Due: December 08, 2016 (Thursday)

**Question 01 (10 points)**

Create the X matrix and print it from SAS, R, and Python.

**SAS code**

**proc** **iml** ;

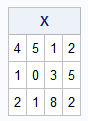
X = {**4** **5** **1** **2**,

**1** **0** **3** **5**,

**2** **1** **8** **2**} ;

print X ;

**run** ;



**R code**

X = matrix(c(4,5,1,2,1,0,3,5,2,1,8,2),

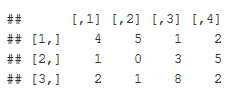
nrow=3,

ncol=4,

byrow=TRUE # fill matrix by rows

)

X

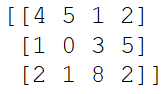


**Python Code**

**import** numpy **as** np

X = np.matrix('4 5 1 2;1 0 3 5;2 1 8 2')

print (X)



**Question 02 (15 points)**

Please watch videos 1 and 2 in week 11 lecture assignment. You can download the code which used for S&P from files tab.

Please do the following with your assigned stock.

|  |  |
| --- | --- |
| **Group** | **Stock** |
| Jim, Vivek , and Brandon | AGIO |

* Download the data.

#Libraries

#install.packages("tseries")

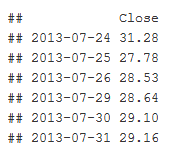
library(tseries)

#Download Stock Data for 'AGIO'

SNPdata <- get.hist.quote('agio',quote="Close")

length(SNPdata)

head(SNPdata)



* Calculate log returns.

#Create Log Returns

SNPret <- log(lag(SNPdata)) - log(SNPdata)

length(SNPret)

head(SNPret)

* Calculate volatility measure.

#Calculate Volatility of the complete dataset

SNPvol <- sd(SNPret) \* sqrt(250) \* 100

SNPvol

## [1] 79.47745

* Calculate volatility over entire length of series for various three different decay factors.

#Calculate Volatility of the complete dataset

SNPvol <- sd(SNPret) \* sqrt(250) \* 100

SNPvol

## Create volatility Look back window function

Vol <- function(d, logrets){

var = 0

lam = 0

varlist <- c()

for (r in logrets) {

lam = lam\*(1 - 1/d) + 1

var = (1 - 1/lam)\*var + (1/lam)\*r^2

varlist <- c(varlist, var)

}

sqrt(varlist)

}

# Volatility estimate over entire time with different decay factors(weights).

volest <- Vol( 10,SNPret)

volest2 <- Vol( 30,SNPret)

volest3 <- Vol(100,SNPret)

* Plot the results, overlaying the volatility curves on the data, just as was done in the S&P example.

# Plot the results with different decays.

plot(volest,type="l",

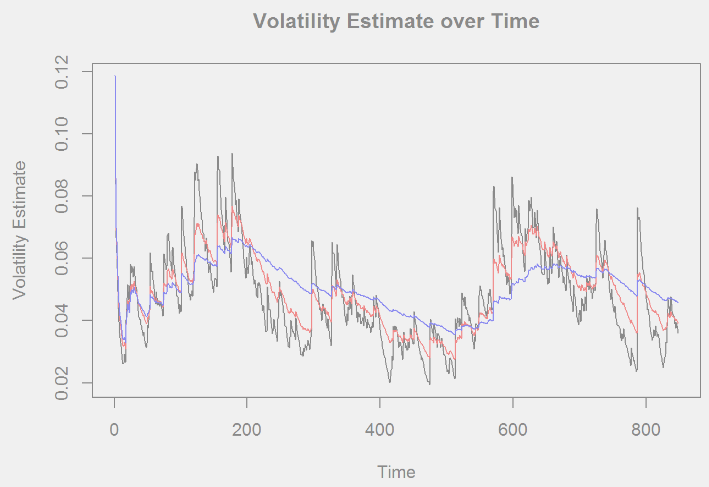
main = "Volatility Estimate over Time",

xlab = 'Time',

ylab = 'Volatility Estimate')

lines(volest2,type="l",col="red")

lines(volest3, type = "l", col="blue")



**Question 03 (20 points)**

The built-in data set called ***Orange*** in R is about the growth of orange trees. The Orange data frame has 3 columns of records of the growth of orange trees.

