Homework 1

CUNY DATA 624 - 2023 Summer I

Group 3: Daniel Sullivan, Jeff Parks, Lwin Shwe, Matthew Katz

Contents

# Exercise Key

**KJ** - *Applied Predictive Modeling* (Kuhn, Johnson)

**HA** - *Forecasting: Principles and Practice, 2nd Ed.* (Hyndman, Athanasopoulos)

# HA Exercise 2.1

Use the help function to explore what the series gold, woolyrnq and gas represent.

1. Use autoplot() to plot each of these in separate plots.
2. What is the frequency of each series? Hint: apply the frequency() function.
3. Use which.max() to spot the outlier in the gold series. Which observation was it?

# code here

# HA Exercise 2.3

Download some monthly Australian retail data from the book website. These represent retail sales in various categories for different Australian states, and are stored in a MS-Excel file.

1. Select one of the time series … explore your chosen retail time series using the following functions: autoplot(), ggseasonplot(), ggsubseriesplot(), gglagplot(), ggAcf()
2. Can you spot any seasonality, cyclicity and trend? What do you learn about the series?

# code here

# HA Exercise 6.2

The plastics data set consists of the monthly sales (in thousands) of product A for a plastics manufacturer for five years.

1. Plot the time series of sales of product A. Can you identify seasonal fluctuations and/or a trend-cycle?
2. Use a classical multiplicative decomposition to calculate the trend-cycle and seasonal indices.
3. Do the results support the graphical interpretation from part a?
4. Compute and plot the seasonally adjusted data.
5. Change one observation to be an outlier (e.g., add 500 to one observation), and recompute the seasonally adjusted data. What is the effect of the outlier?
6. Does it make any difference if the outlier is near the end rather than in the middle of the time series?

# code here

# KJ Exercise 3.1

The UC Irvine Machine Learning Repository contains a data set related to glass identification. The data consist of 214 glass samples labeled as one of seven class categories. There are nine predictors, including the refractive index and percentages of eight elements: Na, Mg, Al, Si, K, Ca, Ba, and Fe.

1. Using visualizations, explore the predictor variables to understand their distributions as well as the relationships between predictors.
2. Do there appear to be any outliers in the data? Are any predictors skewed?
3. Are there any relevant transformations of one or more predictors that might improve the classification model?

# code here  
library(mlbench)  
data(Glass)

# KJ Exercise 3.2

The UC Irvine Machine Learning Repository contains a data set related to soybeans. Data were collected to predict disease in 683 soybeans. The 35 predictors are mostly categorical and include information on the environmen- tal conditions (e.g., temperature, precipitation) and plant conditions (e.g., left spots, mold growth). The outcome labels consist of 19 distinct classes.

1. Investigate the frequency distributions for the categorical predictors. Are any of the distributions degenerate in the ways discussed earlier in this chapter?
2. Roughly 18 % of the data are missing. Are there particular predictors that are more likely to be missing? Is the pattern of missing data related to the classes?
3. Develop a strategy for handling missing data, either by eliminating predictors or imputation.

# code here  
data(Soybean)

# HA Exercise 7.1

Consider the pigs series — the number of pigs slaughtered in Victoria each month.

1. Use the ses() function in R to find the optimal values of α and ℓ0, and generate forecasts for the next four months.
2. Compute a 95% prediction interval for the first forecast using ^y±1.96s where s is the standard deviation of the residuals. Compare your interval with the interval produced by R.

# code here

# HA Exercise 7.2

Write your own function to implement simple exponential smoothing. The function should take arguments y (the time series), alpha (the smoothing parameter α) and level(the initial level ℓ0). It should return the forecast of the next observation in the series. Does it give the same forecast as ses()?

