CSC 360 Assignment 1 Instructions

Harold Nelson

January 31, 2017

This assignment is an analysis of weather in Olympia, Washington. It uses daily data from July 1, 1877 to the middle of July 2017. The purpose. You need to use appropriate ggplot2 and dplyr instructions to answer the questions below. ## Load the required libraries.

library(tidyverse)

## -- Attaching packages ------------------------------------------------------------------------------------------------------------------ tidyverse 1.2.1 --

## v ggplot2 2.2.1 v purrr 0.2.4  
## v tibble 1.4.2 v dplyr 0.7.4  
## v tidyr 0.7.2 v stringr 1.2.0  
## v readr 1.1.1 v forcats 0.2.0

## -- Conflicts --------------------------------------------------------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

## Problem 1

The first step is to load the data. The load command in the following chunk works on my computer, but you will have to modify it. Of course before you can do this you need to download the data from the course onto your computer. After that, you can bring it into RStudio with the following steps.

1. Click on “File”
2. Click on “Open File”
3. Navigate to the file and double-click it.
4. Say yes.
5. Copy the command that this process placed in your console.
6. Paste this on top of the command from my system. Leave this command in your first chunk so that it runs everytime you knit.

Run the commands glimpse() and summary() on your file to verify that your import was successful.

# The load command below must be run every time you run knitr.  
  
load("C:/Users/MSSA/Desktop/olywthr (1).rdata")  
  
glimpse(olywthr)

## Observations: 49,316  
## Variables: 9  
## $ STATION\_NAME <chr> "OLYMPIA PRIEST PT PA WA US", "OLYMPIA PRIEST PT ...  
## $ DATE <date> 1877-07-01, 1877-07-02, 1877-07-03, 1877-07-04, ...  
## $ PRCP <dbl> 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0.00, 0...  
## $ SNOW <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0...  
## $ TMAX <int> 63, 63, 67, 67, 71, 74, 80, 88, 81, 70, 71, 75, 7...  
## $ TMIN <int> 48, 53, 45, 45, 43, 49, 49, 50, 57, 57, 45, 47, 4...  
## $ yr <dbl> 1877, 1877, 1877, 1877, 1877, 1877, 1877, 1877, 1...  
## $ mo <dbl> 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7, 7...  
## $ dy <int> 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15...

summary(olywthr)

## STATION\_NAME DATE PRCP   
## Length:49316 Min. :1877-07-01 Min. :0.0000   
## Class :character 1st Qu.:1913-05-10 1st Qu.:0.0000   
## Mode :character Median :1949-12-24 Median :0.0000   
## Mean :1948-11-12 Mean :0.1409   
## 3rd Qu.:1983-09-26 3rd Qu.:0.1400   
## Max. :2017-07-11 Max. :4.8200   
## SNOW TMAX TMIN yr   
## Min. : 0.00000 Min. : 15.00 Min. :-8.00 Min. :1877   
## 1st Qu.: 0.00000 1st Qu.: 50.00 1st Qu.:34.00 1st Qu.:1913   
## Median : 0.00000 Median : 59.00 Median :41.00 Median :1949   
## Mean : 0.02647 Mean : 60.64 Mean :40.42 Mean :1948   
## 3rd Qu.: 0.00000 3rd Qu.: 71.00 3rd Qu.:47.00 3rd Qu.:1983   
## Max. :14.20000 Max. :104.00 Max. :76.00 Max. :2017   
## mo dy   
## Min. : 1.000 Min. : 1.00   
## 1st Qu.: 4.000 1st Qu.: 8.00   
## Median : 7.000 Median :16.00   
## Mean : 6.516 Mean :15.74   
## 3rd Qu.:10.000 3rd Qu.:23.00   
## Max. :12.000 Max. :31.00

Be careful to keep the R code you need within chunks. Always include the R code and output needed to answer the questions.

## Problem 2

Let’s create a small dataframe to work with containing only data from the years 2014 through 2016. Call this dataframe ‘recent’. Use the filter() command from dplyr to do this. Run the summary() command to verify that your use of filter() worked correctly. Cite two specific results in the output to support your belief that recent is what you wanted.

