Assignment 1

Sahil Shah (netid :sbs554, N12706992)

September 27, 2016

Readme and all the datasets are located in the root directory.

Q1. Data Exploration, Qualitative statistics and Missing Data

1-a) Qualitatively describe the difference(s) between states in Region 1 and Region 10. Using descriptive statistics and a figure if necessary.

Ans:

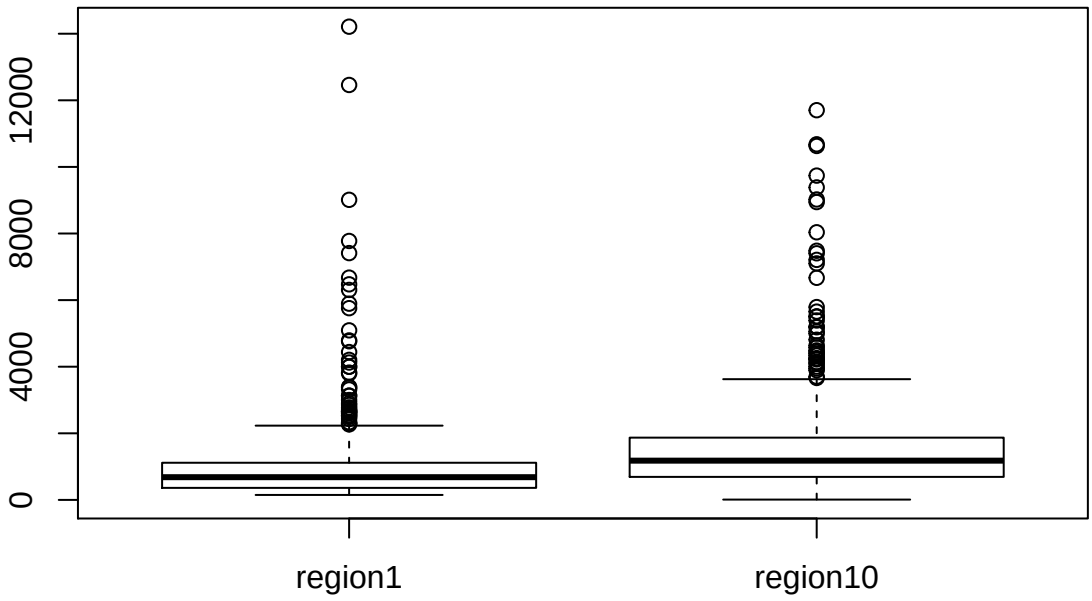
```
#load region1 and region 10 data into a dataframe
region1and10 <- data.frame(rawdata$Date, rawdata$HHS.Region.1..CT..ME..MA..NH..RI..VT.,
                           rawdata$HHS.Region.10..AK..ID..OR..WA.)
colnames(region1and10) <- c("date", "region1", "region10")
#convert the data to numeric
region1and10$region1 <- as.numeric(as.character(region1and10$region1))
region1and10$region10 <- as.numeric(as.character(region1and10$region10))
region1and10$date <-as.Date(as.character(rawdata$Date)) #convert to date format
#this is to treat the region1 and region10 as variables to plot there boxplot and compare
```

```
summary(region1and10)
```

```
##       date                region1           region10
##  Min.   :2003-09-28   Min.   :  149.0   Min.   :   10.0
##  1st Qu.:2006-09-15   1st Qu.:  361.5   1st Qu.:  691.8
##  Median :2009-09-02   Median :  682.5   Median : 1179.0
##  Mean   :2009-09-02   Mean   :  986.4   Mean   : 1589.2
##  3rd Qu.:2012-08-20   3rd Qu.: 1115.0   3rd Qu.: 1868.5
##  Max.   :2015-08-09   Max.   :14211.0   Max.   :11703.0
```

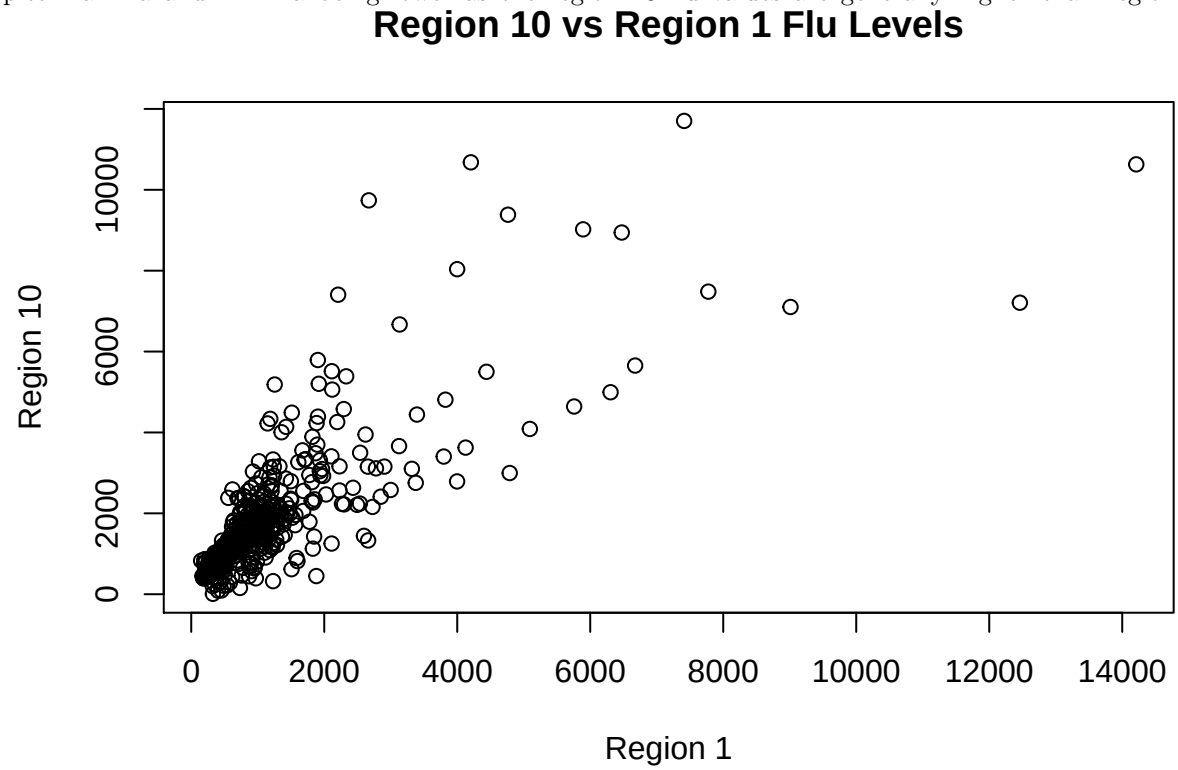
```
library(reshape2)
meltData <- melt(region1and10, id = c("date"))
meltData$value <- as.numeric(as.character(meltData$value))
```

As you can see from the summary we see that region1 has higher minimum and maximum but lower mean and median on the other hand region 10 has higher mean and median than region 1. This is showcased by the fol-

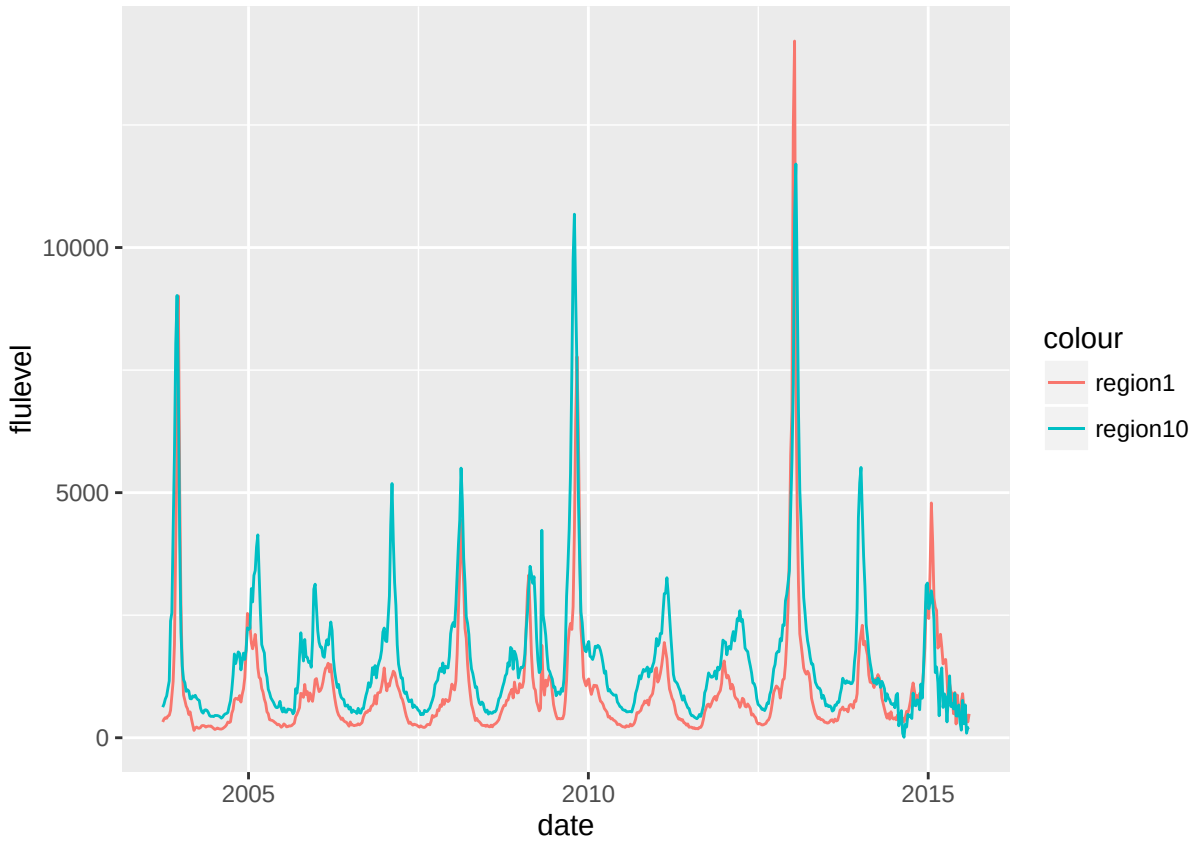


lowing boxplot.

Another interesting thing to observe will be when does this maxima occur with respect to the date and this can be verified by looking at the timeseries of both the regions and we find out that both the maxima occur pretty close to each other. Also the scatterplot and timeseries both confirm why mean is higher for region 10 despite maxima and minima being lower as the region 10 flu values are generally higher than region



1.



1-b) Compare query data from all cities in Arizona over all years using multiple descriptors. What metrics do you use? Reason your approach for dealing with missing data.

```
Arizonacities <- data.frame(rawdata$Date, rawdata$Mesa..AZ, rawdata$Phoenix..AZ,
                            rawdata$Scottsdale..AZ, rawdata$Tempe..AZ, rawdata$Tucson..AZ)
colnames(Arizonacities) <- c("Date", "Mesa..AZ", "Phoenix..AZ", "Scottsdale..AZ",
                             "Tempe..AZ" , "Tucson..AZ")
Arizonacities$Mesa..AZ <- as.numeric(as.character(Arizonacities$Mesa..AZ))
Arizonacities$Phoenix..AZ <- as.numeric(as.character(Arizonacities$Phoenix..AZ))
Arizonacities$Tempe..AZ <- as.numeric(as.character(Arizonacities$Tempe..AZ))
Arizonacities$Tucson..AZ<- as.numeric(as.character(Arizonacities$Tucson..AZ))
Arizonacities$Scottsdale..AZ <- as.numeric(as.character(Arizonacities$Scottsdale..AZ))
Arizonacities$Date = as.Date(as.character(Arizonacities$Date))
```