# code here

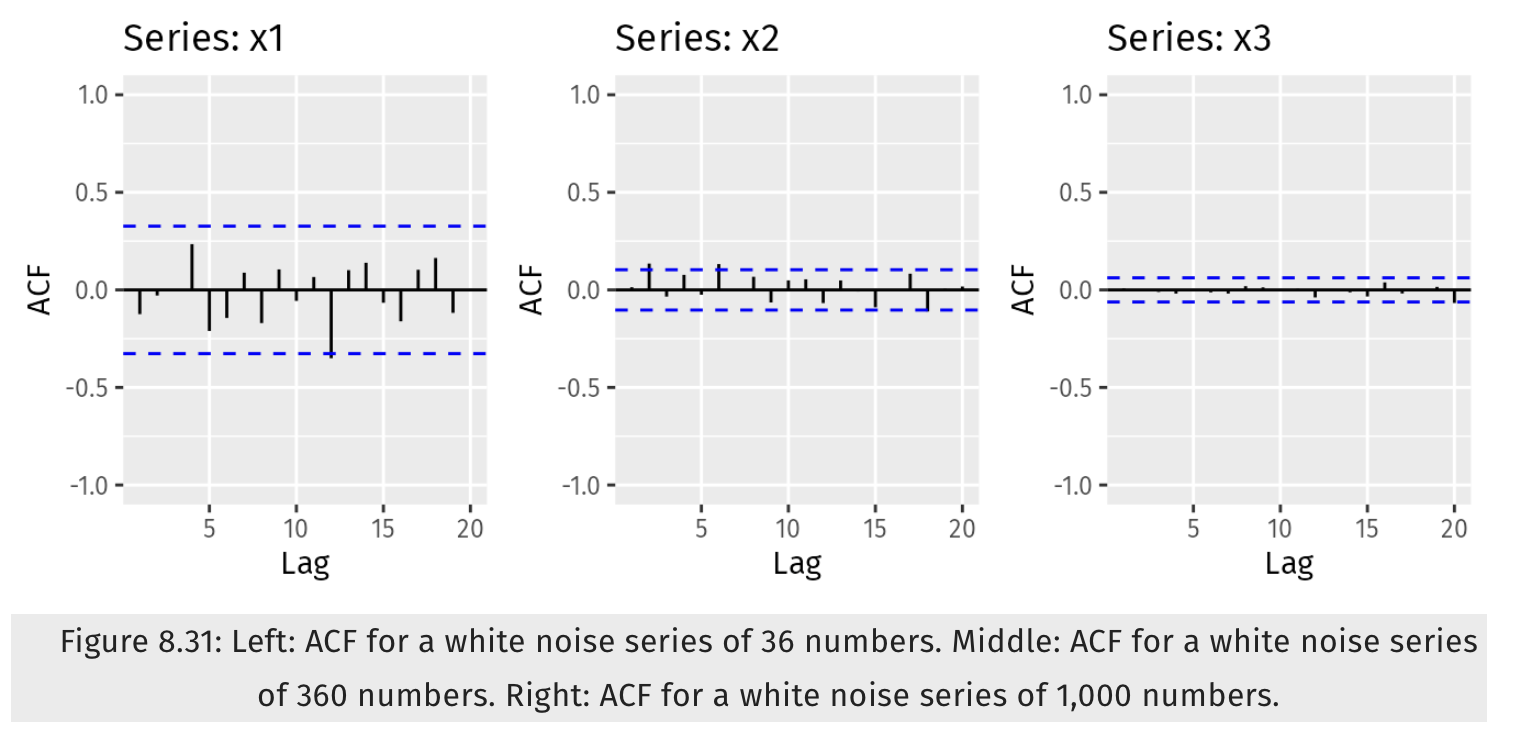
# HA Exercise 7.3

Modify your function from the previous exercise to return the sum of squared errors rather than the forecast of the next observation. Then use the optim() function to find the optimal values of α and ℓ0. Do you get the same values as the ses() function?

# code here

# HA Exercise 8.1

Figure [8.31](https://otexts.com/fpp2/arima-exercises.html#fig:wnacfplus) shows the ACFs for 36 random numbers, 360 random numbers and 1,000 random numbers.



1. Explain the differences among these figures. Do they all indicate that the data are white noise?
2. Why are the critical values at different distances from the mean of zero? Why are the autocorrelations different in each figure when they each refer to white noise?

# code here

# HA Exercise 8.2

A classic example of a non-stationary series is the daily closing IBM stock price series (data set ibmclose). Use R to plot the daily closing prices for IBM stock and the ACF and PACF. Explain how each plot shows that the series is non-stationary and should be differenced.

# code here

# HA Exercise 8.6

Use R to simulate and plot some data from simple ARIMA models.

1. Use the following R code to generate data from an AR(1) model with ϕ1=0.6 and σ2=1. The process starts with y1=0.

y <- ts(numeric(100))  
e <- rnorm(100)  
for(i in 2:100)  
 y[i] <- 0.6\*y[i-1] + e[i]

1. Produce a time plot for the series. How does the plot change as you change ϕ1?
2. Write your own code to generate data from an MA(1) model with θ1=0.6 and σ2=1.
3. Produce a time plot for the series. How does the plot change as you change θ1?
4. Generate data from an ARMA(1,1) model with ϕ1=0.6, θ1=0.6 and σ2=1.
5. Generate data from an AR(2) model with ϕ1=−0.8, ϕ2=0.3 and σ2=1. (Note that these parameters will give a non-stationary series.)
6. Graph the latter two series and compare them.

# code here

# HA Exercise 8.8

Consider austa, the total international visitors to Australia (in millions) for the period 1980-2015.

1. Use auto.arima() to find an appropriate ARIMA model. What model was selected. Check that the residuals look like white noise. Plot forecasts for the next 10 periods.
2. Plot forecasts from an ARIMA(0,1,1) model with no drift and compare these to part a. Remove the MA term and plot again.
3. Plot forecasts from an ARIMA(2,1,3) model with drift. Remove the constant and see what happens.
4. Plot forecasts from an ARIMA(0,0,1) model with a constant. Remove the MA term and plot again.
5. Plot forecasts from an ARIMA(0,2,1) model with no constant.

# code here