DA5020 – Assignment 9

Rebecca Weiss

11/16/2021

Clear the workspace:

```
rm(list = ls())
```

```
#load all necessary libraries
library(tidyverse)
library(lubridate)
library(openintro)
```

1. Load the data into your R environment directly from the URL. Ensure that you inspect the data, so that you know how to identify the necessary columns.

```
url <- 'https://stats.oecd.org/sdmx-json/data/DP_LIVE/.MEATCONSUMP.../OECD?contentType=csv&detail=code&

df <- read.csv(url)

summary(df)
```

```
##     LOCATION           INDICATOR       SUBJECT          MEASURE     FREQUENCY
##  ARG    :  320   MEATCONSUMP:12160   BEEF   :3040   KG_CAP    :6080   A:12160
##  AUS    :  320                       PIG    :3040   THND_TONNE:6080
##  BRA    :  320                       POULTRY:3040
##  BRICS  :  320                       SHEEP  :3040
##  CAN    :  320
##  CHE    :  320
##  (Other):10240
##       TIME          Value           Flag.Codes
##  Min.   :1990   Min.   :     0.00   Mode:logical
##  1st Qu.:2000   1st Qu.:     5.45   NA's:12160
##  Median :2010   Median :    24.77
##  Mean   :2010   Mean   :  2291.58
##  3rd Qu.:2019   3rd Qu.:   433.10
##  Max.   :2029   Max.   :144874.23
##
```

```r
str(df)
```

```
## 'data.frame':    12160 obs. of  8 variables:
##  $ LOCATION  : Factor w/ 38 levels "ARG","AUS","BRA",..: 2 2 2 2 2 2 2 2 2 2 ...
##  $ INDICATOR : Factor w/ 1 level "MEATCONSUMP": 1 1 1 1 1 1 1 1 1 1 ...
##  $ SUBJECT   : Factor w/ 4 levels "BEEF","PIG","POULTRY",..: 1 1 1 1 1 1 1 1 1 1 ...
##  $ MEASURE   : Factor w/ 2 levels "KG_CAP","THND_TONNE": 1 1 1 1 1 1 1 1 1 1 ...
##  $ FREQUENCY : Factor w/ 1 level "A": 1 1 1 1 1 1 1 1 1 1 ...
##  $ TIME      : int  1990 1991 1992 1993 1994 1995 1996 1997 1998 1999 ...
##  $ Value     : num  0 27.9 26.4 26.4 25.7 ...
##  $ Flag.Codes: logi  NA NA NA NA NA NA ...
```

2. Extract the poultry consumption data, from 1994 to 2014, for Mexico, that is measured in thousand tonnes of carcass weight. Pay close attention to the SUBJECT and MEASURE fields to filter the appropriate type of meat and the correct measurement. Visualize the extracted data, using a line chart, and comment on the trend.

```r
# filter data
poultry <- df %>%
  filter(SUBJECT == "POULTRY" & MEASURE == "THND_TONNE"
         & LOCATION == "MEX") %>%
  filter(TIME >= 1994, TIME < 2015) # get correct years only

# view and make sure ranges are correct
head(poultry)
```

```
##   LOCATION   INDICATOR SUBJECT    MEASURE FREQUENCY TIME    Value Flag.Codes
## 1      MEX MEATCONSUMP POULTRY THND_TONNE         A 1994 1369.909         NA
## 2      MEX MEATCONSUMP POULTRY THND_TONNE         A 1995 1515.516         NA
## 3      MEX MEATCONSUMP POULTRY THND_TONNE         A 1996 1505.322         NA
## 4      MEX MEATCONSUMP POULTRY THND_TONNE         A 1997 1750.495         NA
## 5      MEX MEATCONSUMP POULTRY THND_TONNE         A 1998 1931.271         NA
## 6      MEX MEATCONSUMP POULTRY THND_TONNE         A 1999 2080.252         NA
```