Let us briefly look at the summary data of Arizona cities. This will help us get an idea on various metrics like mean, max, median etc.

```
summary(Arizonacities)
```

```
##       Date               Mesa..AZ      Phoenix..AZ      Scottsdale..AZ
##  Min.   :2003-09-28   Min.   :  642   Min.   :  440.0   Min.   :  650
##  1st Qu.:2006-09-15   1st Qu.: 1152   1st Qu.:  913.2   1st Qu.: 1049
##  Median :2009-09-02   Median : 1834   Median : 1579.5   Median : 1809
```

```
##  Mean   :2009-09-02   Mean   : 2345   Mean   : 2102.9   Mean   : 2285
##  3rd Qu.:2012-08-20   3rd Qu.: 2780   3rd Qu.: 2374.5   3rd Qu.: 2679
##  Max.   :2015-08-09   Max.   :18968   Max.   :21643.0   Max.   :19566
##                       NA's   :58                        NA's   :56
##    Tempe..AZ         Tucson..AZ
##  Min.   :  503.0   Min.   :  564
##  1st Qu.:  971.5   1st Qu.: 1065
##  Median : 1706.5   Median : 1730
##  Mean   : 2158.2   Mean   : 2216
##  3rd Qu.: 2441.5   3rd Qu.: 2445
##  Max.   :16468.0   Max.   :17611
##
```

```
cor(na.omit(Arizonacities[,-1]))
```

```
##                 Mesa..AZ Phoenix..AZ Scottsdale..AZ Tempe..AZ Tucson..AZ
## Mesa..AZ       1.0000000   0.9176203      0.9582068 0.9440322  0.9310202
## Phoenix..AZ    0.9176203   1.0000000      0.9341658 0.9419139  0.9326898
## Scottsdale..AZ 0.9582068   0.9341658      1.0000000 0.9444598  0.9391714
## Tempe..AZ      0.9440322   0.9419139      0.9444598 1.0000000  0.9530885
## Tucson..AZ     0.9310202   0.9326898      0.9391714 0.9530885  1.0000000
```

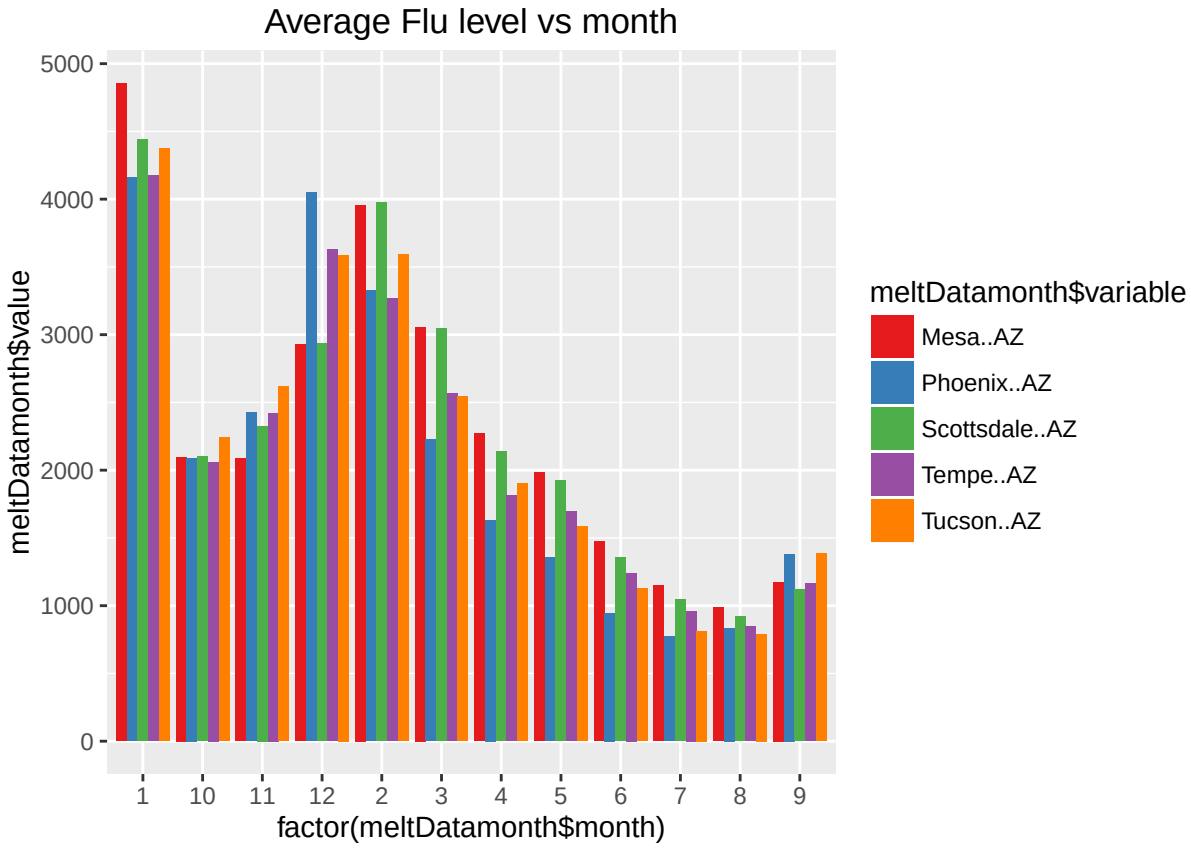
We find out that all this cities are highly comparable. Also we are also able to surmise that although Phoenix has the highest flu level it is lower on every count.

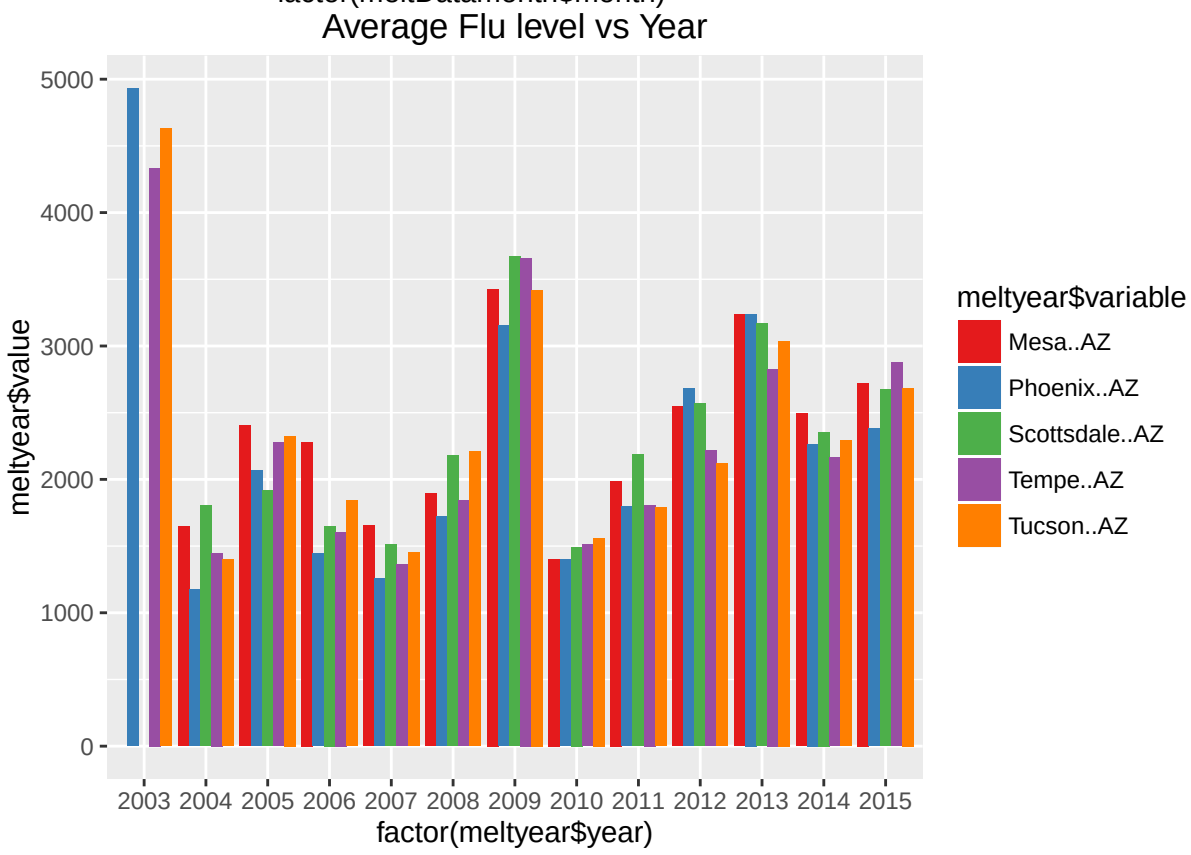
1st Metric : Mean

There are numerous ways to go about this. The basic is ofcourse to look at the means of all cities and compare which leads us to believe mean of each site lies in 2000-2500 range. We can also do monthly average of flu of every city.

```
library(lubridate)
month <- paste(month(Arizonacities$Date))
monthdf<- data.frame(aggregate(Arizonacities, list(month),mean, na.rm = T))
colnames(monthdf)[1]<- "month"
meltDatamonth <- melt(monthdf , id = c("Date", "month"))
year <- paste(year(Arizonacities$Date))
yeardf <- data.frame(aggregate(Arizonacities, list(year),mean, na.rm = T))
colnames(yeardf)[1]<- "year"
meltyear <- melt(yeardf, id = c("Date" , "year"))
```

The next graph shows the monthly and yearly average of Arizona cities for comparison. We are able to see the similarity within the cities. Also we are able to identify that colder months affect the cities equally (like Dec, Jan and Feb).

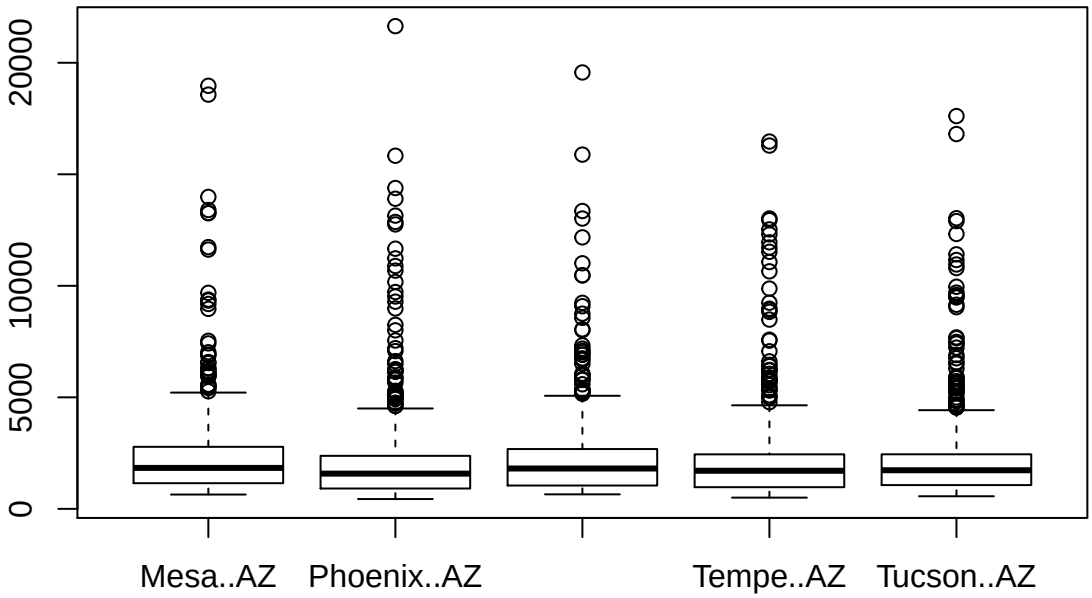

Average Flu level vs month
meltDatamonth$value
5000
4000
3000
2000
1000
0
1
10
11
12
2
3
4
5
6
7
8
9
factor(meltDatamonth$month)
meltDatamonth$variable
Mesa..AZ
Phoenix..AZ
Scottsdale..AZ
Tempe..AZ
Tucson..AZ


Average Flu level vs Year
meltyear$value
5000
4000
3000
2000
1000
0
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
factor(meltyear$year)
meltyear$variable
Mesa..AZ
Phoenix..AZ
Scottsdale..AZ
Tempe..AZ
Tucson..AZ

