Practicum 1

Maitri Shah

```
library(XML)
library(RCurl)
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.6     v purrr   0.3.4
## v tibble  3.1.1     v dplyr   1.0.8
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x tidyr::complete() masks RCurl::complete()
## x dplyr::filter()   masks stats::filter()
## x dplyr::lag()      masks stats::lag()
```

```
library(ggplot2)
library(dplyr)
```

1.Load the data, directly from the URL, into your R environment.

```
NYdata <- read.csv("https://data.ny.gov/api/views/ngbt-9rwf/rows.csv", header = TRUE)
```

2.Evaluate the dataset to determine what data preparation steps are needed and perform them. At a minimum, ensure that you discuss the distribution of the data, outliers and prepare any helpful summary statistics to support your analysis.

```
# Dimensions
dim(NYdata) #The dataset has 7 Columns and 86374 rows
```

```
## [1] 99367     7
```

```
str(NYdata) # There are major categorical data in the set. All columns (County.of.Program.Location, Pro
```

```
## 'data.frame':    99367 obs. of  7 variables:
##  $ Year                       : int  2007 2007 2007 2007 2007 2007 2007 2007 2007 2007 ...
##  $ County.of.Program.Location: chr  "Albany" "Albany" "Albany" "Albany" ...
##  $ Program.Category           : chr  "Crisis" "Crisis" "Crisis" "Crisis" ...
##  $ Service.Type               : chr  "Medical Managed Detoxification" "Medical Managed Detoxification"
##  $ Age.Group                  : chr  "Under 18" "18 through 24" "18 through 24" "18 through 24" ...
##  $ Primary.Substance.Group    : chr  "Heroin" "All Others" "Other Opioids" "Heroin" ...
##  $ Admissions                 : int  4 2 6 132 35 8 1 11 276 135 ...
```

```
summary(NYdata)
```

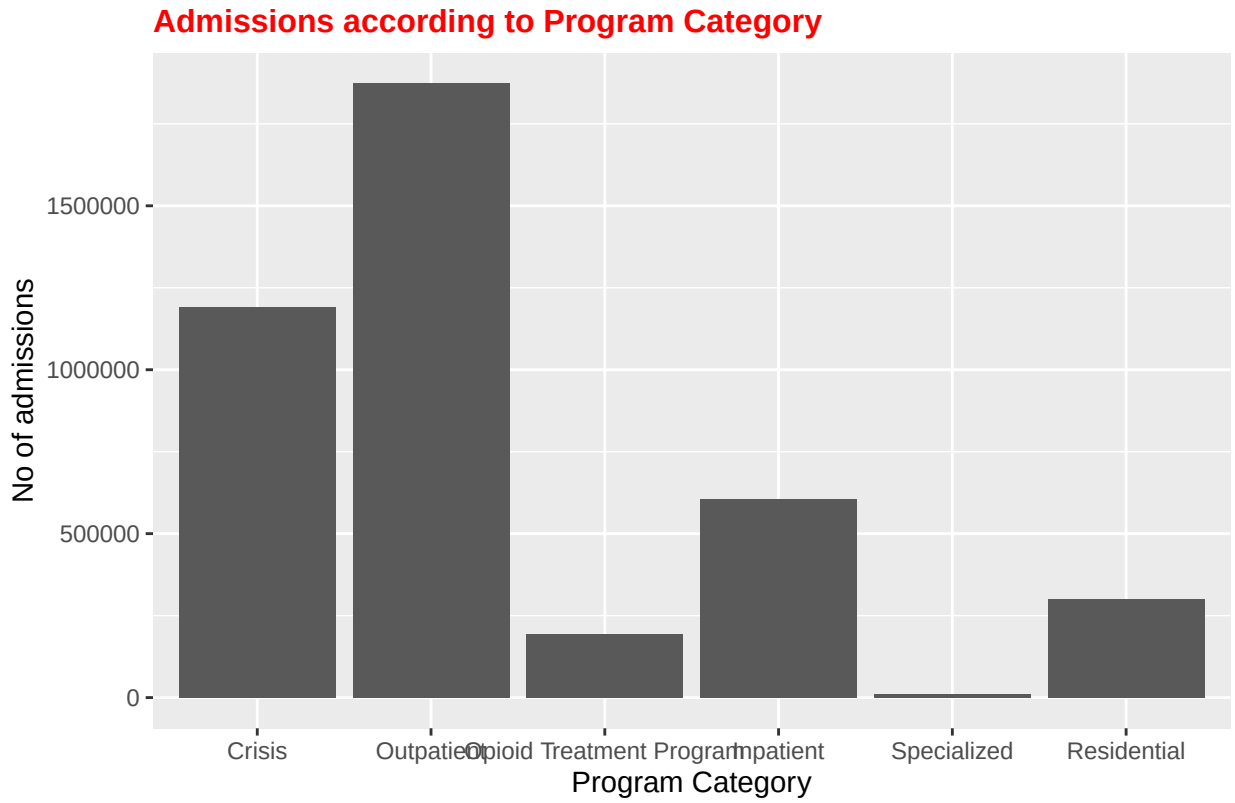
```
##       Year      County.of.Program.Location Program.Category
##  Min.   :2007   Length:99367               Length:99367
##  1st Qu.:2010   Class :character           Class :character
##  Median :2014   Mode  :character           Mode  :character
##  Mean   :2014
##  3rd Qu.:2018
##  Max.   :2021
##  Service.Type        Age.Group         Primary.Substance.Group
##  Length:99367       Length:99367       Length:99367
##  Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character
##
##
##
##    Admissions
##  Min.   :   1.00
##  1st Qu.:   2.00
##  Median :   8.00
##  Mean   :  41.91
##  3rd Qu.:  28.00
##  Max.   :2861.00
```

```
View(NYdata)
```

