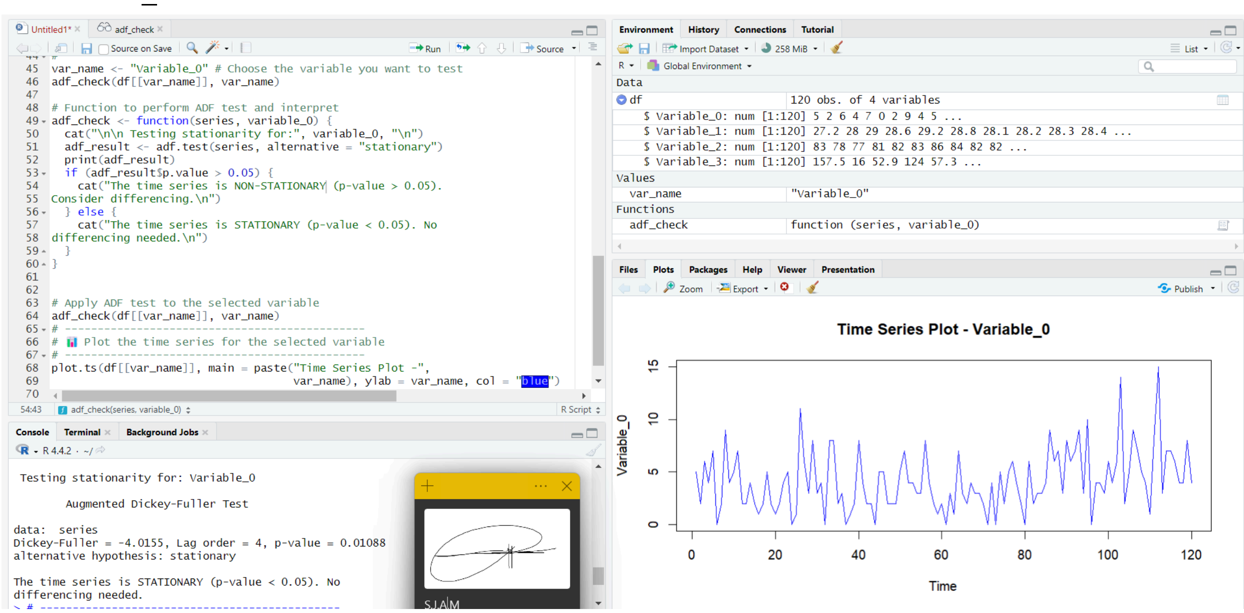
Saint Joy A. Mandalinao

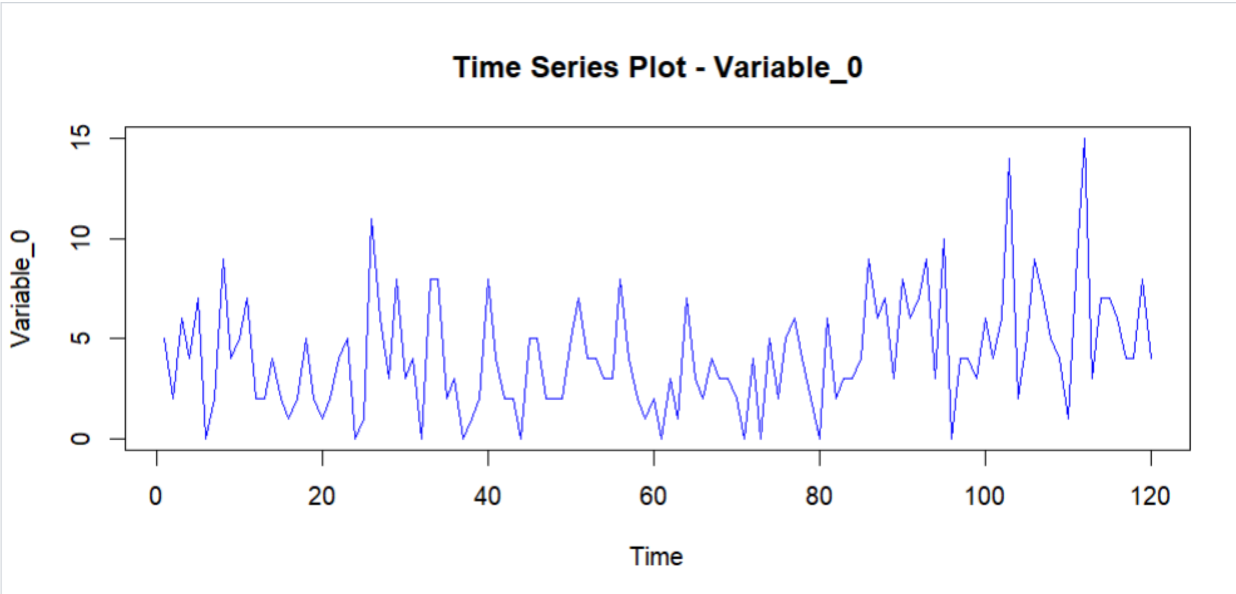
General Objectives:
This exercise aims to help students:
1. Conduct stationarity tests for time series using ADF test.
2. Plot ACF and PACF to determine what model/order to use in the analysis.
3. Fit ARMA model.
4. Conduct Ljung Box Test to determine if the model used is a good fit.

Exercise 5a: Stationarity Test Using ADF Test
Objective: Learn how to conduct a stationarity test for a time series using the ADF test.
Instructions:
1. Load the provided dataset in R.
2. Adjust the code to capture your own directory or local drive where you save the dataset.
3. Change the file name (when necessary).
4. Change the variable name depending on which variable you are performing the stationarity test on.
5. Discuss your results.