2nd Metric: Median, Max and Quartiles

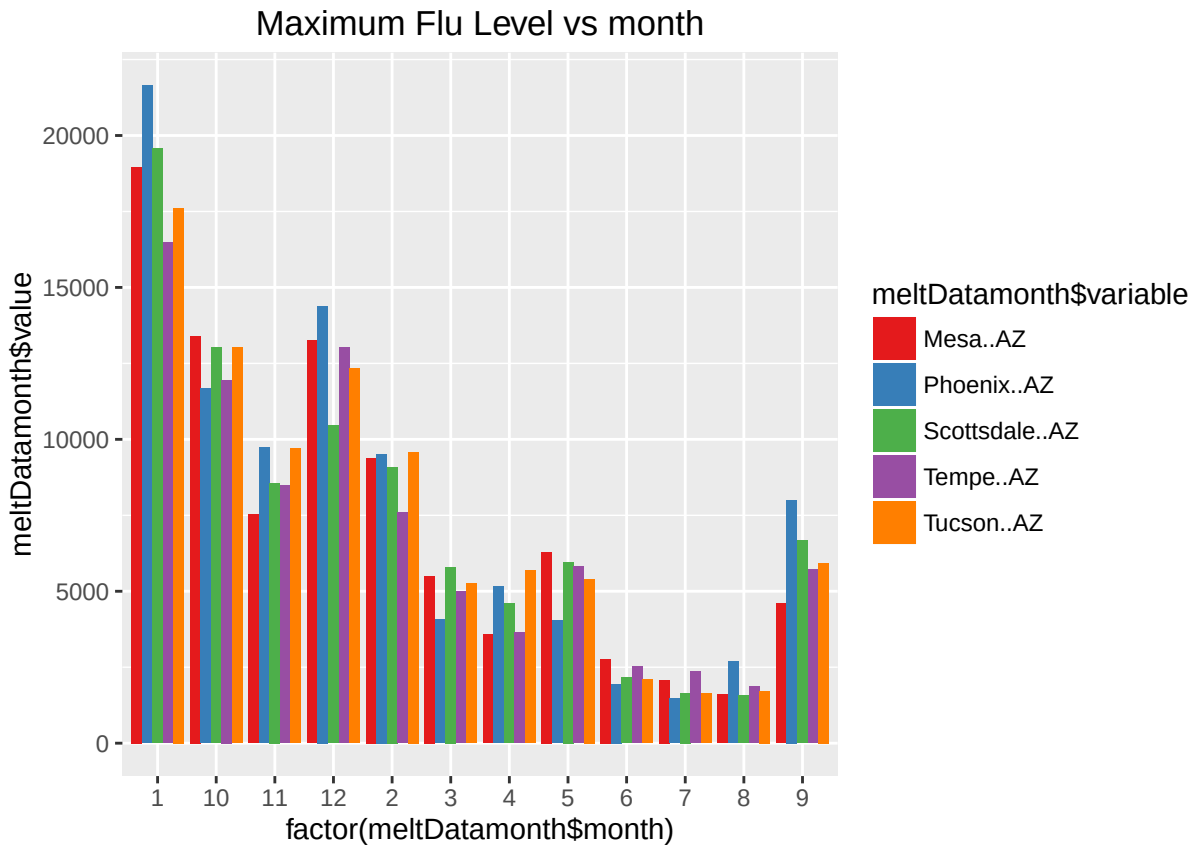
In order to see the effect of medians, max and quartiles, we can visually depict them on a box plot and compare their maximas and medians.

```
meltAz <- melt(Arizonacities , id = c("Date"))
boxplot(meltAz$value ~ meltAz$variable, data = meltAz)
```

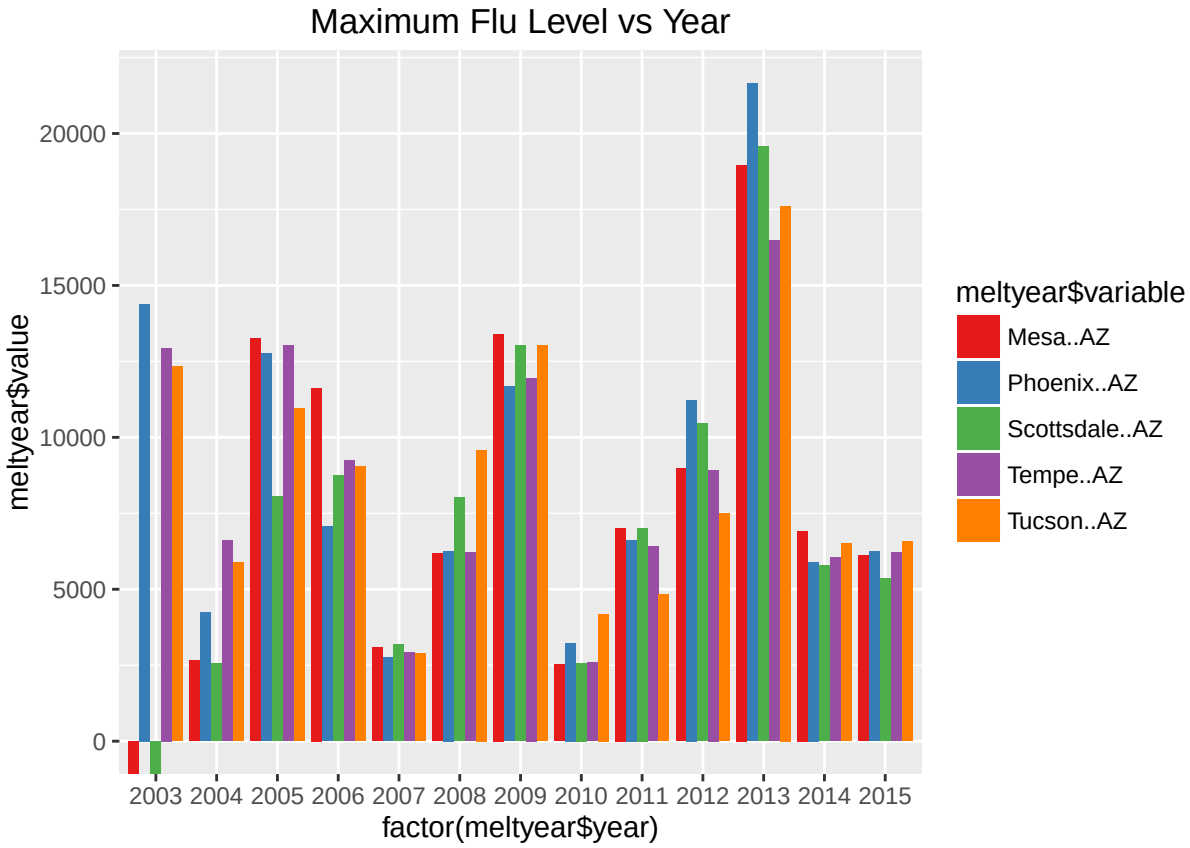


Next we compare each city with their max flu level for every month and every year. This will help us with identifying the coldest month and compare cities with eachother on the basis of severity.

```
library(lubridate)
month <- paste(month(Arizonacities$Date))
monthdf<- data.frame(aggregate(Arizonacities, list(month),max, na.rm = T))
colnames(monthdf)[1]<- "month"
meltDatamonth <- melt(monthdf , id = c("Date", "month"))
year <- paste(year(Arizonacities$Date))
yeardf <- data.frame(aggregate(Arizonacities, list(year),max, na.rm = T))
colnames(yeardf)[1]<- "year"
meltyear <- melt(yeardf, id = c("Date" , "year"))
ggplot(meltDatamonth, aes(factor(meltDatamonth$month), meltDatamonth$value,
                          fill = meltDatamonth$variable)) +
  geom_bar(stat="identity", position = "dodge") +
  scale_fill_brewer(palette = "Set1") +
  ggtitle('Maximum Flu Level vs month')
```




```
ggplot(meltyear, aes(factor(meltyear$year), meltyear$value, fill = meltyear$variable)) +
  geom_bar(stat="identity", position = "dodge") +
  scale_fill_brewer(palette = "Set1") +
  ggtitle('Maximum Flu Level vs Year')
```



Missing Values:

The missing values have simply been neglected. This was done by using *na.rm* option and setting it to *True*. The reasons to do are as follows:
1. The way to include them would have involved assuming it on the basis of previous weeks. But several months(almost 2 years) data is missing. So approximation wouldn't have done it justice. Also it just so happens 2003 year data is missing and if you look at 2003 it is a record year in terms of average flu level and thus any approximation would probably have been inaccurate.
2. The other reason, albeit an obvious one, is that the duration of dataset is high enough that we can afford the luxury to neglect few cells.

1-c) Find the population of the states (give a complete citation/credit for your source), and create a comparison of population vs. peak flu trend value for the most recent year. Is the relationship significant? Does it depend on if you consider the data as continuous or categorical (You will have to decide how to bin the data)?

Source of population Data: https://www.census.gov/popest/data/national/totals/2015/files/NST-EST2015-alldata.csv

The following steps are listed to plot the Population v Peak Flu graph

```
#only get 2015 data into the dataframe
GFTdata <- read.csv("data.txt", head = TRUE, sep = ",", skip=588)
#this step extracts the first row i.e the names of the columns
```

```
header <- read.csv("data.txt", nrows = 1, header = FALSE, sep =',',
                   stringsAsFactors = FALSE)
colnames(GFTdata) <- unlist(header) #this step attaches those names to GFTdata dataframe
GFTdata <- GFTdata[3:53] #only get the 50 states and ignore the rest columns
colMax <- function(data) sapply(data, max, na.rm = TRUE) #find the maximum of the columns
Peakflu <- data.frame(colMax(GFTdata)) #apply the maximum function to the GFTdata dataframe
colnames(Peakflu)[1] <- "PeakLevel"
Peakflu[,2] <- colnames(GFTdata)
colnames(Peakflu)[2] <- "names"
Populationdata<-read.csv("NST-EST2015-alldata.csv",head=TRUE,sep=",")
#this steps get the requisite fields into a new dataframe namely state name and the population
Populationstates<- data.frame(Populationdata[,5], Populationdata[,13])
colnames(Populationstates)[1] <- "name"
colnames(Populationstates)[2] <- "population2015"
#this is a step to merge two dataframe on a common field. this is essentially an inner join.[1]
mergeState<-data.frame(merge(Peakflu ,Populationstates, by.x=c("names"),by.y=c("name")))
```

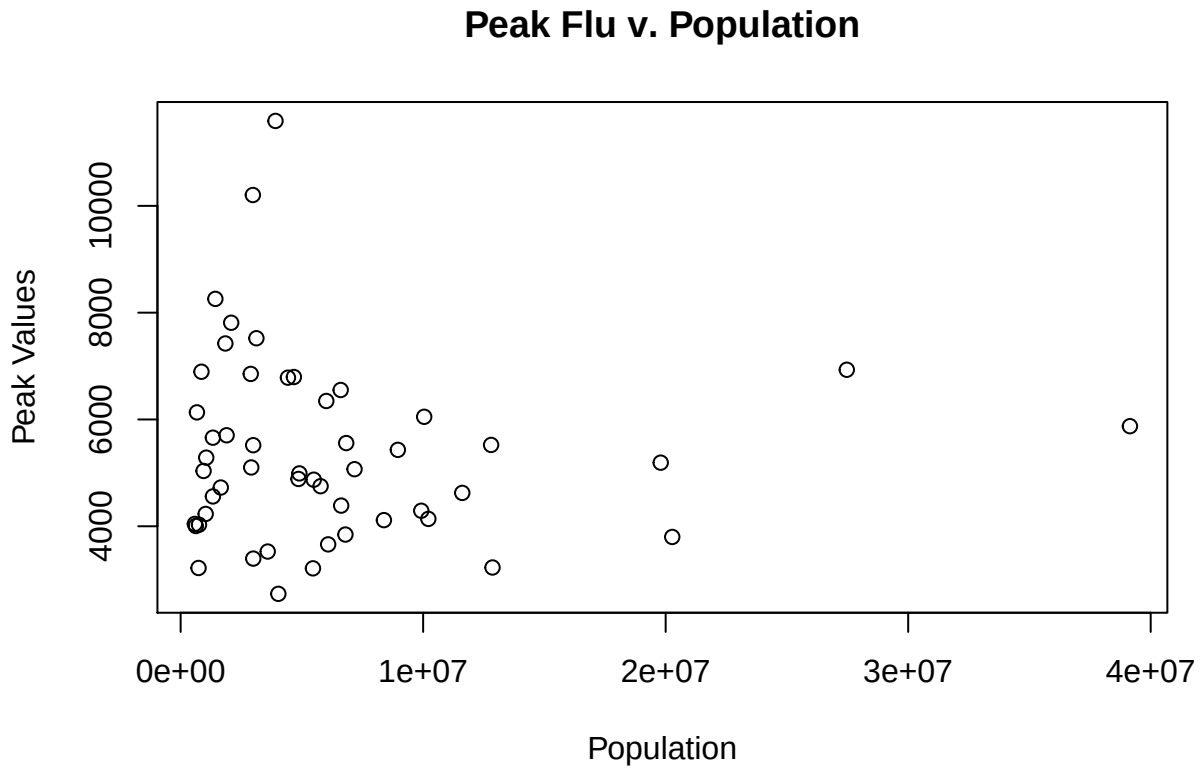
Continuous Data

```
kable(mergeState, format = "markdown")
```

names	PeakLevel	population2015
Alabama	4886	4858979
Alaska	3219	738432
Arizona	5558	6828065
Arkansas	10203	2978204
California	5874	39144818
Colorado	3212	5456574
Connecticut	3527	3590886
Delaware	5036	945934
District of Columbia	6133	672228
Florida	3800	20271272
Georgia	4138	10214860
Hawaii	8258	1431603
Idaho	4725	1654930
Illinois	3228	12859995
Indiana	4387	6619680
Iowa	7521	3123899
Kansas	5101	2911641
Kentucky	6783	4425092
Louisiana	6795	4670724
Maine	5659	1329328
Maryland	6346	6006401
Massachusetts	3845	6794422
Michigan	4291	9922576
Minnesota	4872	5489594
Mississippi	5518	2992333
Missouri	3661	6083672
Montana	4231	1032949
Nebraska	5704	1896190

names	PeakLevel	population2015
Nevada	6853	2890845
New Hampshire	4559	1330608
New Jersey	5431	8958013
New Mexico	7811	2085109
New York	5191	19795791
North Carolina	6050	10042802
North Dakota	4029	756927
Ohio	4625	11613423
Oklahoma	11590	3911338
Oregon	2735	4028977
Pennsylvania	5523	12802503
Rhode Island	5285	1056298
South Carolina	4990	4896146
South Dakota	6894	858469
Tennessee	6552	6600299
Texas	6931	27469114
Utah	3395	2995919
Vermont	4005	626042
Virginia	4115	8382993
Washington	5069	7170351
West Virginia	7423	1844128
Wisconsin	4750	5771337
Wyoming	4044	586107

```
plot(mergeState$population2015,mergeState$PeakLevel,ylab="Peak Values", xlab="Population",
     main="Peak Flu v. Population")
```



```
cor(mergeState[,-1])
```