# Place your R code here.  
recent <- filter(olywthr, yr >= 2014 & yr<= 2016)  
summary(recent)

## STATION\_NAME DATE PRCP   
## Length:1096 Min. :2014-01-01 Min. :0.0000   
## Class :character 1st Qu.:2014-10-01 1st Qu.:0.0000   
## Mode :character Median :2015-07-02 Median :0.0000   
## Mean :2015-07-02 Mean :0.1547   
## 3rd Qu.:2016-04-01 3rd Qu.:0.1600   
## Max. :2016-12-31 Max. :2.8900   
## SNOW TMAX TMIN yr   
## Min. :0.000000 Min. :29.00 Min. :13.00 Min. :2014   
## 1st Qu.:0.000000 1st Qu.:52.00 1st Qu.:36.00 1st Qu.:2014   
## Median :0.000000 Median :61.00 Median :43.00 Median :2015   
## Mean :0.003832 Mean :62.85 Mean :42.01 Mean :2015   
## 3rd Qu.:0.000000 3rd Qu.:73.00 3rd Qu.:49.00 3rd Qu.:2016   
## Max. :3.200000 Max. :98.00 Max. :61.00 Max. :2016   
## mo dy   
## Min. : 1.000 Min. : 1.00   
## 1st Qu.: 4.000 1st Qu.: 8.00   
## Median : 7.000 Median :16.00   
## Mean : 6.522 Mean :15.73   
## 3rd Qu.:10.000 3rd Qu.:23.00   
## Max. :12.000 Max. :31.00

# The TMAX min is 29 and Max is 98 for the years of 2014 - 2016.  
# The mean days that were reported is 15.73 days.

# The yr column show a min value of 2014 and a max value of 2016. While the mo column shows a max of 12 months that are counted.

## Problem 3

Provide the basic descriptive statistics and a histogram for maximum daily temperature (TMAX) in recent. You can use summary() but you need to add the interquartile range and the standard deviation. Is this distribution symmetric? Make two correct statements about TMAX. Use ggplot2 commands to produce the graphics.

# Place the R code you need to answer this question in this chunk.  
summary(recent$TMAX)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 29.00 52.00 61.00 62.85 73.00 98.00

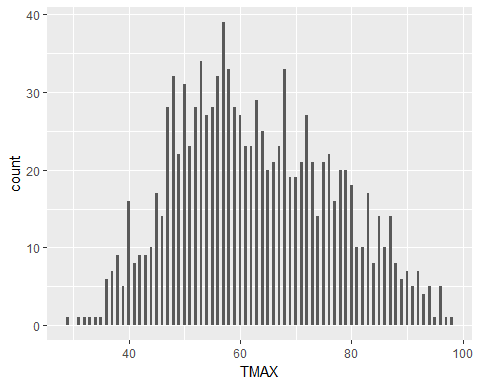
sd(recent$TMAX)

## [1] 13.9333

IQR(recent$TMAX)

## [1] 21

ggplot(recent, aes(TMAX)) + geom\_histogram(binwidth=0.5)



#The distribution is more one sided than symmetric. The data starts out high and gets lower  
#As the histogram progresses from left to right, it grows to the max then drops back down to the min.  
#The min is 29, mean is 62.85, and max is 98.

## Problem 5

Applying filter to recent, Create a smaller dataframe (sepoct) containing only observations from the months of September and October. Use summary() to verify your results and cite two items in the output which are consistent with success. Use tapply() with summary() to compare the TMAX values from these two months. Produce a side-by-side boxplot. Make two correct statements to describe your results.

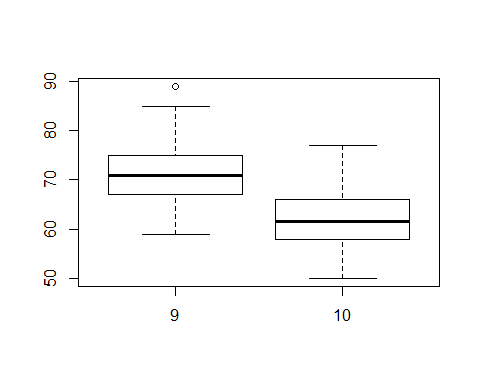
# Place your R code here.  
sepoct <- filter(recent, mo == 9:10)  
summary(sepoct)

## STATION\_NAME DATE PRCP SNOW   
## Length:91 Min. :2014-09-02 Min. :0.0000 Min. :0   
## Class :character 1st Qu.:2014-10-16 1st Qu.:0.0000 1st Qu.:0   
## Mode :character Median :2015-09-29 Median :0.0100 Median :0   
## Mean :2015-09-27 Mean :0.1849 Mean :0   
## 3rd Qu.:2016-09-14 3rd Qu.:0.1850 3rd Qu.:0   
## Max. :2016-10-30 Max. :2.0100 Max. :0   
## TMAX TMIN yr mo   
## Min. :50.00 Min. :29.00 Min. :2014 Min. : 9.000   
## 1st Qu.:61.00 1st Qu.:42.00 1st Qu.:2014 1st Qu.: 9.000   
## Median :66.00 Median :46.00 Median :2015 Median :10.000   
## Mean :66.75 Mean :45.96 Mean :2015 Mean : 9.505   
## 3rd Qu.:71.50 3rd Qu.:50.50 3rd Qu.:2016 3rd Qu.:10.000   
## Max. :89.00 Max. :59.00 Max. :2016 Max. :10.000   
## dy   
## Min. : 1.00   
## 1st Qu.: 8.00   
## Median :16.00   
## Mean :15.67   
## 3rd Qu.:23.00   
## Max. :31.00

tapply(sepoct$TMAX, sepoct$mo, summary)

## $`9`  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 59.00 67.00 71.00 70.91 75.00 89.00   
##   
## $`10`  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 50.00 58.00 61.50 62.67 66.00 77.00

boxplot(sepoct$TMAX~sepoct$mo)



#During the month of September there was a higher TMAX ratio than October.  
#The lowest number of accounted days at any one time was 1.