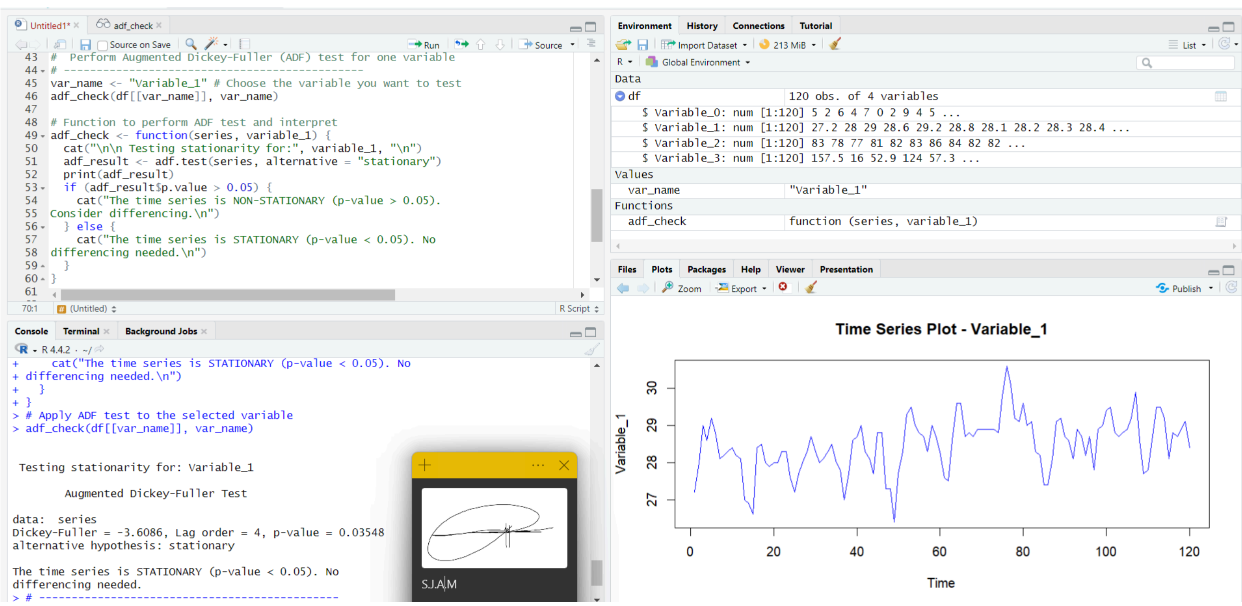
RESULTS AND DISCUSSIONS

VARIABLE_0

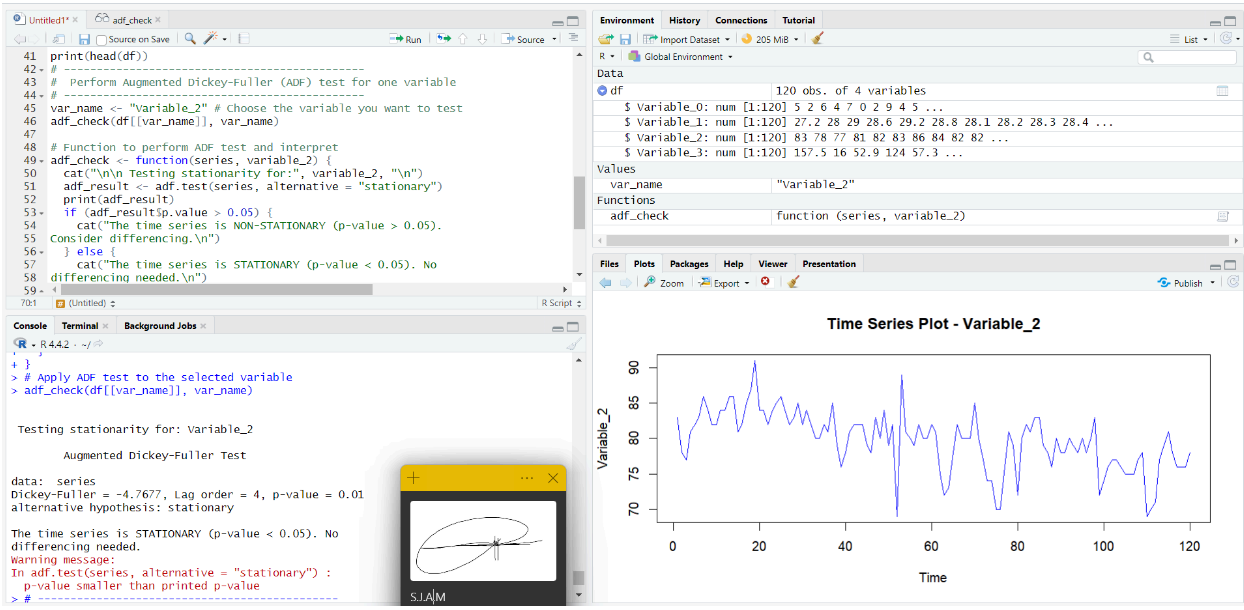


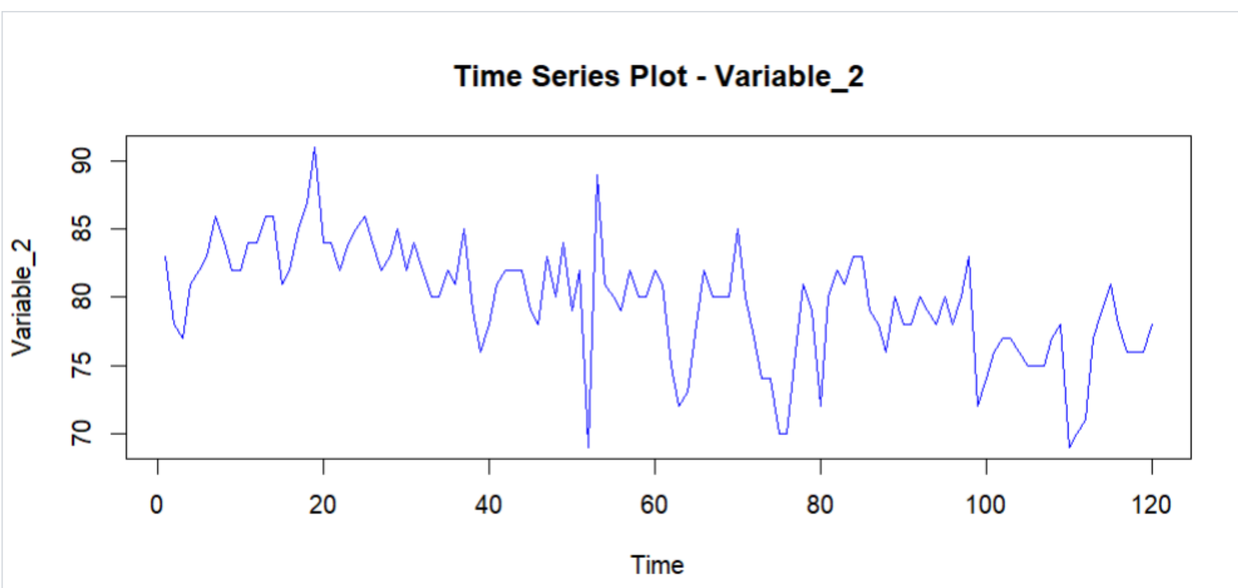

Time Series Plot - Variable_0
Variable_0
0
5
10
15
0
20
40
60
80
100
120
Time

VARIABLE_1

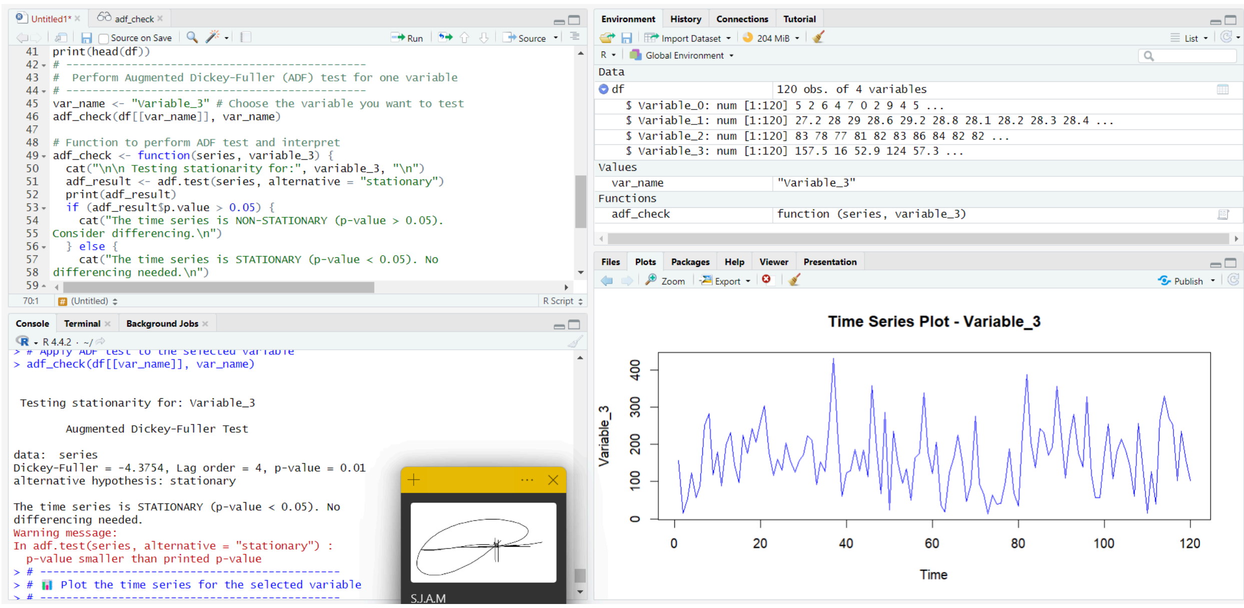

Untitled1*
adf_check
Source on Save
Run
Source
# Perform Augmented Dickey-Fuller (ADF) test for one variable
# --
var_name <- "Variable_1" # Choose the variable you want to test
adf_check(df[[var_name]], var_name)
# Function to perform ADF test and interpret
adf_check <- function(series, variable_1) {
cat("\n\n Testing stationarity for:", variable_1, "\n")
adf_result <- adf.test(series, alternative = "stationary")
print(adf_result)
if (adf_result$p.value > 0.05) {
cat("The time series is NON-STATIONARY (p-value > 0.05).
Consider differencing.\n")
} else {
cat("The time series is STATIONARY (p-value < 0.05). No
differencing needed.\n")
}
}
(Untitled)
R Script
Console
Terminal
Background Jobs
R 4.4.2 · ~/
+ cat("The time series is STATIONARY (p-value < 0.05). No
+ differencing needed.\n")
+ }
+ }
> # Apply ADF test to the selected variable
> adf_check(df[[var_name]], var_name)
Testing stationarity for: Variable_1
Augmented Dickey-Fuller Test
data: series
Dickey-Fuller = -3.6086, Lag order = 4, p-value = 0.03548
alternative hypothesis: stationary
The time series is STATIONARY (p-value < 0.05). No
differencing needed.
> # --
S.J.A|M
Environment
History
Connections
Tutorial
Import Dataset
213 MiB
List
Global Environment
Data
df
120 obs. of 4 variables
$ Variable_0: num [1:120] 5 2 6 4 7 0 2 9 4 5 ...
$ Variable_1: num [1:120] 27.2 28 29 28.6 29.2 28.8 28.1 28.2 28.3 28.4 ...
$ Variable_2: num [1:120] 83 78 77 81 82 83 86 84 82 82 ...
$ Variable_3: num [1:120] 157.5 16 52.9 124 57.3 ...
Values
var_name
"Variable_1"
Functions
adf_check
function (series, variable_1)
Files
Plots
Packages
Help
Viewer
Presentation
Zoom
Export
Publish
Time Series Plot - Variable_1
Variable_1
27
28
29
30
0
20
40
60
80
100
120
Time

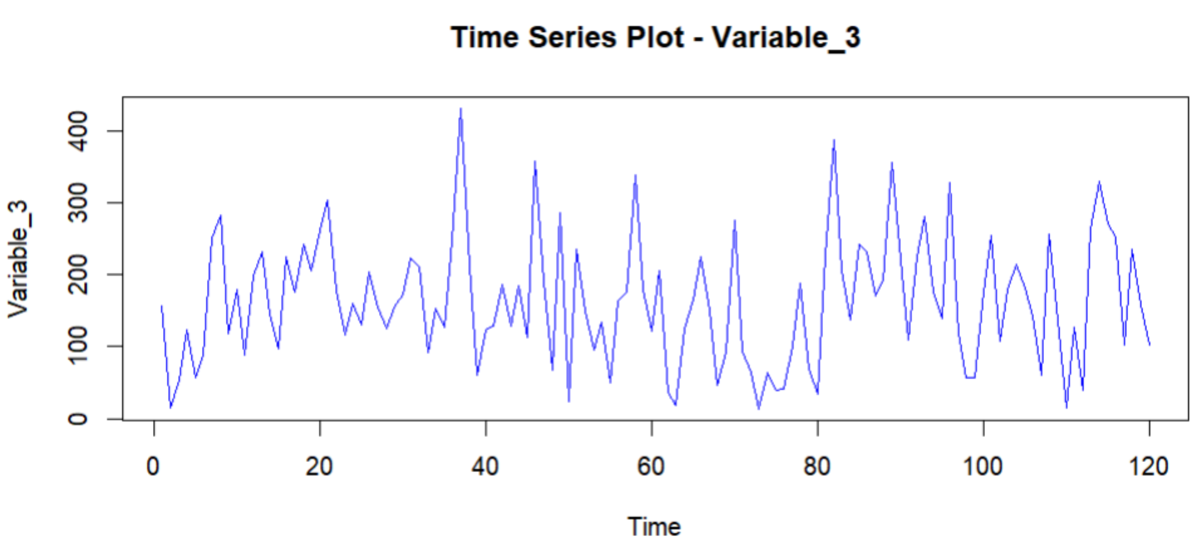
VARIABLE_2


print(head(df))
# ---
# Perform Augmented Dickey-Fuller (ADF) test for one variable
# ---
var_name <- "Variable_2" # Choose the variable you want to test
adf_check(df[[var_name]], var_name)
# Function to perform ADF test and interpret
adf_check <- function(series, variable_2) {
cat("\n\n Testing stationarity for:", variable_2, "\n")
adf_result <- adf.test(series, alternative = "stationary")
print(adf_result)
if (adf_result$p.value > 0.05) {
cat("The time series is NON-STATIONARY (p-value > 0.05). Consider differencing.\n")
} else {
cat("The time series is STATIONARY (p-value < 0.05). No differencing needed.\n")
Testing stationarity for: Variable_2
Augmented Dickey-Fuller Test
data: series
Dickey-Fuller = -4.7677, Lag order = 4, p-value = 0.01
alternative hypothesis: stationary
The time series is STATIONARY (p-value < 0.05). No differencing needed.
Warning message:
In adf.test(series, alternative = "stationary") :
p-value smaller than printed p-value
df 120 obs. of 4 variables
$ Variable_0: num [1:120] 5 2 6 4 7 0 2 9 4 5 ...
$ Variable_1: num [1:120] 27.2 28 29 28.6 29.2 28.8 28.1 28.2 28.3 28.4 ...
$ Variable_2: num [1:120] 83 78 77 81 82 83 86 84 82 82 ...
$ Variable_3: num [1:120] 157.5 16 52.9 124 57.3 ...
var_name "Variable_2"
adf_check function (series, variable_2)
Time Series Plot - Variable_2
Variable_2
Time
S.J.A.M


Time Series Plot - Variable_2
Variable_2
70
75
80
85
90
0
20
40
60
80
100
120
Time

VARIABLE_3





Discussion Questions:

1. What time series are stationary? What time series are non-stationary?

Using the ADF test, we have these values as its results:

- Variable_0: Dickey-Fuller = -4.0155, p-value = 0.01088. Since the p-value < 0.05, the time series is stationary.
- Variable_1: Dickey-Fuller = -3.6086, p-value = 0.03548. Since the p-value < 0.05, the time series is stationary.
- Variable_2: Dickey-Fuller = -4.7677, p-value = 0.01. Since the p-value < 0.05, the time series is stationary.
- Variable_3: Dickey-Fuller = -4.3754, p-value = 0.01. Since the p-value < 0.05, the time series is stationary.

Hence, the results clearly indicate that all four time series are stationary, with none exhibiting non-stationary behavior.

2. What is the basis to conclude if a time series is stationary or not?

The Augmented Dickey-Fuller (ADF) test determines the stationarity. It tests the null hypothesis of a unit root (non-stationarity) against the alternative hypothesis of stationarity. If the p-value < 0.05, it rejects the null hypothesis and suggests the series is stationary. All variables are having p-values < 0.05, which means stationarity. In our results, the Augmented Dickey-Fuller (ADF) test results show that all four time series—Variable_0, Variable_1, Variable_2, and Variable_3—are stationary. Because the p-value for each variable was less than 0.05,therefore we conclude that the null hypothesis of non-stationarity can be rejected. The p-values were 0.01088 for Variable_0, 0.03548 for Variable_1, 0.01 for Variable_2, and 0.01 for Variable_3. Since all p-values are below the cut-off value of 0.05, we conclude that none of the series have trends or changing variance with time and can be modeled without differencing.

3. Discuss each plot and how these visual representations can support the numerical results.

For Plot - Variable_0:

The plot of Variable_0 shows fluctuations centered around 5, with values ranging from 0 to 15. There is no visible trend, the variance appears steady, and occasional spikes are observed. These characteristics support the stationarity conclusion (p-value 0.01088 < 0.05), which indicates a stable mean and variance over time.

For Plot - Variable_1:

In the plot of Variable_1, the series varies around 28, within a range of 27 to 30. The pattern shows no trend and maintains a constant variance, which supports the conclusion of stationarity (p-value 0.03548 < 0.05), suggesting a consistent mean and variance.

For Plot - Variable_2:

The visual representation of Variable_2 indicates fluctuations around 80, spanning from 70 to 90. While some spikes are present, the series shows no trend and retains a consistent variance. This confirms the stationarity of the series (p-value 0.01 < 0.05) with a constant mean and variance.

For Plot - Variable_3:

Lastly, the plot for Variable_3 displays variation around 150, ranging from 0 to 350. Although the series has larger fluctuations and occasional spikes, no trend is evident. The stationarity result (p-value 0.01 < 0.05) suggests that the mean remains stable, even with variable fluctuations in variance.