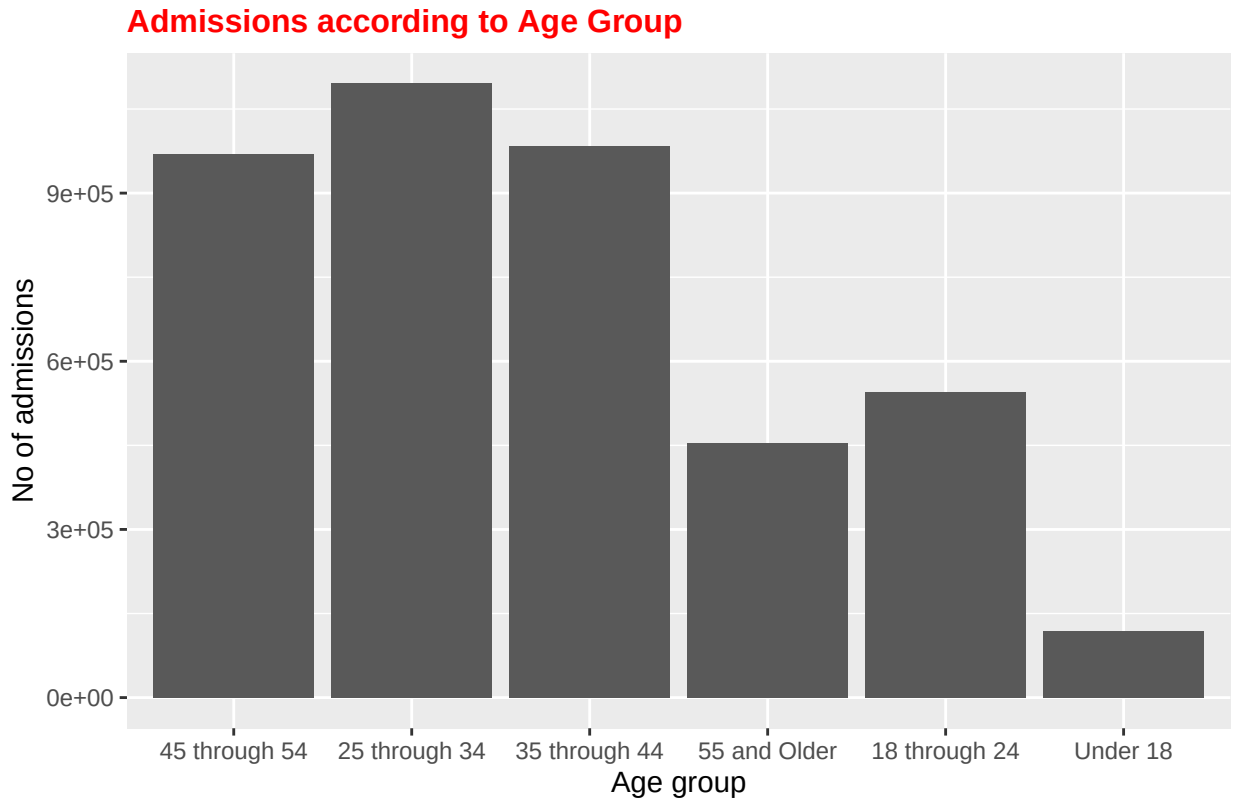
```
## Warning in system2("/usr/bin/otool", c("-L", shQuote(DSO)), stdout = TRUE):
## running command ''/usr/bin/otool' -L '/Library/Frameworks/R.framework/Resources/
## modules/R_de.so'' had status 1
```

```
# Missing Values
na <- table(is.na(NYdata)) #No null values in the data set

#Finding counts

#1)
ggplot(data= NYdata, aes(x= reorder(Program.Category,-Admissions), y= Admissions)) +
  geom_histogram(stat= "identity") +
  labs(title= "Admissions according to Program Category",
       caption = "Chart to visualize no of admissions for each program category") +
  theme(plot.title = element_text(color = "red", size = 12, face = "bold"), plot.caption = element_text
  xlab("Program Category") + ylab("No of admissions") #Maximum admissions are for Outpatient and minimu
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```