#Libraries

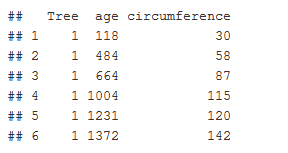
library(plyr)

library(ggplot2)

#Load Orange Data (included with R)

data("Orange")

head (Orange)



Variable description

**Tree** : an ordered factor indicating the tree on which the measurement is made. The ordering

is according to increasing maximum diameter.

**age** : a numeric vector giving the age of the tree (days since 1968/12/31)

**circumference** : a numeric vector of trunk circumferences (mm). This is probably

“circumference at breast height”, a standard measurement in forestry.

1. Calculate the mean and the median of the trunk circumferences for different size of the trees. (Tree)

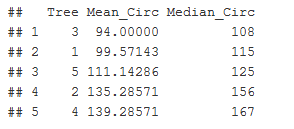
summOrange <- ddply(Orange, .(Tree), summarize,

Mean\_Circ =mean(circumference),

Median\_Circ=median(circumference)

)

summOrange



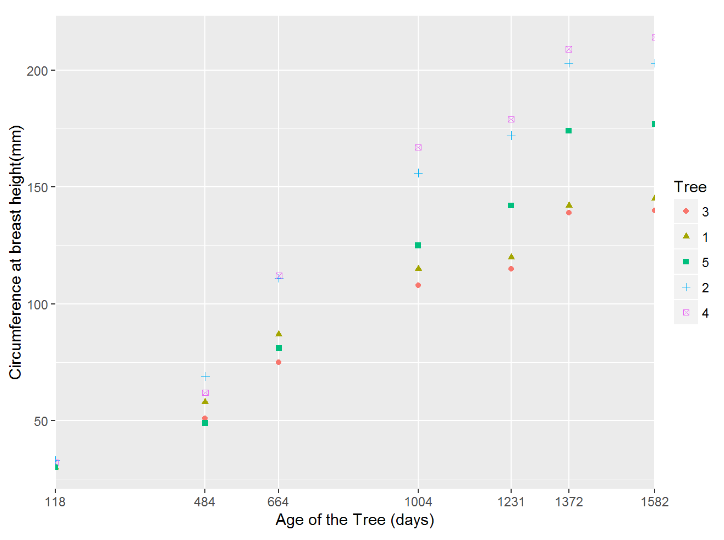
1. Make a scatter plot of the trunk circumferences against the age of the tree. Use different plotting symbols for different size of trees.

ggplot(Orange, aes(x=Orange$age, y=Orange$circumference)) +

geom\_point(aes(shape=Tree, color=Tree)) +

scale\_x\_discrete(name="Age of the Tree (days)",limits=c(118,484,664,1004,1231,1372,1582)) +

scale\_y\_continuous(name="Circumference at breast height(mm)")



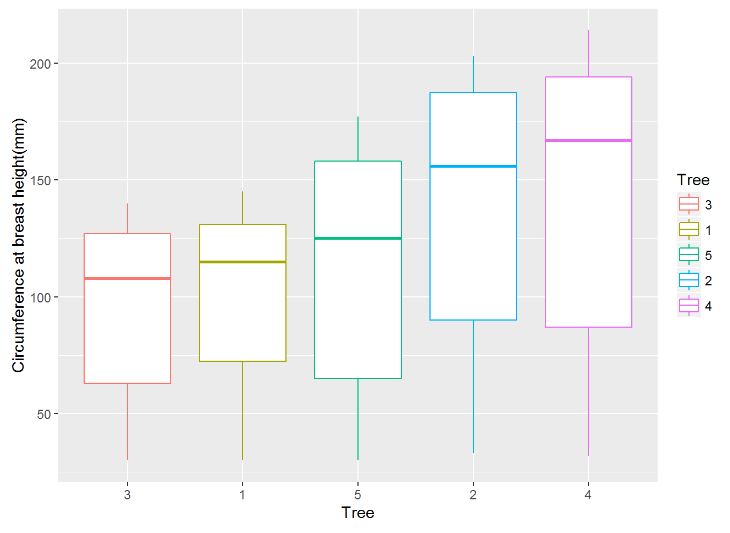
1. Display the trunk circumferences on a comparative boxplot against tree. Be sure you order the boxplots in the increasing order of maximum diameter.