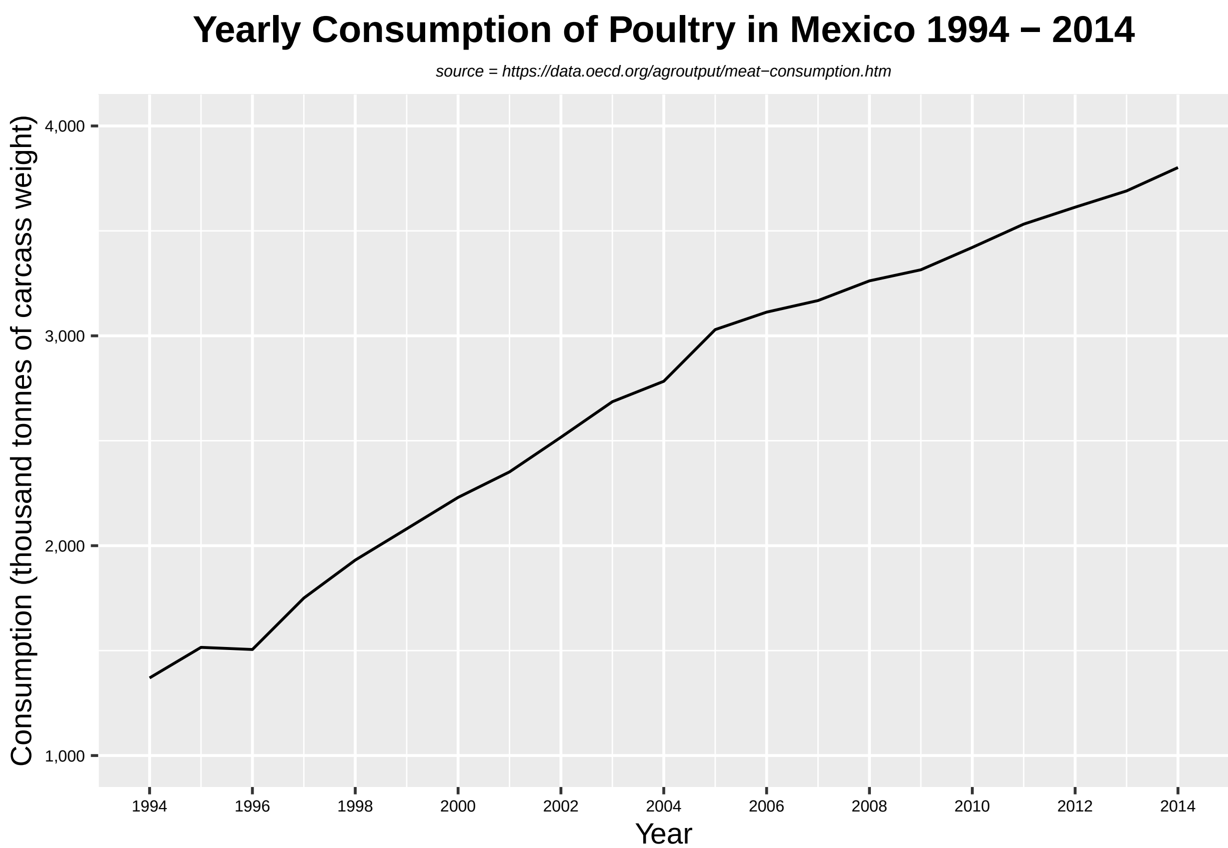
```r
summary(poultry)
```

```
##     LOCATION        INDICATOR      SUBJECT          MEASURE    FREQUENCY
##  MEX    :21   MEATCONSUMP:21   BEEF    : 0   KG_CAP     : 0   A:21
##  ARG    : 0                    PIG     : 0   THND_TONNE:21
##  AUS    : 0                    POULTRY:21
##  BRA    : 0                    SHEEP   : 0
##  BRICS  : 0
##  CAN    : 0
##  (Other): 0
##       TIME          Value        Flag.Codes
##  Min.   :1994   Min.   :1370   Mode:logical
##  1st Qu.:1999   1st Qu.:2080   NA's:21
##  Median :2004   Median :2783
```

```
##  Mean   :2004   Mean   :2698
##  3rd Qu.:2009   3rd Qu.:3315
##  Max.   :2014   Max.   :3802
##
```