```
#2)
ggplot(data= NYdata, aes(x= reorder(Age.Group,-Admissions), y= Admissions)) +
  geom_bar(stat= "identity") +
   labs(title= "Admissions according to Age Group",
        caption = "Chart to visualize no of admissions for each age group") +
  theme(plot.title = element_text(color = "red", size = 12, face = "bold"), plot.caption = element_text
  xlab("Age group") + ylab("No of admissions") #Maximum admissions are for age group 25 thru 34 and min
```




```
#3)
ggplot(data= NYdata, aes(x= reorder(Primary.Substance.Group,-Admissions), y= Admissions)) +
  geom_histogram(stat= "identity") +
   theme(axis.text.x = element_text(angle= 90, hjust=0))+
  labs(title= "Admissions according to Primary Substance Group",
       caption = "Chart to visualize no of admissions for each Primary Substance Group") +
  theme(plot.title = element_text(color = "red", size = 12, face = "bold"), plot.caption = element_text
  xlab("Primary Substance Group") + ylab("No of admissions") #Maximum admissions are for Alcohol and mi
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```

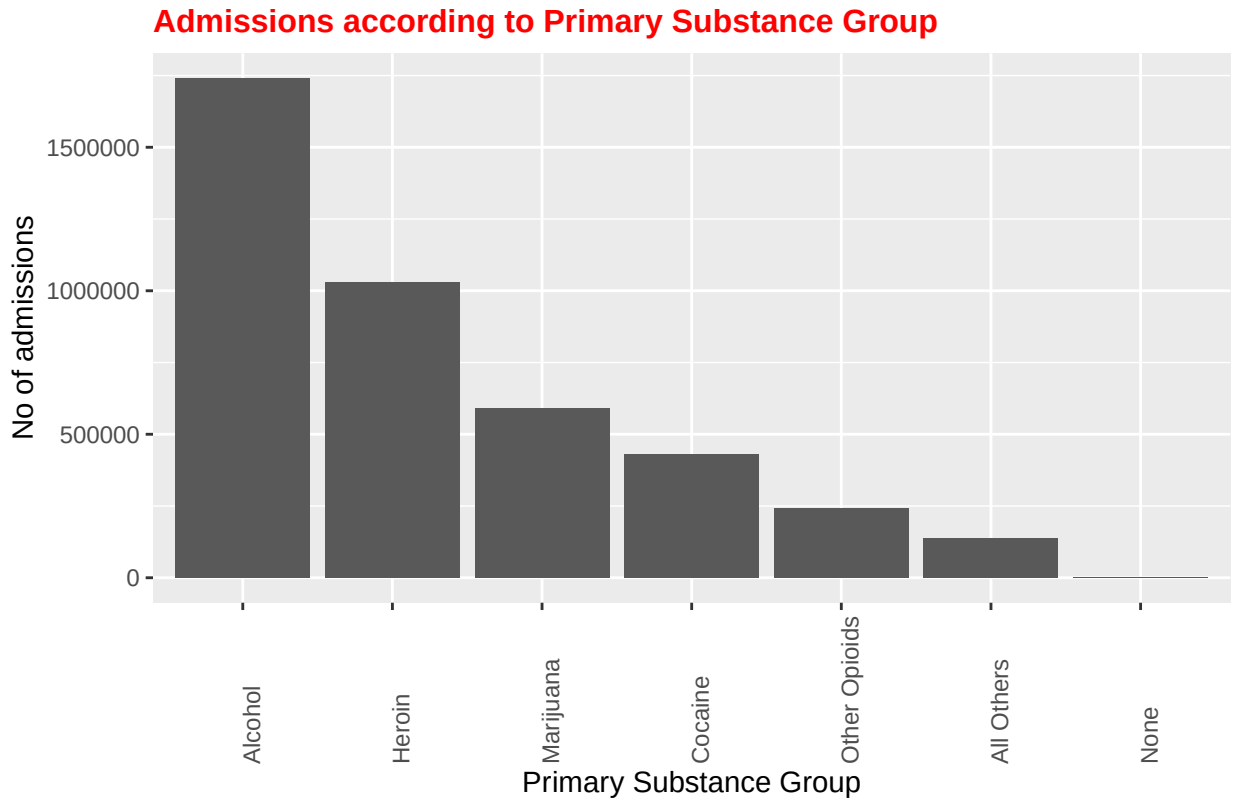


Chart to visualize no of admissions for each Primary Substance Group

```
#This might mean that ages 25 through 34 consume alcohol and do outpatient visits.

# Outliers for no. of Admissions using box plot
ggplot(NYdata) +
  aes(x = "", y = Admissions) +
  geom_boxplot(fill = "#0c4c8a") +
  theme_minimal()+
  labs(title= "Outliers for number of admissions",
       caption = "Chart to visualize outliers for no of admissions") +
  theme(plot.title = element_text(color = "red", size = 12, face = "bold"), plot.caption = element_text
  xlab("Data distribution") + ylab("No of admissions")
```