## Problem 6

Use tapply() with summary() to compare the TMAX values from these two months. Produce a side-by-side boxplot. Make two correct statements to describe your results.

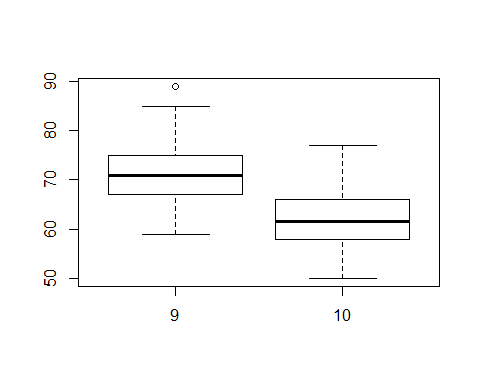
sepoct <- filter(recent, mo == 9:10)  
summary(sepoct)

## STATION\_NAME DATE PRCP SNOW   
## Length:91 Min. :2014-09-02 Min. :0.0000 Min. :0   
## Class :character 1st Qu.:2014-10-16 1st Qu.:0.0000 1st Qu.:0   
## Mode :character Median :2015-09-29 Median :0.0100 Median :0   
## Mean :2015-09-27 Mean :0.1849 Mean :0   
## 3rd Qu.:2016-09-14 3rd Qu.:0.1850 3rd Qu.:0   
## Max. :2016-10-30 Max. :2.0100 Max. :0   
## TMAX TMIN yr mo   
## Min. :50.00 Min. :29.00 Min. :2014 Min. : 9.000   
## 1st Qu.:61.00 1st Qu.:42.00 1st Qu.:2014 1st Qu.: 9.000   
## Median :66.00 Median :46.00 Median :2015 Median :10.000   
## Mean :66.75 Mean :45.96 Mean :2015 Mean : 9.505   
## 3rd Qu.:71.50 3rd Qu.:50.50 3rd Qu.:2016 3rd Qu.:10.000   
## Max. :89.00 Max. :59.00 Max. :2016 Max. :10.000   
## dy   
## Min. : 1.00   
## 1st Qu.: 8.00   
## Median :16.00   
## Mean :15.67   
## 3rd Qu.:23.00   
## Max. :31.00

tapply(sepoct$TMAX, sepoct$mo, summary)

## $`9`  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 59.00 67.00 71.00 70.91 75.00 89.00   
##   
## $`10`  
## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 50.00 58.00 61.50 62.67 66.00 77.00

boxplot(sepoct$TMAX~sepoct$mo)



#The average TMAX in the month of september is higher than that of october.  
#The min TMAX in both months are within the 50's.

## Problem 7

Create a new boolean variable QRain in the dataframe sepoct. If PRCP is greater than 0, the variable should be set to TRUE. Otherwise it should be set to FALSE. Produce a table and a barplot of QRain. Use ggplot2 to create the graphic.

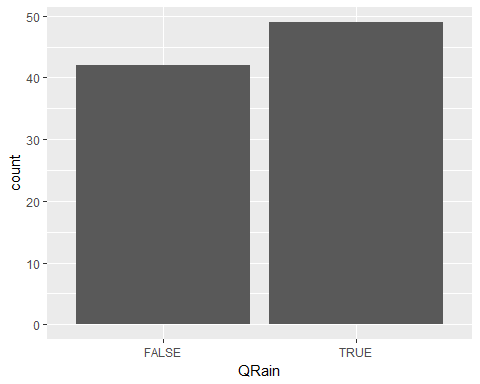
sepoct$QRain <- ifelse( sepoct$PRCP > 0,"TRUE","FALSE")  
print(sepoct$QRain)