ggplot(Orange, aes(x = Orange$Tree, y = Orange$circumference)) +

geom\_boxplot(aes(group=Tree, color=Tree)) +

scale\_x\_discrete(name="Tree") +

scale\_y\_continuous(name="Circumference at breast height(mm)")



**Question 04 (45 points)**

Download “Temp” data set (check your SMU email).

#Libraries

#install.packages("DataCombine")

library(DataCombine)

#Import Temp Data

x <- getURL("https://raw.githubusercontent.com/blawrenceDS/CaseStudy02/master/TEMP.csv")

raw\_temp\_data <- read.csv(text = x, header = TRUE)

#Import records with '/', which are post 1990 data.

Temp\_data <- raw\_temp\_data[grep("/",raw\_temp\_data$Date),]

#Assign Field names

names(Temp\_data) <- c("Date", "Monthly\_Avg\_Temp", "Uncertainty", "Country" )

#Convert Country and date to Character and date

Temp\_data$Country <- as.character(Temp\_data$Country)

Temp\_data$Date <- as.Date(Temp\_data$Date, "%m/%d/%Y")

#Remove entries with no temp readings.

Temp\_data <- Temp\_data[!is.na(Temp\_data$Monthly\_Avg\_Temp),]

1. Find the difference between the maximum and the minimum monthly average temperatures for each country and report/visualize top 20 countries with the maximum differences for the period **since 1900**.

## Part i

# Generating a dataset with max and min temp for each country over years.

min\_max\_data <- merge(

x = aggregate(Temp\_data$Monthly\_Avg\_Temp, by = list(Temp\_data$Country), max),

y = aggregate(Temp\_data$Monthly\_Avg\_Temp, by = list(Temp\_data$Country), min),

by = "Group.1")

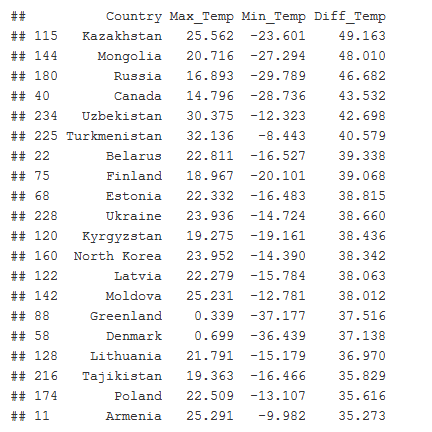
names(min\_max\_data) <- c("Country", "Max\_Temp", "Min\_Temp")

# Derive the max-min temp field.

min\_max\_data$Diff\_Temp <- min\_max\_data$Max\_Temp - min\_max\_data$Min\_Temp

# Report top 20 countries with max difference b/w max and min temp.

head(min\_max\_data[order(-min\_max\_data$Diff\_Temp),],20)



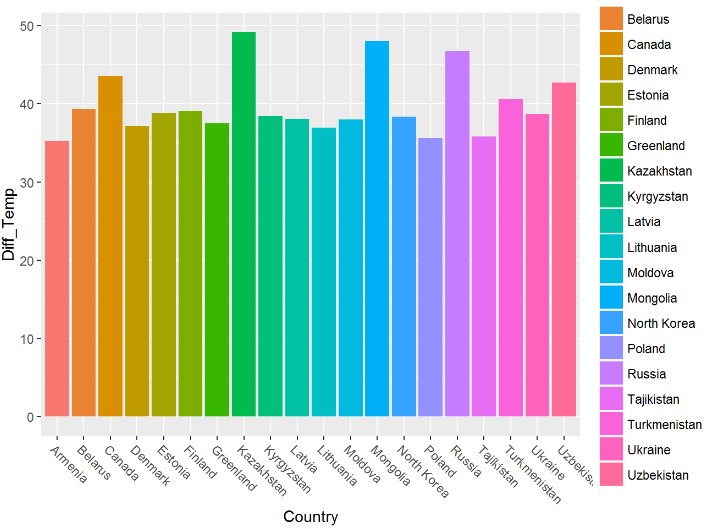
#Visualize top 20

top20 <- head(min\_max\_data[order(-min\_max\_data$Diff\_Temp),],20)

ggplot(data=top20, aes(x=Country, y=Diff\_Temp, fill=Country)) +

geom\_bar(stat="identity") +

theme(axis.text.x=element\_text(angle = -45, hjust = 0))