Exercise 5b: ACF and PACF Unset Unset

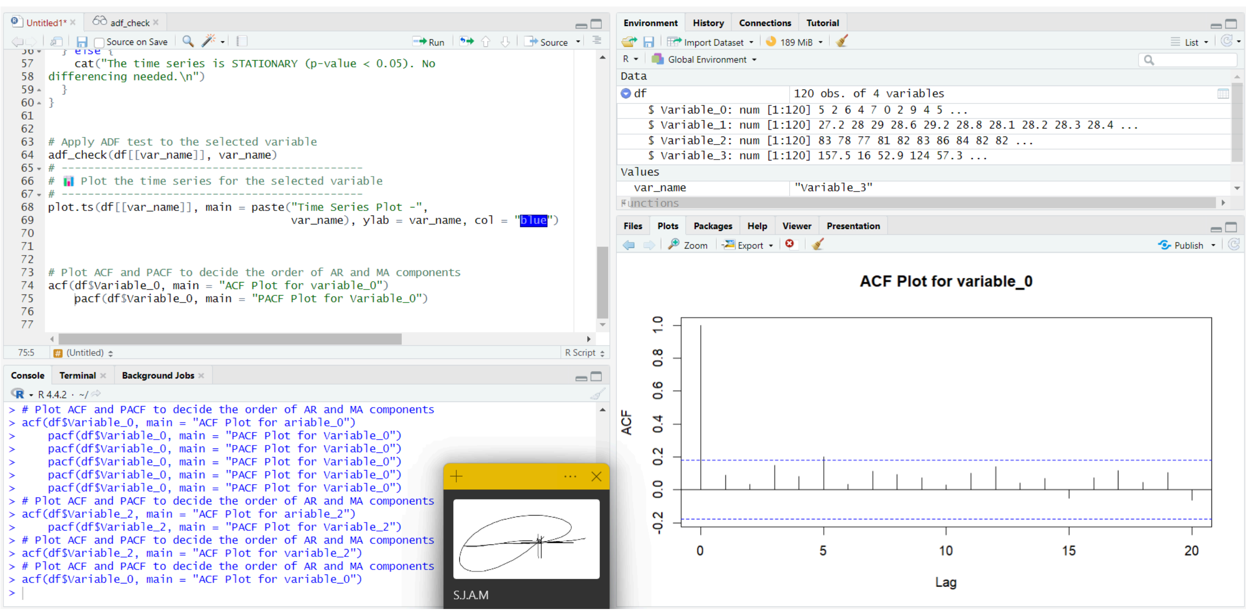
Objective: Learn how to plot ACF and PACF to determine what model/order to use in the analysis

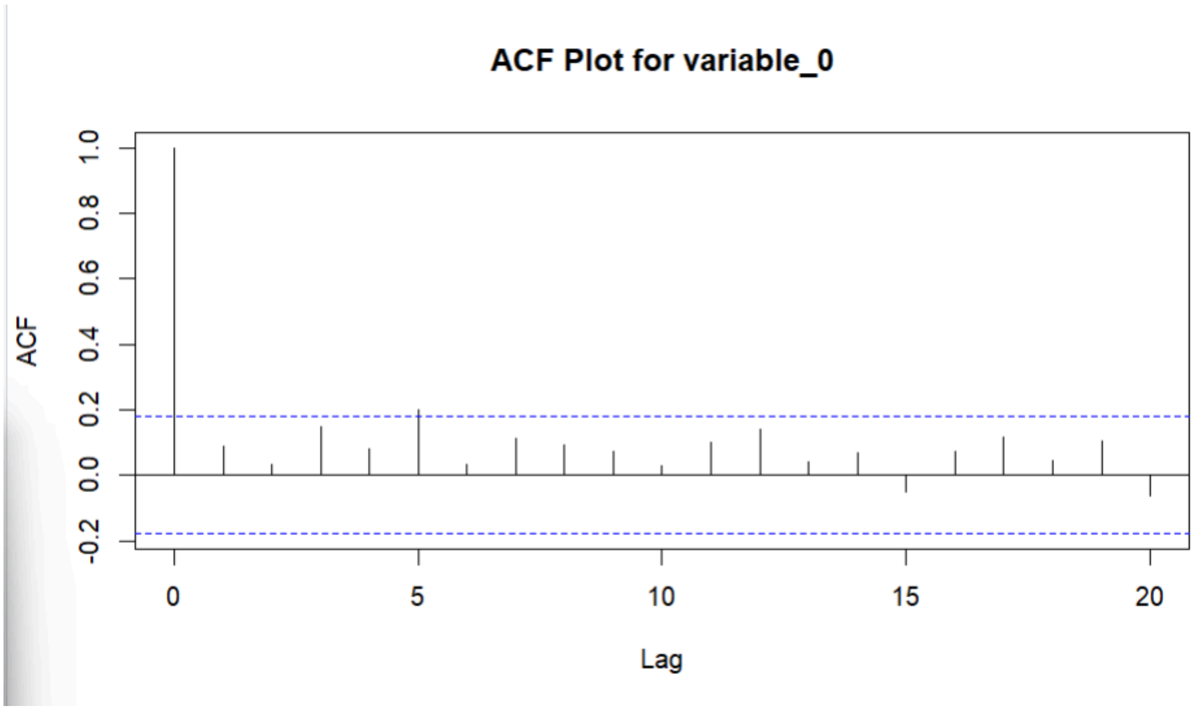
Instructions:

1. Run the code to plot the ACF and PACF for each time series.

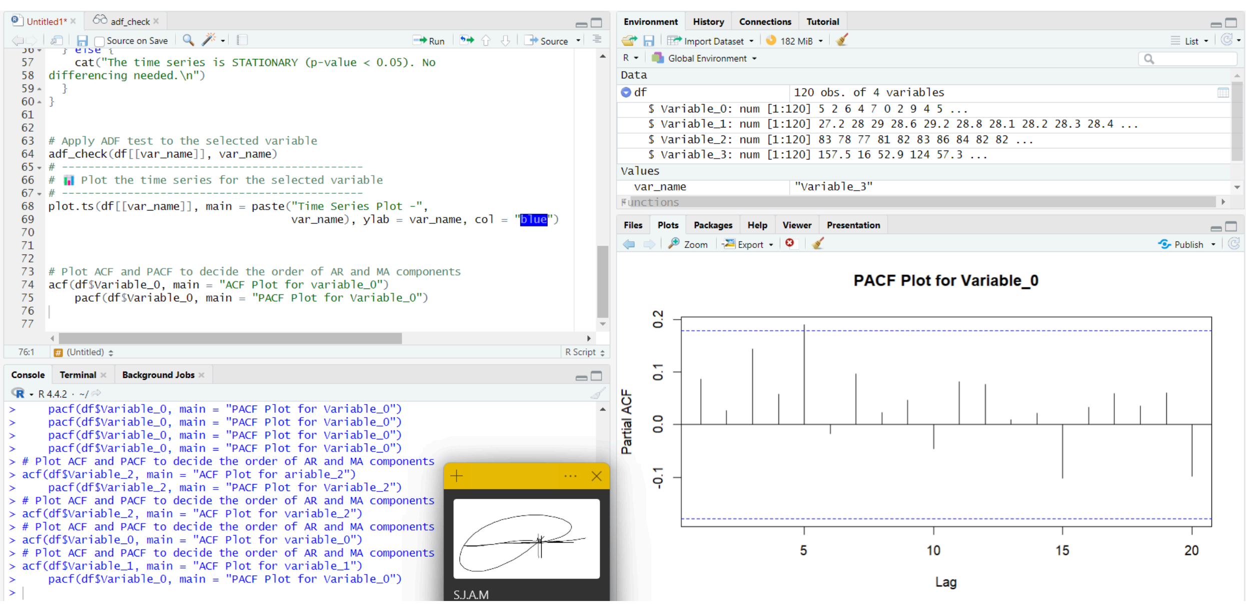
ACF

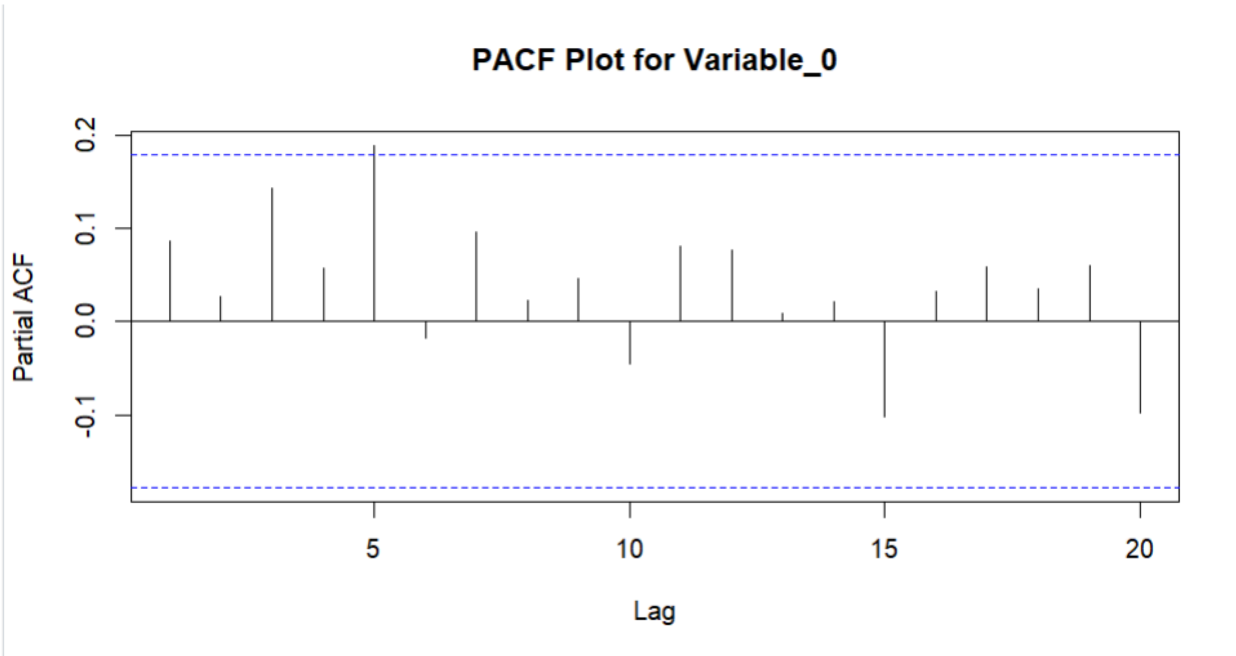
VARIABLE 0



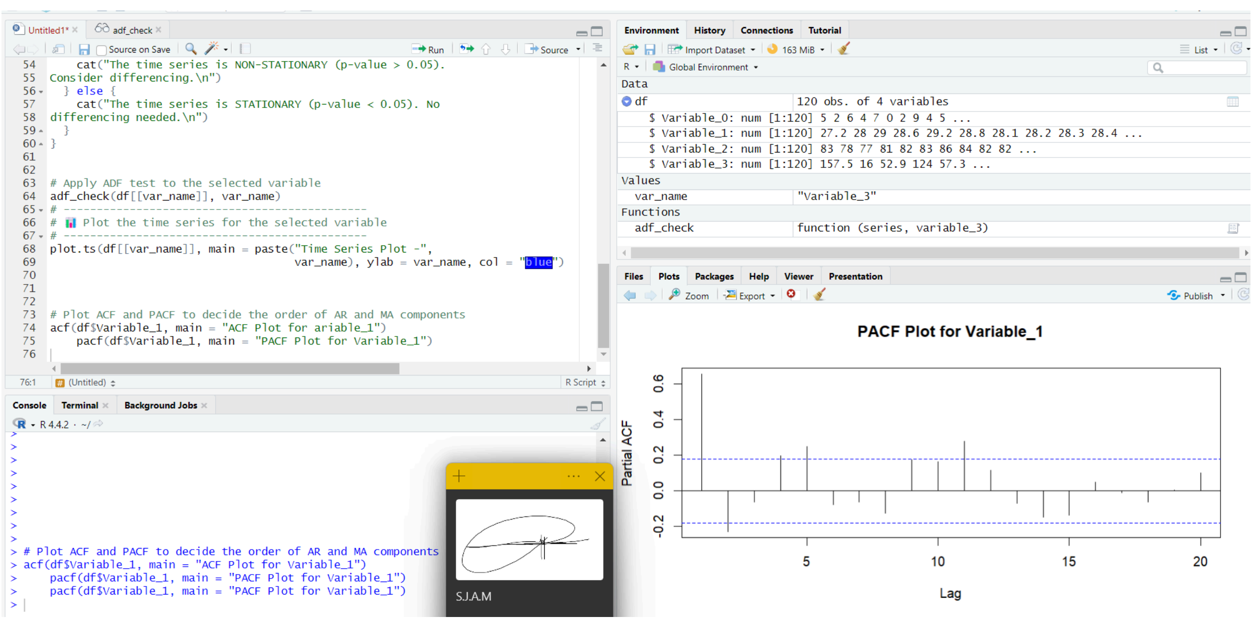


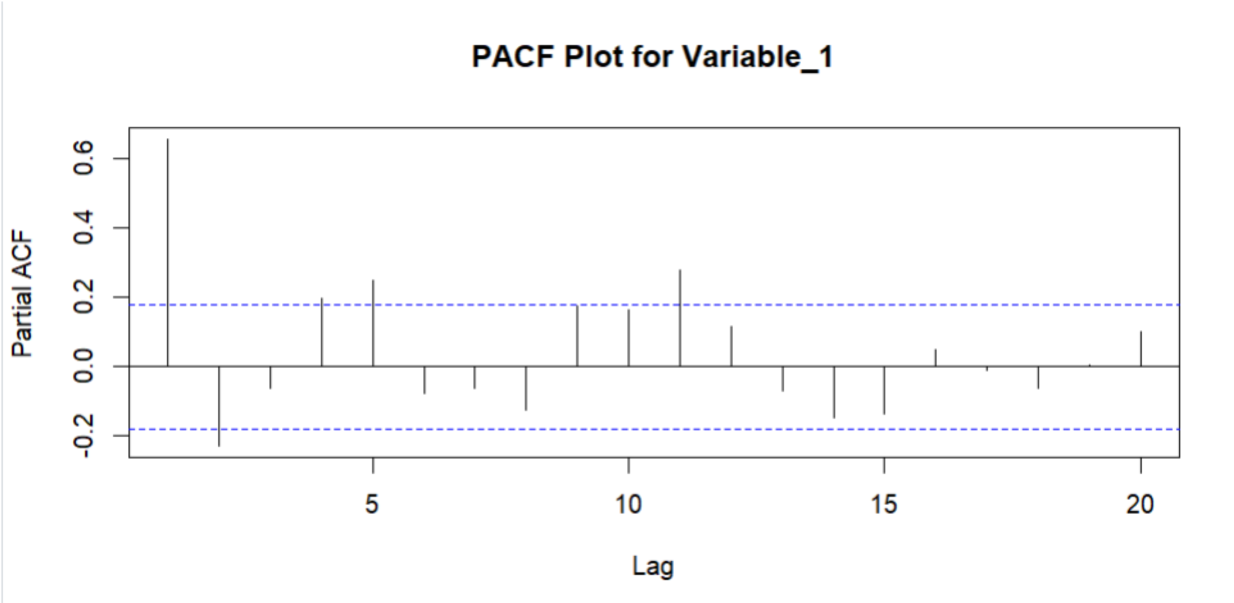
PACF
VARIABLE 0



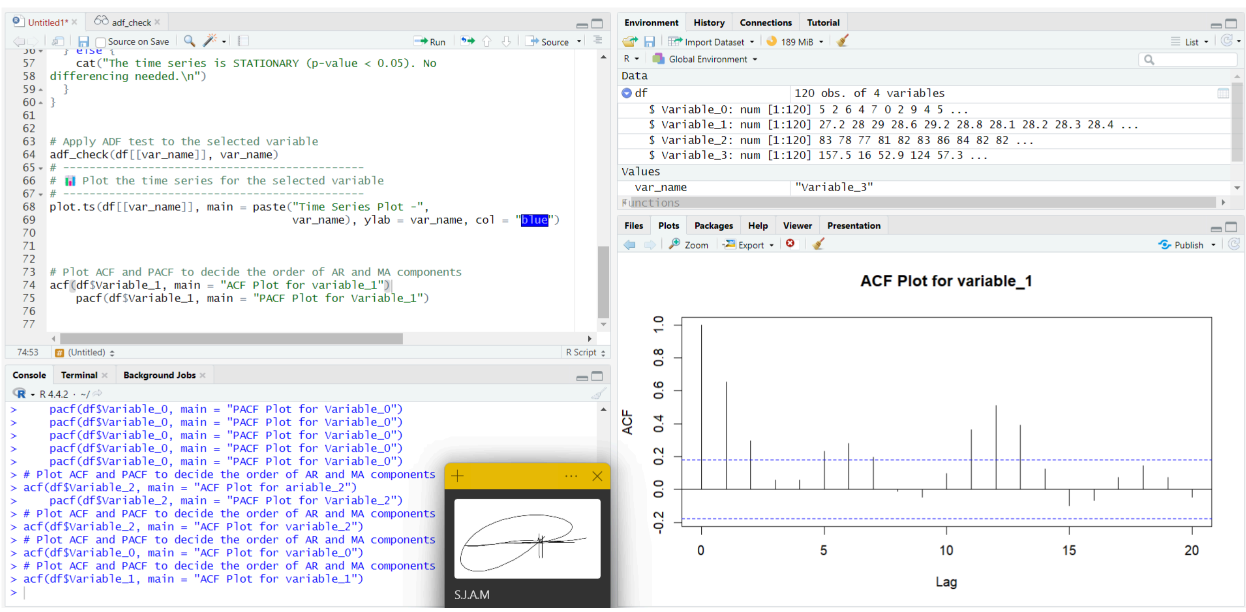


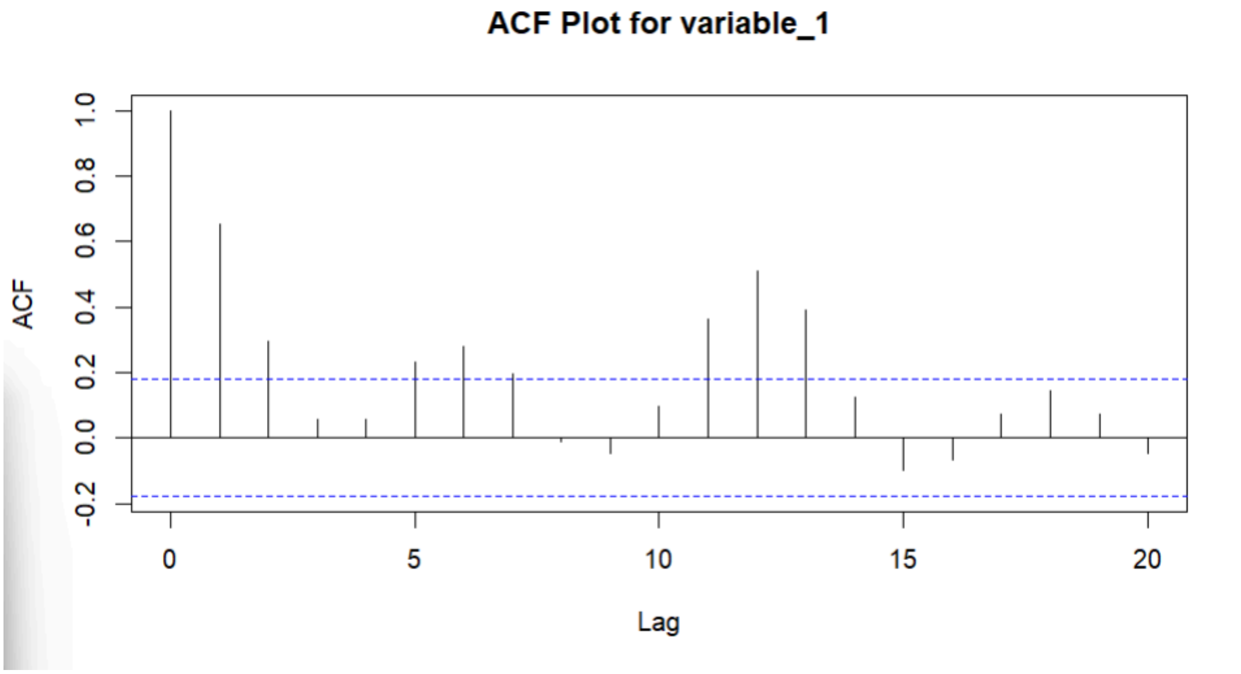
PACF
VARIABLE 1



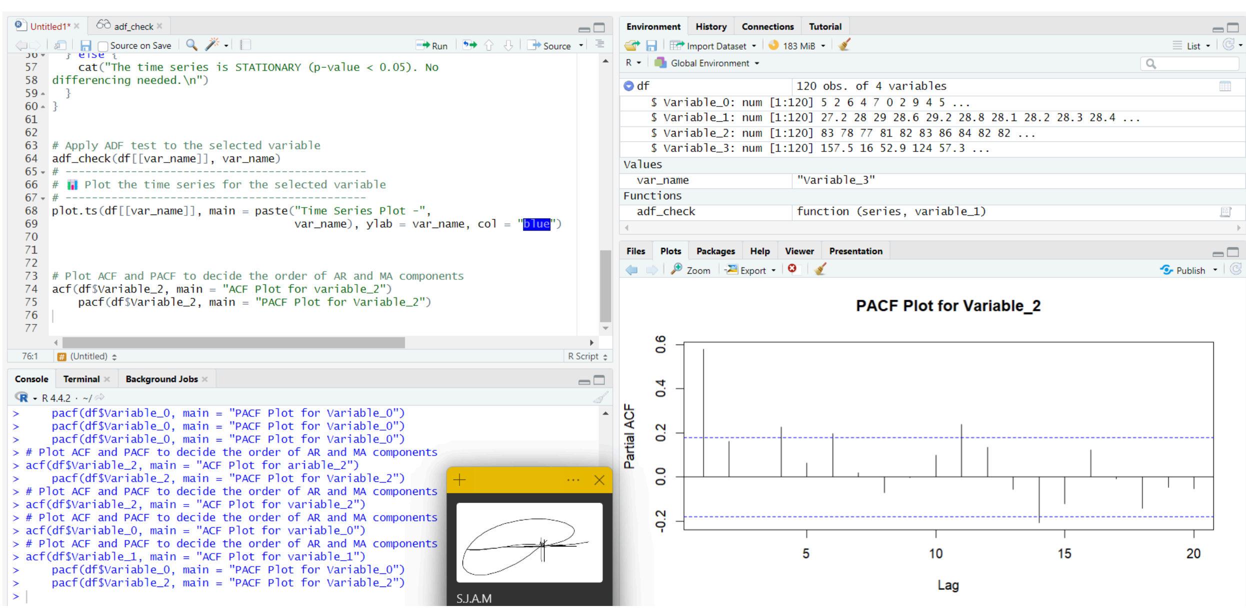


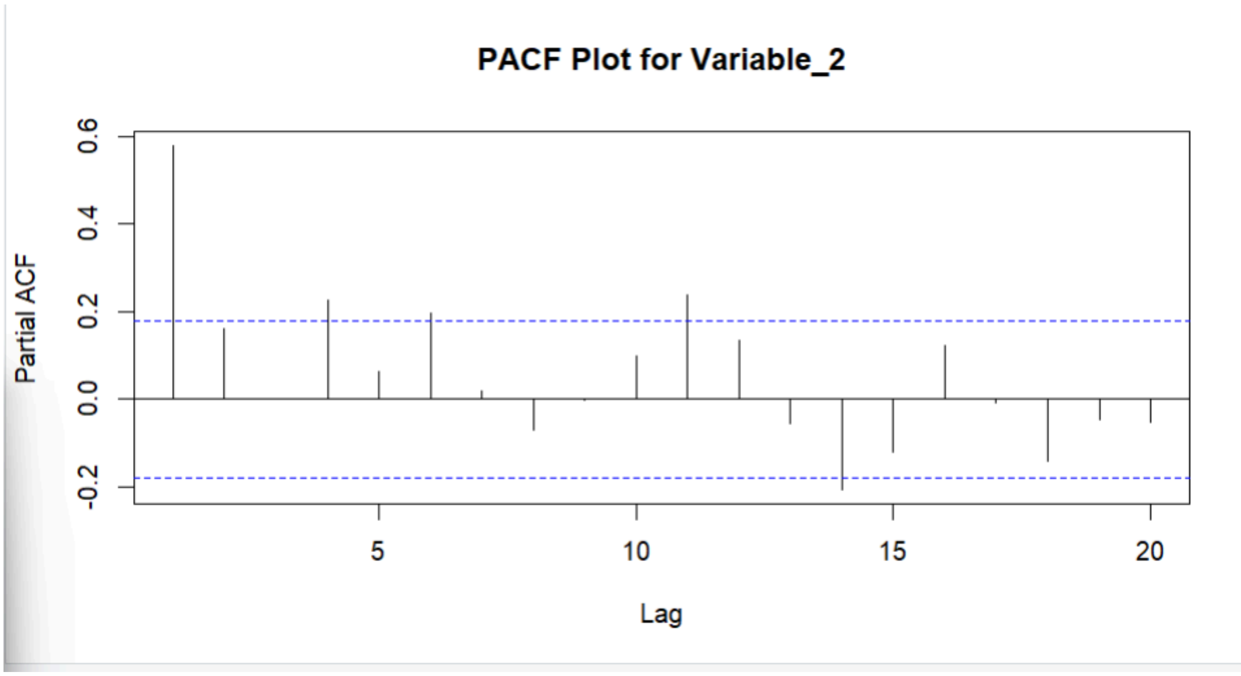
ACF VARIABLE 1



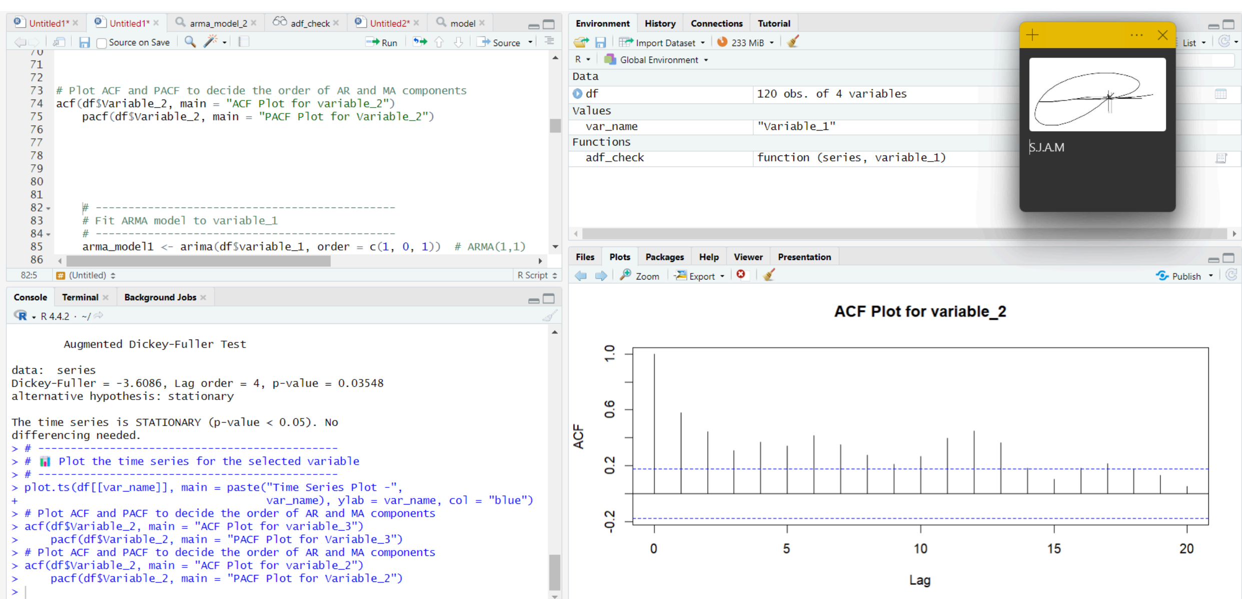


PACF
VARIABLE 2

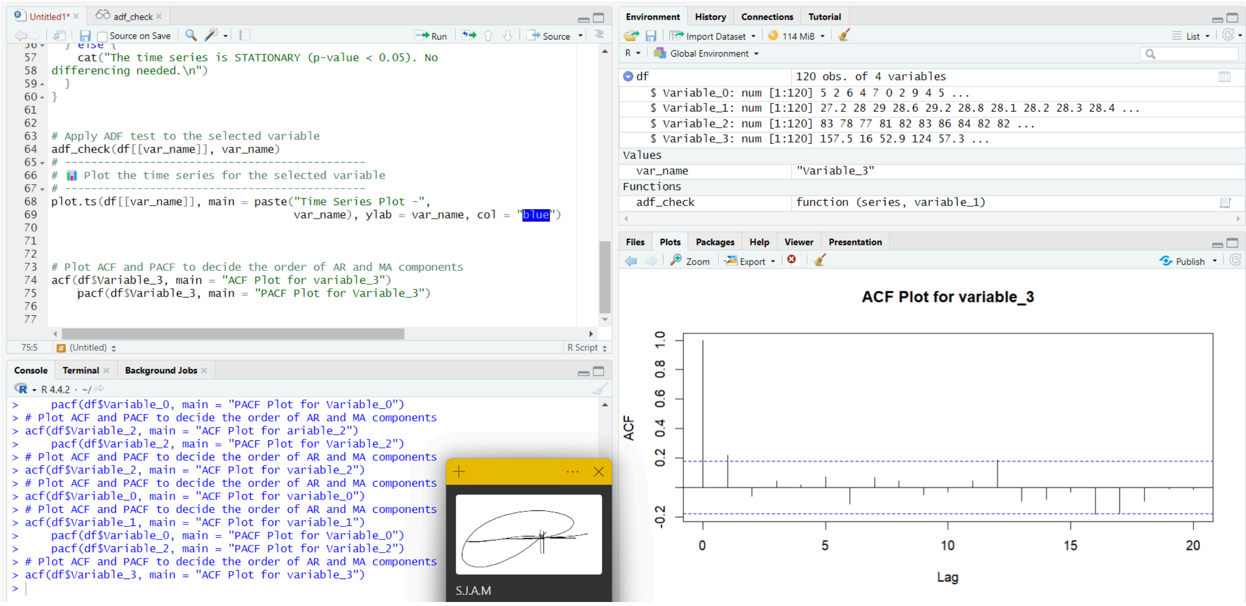


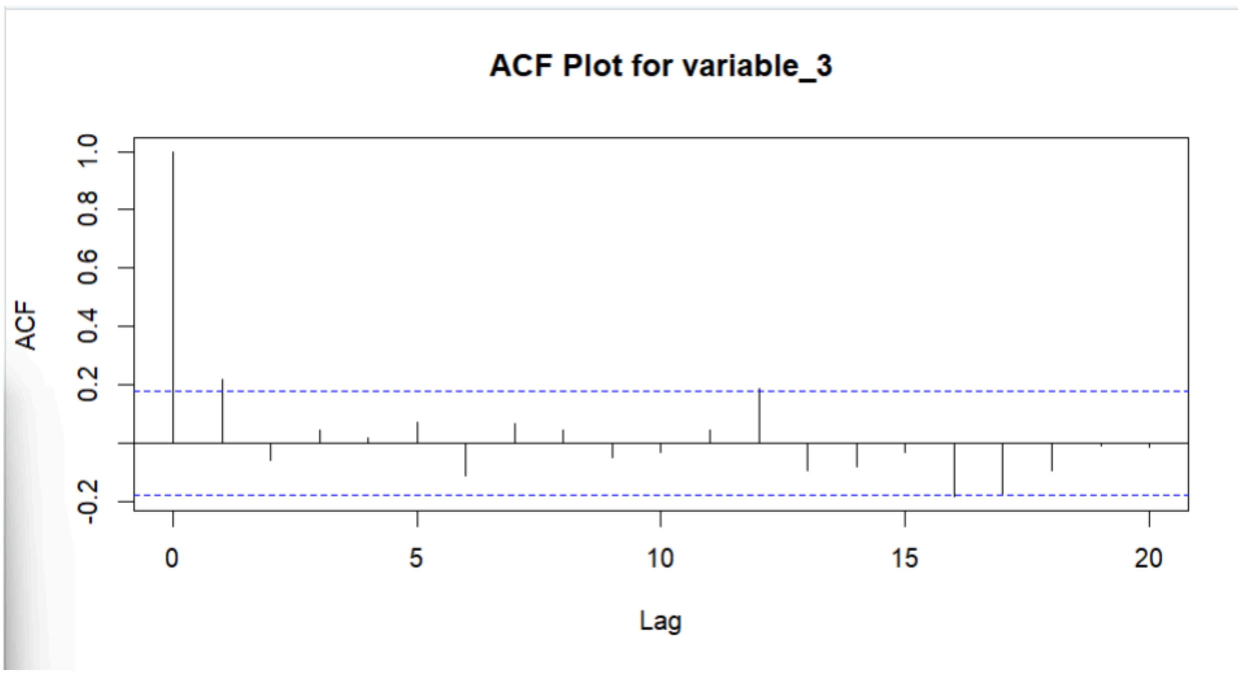


ACF
VARIABLE 2

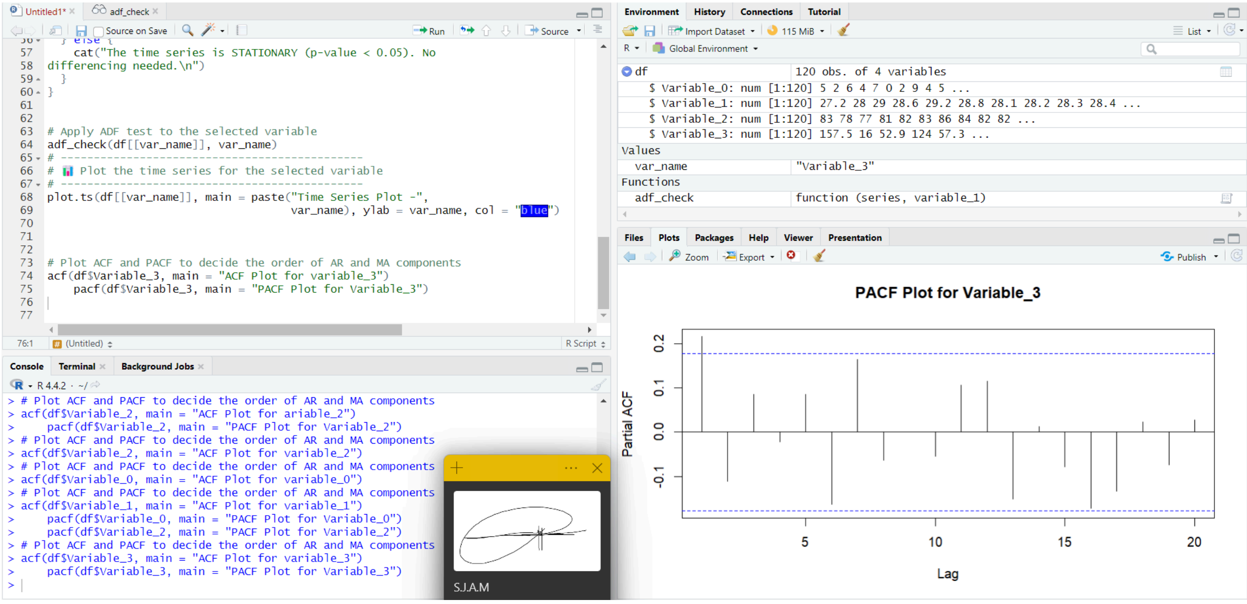


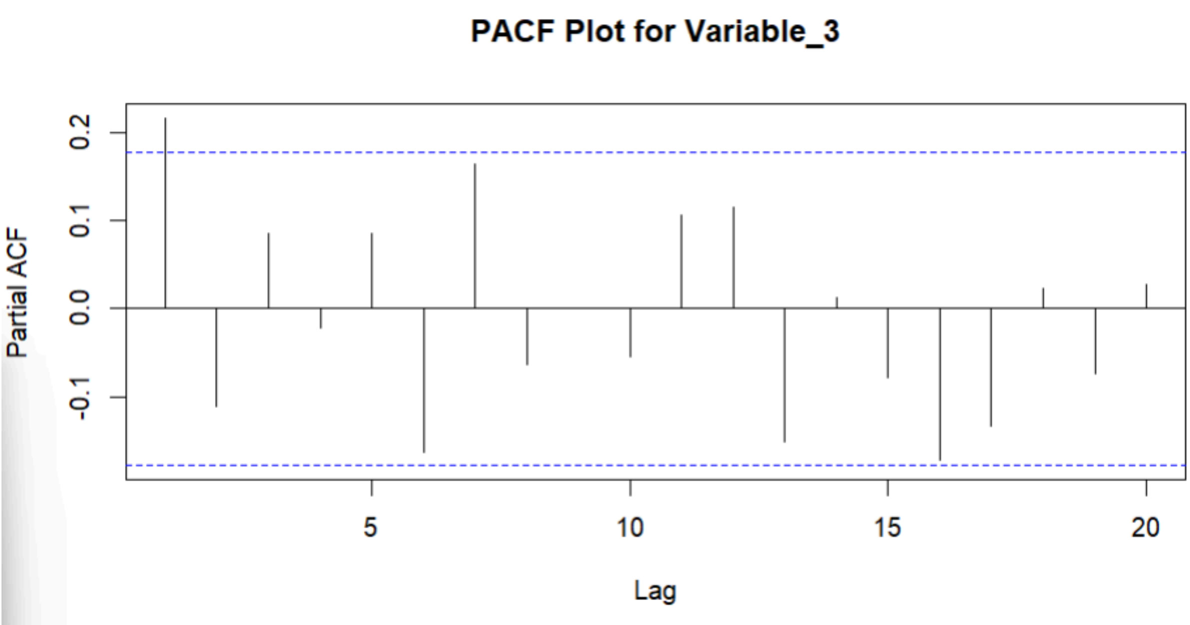
ACF VARIABLE 3





PACF VARIABLE 3


Untitled1*
adf_check
Source on Save
Run
Source
cat("The time series is STATIONARY (p-value < 0.05). No differencing needed.\n")
}
}
# Apply ADF test to the selected variable
adf_check(df[[var_name]], var_name)
# --
# Plot the time series for the selected variable
# --
plot.ts(df[[var_name]], main = paste("Time Series Plot -", var_name), ylab = var_name, col = "blue")
# Plot ACF and PACF to decide the order of AR and MA components
acf(df$Variable_3, main = "ACF Plot for variable_3")
pacf(df$Variable_3, main = "PACF Plot for Variable_3")
76:1
(Untitled)
R Script
Console
Terminal
Background Jobs
R 4.4.2 · ~/
> # Plot ACF and PACF to decide the order of AR and MA components
> acf(df$Variable_2, main = "ACF Plot for ariable_2")
>   pacf(df$Variable_2, main = "PACF Plot for Variable_2")
> # Plot ACF and PACF to decide the order of AR and MA components
> acf(df$Variable_2, main = "ACF Plot for variable_2")