From the output of `str(poultry)`, we can confirm that the, SUBJECT, MEASURE, LOCATION and TIME variables are all within the correct ranges the question asked for. Now we will plot it:

```
ggplot(data = poultry, aes(x = TIME, y = Value)) +
  geom_line() +
  labs(title = "Yearly Consumption of Poultry in Mexico 1994 - 2014",
       subtitle = "source = https://data.oecd.org/agroutput/meat-consumption.htm",
       x = "Year",
       y = "Consumption (thousand tonnes of carcass weight)") +
  theme(plot.subtitle=element_text(size=6, hjust=0.5, face="italic", color="black")) +
  theme(plot.title=element_text(size=14, hjust=0.5, face="bold", color="black"))  +
  scale_y_continuous(labels = scales::comma, limits = c(1000, 4000)) +
  scale_x_continuous(limits = c(1994, 2014), breaks = scales::breaks_width(2)) +
  theme(axis.text = element_text(size=6, hjust=0.5, color="black"))
```



From the output of this graph, we see that over time the amount of poultry consumed in Mexico from 1994 - 2014 increases overall. Most notably, the increases in consumption happen consistently, with the exception of from 1995 - 1995, where it appears consumption remains flat and then takes off after 1996.

Use the extracted poultry data to answer the questions below.

3. Forecast the poultry consumption for 2014, using a simple moving average of the following four time periods: 2010, 2011, 2012 and 2013. After which, calculate the error (i.e. the difference between the actual and the predicted values). Evaluate the results; how does it compare to the actual data for 2014?

```
# select variables, filter for years, average
mavg_vals <- poultry %>%
  select(TIME, Value) %>%
  filter(TIME < 2014, TIME > 2009)

mavg = mean(mavg_vals$Value)

# view
# mavg

actual <- poultry %>%
  filter(TIME == 2014) %>%
  select(Value)

error = actual - mavg
```

Using the moving average to forecast the value for 2014, we get 3564.14575, which is slightly lower than the actual 2014 value, 3801.833. The error from the forecast and actual value = 237.68725 thousand tonnes of carcass weight.

4. Forecast the poultry consumption for 2014, using a three year weighted moving average. Apply the following weights: 5, 7, and 15 for the respective years 2011, 2012, and 2013. After which, calculate the error and evaluate the result from your prediction.

```
# get rid of 2010 year, add weights
new <-mavg_vals %>%
  filter(TIME > 2010) %>%
  mutate("Weight" = c(5, 7, 15), "Weight_Value" = Value * Weight)

weight_avg = sum(new$Weight_Value)/sum(new$Weight)
error_weight = actual - weight_avg
```

As we can see from the output calculations, when adding weights and using the years 2011, 2012, and 2013 for a 3 year weighted moving average, we get a forecast of 3640.9651852 for 2014. The forecasted value is still lower than actual value, 3801.833, with an error of 160.867814814815. Thus, the forecasted value using this method is slightly closer to the actual value for 2014 than the moving average from 2010, 2011, 2012 and 2013, as seen in question 3.