Select a subset of data called “UStemp” where US land temperatures from 01/01/1990 in Temp data. Use UStemp dataset to answer the followings.

## Part ii

#Subset only United States records

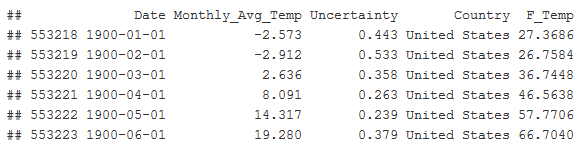
UStemp <- Temp\_data[Temp\_data$Country == "United States",]

1. Create a new column to display the monthly average land temperatures in Fahrenheit (**°F)**.

# a. Centigrade to Fahrenheit conversion

UStemp$F\_Temp <- (UStemp$Monthly\_Avg\_Temp\*(9/5)) + 32

head(UStemp)



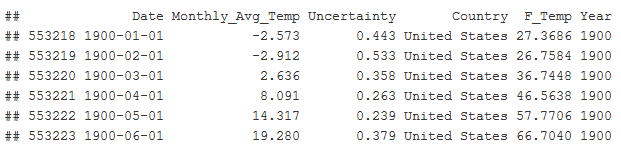
1. Calculate average land temperature by year and plot it. The original file has the average land temperature by month.

# b. Derive the Year

UStemp$Year <- strftime(UStemp$Date, "%Y")

UStemp$Year <- as.numeric(UStemp$Year)

head(UStemp)



# Aggregate the average temp per year

year\_avg\_temp <- ddply(UStemp, .(Year), summarize, Mean\_F\_Temp=mean(F\_Temp))

head (year\_avg\_temp)

# Plot the result

plot(year\_avg\_temp,

main = 'Average annual Temperatures in United States',

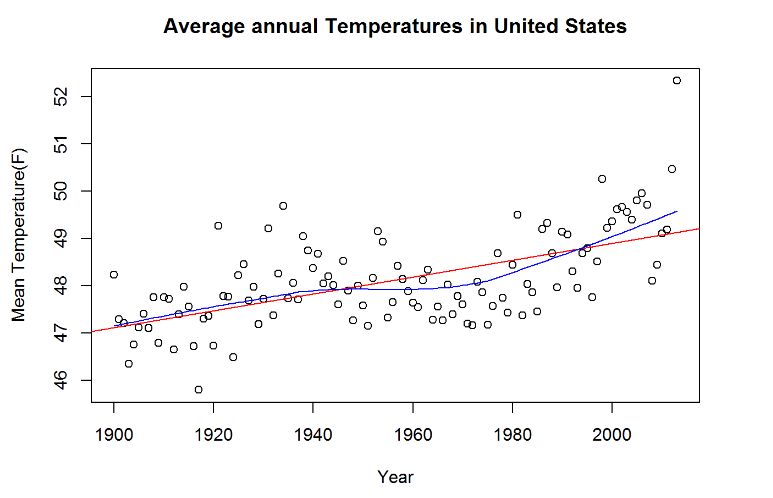
xlab = 'Year',

ylab = 'Mean Temperature(F)')

# Add fit lines

abline(lm(year\_avg\_temp$Mean\_F\_Temp ~ year\_avg\_temp$Year), col="red") # regression line (y~x)

lines(lowess(year\_avg\_temp$Year , year\_avg\_temp$Mean\_F\_Temp), col="blue") # lowest line (x,y)



1. Calculate the one year difference of average land temperature by year and provide the maximum difference (value) with corresponding two years.