> # Plot ACF and PACF to decide the order of AR and MA components
> acf(df$Variable_0, main = "ACF Plot for variable_0")
> # Plot ACF and PACF to decide the order of AR and MA components
> acf(df$Variable_1, main = "ACF Plot for variable_1")
>   pacf(df$Variable_0, main = "PACF Plot for Variable_0")
>   pacf(df$Variable_2, main = "PACF Plot for Variable_2")
> # Plot ACF and PACF to decide the order of AR and MA components
> acf(df$Variable_3, main = "ACF Plot for variable_3")
>   pacf(df$Variable_3, main = "PACF Plot for Variable_3")
S.J.A.M
Environment
History
Connections
Tutorial
Import Dataset
115 MiB
List
Global Environment
df   120 obs. of 4 variables
$ Variable_0: num [1:120] 5 2 6 4 7 0 2 9 4 5 ...
$ Variable_1: num [1:120] 27.2 28 29 28.6 29.2 28.8 28.1 28.2 28.3 28.4 ...
$ Variable_2: num [1:120] 83 78 77 81 82 83 86 84 82 82 ...
$ Variable_3: num [1:120] 157.5 16 52.9 124 57.3 ...
Values
var_name   "Variable_3"
Functions
adf_check   function (series, variable_1)
Files
Plots
Packages
Help
Viewer
Presentation
Zoom
Export
Publish
PACF Plot for Variable_3
Partial ACF
Lag


PACF Plot for Variable_3
Partial ACF
0.2
0.1
0.0
-0.1
5
10
15
20
Lag

1. Discuss your observations of the graphs.

Variable_0:

The ACF graph displays a notable spike at lag 5, while most other values remain within the confidence bounds, indicating MA(5) as a suitable choice. Similarly, the PACF graph shows a prominent spike at lag 5, with the rest of the values mostly within the confidence interval, suggesting a potential AR(1) structure without strong higher-order terms. Therefore, an appropriate ARMA model to apply is (5,5).