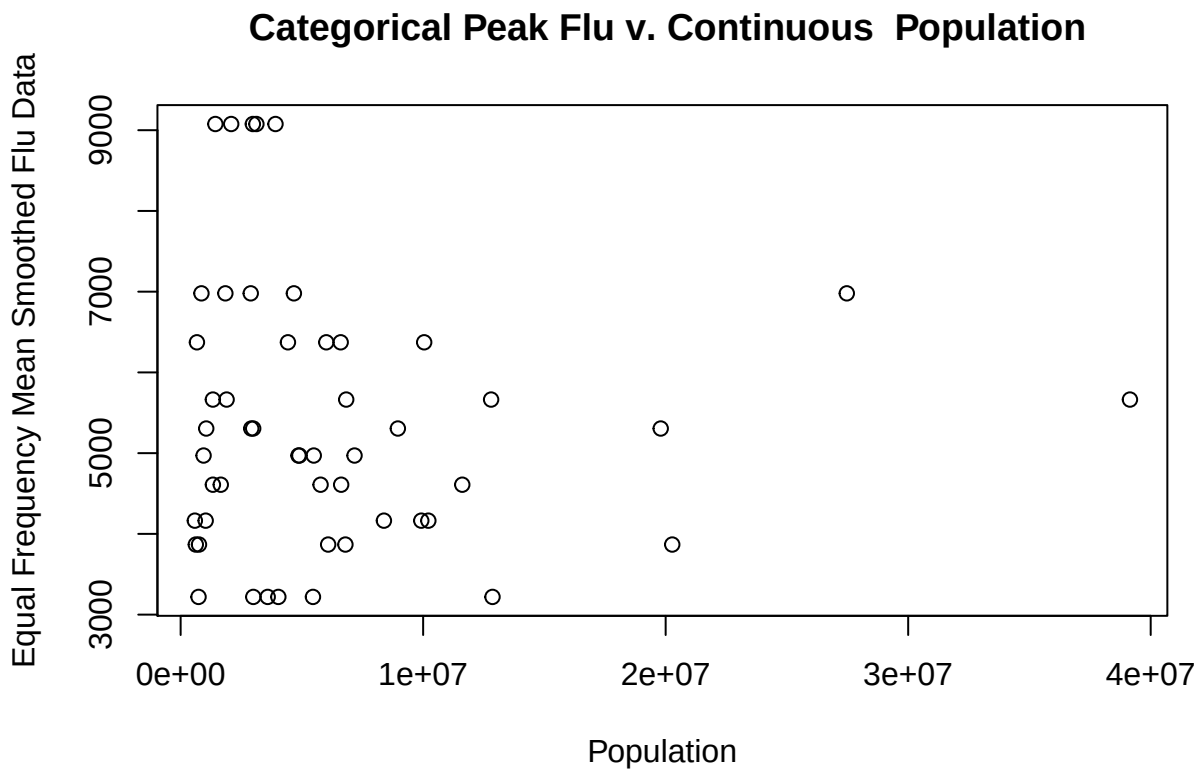
```
##                  PeakLevel population2015
## PeakLevel       1.00000000    -0.04202087
## population2015 -0.04202087     1.00000000
```

As we can see from the graph, treating the data as continuous, there is **no direct relation** between Population and Peak Flu. This is further verified by finding out the correlation coefficient which comes out to be very low (-0.0420)

Categorical Data

In order to treat the data as categorical we will need to perform binning on the data. Now let us perform Equal Frequency Binning on Flu data. We bin the data into 10 categories of equal frequency and then smooth individual categories with respect to mean.

```
require(ggplot2)
mergedstate1 <- mergeState
mergedstate1[,"FluBin"] <-(cut_number(mergedstate1$PeakLevel, n = 10)) #this creates the bins
#this performs smoothing based on means
df0 <- data.frame(aggregate(mergedstate1$PeakLevel, list(mergedstate1$FluBin),mean, na.rm = T))
mergedstate1$FluBin<- as.character(mergedstate1$FluBin)
df0$Group.1 <- as.character(df0$Group.1)
require(plyr)
mergedstate1$FluBin <- mapvalues(mergedstate1$FluBin, from=df0$Group.1, to=df0$x)
plot(mergedstate1$population2015, mergedstate1$FluBin,
     ylab = "Equal Frequency Mean Smoothed Flu Data", xlab = "Population",
     main = " Categorical Peak Flu v. Continuous  Population" )
```



```
mergedstate1$FluBin<-as.numeric(mergedstate1$FluBin)
cor((mergedstate1[,-1]))
```

```
##                  PeakLevel population2015      FluBin
## PeakLevel       1.00000000    -0.04202087  0.95350681
## population2015 -0.04202087     1.00000000 -0.05099818
## FluBin          0.95350681    -0.05099818  1.00000000
```

We see that there is no relation ship between Population and Peak flu even if the fludata is categorical which is further proved with correlation coefficient of -0.05099 between population2015 and Bin.

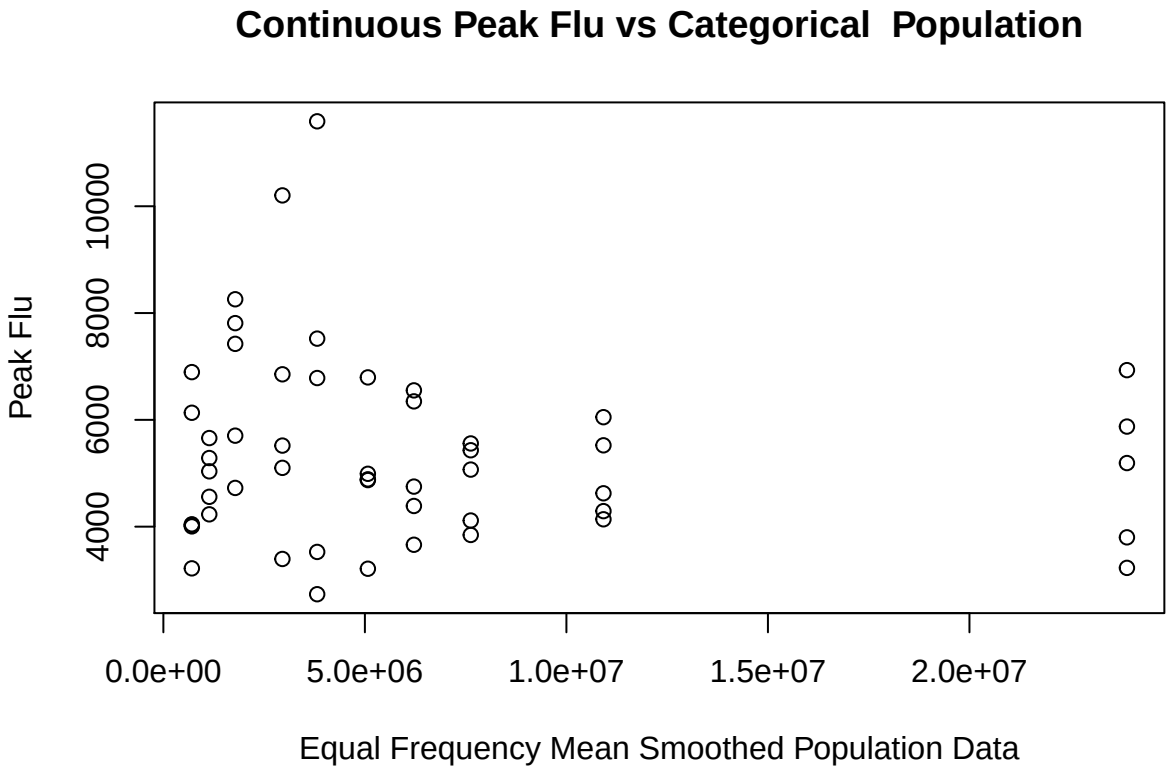
Lets see what happens if we also consider the Population to be categorical.

```
require(ggplot2)
#mergedstate1 <- mergeState
mergedstate1[,"PopBin"] <-(cut_number(mergedstate1$population2015, n = 10))
df2 <- data.frame(aggregate(mergedstate1$population2015, list(mergedstate1$PopBin),mean, na.rm = T))
mergedstate1$PopBin<- as.character(mergedstate1$PopBin)
df2$Group.1 <- as.character(df2$Group.1)
require(plyr)
mergedstate1$PopBin <- mapvalues(mergedstate1$PopBin, from=df2$Group.1, to=df2$x)
mergedstate1$PopBin <- as.numeric(mergedstate1$PopBin)
kable(mergedstate1, format = "markdown")
```

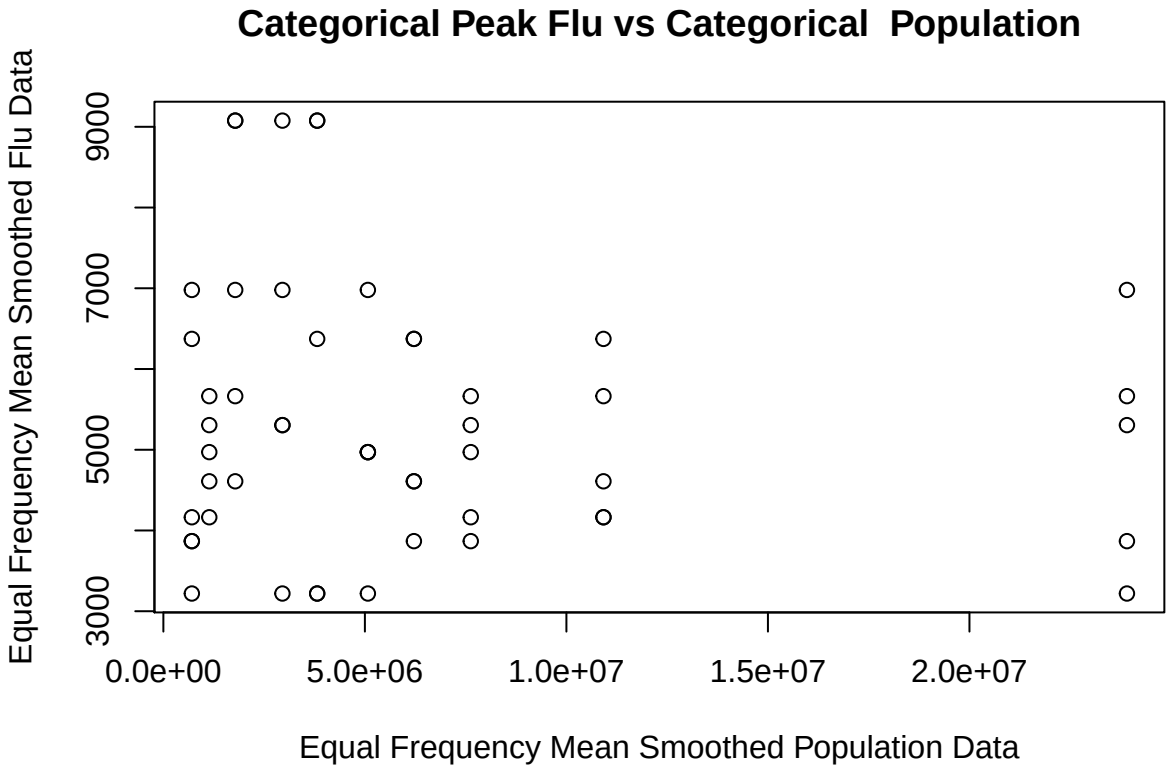
names	PeakLevel	population2015	FluBin	PopBin
Alabama	4886	4858979	4970.600	5074403.4
Alaska	3219	738432	3219.333	706367.5
Arizona	5558	6828065	5663.600	7626768.8
Arkansas	10203	2978204	9076.600	2953788.4
California	5874	39144818	5663.600	23908198.0
Colorado	3212	5456574	3219.333	5074403.4
Connecticut	3527	3590886	3219.333	3816038.4
Delaware	5036	945934	4970.600	1139023.4
District of Columbia	6133	672228	6372.800	706367.5
Florida	3800	20271272	3868.000	23908198.0
Georgia	4138	10214860	4163.800	10919232.8
Hawaii	8258	1431603	9076.600	1782392.0
Idaho	4725	1654930	4609.200	1782392.0
Illinois	3228	12859995	3219.333	23908198.0
Indiana	4387	6619680	4609.200	6216277.8
Iowa	7521	3123899	9076.600	3816038.4
Kansas	5101	2911641	5305.200	2953788.4
Kentucky	6783	4425092	6372.800	3816038.4
Louisiana	6795	4670724	6979.200	5074403.4
Maine	5659	1329328	5663.600	1139023.4
Maryland	6346	6006401	6372.800	6216277.8
Massachusetts	3845	6794422	3868.000	7626768.8
Michigan	4291	9922576	4163.800	10919232.8
Minnesota	4872	5489594	4970.600	5074403.4
Mississippi	5518	2992333	5305.200	2953788.4
Missouri	3661	6083672	3868.000	6216277.8
Montana	4231	1032949	4163.800	1139023.4