(for example, year 2000: add all 12 monthly averages and divide by 12 to get average temperature in 2000. You can do the same thing for all the available years. Then you can calculate the one year difference as 1991-1990, 1992-1991, etc)

# c. Derive change in annual average temp from previous year.

# Sort by Year to new data frame

comp\_years <- year\_avg\_temp[order(year\_avg\_temp$Year),]

comp\_years <- slide(comp\_years, Var="Year" , slideBy=-1)

comp\_years <- slide(comp\_years, Var="Mean\_F\_Temp", slideBy=-1)

# Rename lag variables

comp\_years <- rename(comp\_years, c("Year-1"="Prev\_Year", "Mean\_F\_Temp-1"="Prev\_F\_Temp"))

# Calculate difference between current and previous year avg. temp

# Requesting the Max Difference so get absolute value of difference

comp\_years$F\_Temp\_diff <- abs(comp\_years$Mean\_F\_Temp - comp\_years$Prev\_F\_Temp)

# Year with max Change in temp. (Drop record with NA due to missing value for lag)

comp\_years <- comp\_years[!is.na(comp\_years$F\_Temp\_diff),]

comp\_years <- comp\_years[order(comp\_years$F\_Temp\_diff),]

tail(comp\_years,1)



1. Download “CityTemp” data set (check your SMU email). Find the difference between the maximum and the minimum temperatures for each major city and report/visualize top 20 cities with maximum differences for the period since 1900.

#Import CityTemp Data from GitHub

x2 <- getURL("https://raw.githubusercontent.com/blawrenceDS/CaseStudy02/master/CityTemp.csv")

raw\_city\_temp\_data <- read.csv(text = x2, header = TRUE)

#Import records with '/', which get us only post 1990 data.

city\_Temp\_data <- raw\_city\_temp\_data[grep("/",raw\_city\_temp\_data$Date),]

#Assign Field names

names(city\_Temp\_data) <- c("Date", "Monthly\_Avg\_Temp",

"Uncertainty","City","Country","Latitude","Longitude" )

# Data Conversions

city\_Temp\_data$Country <- as.character(city\_Temp\_data$Country)

city\_Temp\_data$City <- as.character(city\_Temp\_data$City)

city\_Temp\_data$Date <- as.Date(city\_Temp\_data$Date, "%m/%d/%Y")

head(city\_Temp\_data)

# Remove latitude and longitude fields since not needed

city\_Temp\_data <- city\_Temp\_data[,c("Date", "Monthly\_Avg\_Temp", "Uncertainty","City","Country")]

# Remove entries with no temp recordings

city\_Temp\_data <- city\_Temp\_data[!is.na(city\_Temp\_data$Monthly\_Avg\_Temp),]

# Convert to Fahrenheit to compare to USA dataset

city\_Temp\_data$Monthly\_Avg\_F\_Temp <- (city\_Temp\_data$Monthly\_Avg\_Temp\*(9/5)) + 32

# Generate a dataset with max and min temp for each City and country over years

city\_min\_max\_data <- merge(

x = aggregate(city\_Temp\_data$Monthly\_Avg\_Temp,

by = list(city\_Temp\_data$City,city\_Temp\_data$Country), max),

y = aggregate(city\_Temp\_data$Monthly\_Avg\_Temp,

by = list(city\_Temp\_data$City,city\_Temp\_data$Country), min),

by = c("Group.1","Group.2"))

names(city\_min\_max\_data) <- c("City", "Country", "Max\_Temp", "Min\_Temp")

head(city\_min\_max\_data)