Variable_1:

In the ACF plot, there are multiple visible spikes, but the most dominant one is at lag 1, and the rest fall largely within the confidence limits, suggesting MA(1). The PACF plot shows significant spikes at lags 1, 2, 4, 5, and 11, but the spike at lag 1 is the strongest, indicating AR(1) as a fitting choice. Thus, an ARMA(1,1) model is recommended.

Variable_2:

The ACF graph highlights several spikes, with the largest at lag 1, and the majority of the values staying within the confidence interval—pointing to MA(1). The PACF also presents several spikes, but the most dominant one is at lag 1, supporting the selection of AR(1). Hence, ARMA(1,1) is a suitable model for this variable.

Variable_3:

For Variable_3, the ACF reveals several spikes, the strongest of which is at lag 1, while most other values lie within the confidence bounds, indicating MA(1). Similarly, the PACF exhibits multiple spikes, but the most significant one is again at lag 1, suggesting AR(1). Therefore, the ARMA(1,1) model is considered appropriate.

2. From the plots, decide what model/order to use in the analysis For example: ARMA (1, 1)

Based on the analysis of the autocorrelation and partial autocorrelation plots, Variable_0 is best modeled using an ARMA(5,5) model. This choice is supported by the significant spikes at lag 5 in both the ACF and PACF plots, indicating the presence of both AR and MA components of order 5. For Variable_1, the ARMA(1,1) model is appropriate, as the ACF shows a dominant spike at lag 1 and the PACF highlights significant spikes at multiple lags, with lag 1 being the most prominent. Similarly, Variable_2 is well-suited to an ARMA(1,1) model, with both the ACF and PACF indicating strong spikes at lag 1, suggesting a combination of short-term AR and MA effects. Finally, Variable_3 also fits an ARMA(1,1) model, as both plots display their most significant spikes at lag 1, reinforcing the selection of this model to capture its time series behavior.

Exercise 5c: Model Fitting

Objective: Learn how to fit an ARMA model for a time series.

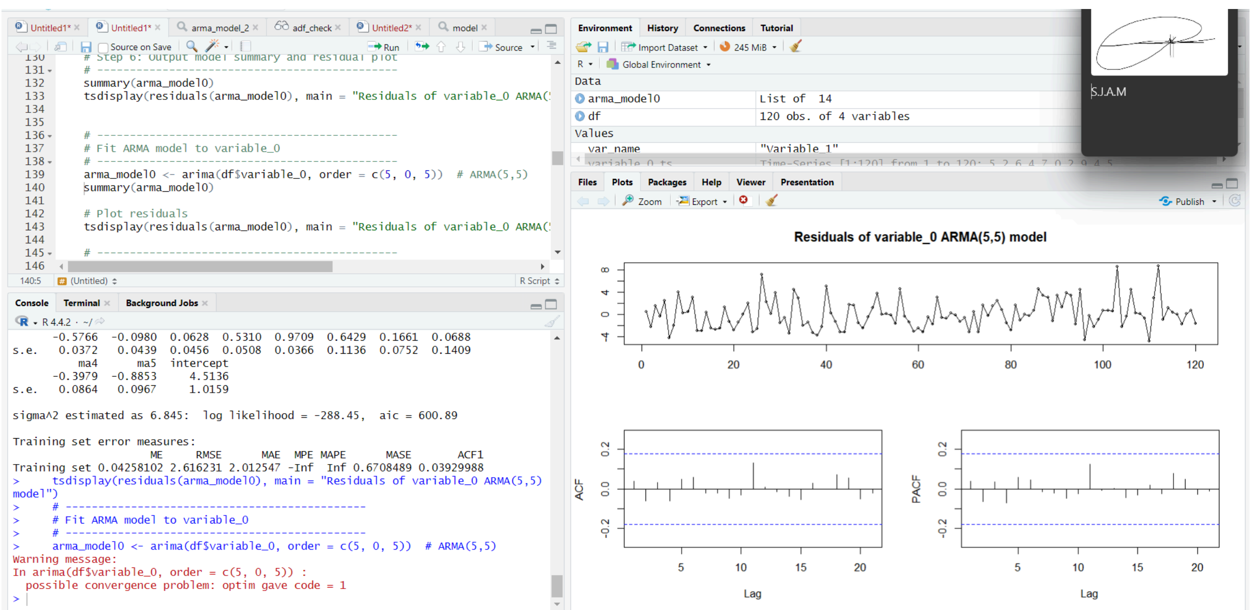
Instructions:

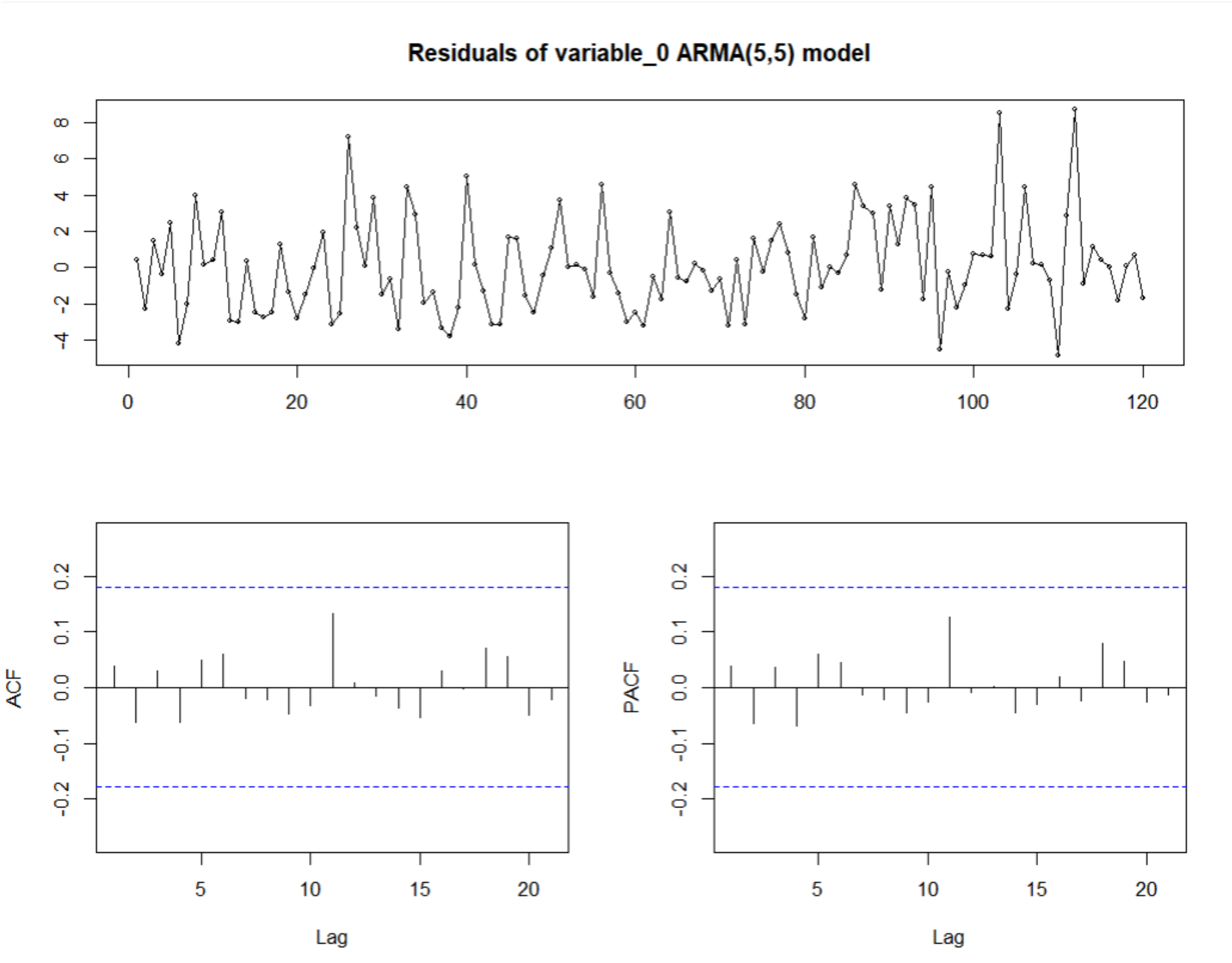
1. Fit an ARMA model to a stationary time series using your results in 5b. Change the order in the sample code.

2. Analyze the model's residuals to assess its fit.

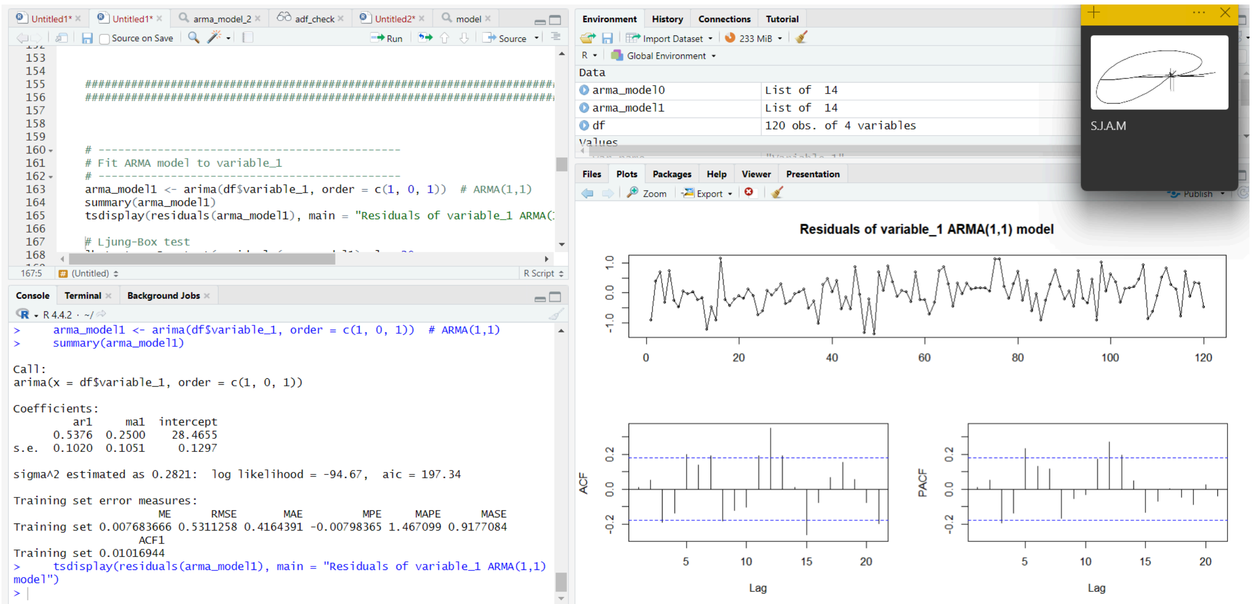
3. Interpret the estimated coefficients and discuss the validity of the model.