names	PeakLevel	population2015	FluBin	PopBin
Nebraska	5704	1896190	5663.600	1782392.0
Nevada	6853	2890845	6979.200	2953788.4
New Hampshire	4559	1330608	4609.200	1139023.4
New Jersey	5431	8958013	5305.200	7626768.8
New Mexico	7811	2085109	9076.600	1782392.0
New York	5191	19795791	5305.200	23908198.0
North Carolina	6050	10042802	6372.800	10919232.8
North Dakota	4029	756927	3868.000	706367.5
Ohio	4625	11613423	4609.200	10919232.8
Oklahoma	11590	3911338	9076.600	3816038.4
Oregon	2735	4028977	3219.333	3816038.4
Pennsylvania	5523	12802503	5663.600	10919232.8
Rhode Island	5285	1056298	5305.200	1139023.4
South Carolina	4990	4896146	4970.600	5074403.4
South Dakota	6894	858469	6979.200	706367.5
Tennessee	6552	6600299	6372.800	6216277.8
Texas	6931	27469114	6979.200	23908198.0
Utah	3395	2995919	3219.333	2953788.4
Vermont	4005	626042	3868.000	706367.5
Virginia	4115	8382993	4163.800	7626768.8
Washington	5069	7170351	4970.600	7626768.8
West Virginia	7423	1844128	6979.200	1782392.0
Wisconsin	4750	5771337	4609.200	6216277.8
Wyoming	4044	586107	4163.800	706367.5

```
plot( mergedstate1$PopBin, mergedstate1$PeakLevel, ylab = "Peak Flu",
     xlab = "Equal Frequency Mean Smoothed Population Data",
     main = "Continuous Peak Flu vs Categorical  Population")
```




```
plot(mergedstate1$PopBin, mergedstate1$FluBin, ylab = "Equal Frequency Mean Smoothed Flu Data",
     xlab = "Equal Frequency Mean Smoothed Population Data",
     main = "Categorical Peak Flu vs Categorical  Population")
```



First we need to understand treating the data as continuous is always a far better idea as it preserves

the nuance of the dataset. Also in this case it doesn't matter if it is categorical or continuous as there is intuitively no relation between Population and peak flu level. Nevertheless, it **does not** depend if we consider the data to be categorical Still there is no relation between Population and Peak Flu even if consider either or both categorical. This is reinforced by the correlation matrix of Final Dataframe which shows very low correlation coefficient. (Note: the high correlation coefficient are between binned data and their corresponding continuous data which is expected)

```
cor(mergedstate1[,-1])
```

```
##                  PeakLevel population2015      FluBin     PopBin
## PeakLevel       1.00000000    -0.04202087  0.95350681 -0.1224013
## population2015 -0.04202087     1.00000000 -0.05099818  0.9175922
## FluBin          0.95350681    -0.05099818  1.00000000 -0.1265753
## PopBin         -0.12240129     0.91759219 -0.12657528  1.0000000
```

1-d)For this question, download the flu data for all of the countries. Plot the center latitude for the country versus peak week of flu in the most recent year of data. Is there any relationship? In your response remember to credit your source for the latitude information.

Simple google search for latitude information of each country leads us to this result as the **source**: https://developers.google.com/public-data/docs/canonical/countries_csv

The problem is, however, that the table is embedded within the webpage. We will need to do some web-scraping for this. [2]

```
library("XML")
library("RCurl")
URL<-getURL("https://developers.google.com/public-data/docs/canonical/countries_csv")
htmltable <-data.frame(readHTMLTable(URL,header = TRUE,as.data.frame = TRUE,which=1))
countrylatitude <- htmltable[,c("latitude","name")]
#load world data from GFT and skip directly to the latest year i.e. 2015
worlddata <- read.csv("worlddata.txt", head=TRUE, sep = ",", skip=627)
header <- read.csv("worlddata.txt", nrows = 1, header = FALSE, sep =',', stringsAsFactors = FALSE)
colnames(worlddata) <- unlist(header)
colMax <- function(data) sapply(data, max, na.rm = TRUE)
Peakworld <- data.frame(colMax(worlddata[,2:30]))#apply maximum function on all columns except Date
colnames(Peakworld)[1] <- "Peaklevel"
Peakworld[,2] <- rownames(Peakworld)
colnames(Peakworld)[2] <- "names"
mergedcountries<-data.frame(merge(Peakworld , countrylatitude, by.x=c("names"),by.y=c("name"))) #[1]
kable(mergedcountries, format = "markdown")
```

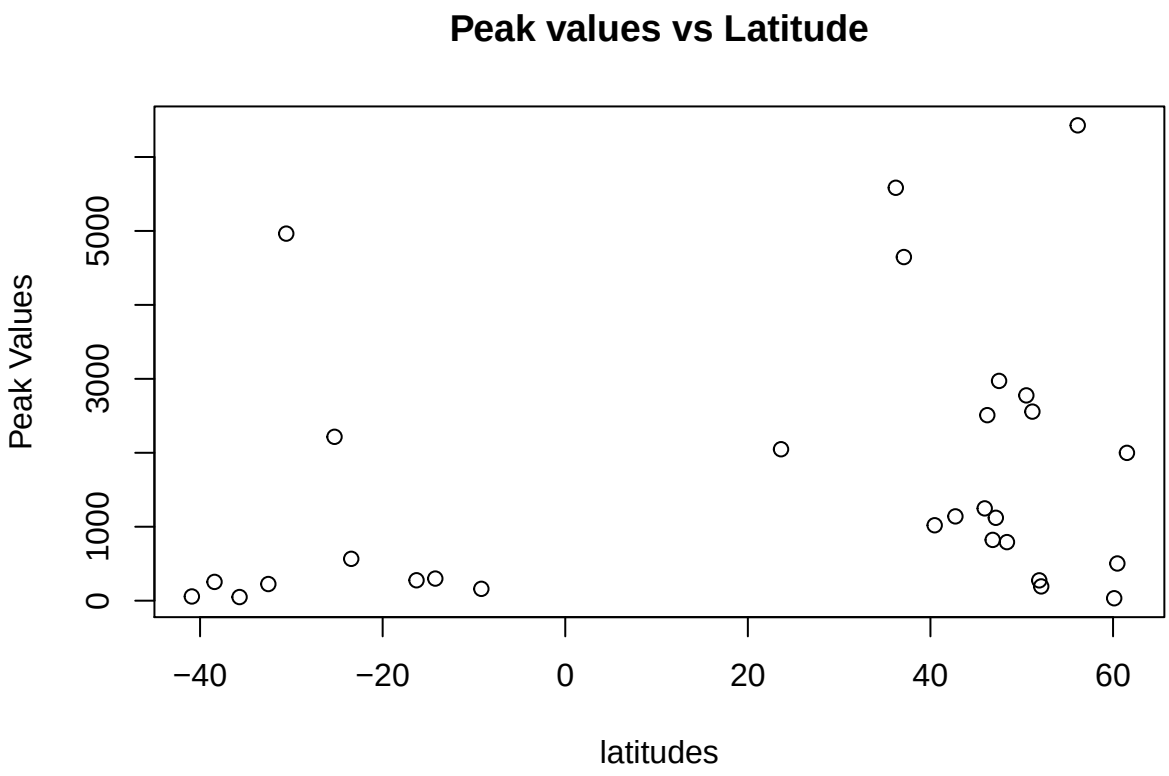
names	Peaklevel	latitude
Argentina	254	-38.416097
Australia	2216	-25.274398
Austria	2972	47.516231
Belgium	2776	50.503887
Bolivia	276	-16.290154
Brazil	299	-14.235004
Bulgaria	1140	42.733883

names	Peaklevel	latitude
Canada	6428	56.130366
Chile	49	-35.675147
France	2509	46.227638
Germany	2557	51.165691
Hungary	1121	47.162494
Japan	5584	36.204824
Mexico	2048	23.634501
Netherlands	193	52.132633
New Zealand	57	-40.900557
Norway	503	60.472024
Paraguay	566	-23.442503
Peru	160	-9.189967
Poland	275	51.919438
Romania	1248	45.943161
Russia	1999	61.52401
South Africa	4964	-30.559482
Spain	1020	40.463667
Sweden	33	60.128161
Switzerland	822	46.818188
Ukraine	792	48.379433
United States	4647	37.09024
Uruguay	225	-32.522779

```
mergedcountries$latitude = as.numeric(as.character(mergedcountries$latitude))
```

Peak Flu vs Latitude

```
plot(mergedcountries$latitude, mergedcountries$Peaklevel,ylab="Peak Values",
     xlab="latitudes", main="Peak values vs Latitude")
```




```
cor(mergedcountries[,-1])
```

```
##           Peaklevel  latitude
## Peaklevel 1.0000000 0.2478831
## latitude  0.2478831 1.0000000
```

Just looking at the graphs I think there is no relation between latitude and peak week values. As the values are pretty much scattered. For instance between latitudes -40 to -20 we find the values range from 0 to 5000. This intuition is confirmed by the correlation function which gives a pretty low coefficient of 0.247 and hence this confirms that there is little to no relation between latitudes and flu levels.

Peak Date of Flu vs Latitude

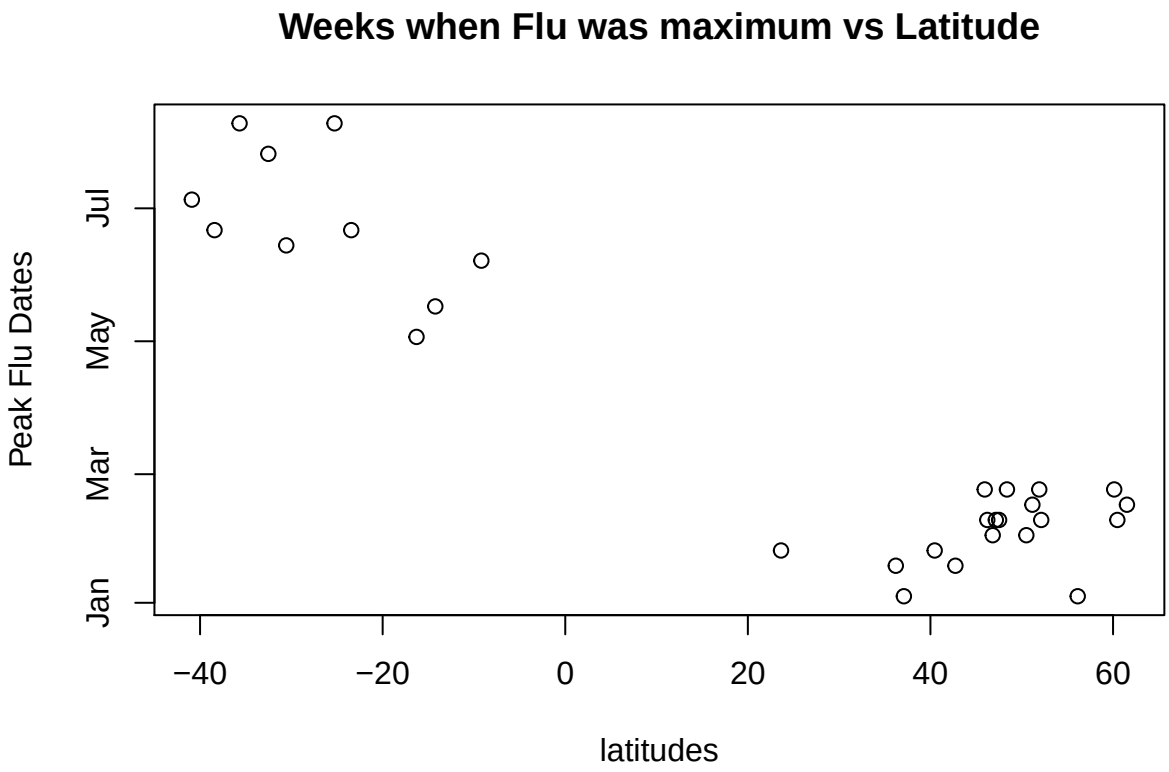
Alternatively, Lets see when these Peaks occur within a year and see if there is some relation between timing of the peaks and latitudes.

```
Peakweek1 <- data.frame(as.Date(worlddata$Date[apply(worlddata[,2:30],2,which.max)]))
colnames(Peakweek1)[1] <- 'Date'
Peakweek1[,"names"] = colnames(worlddata)[2:30]
mergedcountries1<-data.frame(merge(Peakweek1 , countrylatitude, by.x=c("names"),by.y=c("name"))) #[1]
mergedcountries1$latitude = as.numeric(as.character(mergedcountries1$latitude))
kable(mergedcountries1, format = "markdown")
```