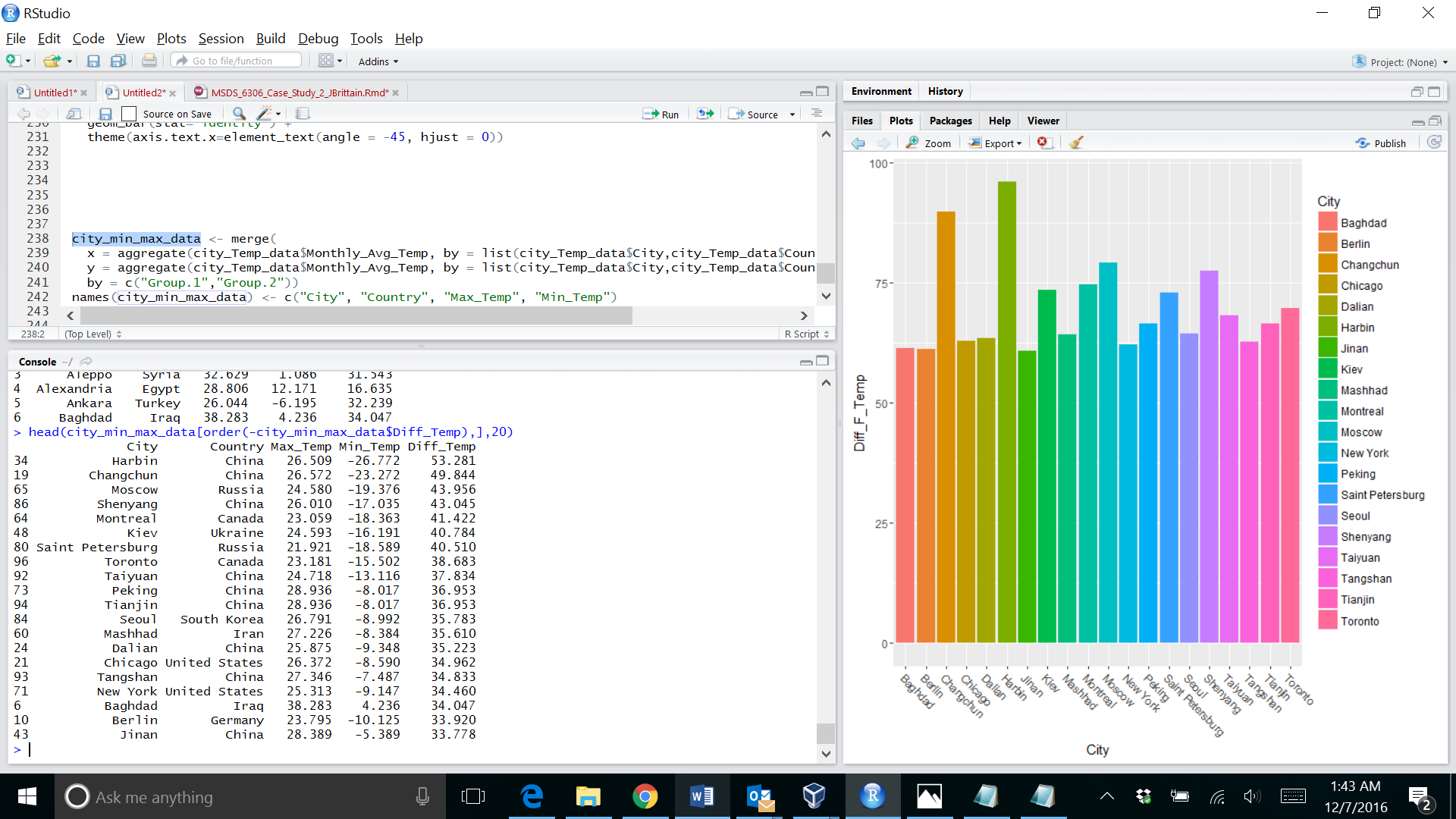
# Derive the max-min temp field for each city

# Requesting the Max Difference so get absolute value of difference

city\_min\_max\_data$Diff\_Temp <- city\_min\_max\_data$Max\_Temp - city\_min\_max\_data$Min\_Temp

# Top 20 Cities with highest difference in max and min temp.

head(city\_min\_max\_data[order(-city\_min\_max\_data$Diff\_Temp),],20)