VARIABLE_0

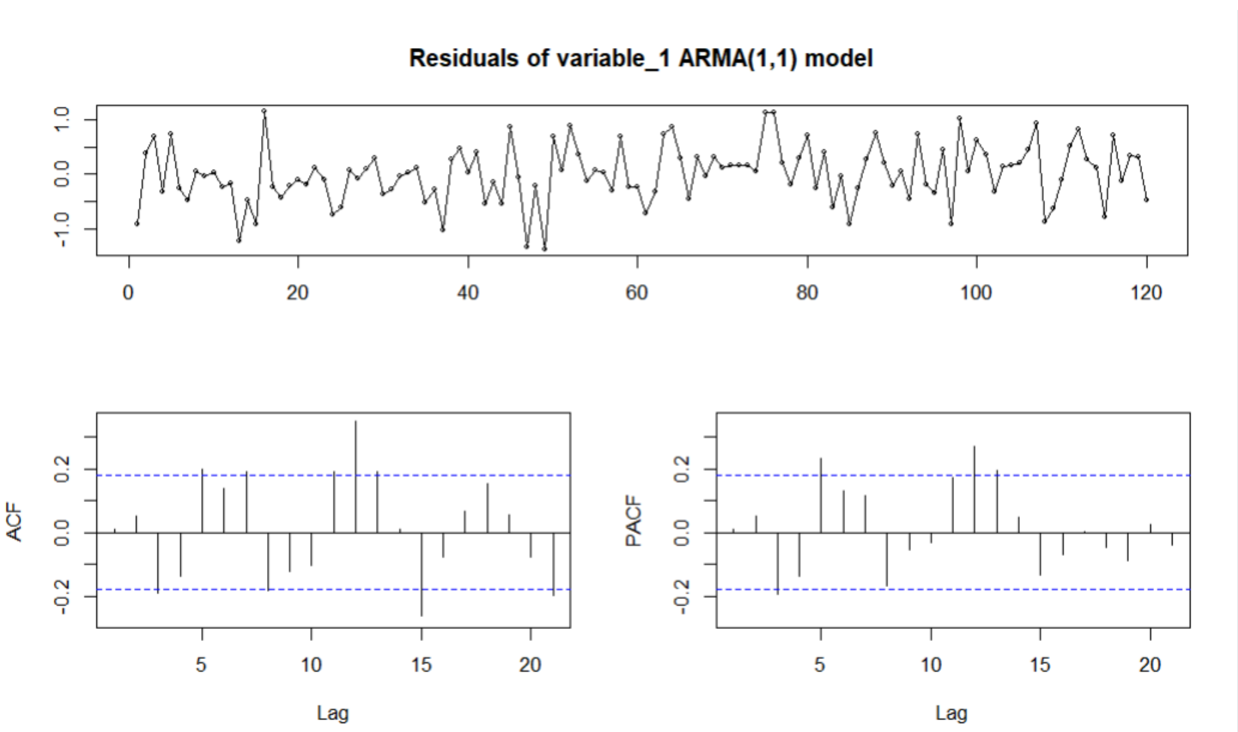



Residuals of variable_0 ARMA(5,5) model
ACF
PACF
Lag
Lag

VARIABLE_1

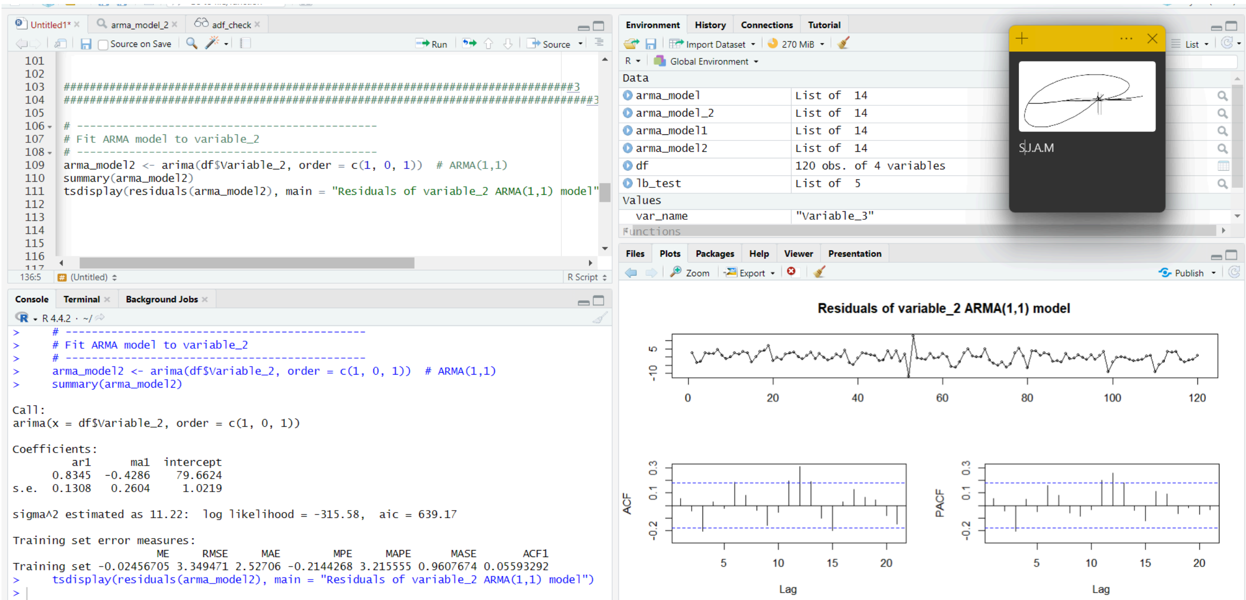


# Fit ARMA model to variable_1
arma_model1 <- arima(df$variable_1, order = c(1, 0, 1)) # ARMA(1,1)
summary(arma_model1)
tsdisplay(residuals(arma_model1), main = "Residuals of variable_1 ARMA(
Ljung-Box test
Environment History Connections Tutorial
Import Dataset 233 MiB Global Environment
Data
arma_model0 List of 14
arma_model1 List of 14
df 120 obs. of 4 variables
Values
S.J.A.M
Console Terminal Background Jobs
R 4.4.2 · ~/
> arma_model1 <- arima(df$variable_1, order = c(1, 0, 1)) # ARMA(1,1)
> summary(arma_model1)
Call:
arima(x = df$variable_1, order = c(1, 0, 1))
Coefficients:
ar1 ma1 intercept
0.5376 0.2500 28.4655
s.e. 0.1020 0.1051 0.1297
sigma^2 estimated as 0.2821: log likelihood = -94.67, aic = 197.34
Training set error measures:
ME RMSE MAE MPE MAPE MASE
Training set 0.007683666 0.5311258 0.4164391 -0.00798365 1.467099 0.9177084
ACF1
Training set 0.01016944
> tsdisplay(residuals(arma_model1), main = "Residuals of variable_1 ARMA(1,1) model")
Files Plots Packages Help Viewer Presentation
Zoom Export
Residuals of variable_1 ARMA(1,1) model
ACF
PACF
Lag
Lag

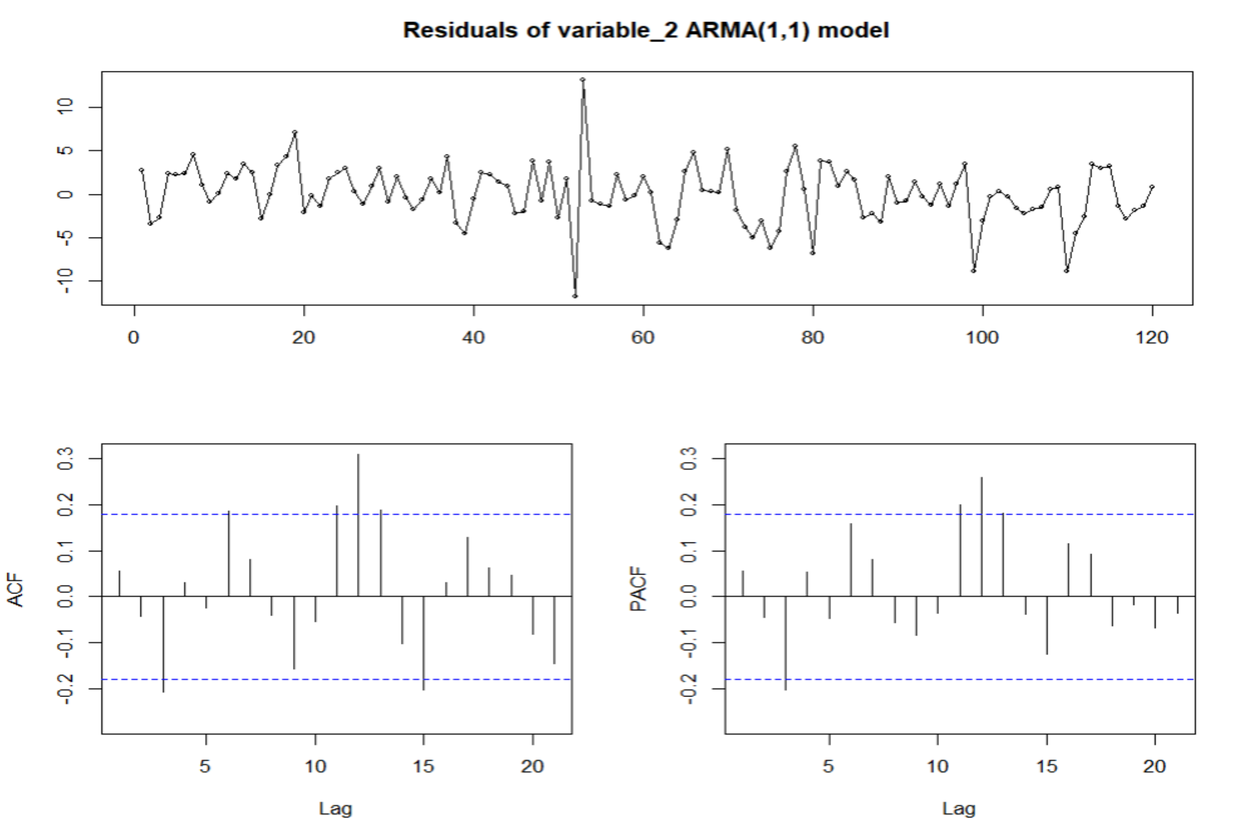


Residuals of variable_1 ARMA(1,1) model
ACF
PACF
Lag
Lag

VARIABLE_2

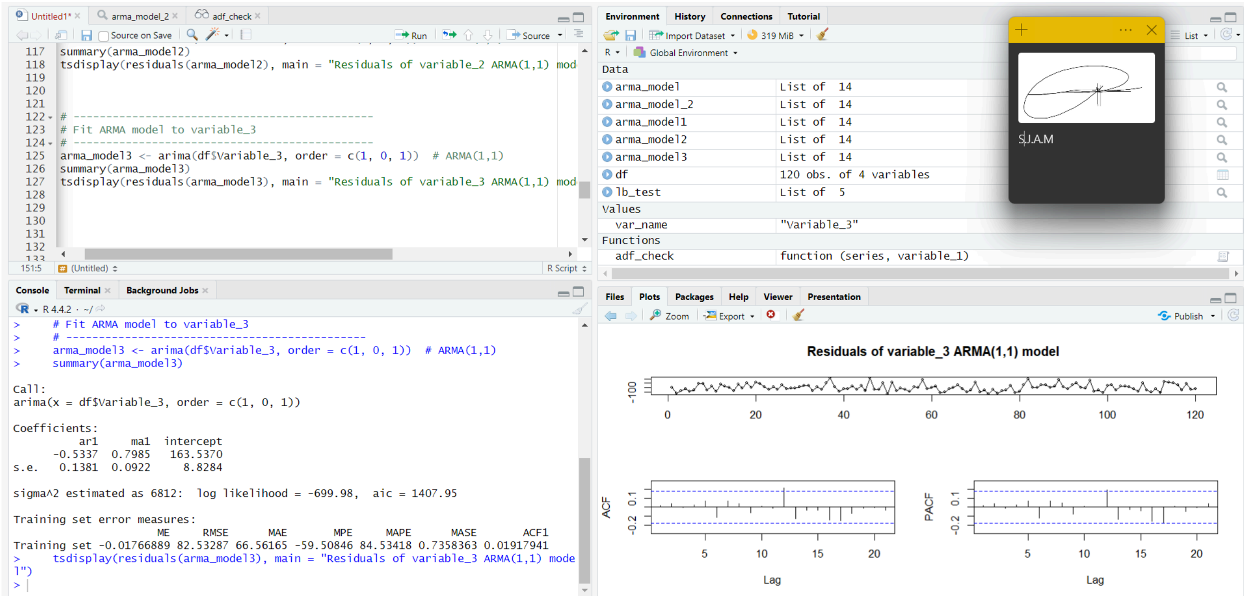


# Fit ARMA model to variable_2
arma_model2 <- arima(df$Variable_2, order = c(1, 0, 1)) # ARMA(1,1)
summary(arma_model2)
tsdisplay(residuals(arma_model2), main = "Residuals of variable_2 ARMA(1,1) model")
Call:
arima(x = df$Variable_2, order = c(1, 0, 1))
Coefficients:
ar1 ma1 intercept
0.8345 -0.4286 79.6624
s.e. 0.1308 0.2604 1.0219
sigma^2 estimated as 11.22: log likelihood = -315.58, aic = 639.17
Training set error measures:
ME RMSE MAE MPE MAPE MASE ACF1
Training set -0.02456705 3.349471 2.52706 -0.2144268 3.215555 0.9607674 0.05593292
Data
arma_model List of 14
arma_model_2 List of 14
arma_model1 List of 14
arma_model2 List of 14
df 120 obs. of 4 variables
lb_test List of 5
Values
var_name "Variable_3"
Residuals of variable_2 ARMA(1,1) model