names	Date	latitude
Argentina	2015-06-21	-38.416097
Australia	2015-08-09	-25.274398

names	Date	latitude
Austria	2015-02-08	47.516231
Belgium	2015-02-01	50.503887
Bolivia	2015-05-03	-16.290154
Brazil	2015-05-17	-14.235004
Bulgaria	2015-01-18	42.733883
Canada	2015-01-04	56.130366
Chile	2015-08-09	-35.675147
France	2015-02-08	46.227638
Germany	2015-02-15	51.165691
Hungary	2015-02-08	47.162494
Japan	2015-01-18	36.204824
Mexico	2015-01-25	23.634501
Netherlands	2015-02-08	52.132633
New Zealand	2015-07-05	-40.900557
Norway	2015-02-08	60.472024
Paraguay	2015-06-21	-23.442503
Peru	2015-06-07	-9.189967
Poland	2015-02-22	51.919438
Romania	2015-02-22	45.943161
Russia	2015-02-15	61.524010
South Africa	2015-06-14	-30.559482
Spain	2015-01-25	40.463667
Sweden	2015-02-22	60.128161
Switzerland	2015-02-01	46.818188
Ukraine	2015-02-22	48.379433
United States	2015-01-04	37.090240
Uruguay	2015-07-26	-32.522779

```
plot(mergedcountries1$latitude,mergedcountries1$Date,ylab="Peak Flu Dates",
     xlab="latitudes", main="Weeks when Flu was maximum vs Latitude")
```

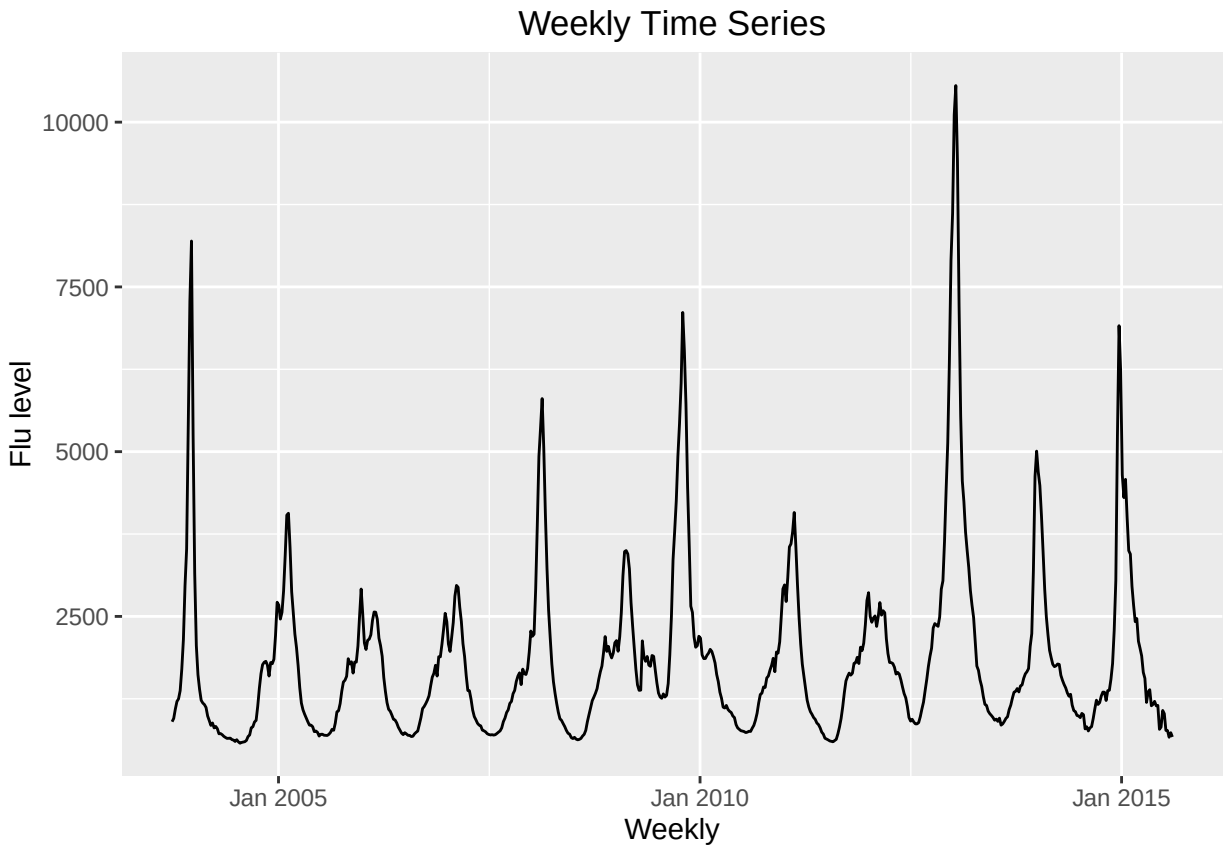


This is really interesting. We observe that for latitudes 20 to 60, Jan - March is the time of the year when peaks occur whereas for latitudes -40 to 0, May to December is the time when flu level is highest. Thus **there is indeed some relation** between Latitudes and Week of the year when peaks are observed.

Q2. Noise: Average the United States data to a monthly or lower frequency. How do the time series compare as you go to two lower frequencies (what metrics do you use to compare)?

First lets see the current time series with the weekly frequency. I am going to plot the United States column within the United states data frame(rawdata here)

```
library(ggplot2)
rawdata$Date <-as.Date(as.character(rawdata$Date))
ggplot(rawdata, aes(rawdata$Date, rawdata$United.States)) +
  geom_line() + scale_x_date(date_labels = "%b %Y") + xlab("Weekly") + ylab("Flu level") +
  ggtitle("Weekly Time Series")
```



This tells us although we are able to identify the peaks and low we still have a lot of noise we the value fluctuating between each maxima and minima. Next let us reduce the frequency to monthly. And see where it leads us in terms of time series.

```
monthly <- paste(paste(year(rawdata$Date),month(rawdata$Date), sep = "-"))
```

Now lets look at the metrics we can use to analyse. I see two metrics of use: **mean** and **max**. Mean will work when we need to get the general idea of the data of the country and will be more accurate representation of flu level as time progresses.

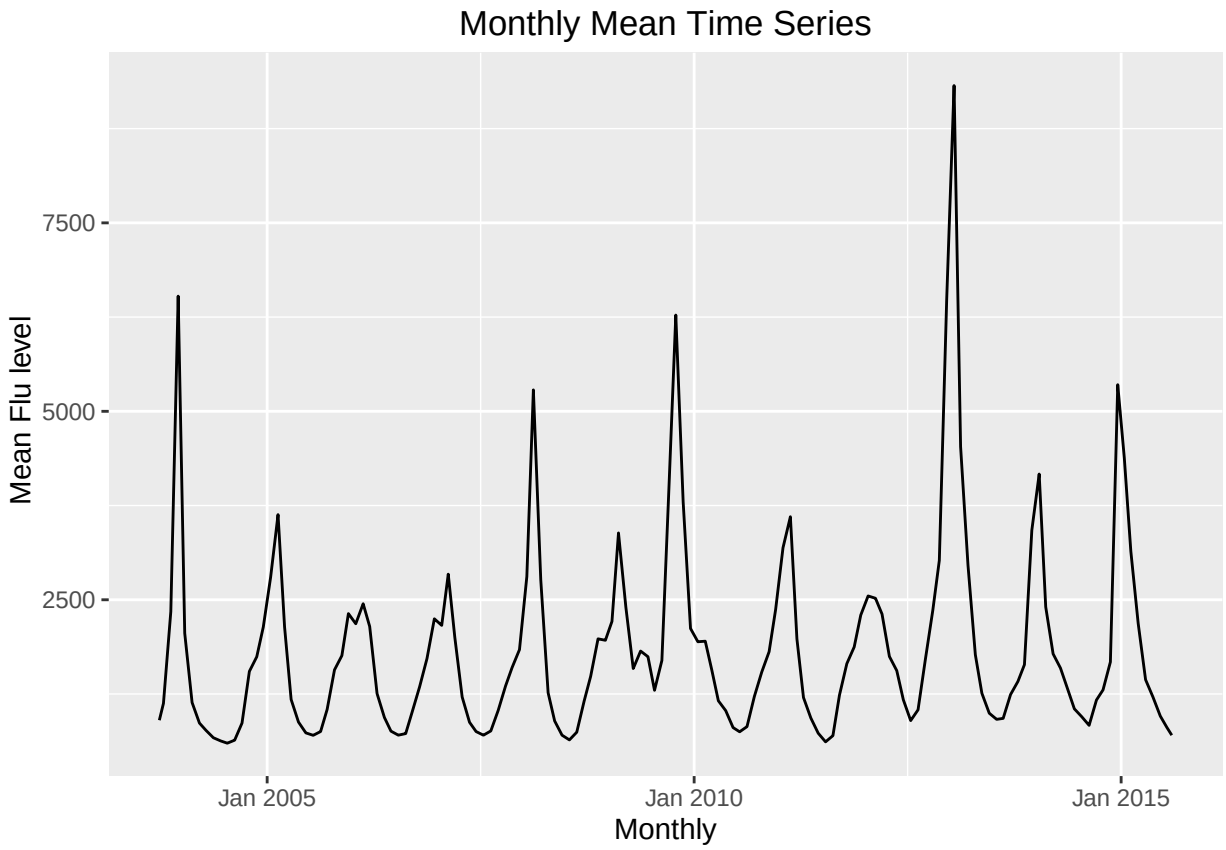
```
monthlydata <- data.frame(aggregate(rawdata, list(monthly),mean, na.rm = T))
```

The 2nd metric of interest is maximum: This is helpful as it helps us predict the severity of the flu. We can find out how severe can flu get and we can prepare for the worst in the future.

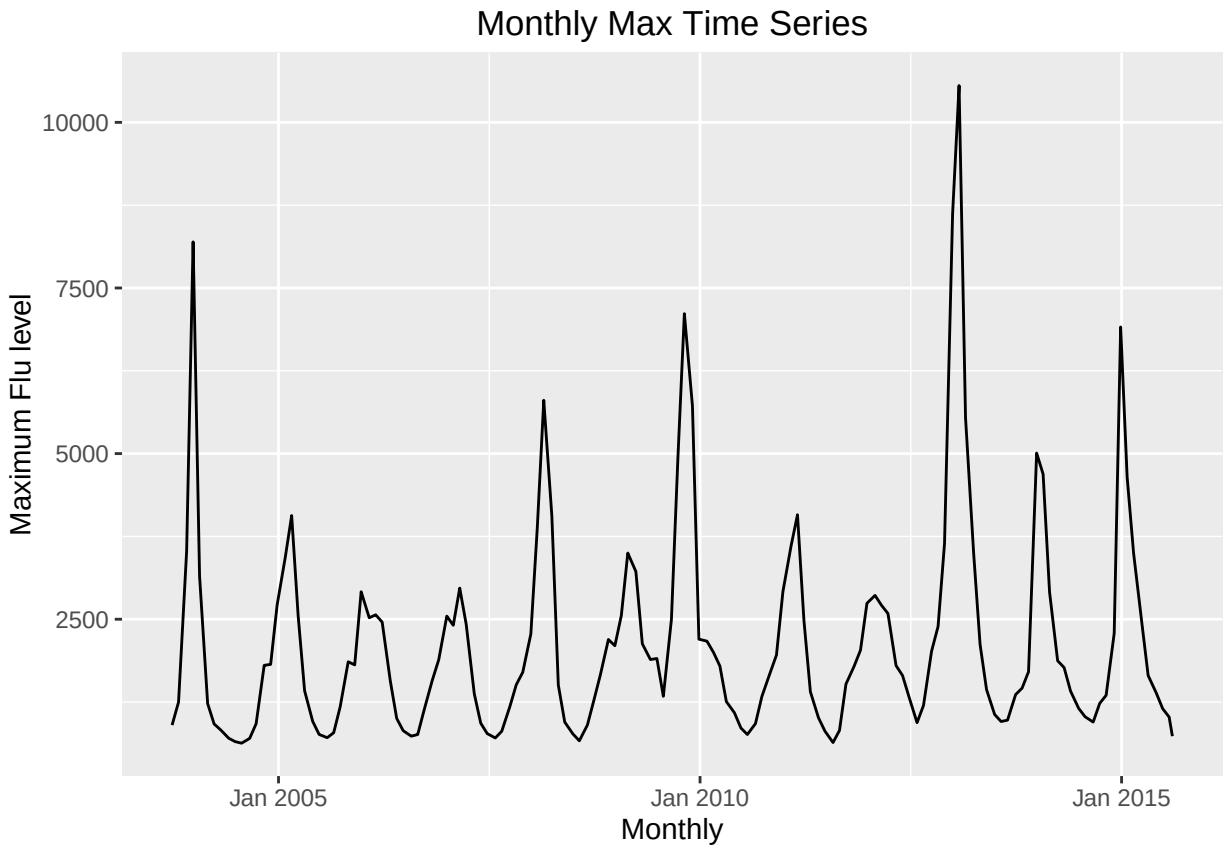
```
monthlymaxdata <- data.frame(aggregate(rawdata, list(monthly),FUN = max, na.rm = T))
```