5. Forecast the poultry consumption for 2014 using exponential smoothing (alpha is 0.9). Comment on the prediction for 2014 with the actual value. Note: use data from 1994 to 2013 to build your model.

```
# select variables we want, drop 2014
smooth_df <- poultry %>%
  select(TIME, Value) %>%
  filter(TIME != 2014)

# add new values to df
smooth_df$Ft <- 0
smooth_df$E <- 0

# have to calculate first row manually
smooth_df$Ft[1] <- smooth_df[1,2]

# iterate over the rest of the rows
for (i in 2:nrow(smooth_df)) {
  smooth_df$Ft[i] <- smooth_df$Ft[i-1] + 0.9*smooth_df$E[i-1]
  smooth_df$E[i] <- smooth_df[i,2] - smooth_df$Ft[i]
}

# view Error Ft calculated
smooth_df
```

```
##    TIME    Value       Ft         E
## 1  1994 1369.909 1369.909   0.00000
## 2  1995 1515.516 1369.909 145.60700
## 3  1996 1505.322 1500.955   4.36670
## 4  1997 1750.495 1504.885 245.60967
## 5  1998 1931.271 1725.934 205.33697
## 6  1999 2080.252 1910.737 169.51470
## 7  2000 2229.966 2063.301 166.66547
## 8  2001 2351.655 2213.299 138.35555
## 9  2002 2516.807 2337.819 178.98755
## 10 2003 2686.007 2498.908 187.09876
## 11 2004 2783.345 2667.297 116.04788
## 12 2005 3029.620 2771.740 257.87979
## 13 2006 3112.794 3003.832 108.96198
## 14 2007 3167.940 3101.898  66.04220
## 15 2008 3261.932 3161.336 100.59622
## 16 2009 3314.587 3251.872  62.71462
## 17 2010 3421.165 3308.316 112.84946
## 18 2011 3532.197 3409.880 122.31695
## 19 2012 3612.905 3519.965  92.93969
## 20 2013 3690.316 3603.611  86.70497
```

```
n <- nrow(smooth_df)
f_exp <- smooth_df$Ft[n] + 0.9*smooth_df$E[n]
error_smooth <- actual - f_exp
```

Using the data from 1994-2013 for exponential smoothing with an alpha = 0.90, the forecast of the value of consumption in 2014 = 3681.6455031, and the actual value = 3801.833. The error between the exponentially smoothed value and the actual for 2014 = 120.18749694622. As we can see from the output, the exponential smoothing method with an alpha = 0.90 has the lowest difference between the predicted vs actual 2014 value, and is just ~120 thousand tonnes lower than the true value.

6. Build a simple linear regression model using the TIME and VALUE for all data from 1994 to 2013. After which, forecast the poultry consumption for 2014 to 2016. Comment on the results. Note: Your predictions should be calculated using the coefficients. Do not use any libraries to make your predictions.

```r
# select variables we want, drop 2014
slr <- poultry %>%
  select(TIME, Value) %>%
  filter(TIME != 2014)

# build linear regression model
model <- lm(Value ~ TIME, data = slr)
summary(model)
```