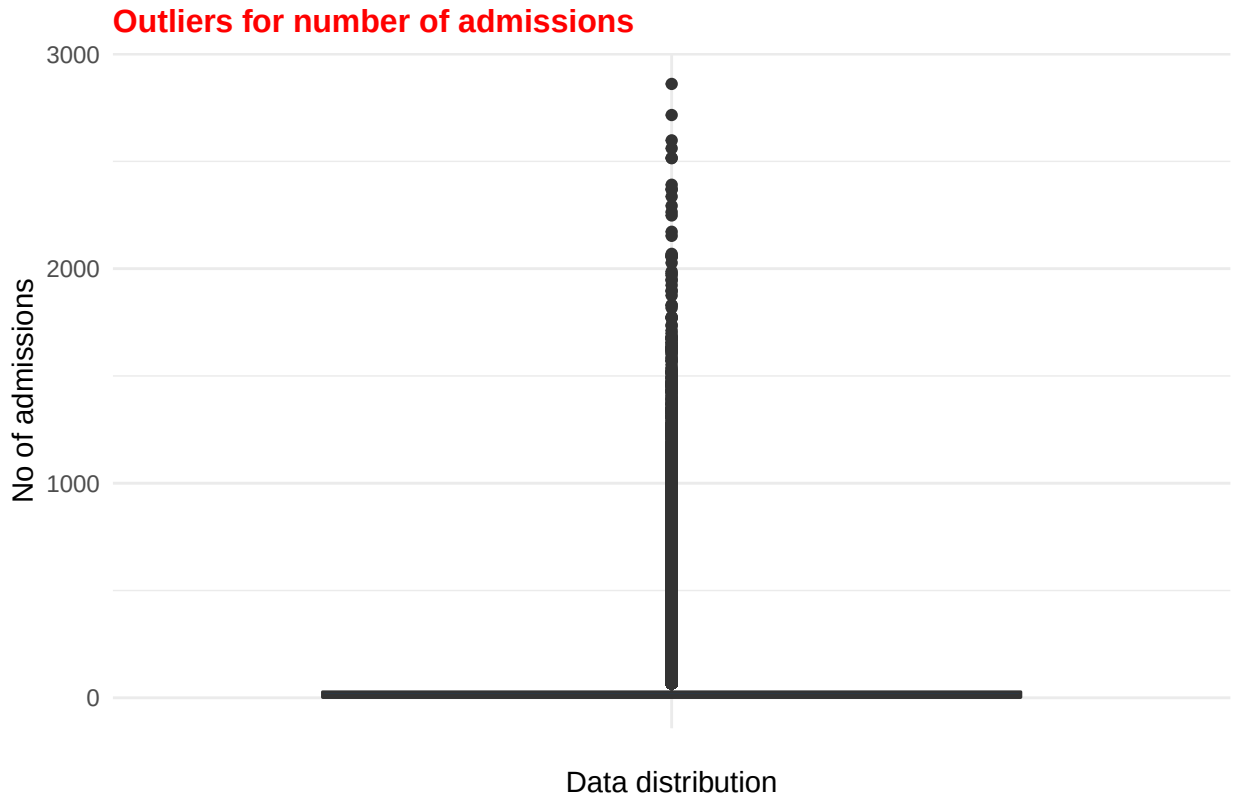


Chart to visualize outliers for no of admissions

```
#Outliers using which()

#Calculate mean, sd and z score
mean_df <- mean(NYdata$Admissions)
sd_df <- sd (NYdata$Admissions)
z_df <- abs((mean_df- NYdata$Admissions)/ sd_df)

#Outliers are 3 z- scores away from either side of the mean
outliers <- NYdata[which(z_df >3),]
View(outliers)
```

```
## Warning in system2("/usr/bin/otool", c("-L", shQuote(DSO)), stdout = TRUE):
## running command ''/usr/bin/otool' -L '/Library/Frameworks/R.framework/Resources/
## modules/R_de.so'' had status 1
```

```
summary(outliers)
```

```
##       Year      County.of.Program.Location Program.Category
##  Min.   :2007   Length:1917                Length:1917
##  1st Qu.:2010   Class :character           Class :character
##  Median :2013   Mode  :character           Mode  :character
##  Mean   :2013
##  3rd Qu.:2017
##  Max.   :2021
##  Service.Type        Age.Group         Primary.Substance.Group   Admissions
```

```
##  Length:1917        Length:1917        Length:1917            Min.   : 411
##  Class :character   Class :character   Class :character       1st Qu.: 508
##  Mode  :character   Mode  :character   Mode  :character       Median : 656
##                                                              Mean   : 754
##                                                              3rd Qu.: 882
##                                                              Max.   :2861
```

```
#There are 1680 outliers in the set that are 3 SD away from mean. The minimum no. of admissions in outl
```