Let us see both graphs and see how has the noise been affected with the weekly data…

```
ggplot(monthlydata, aes(monthlydata$Date, monthlydata$United.States)) +
  geom_line() + scale_x_date(date_labels = "%b %Y") + xlab("Monthly") + ylab(" Mean Flu level") +
  ggtitle("Monthly Mean Time Series")
```

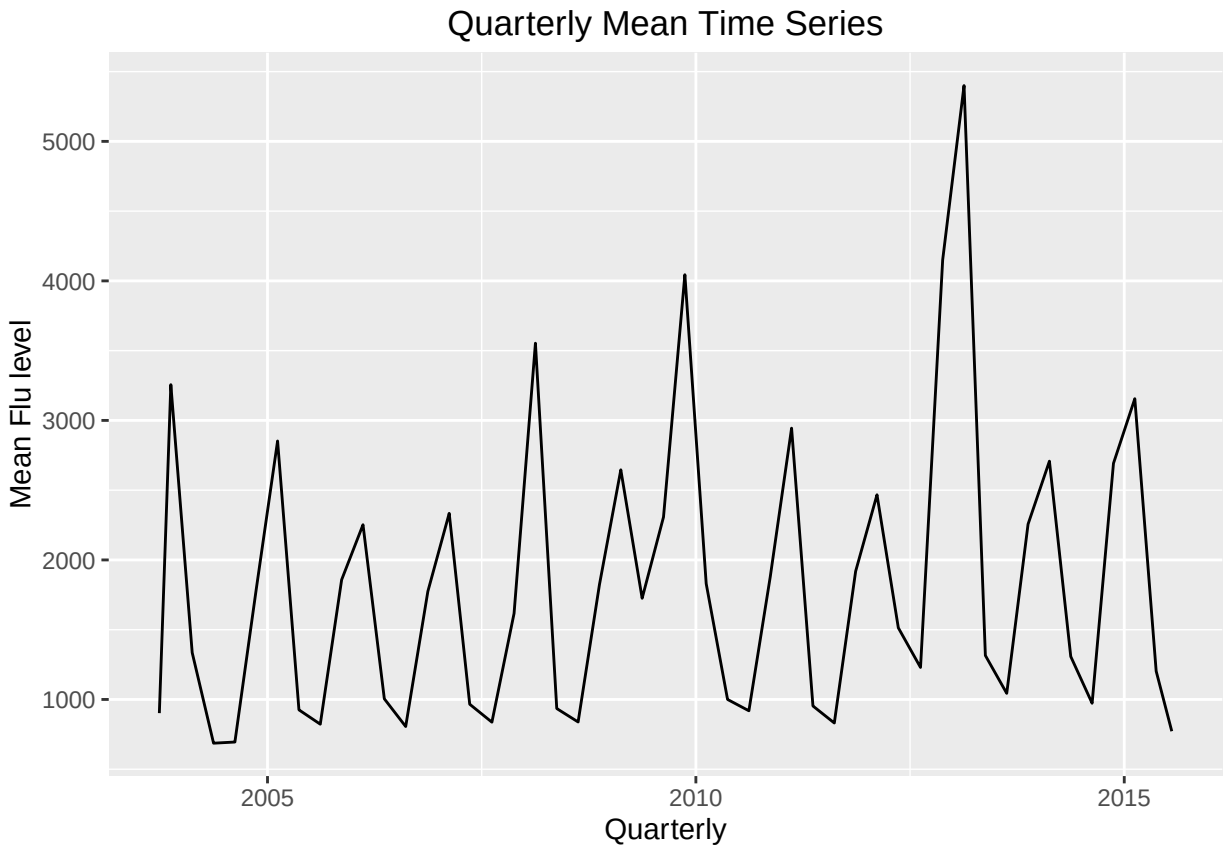



```
ggplot(monthlymaxdata, aes(monthlymaxdata$Date, monthlymaxdata$United.States)) +
  geom_line() + scale_x_date(date_labels = "%b %Y") + xlab("Monthly") + ylab("Maximum Flu level") +
  ggtitle("Monthly Max Time Series")
```

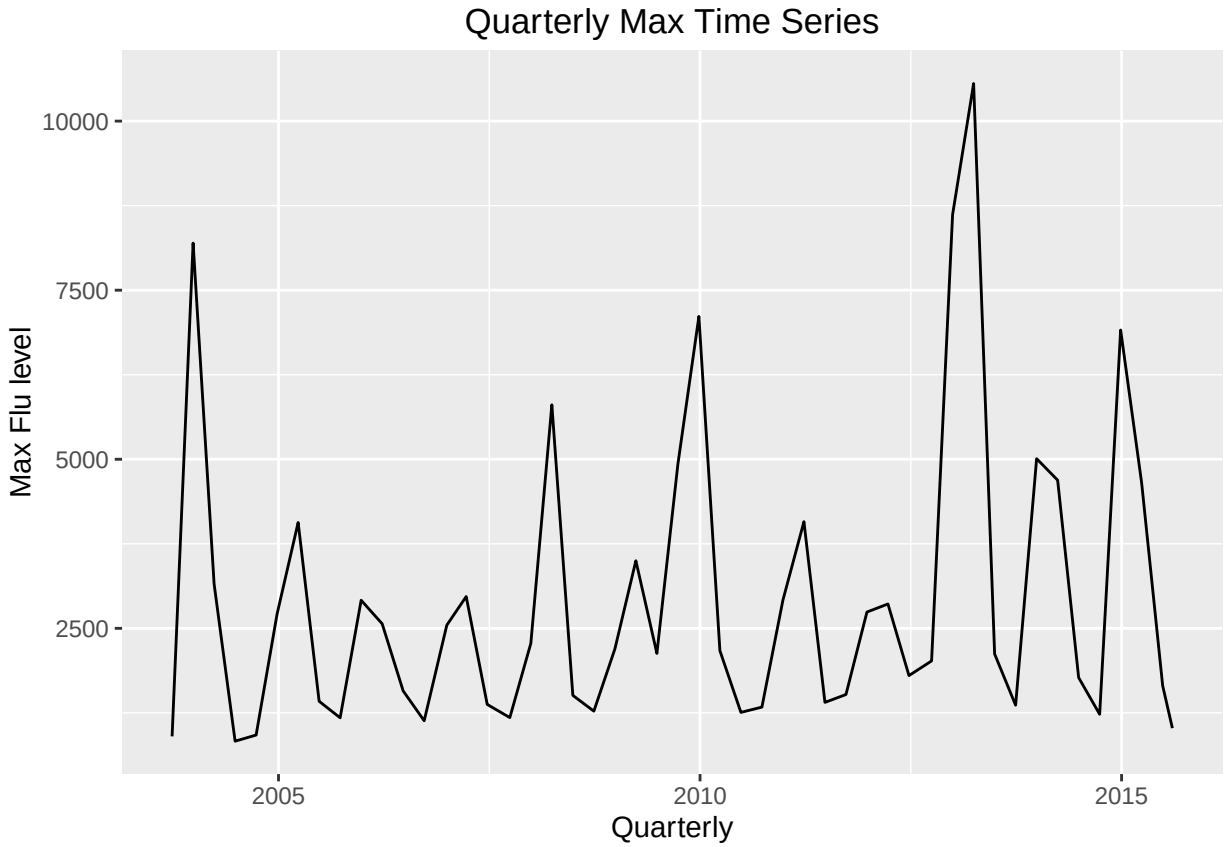


Let us further reduce the frequency to quarterly and repeat the above process again

```
quarterly <- paste(paste(year(rawdata$Date),quarter(rawdata$Date), sep = "-"))
quarterlydata <- data.frame(aggregate(rawdata, list(quarterly),mean, na.rm = T))
quarterlymaxdata <- data.frame(aggregate(rawdata, list(quarterly),FUN = max, na.rm = T))
ggplot(quarterlydata, aes(quarterlydata$Date, quarterlydata$United.States)) +
  geom_line() + scale_x_date(date_labels = "%Y") + xlab("Quarterly") + ylab("Mean Flu level")+
  ggtitle("Quarterly Mean Time Series")
```

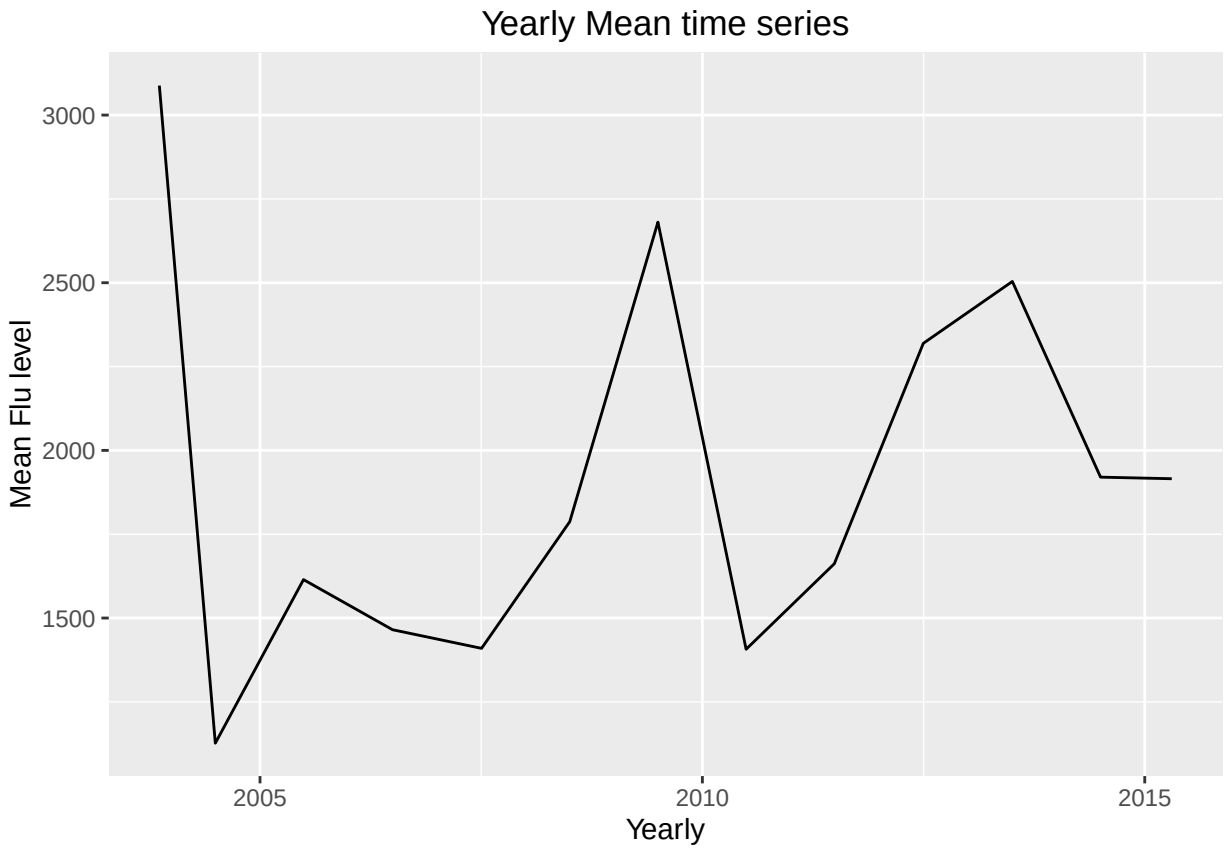



```
ggplot(quarterlymaxdata, aes(quarterlymaxdata$Date, quarterlymaxdata$United.States)) +
  geom_line() + scale_x_date(date_labels = "%Y") + xlab("Quarterly") + ylab("Max Flu level") +
  ggtitle("Quarterly Max Time Series")
```