## [1] "FALSE" "FALSE" "FALSE" "FALSE" "FALSE" "FALSE" "FALSE" "TRUE"   
## [9] "FALSE" "FALSE" "TRUE" "TRUE" "TRUE" "FALSE" "FALSE" "FALSE"  
## [17] "FALSE" "FALSE" "FALSE" "FALSE" "TRUE" "TRUE" "TRUE" "TRUE"   
## [25] "TRUE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE"   
## [33] "TRUE" "FALSE" "FALSE" "FALSE" "FALSE" "FALSE" "FALSE" "TRUE"   
## [41] "FALSE" "FALSE" "FALSE" "TRUE" "FALSE" "FALSE" "TRUE" "FALSE"  
## [49] "FALSE" "FALSE" "TRUE" "TRUE" "FALSE" "FALSE" "TRUE" "TRUE"   
## [57] "FALSE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE"   
## [65] "FALSE" "FALSE" "FALSE" "FALSE" "FALSE" "TRUE" "TRUE" "FALSE"  
## [73] "TRUE" "FALSE" "TRUE" "FALSE" "TRUE" "TRUE" "TRUE" "TRUE"   
## [81] "FALSE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE" "TRUE"   
## [89] "TRUE" "TRUE" "TRUE"

table(sepoct$QRain)

##   
## FALSE TRUE   
## 42 49

ggplot(sepoct, aes(QRain)) + geom\_bar()



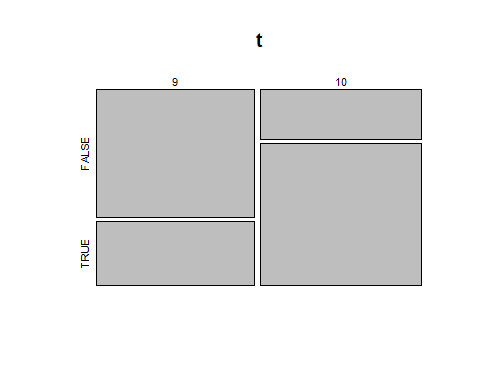
## Problem 8

Produce a table and a mosaicplot to describe the relationship between the variables QRain and mo in the dataframe sepoct. Describe what you see.

table(sepoct$QRain, sepoct$mo)

##   
## 9 10  
## FALSE 30 12  
## TRUE 15 34

t <- table(sepoct$mo, sepoct$QRain)  
mosaicplot(t)

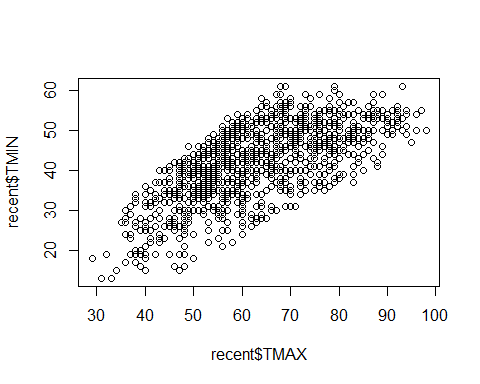


#I see two true and false boxes for both months. Month 9 has a higher false and lower true than month ten, month 10 has a low false and higher true.

## Problem 7

Produce a scatterplot to describe the relationship between TMAX and TMIN using the data in recent. Compute the correlation coefficient. Describe the meaning of the the correlation coefficient. Does it agree with the visual results in the plot?

plot(recent$TMAX, recent$TMIN)



cor(recent$TMAX, recent$TMIN)

## [1] 0.6826072

#Yes it does agree with the visual results in the plot.

## Problem 8

Produce a linear model using the dataframe recent which could be used to predict the value of TMIN from a given value of TMAX.

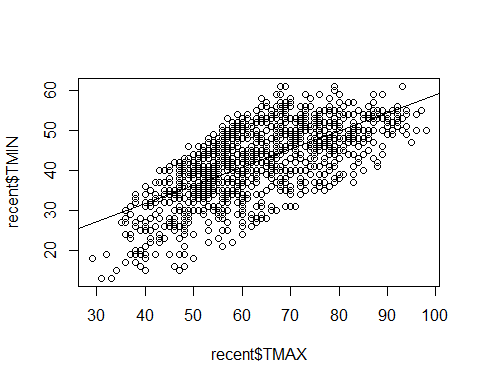
Display the summary results of the linear model.

Use the results of the model to predict the value of TMIN if the value of TMAX is 106. Show the R code you used to make this prediction.

Sally <- lm(recent$TMIN~recent$TMAX)  
summary(Sally)

##   
## Call:  
## lm(formula = recent$TMIN ~ recent$TMAX)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -19.8449 -4.3135 0.5061 4.7610 16.6648   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 13.60479 0.94171 14.45 <2e-16 \*\*\*  
## recent$TMAX 0.45192 0.01463 30.89 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 6.744 on 1094 degrees of freedom  
## Multiple R-squared: 0.466, Adjusted R-squared: 0.4655   
## F-statistic: 954.5 on 1 and 1094 DF, p-value: < 2.2e-16

plot(recent$TMIN~recent$TMAX)  
abline(Sally)



b = 13.60479  
m = 0.45192  
TMAX = 106  
TMIN = m\*TMAX + b  
print(TMIN)

## [1] 61.50831