3.(30 pts) Structure the data relationally, at a minimum, you should have four tibbles or data frames as follows: • county which contains the name of all counties and their respective county code (which is the primary key).

```
# 1. County tibble
Countydf <- NYdata %>%
  distinct(County.of.Program.Location) %>% # To select distinct County names from the data set.
  mutate(County_code = c("AL","AG","BR","BM","CA","CY","CH","CM","CN","CL","CO","CR","DE","DU","ER","ES
                  "FR","FU","GE","GR","HE","JE","KI","LE","LI","MA","MO","MG","NA","NY","NI",
                  "ON","OD","OT","OR","OL","OS","OG","PU","QU","RE","RM","RO","SL","SA","SC","SH",
                  "SY","SE","ST","SU","SV","TI","TO","UL","WR","WS","WA","WE","WY","YA")) %>%#Adding co
  select(County_code, County.of.Program.Location) # To add county_code as the first column

county <- as_tibble(Countydf) # Converting data frame to a tibble

#  Code to check primary key
county %>%
  count(County_code) %>%
  filter(n > 1)
```

```
## # A tibble: 0 x 2
## # ... with 2 variables: County_code <chr>, n <int>
```

```
# 2. program_category tibble
Program_categorydf <- NYdata %>%
  distinct(Program.Category) %>% # To select distinct Program category from the data set.
  mutate(Program_code= c("CR", "IP", "OTP", "OP", "RE", "SP")) %>% # To add Program codes
  select(Program_code,Program.Category) # To add program code as the first column

Program_category <- as_tibble(Program_categorydf) # Converting data frame to a tibble
#  Code to check primary key
Program_category %>%
  count(Program_code) %>%
  filter(n>1)
```

```
## # A tibble: 0 x 2
## # ... with 2 variables: Program_code <chr>, n <int>
```

```
# 3. primary_substance_group tibble
Primary_substance_groupdf <- NYdata %>%
  distinct(Primary.Substance.Group) %>% # To select distinct Substance group from the data set.
  mutate(Substance_code = c("H","A","OO","AO","C","M","N")) %>% # To add substance codes.
  select(Substance_code,Primary.Substance.Group) # To add substance code as the first column
```

```
Primary_substance_group <- as_tibble(Primary_substance_groupdf) # Converting data frame to a tibble
#  Code to check primary key
Primary_substance_group %>%
  count(Substance_code) %>%
  filter(n>1)
```

```
## # A tibble: 0 x 2
## # ... with 2 variables: Substance_code <chr>, n <int>
```

```
# 4. admissions_data tibble
Admissions_datadf <- NYdata %>%
  left_join(county, by = "County.of.Program.Location") %>% # To Join the county tibble
  left_join(Program_category, by = "Program.Category") %>% # To Join Program category tibble
  left_join(Primary_substance_group, by = "Primary.Substance.Group") %>% # To Join Primary substance gr
  # Renaming the Code columns of 3 tibbles to resemble the original data frame
  rename("County_of_Program_Location" = County_code,
         "Program_Category" = Program_code,
         "Primary_Substance_Group" = Substance_code) %>%
  # Selecting all the columns excluding data in the county, program_category and primary substance grou
  select(Year,County_of_Program_Location,Program_Category,Service.Type,Age.Group,Primary_Substance_Grou
# Converting data frame to a tibble
Admissions_data <- as_tibble(Admissions_datadf)
```

4.(15 pts) Create a function called annualAdmissions() that derives the total number of reported admissions that transpired each year, for the entire state of NY and displays the results using a line chart. Annotate the chart to show the year with the highest number of admissions. Note: the year should be on the x-axis and the number of admissions on the y-axis. Explain the chart.

```
annualAdmission <- function(ad)
  #Creating a function called annualAdmission
{
  totadmin <- ad %>%
    mutate(ad_year = format(ad$Year)) %>%
    group_by(ad_year) %>%
    summarise( ad_mission = sum(Admissions))
    print(totadmin)
  max_ad <- totadmin %>% slice_max(ad_mission)
  #Plotting number of admission each year for the state of New York
  ggplot(totadmin, aes(x=ad_year,y=ad_mission, group=1)) + geom_line(colour = '#CC0033') + geom_vline(x
    theme(
  plot.title = element_text(color = "red", size = 12, face = "bold"),
  plot.caption = element_text(color = "green", size= 12, face = "italic")
)
}
annualAdmission(NYdata)
```

```
## # A tibble: 15 x 2
##    ad_year ad_mission
##    <chr>        <int>
##  1 2007        305032
##  2 2008        310234
##  3 2009        311717
```

```
##  4 2010         309703
##  5 2011         303016
##  6 2012         293009
##  7 2013         284252
##  8 2014         282088
##  9 2015         279797
## 10 2016         277690
## 11 2017         276683
## 12 2018         275624
## 13 2019         268856
## 14 2020         200461
## 15 2021         186693
```