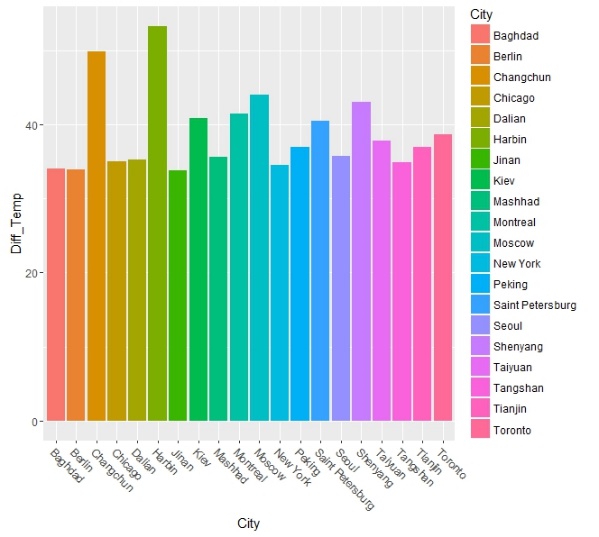
#Visualize top 20 Cities

top20cities <- head(city\_min\_max\_data[order(-city\_min\_max\_data$Diff\_Temp),],20)

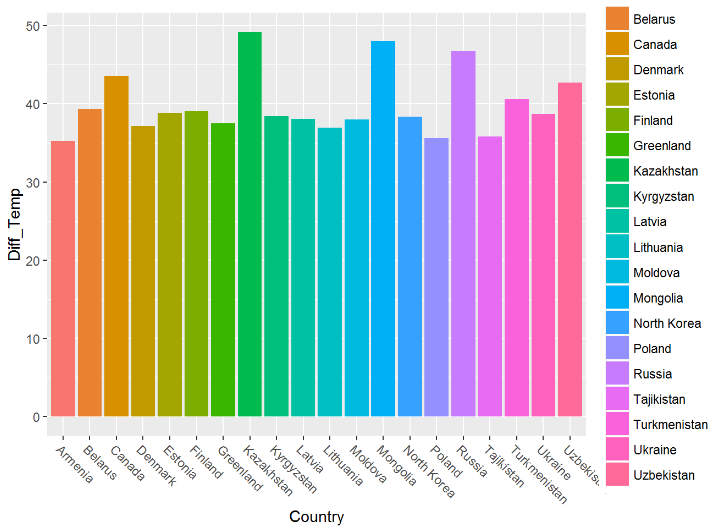
ggplot(data=top20cities, aes(x=City, y=Diff\_Temp, fill=City)) +

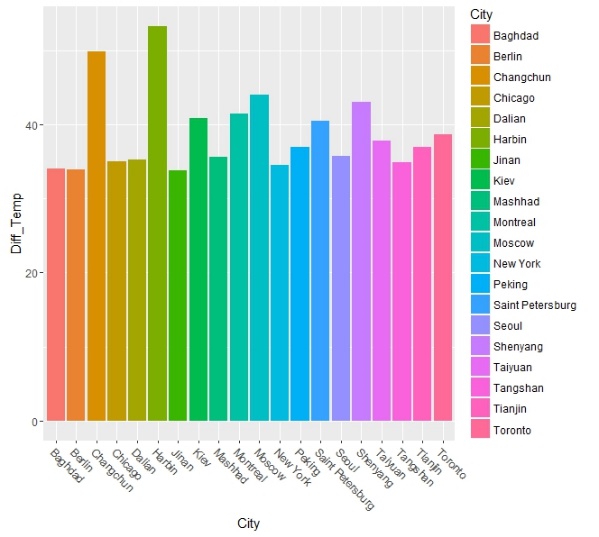
geom\_bar(stat="identity") +

theme(axis.text.x=element\_text(angle = -45, hjust = 0))



1. Compare the two graphs in (i) and (iii) and comment it.





The graphs differ greatly on the scale because the cities we compared intrayear monthly averages for cities and annual averages for countries. So, for the city data we see a huge difference in temperature for Harbin, China of 53.28°C but that represents the difference in highest month on record and lowest month (since 1900). For the Country data, we see Kazakhstan come in with a high difference of 49.2 °C. This represents a difference in the highest and lowest average temp for the year so this flattens out the temperature differences within a given year. Therefore, the comparison methodology for the cities data is more sensitive to outliers in the form of short term temperature anomalies than the country data methodology.

**Question 05 (10 points)**

Christmas Bonus

*A copy of this file, source code, and raw data is available at* [*https://github.com/blawrenceDS/CaseStudy02*](https://github.com/blawrenceDS/CaseStudy02)