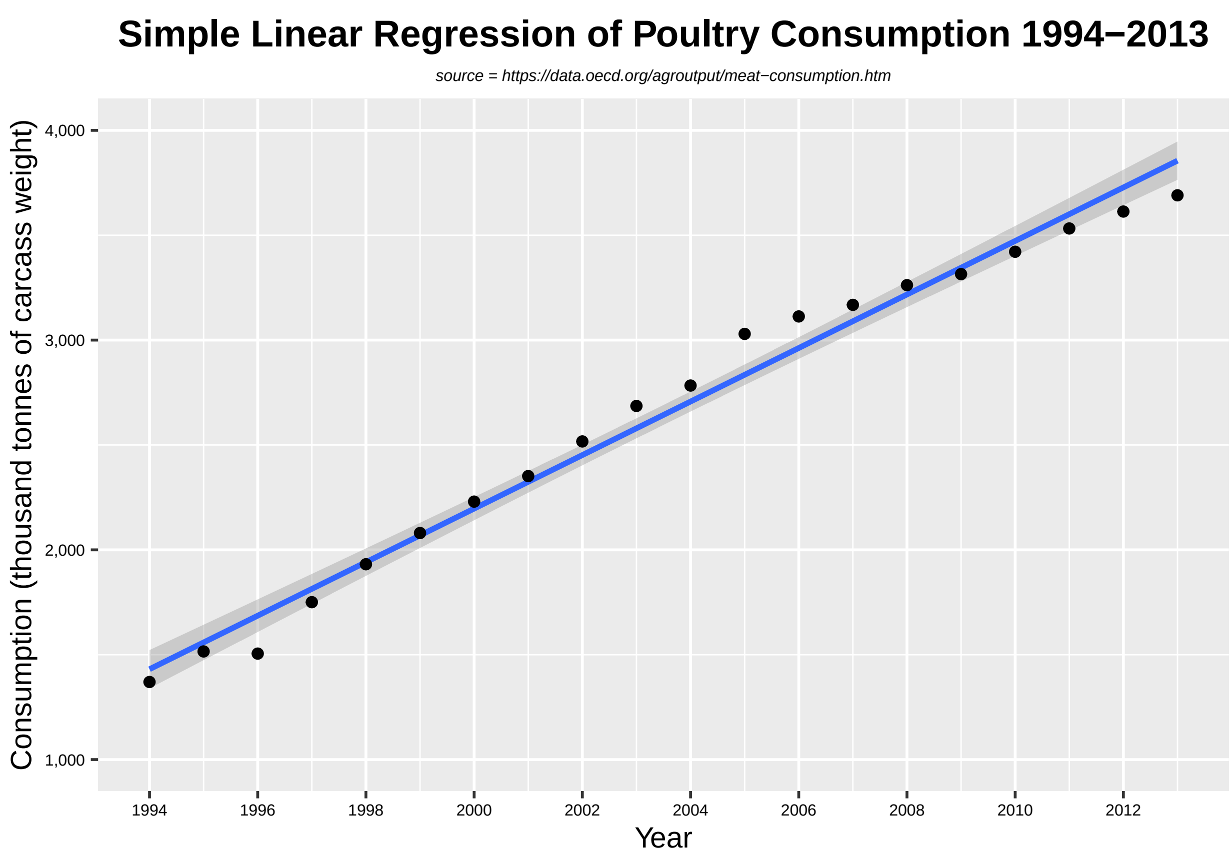
```
##
## Call:
## lm(formula = Value ~ TIME, data = slr)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -180.787  -61.539    0.622   67.854  195.002
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -2.530e+05  7.840e+03  -32.27   <2e-16 ***
## TIME         1.276e+02  3.913e+00   32.61   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 100.9 on 18 degrees of freedom
## Multiple R-squared:  0.9834, Adjusted R-squared:  0.9824
## F-statistic:  1064 on 1 and 18 DF,  p-value: < 2.2e-16
```

```r
# build to estimate value using output of model
slinreg <- function(x)  {
  coeff = as.integer(model$coefficients[2])
  int = as.integer(model$coefficients[1])
  ypred = (coeff * x) + int
  return(ypred)
}

# apply function to get predictions for years
years <- c(2014, 2015, 2016)
preds <- sapply(years, slinreg)

# View predicted values as a tibble for each year
tibble("2014" = preds[1], "2015" = preds[2], "2016" = preds[3])
```

```
## # A tibble: 1 x 3
##   `2014` `2015` `2016`
##    <dbl>  <dbl>  <dbl>
## 1   2751   2878   3005
```

```r
# visualize data used to build linear regression (extra)
ggplot(slr, aes(x=TIME, y=Value)) +
  geom_smooth(method = 'lm') +
  geom_point() +
  labs(title = "Simple Linear Regression of Poultry Consumption 1994-2013",
       subtitle = "source = https://data.oecd.org/agroutput/meat-consumption.htm",
       x = "Year",
       y = "Consumption (thousand tonnes of carcass weight)") +
  theme(plot.subtitle=element_text(size=6, hjust=0.5, face="italic", color="black")) +
  theme(plot.title=element_text(size=14, hjust=0.5, face="bold", color="black"))  +
  scale_y_continuous(labels = scales::comma, limits = c(1000, 4000)) +
  scale_x_continuous(limits = c(1994, 2013), breaks = scales::breaks_width(2)) +
  theme(axis.text = element_text(size=6, hjust=0.5, color="black"))
```

```
## `geom_smooth()` using formula 'y ~ x'
```



The tibble output shows that based on the simple linear regression model, the predictions for 2014 = 2751, 2015 = 2878, and 2016 = 3005. If we look at the graph with the smooth linear regression line, this makes sense; it look like from 2009 and on there are a general trend for the values to deviate below the trendline, so it is possible the values are smaller. One other thing to keep in mind as well is that the actual value for 2014 = 3801.833, which is much higher than the predicted value for 2014. Additionally, the moving averages were much closer to the actual value, suggesting that may be a better method for forecasting, at least for the year 2014.