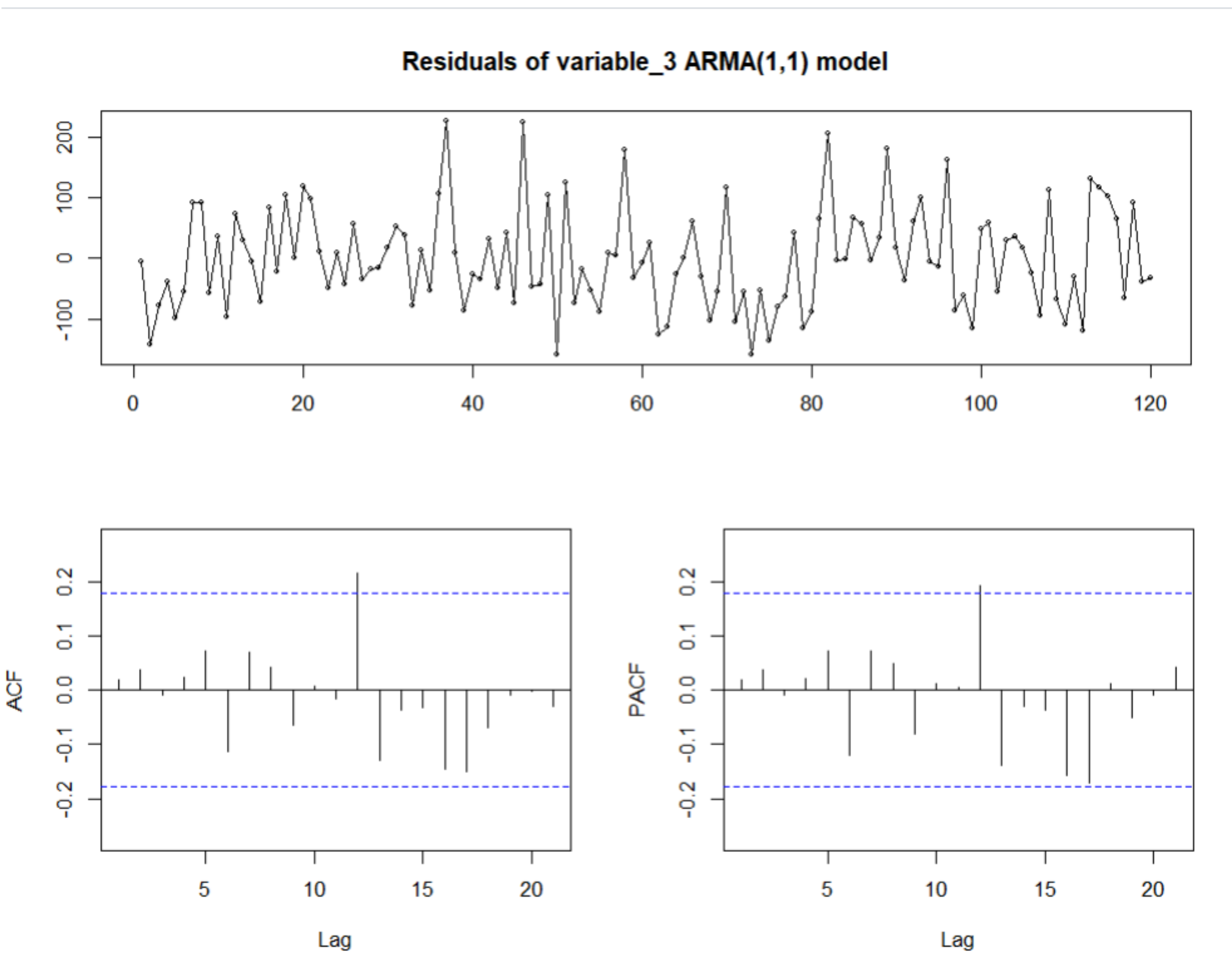


Residuals of variable_2 ARMA(1,1) model
-10
-5
0
5
10
0
20
40
60
80
100
120
ACF
PACF
-0.2
-0.1
0.0
0.1
0.2
0.3
5
10
15
20
Lag

VARIABLE_3



summary(arma_model2)
tsdisplay(residuals(arma_model2), main = "Residuals of variable_2 ARMA(1,1) mod
# --
# Fit ARMA model to variable_3
# --
arma_model3 <- arima(df$variable_3, order = c(1, 0, 1)) # ARMA(1,1)
summary(arma_model3)
tsdisplay(residuals(arma_model3), main = "Residuals of variable_3 ARMA(1,1) mod
Environment
History
Connections
Tutorial
Global Environment
Data
arma_model List of 14
arma_model_2 List of 14
arma_model1 List of 14
arma_model2 List of 14
arma_model3 List of 14
df 120 obs. of 4 variables
lb_test List of 5
Values
var_name "variable_3"
Functions
adf_check function (series, variable_1)
S.J.A.M
Console
Terminal
Background Jobs
Call:
arima(x = df$variable_3, order = c(1, 0, 1))
Coefficients:
ar1 ma1 intercept
-0.5337 0.7985 163.5370
s.e. 0.1381 0.0922 8.8284
sigma^2 estimated as 6812: log likelihood = -699.98, aic = 1407.95
Training set error measures:
ME RMSE MAE MPE MAPE MASE ACF1
Training set -0.01766889 82.53287 66.56165 -59.50846 84.53418 0.7358363 0.01917941
Files
Plots
Packages
Help
Viewer
Presentation
Residuals of variable_3 ARMA(1,1) model
Lag



Residuals of variable_3 ARMA(1,1) model
-100
0
100
200
0
20
40
60
80
100
120
ACF
PACF
-0.2
-0.1
0.0
0.1
0.2
5
10
15
20
Lag

Discussion Questions:

1. What ARMA model did you use for each time series?

Variable_0 ARMA(5,5)
Variable_1 ARMA(1,1)
Variable_2 ARMA(1,1)
Variable_3 ARMA(1,1)

2. How do residuals help assess the goodness of fit of an ARMA model?

Residuals are key indicators for evaluating how well an ARMA model fits a time series. If the model is a good fit, the residuals should resemble white noise—they should average close to zero, maintain constant variance, and show no significant autocorrelation. This is typically confirmed when the residual ACF plot shows all spikes within the confidence bounds and the Ljung-Box test yields a high p-value. Additionally, normally distributed residuals further support the model's validity. On the other hand, visible trends or autocorrelation in the residuals imply that the model may not fully capture the series' structure, indicating a need to revise the AR or MA terms.

Exercise 4d: Ljung-Box Test

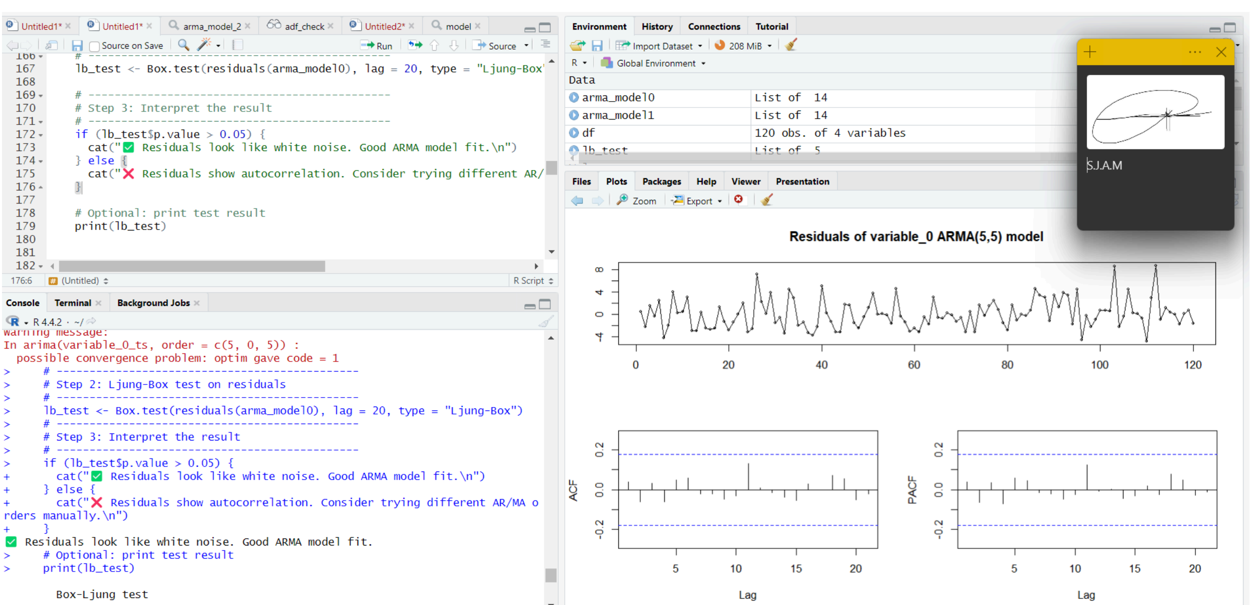
Objective:

Learn how to conduct Ljung Box Test to determine if the model used is a good fit.

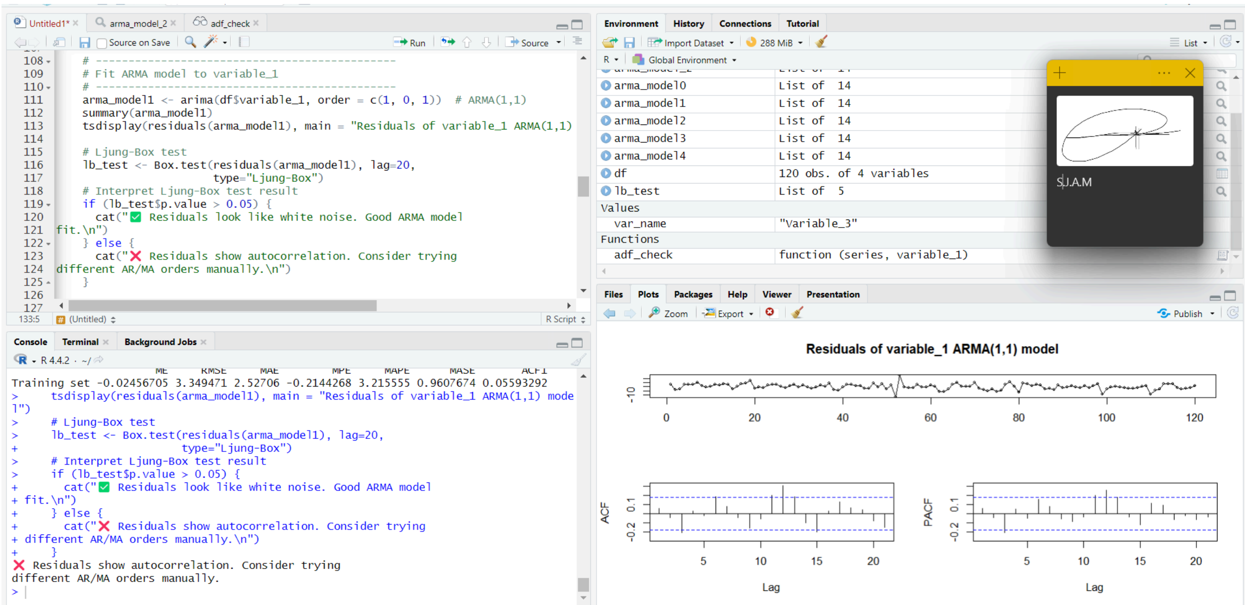
Instructions:

1. Run the code to conduct Ljung Box Test to determine if the model used is a good fit.

VARIABLE_0



VARIABLE_1



VARIABLE_2

VARIABLE_3

Discussion Questions:

1. What is the behavior of the residuals?

Variable 0: ARMA(5,5) – The ACF and PACF plots show that most values fall within the confidence intervals, suggesting there is no significant autocorrelation and that the model fits well. The residuals fluctuate around zero, staying roughly between -4 and 8, with random movements and no visible trend.

Variable 1: ARMA(1,1) – The residuals hover around zero within an approximate range of ±1 and appear random with no clear trend. However, the ACF and PACF plots reveal some spikes outside the confidence bounds, indicating residual autocorrelation, which suggests the model may not provide a good fit.

Variable 2: ARMA(1,1) – The residuals are centered around zero, ranging between -10 and 10, and exhibit random behavior without a consistent trend. Still, the ACF and PACF plots show some values outside the confidence limits, pointing to residual autocorrelation and indicating the model might not be an ideal fit.

Variable 3: ARMA(1,1) – The residuals vary around zero within a wider band of approximately ±100, showing random fluctuations and no discernible trend. Most values in the ACF and PACF plots lie within the confidence intervals, suggesting no significant autocorrelation and supporting that the model fits well.

2. Is the ARMA model a good fit for the time series?

> **Yes**, the ARMA(5,5) model is appropriate for **Variable_0**. The residuals move randomly around zero with no visible trend, and the ACF and PACF plots show most values within the confidence intervals, suggesting no notable autocorrelation. The Ljung-Box test further confirms that the model fits the data well.
>
> **No**, the ARMA(1,1) model does not suit **Variable_1**, as the ACF and PACF plots show spikes beyond the confidence bounds (dashed blue lines), revealing residual autocorrelation that the model fails to address.
>
> **No**, the ARMA(1,1) model is also not a good fit for **Variable_2**. Similar to Variable_1, the ACF and PACF plots include values beyond the confidence limits, indicating residual autocorrelation that the model does not adequately capture.
>
> **Yes**, the ARMA(1,1) model fits **Variable_3** well. The residuals fluctuate randomly around zero with no evident trend, and the ACF and PACF plots mostly lie within the confidence bounds, implying no significant autocorrelation. This is further supported by the Ljung-Box test, which also suggests a good model fit.