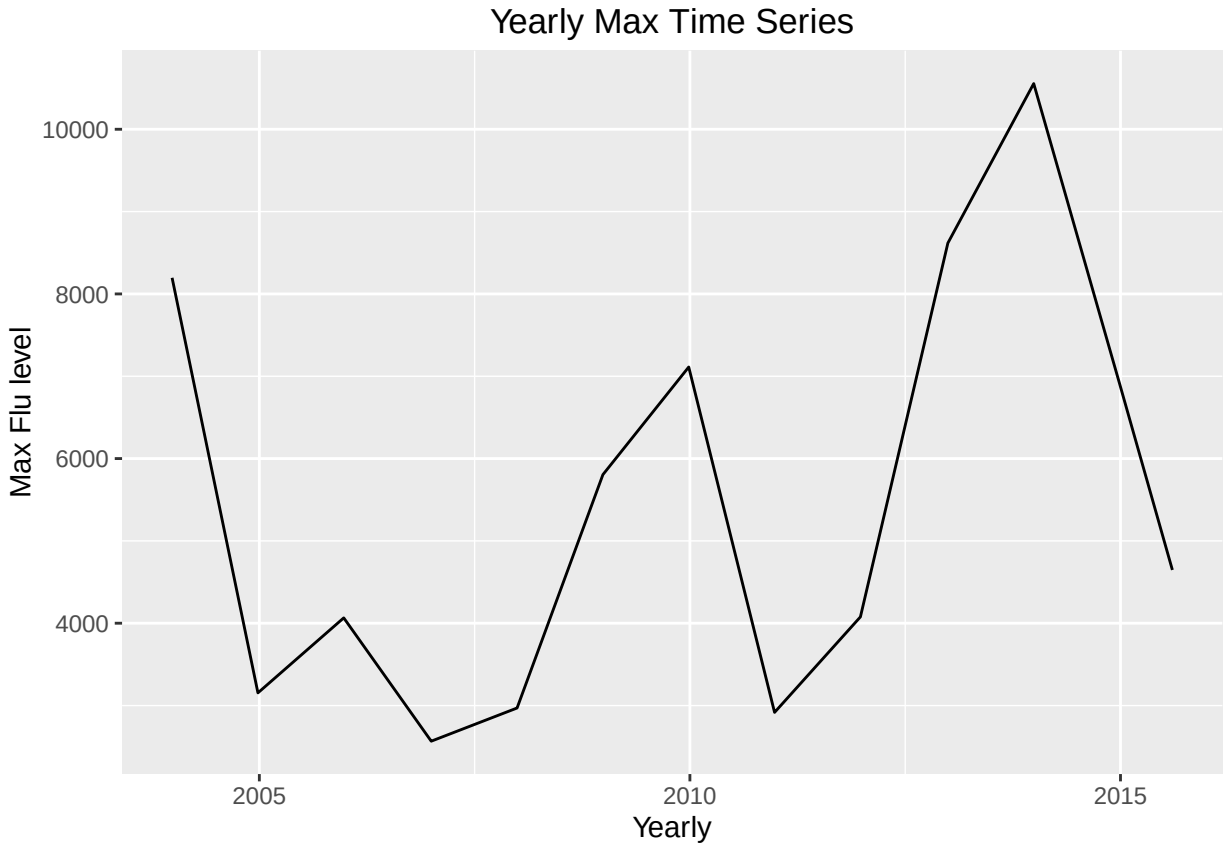


We clearly see the noise disappearing and the edges smoothening however with loss of accuracy. Now lets go to one step further and move to yearly...

```
yearly <- paste(year(rawdata$Date))
yearlydata <- data.frame(aggregate(rawdata, list(yearly),mean, na.rm = T))
yearlymaxdata  <- data.frame(aggregate(rawdata, list(yearly),FUN = max, na.rm = T))
ggplot(yearlydata, aes(yearlydata$Date, yearlydata$United.States)) +
  geom_line() + scale_x_date(date_labels = "%Y") + xlab("Yearly") + ylab("Mean Flu level") +
  ggtitle("Yearly Mean time series")
```




```
ggplot(yearlymaxdata, aes(yearlymaxdata$Date, yearlymaxdata$United.States)) +
  geom_line() + scale_x_date(date_labels = "%Y") + xlab("Yearly") + ylab("Max Flu level") +
  ggtitle("Yearly Max Time Series")
```



Here we see completely different nature of the graph and a lot of actual relevant information is lost. However this does help us to visualize how flu level has progressed yearly and let us get idea which year was the worst in terms of Flu.

Conclusion

As we reduce frequency the fluctuations between maximas and minimas disappear and the graphs smoothens. In other words the noise disappears. However as frequency gets too low (yearly) the entire shape of the graph changes.

Q3. Web Scraping

3-1) Read the Vaccine Status data from the table on the above website into an R data frame. There are many packages to use, I suggest you try the XML package which has useful functions such as htmlParse() to read in HTML documents and readHTML_Table().

```
newurl <- "http://www.cdc.gov/mmwr/preview/mmwrhtml/mm6401a4.htm?s_cid=mm6401a4_w"
newurltable <- readHTMLTable(newurl, header=T, which=1,stringsAsFactors=F)
newurltable <- newurltable[1:40,]  #remove extraneous rows
```

If we look at the table in the url: we see we are only interested in Vaccination Status. This will take some parsing but we have already loaded the table now we can use internal R functions to get the desired dataframe.

```
vaccine_status <- data.frame(newurltable$V1)
vaccine_status[,2] <-data.frame(newurltable$V7)
vaccine_status[,3]<-data.frame(newurltable$V8)
vaccine_status[,4]<-data.frame(newurltable$V9)
vaccine_status[,5]<-data.frame(newurltable$V10)
colnames(vaccine_status)<- c("Characteristics", "No.", "Total", "%", "p value")
vaccine_status <- vaccine_status[-(1:3),]
kable(vaccine_status, format = "markdown")
```

	Characteristics	No.	Total	%	p value
4	Overall	1,236	2,321	(53)	
5	Study site				<0.001
6	Michigan	258	488	(53)	
7	Pennsylvania	210	461	(46)	
8	Texas	252	507	(50)	
9	Washington	313	475	(66)	
10	Wisconsin	203	390	(52)	
11	Sex				0.01
12	Male	499	996	(50)	
13	Female	737	1325	(56)	
14	Age group (yrs)				<0.001
15	6 mos–8	302	638	(47)	
16	9–17	142	355	(40)	
17	18–49	307	668	(46)	
18	50–64	240	359	(67)	
19	65	245	301	(81)	
20	Race/Ethnicity§				<0.001
21	White	970	1734	(56)	
22	Black	61	179	(34)	
23	Other race	107	210	(51)	
24	Hispanic	95	189	(50)	
25	Self-rated health status¶				0.01
26	Fair or poor	66	113	(58)	
27	Good	281	496	(57)	
28	Very good	442	802	(55)	
29	Excellent	443	902	(49)	
30	Illness onset to enrollment (days)				0.15
31	<3	420	805	(52)	
32	3–4	458	883	(52)	
33	5–7	358	633	(57)	
34	Influenza test result				
35	Negative	771	1,371	(56)	
36	Influenza B positive**	17	35	(49)	
37	Influenza A positive**	448	916	(49)	
38	A (H1N1)pdm09	0	0	(0)	
39	A (H3N2)	407	842	(48)	
40	A subtype pending	41	74	(55)	

3-2) Find another example of a table somewhere on the web to load into R (Reminder, everyone must complete this assignment independently including finding a unique table to download). Provide the link to where the table is found along with your code.

```
newurl <- getURL("https://en.wikipedia.org/wiki/List_of_highest-grossing_films")
newurltable <- data.frame(readHTMLTable(newurl, header=T, which=1,stringsAsFactors=F))
newurltable <- newurltable[,-6]
kable(newurltable, format = "markdown")
```

Rank	Peak	Title	Worldwide.gross	Year
1	1	Avatar	$2,787,965,087	2009
2	1	Titanic	$2,186,772,302	1997
3	3	Star Wars: The Force Awakens	$2,068,223,624	2015
4	3	Jurassic World	$1,670,400,637	2015
5	3	The Avengers	$1,519,557,910	2012
6	4	Furious 7	$1,516,045,911	2015
7	5	Avengers: Age of Ultron	$1,405,413,868	2015
8	3	Harry Potter and the Deathly Hallows – Part 2	$1,341,511,219	2011
9F	5	Frozen	$1,287,000,000	2013
10	5	Iron Man 3	$1,215,439,994	2013
11	10	Minions	$1,159,398,397	2015
12	12	Captain America: Civil War	$1,152,765,346	2016
13	4	Transformers: Dark of the Moon	$1,123,794,079	2011
14	2	The Lord of the Rings: The Return of the King	$1,119,929,521	2003
15	7	Skyfall	$1,108,561,013	2012
16	10	Transformers: Age of Extinction	$1,104,054,072	2014
17	7	The Dark Knight Rises	$1,084,939,099	2012
18	4TS3	Toy Story 3	$1,066,969,703	2010
19	3	Pirates of the Caribbean: Dead Man's Chest	$1,066,179,725	2006
20	6	Pirates of the Caribbean: On Stranger Tides	$1,045,713,802	2011
21	1	Jurassic Park	$1,029,939,903	1993
22	2	Star Wars: Episode I – The Phantom Menace	$1,027,044,677	1999
23	5	Alice in Wonderland	$1,025,467,110	2010
24	24	Zootopia	$1,023,589,163	2016
25	14	The Hobbit: An Unexpected Journey	$1,021,103,568	2012
26	4	The Dark Knight	$1,004,558,444	2008
27	2	Harry Potter and the Philosopher's Stone	$974,755,371	2001
28	19DM2	Despicable Me 2	$970,761,885	2013
29	29	Finding Dory	$970,107,000	2016
30	2	The Lion King	$968,483,777	1994
31	30	The Jungle Book	$965,810,164	2016
32	5	Pirates of the Caribbean: At World's End	$963,420,425	2007
33	10	Harry Potter and the Deathly Hallows – Part 1	$960,283,305	2010
34	24	The Hobbit: The Desolation of Smaug	$958,366,855	2013
35	26	The Hobbit: The Battle of the Five Armies	$956,019,788	2014
36	8FN	Finding Nemo	$940,335,536	2003
37	6	Harry Potter and the Order of the Phoenix	$939,885,929	2007
38	8	Harry Potter and the Half-Blood Prince	$934,416,487	2009
39	4	The Lord of the Rings: The Two Towers	$926,047,111	2002
40	6	Shrek 2	$919,838,758	2004

Rank	Peak	Title	Worldwide.gross	Year
41	8	Harry Potter and the Goblet of Fire	$896,911,078	2005
42	10	Spider-Man 3	$890,871,626	2007
43	15	Ice Age: Dawn of the Dinosaurs	$886,686,817	2009
44	40	Spectre	$880,674,609	2015
45	6	Harry Potter and the Chamber of Secrets	$878,979,634	2002
46	29	Ice Age: Continental Drift	$877,244,782	2012
47	45	Batman v Superman: Dawn of Justice	$873,260,194	2016
48	5	The Lord of the Rings: The Fellowship of the Ring	$871,530,324	2001
49	34	The Hunger Games: Catching Fire	$865,011,746	2013
50	43	Inside Out	$857,611,174	2015

Q4. Go Viral Study

I have already signed up for the study and will be reporting symptoms in the coming weeks.

References

Parts of homeworks where briefly discussed with Vivek Ghatala. Only approach was discussed, the implementation was on my own.

1. http://stackoverflow.com/questions/1299871/how-to-join-merge-data-frames-inner-outer-left-right
2. http://stackoverflow.com/questions/1395528/scraping-html-tables-into-r-data-frames-using-the-xml-package
3. https://www.r-bloggers.com/from-continuous-to-categorical/
4. http://stackoverflow.com/questions/17721126/simplest-way-to-do-grouped-barplot
5. https://www.r-bloggers.com/scraping-table-from-any-web-page-with-r-or-cloudstat/
6. http://www.columbia.edu/~cjd11/charles_dimaggio/DIRE/styled-4/styled-6/code-13/
7. http://stackoverflow.com/questions/11722568/roughly-equal-binning-of-frequencies
8. http://stackoverflow.com/questions/22579390/get-values-in-one-column-that-correspond-with-max-value-of-other-col