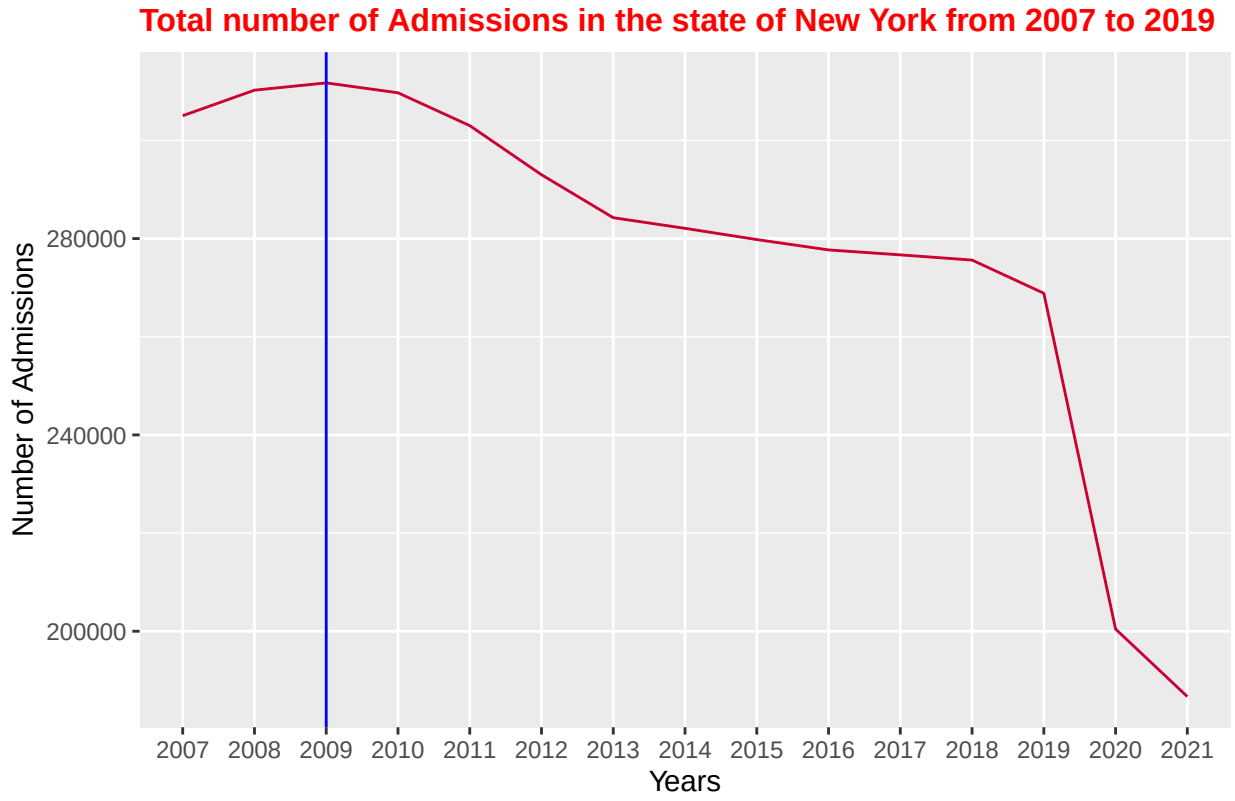


Chart to show the year with the highest number of admissions

5. (10 pts) Analyze the percentage of admissions for each county and visualize the results for the top 10 counties using a bar chart. Explain the results. Note: ensure that you join any related dataframes/tibbles.

```
proportion <- NYdata %>%
  group_by (County.of.Program.Location)%>%
  summarize (Total= sum(Admissions)) %>%
  mutate(Proportion = Total/sum(Total)*100) %>%
  arrange(desc(Proportion))
View(proportion)
```

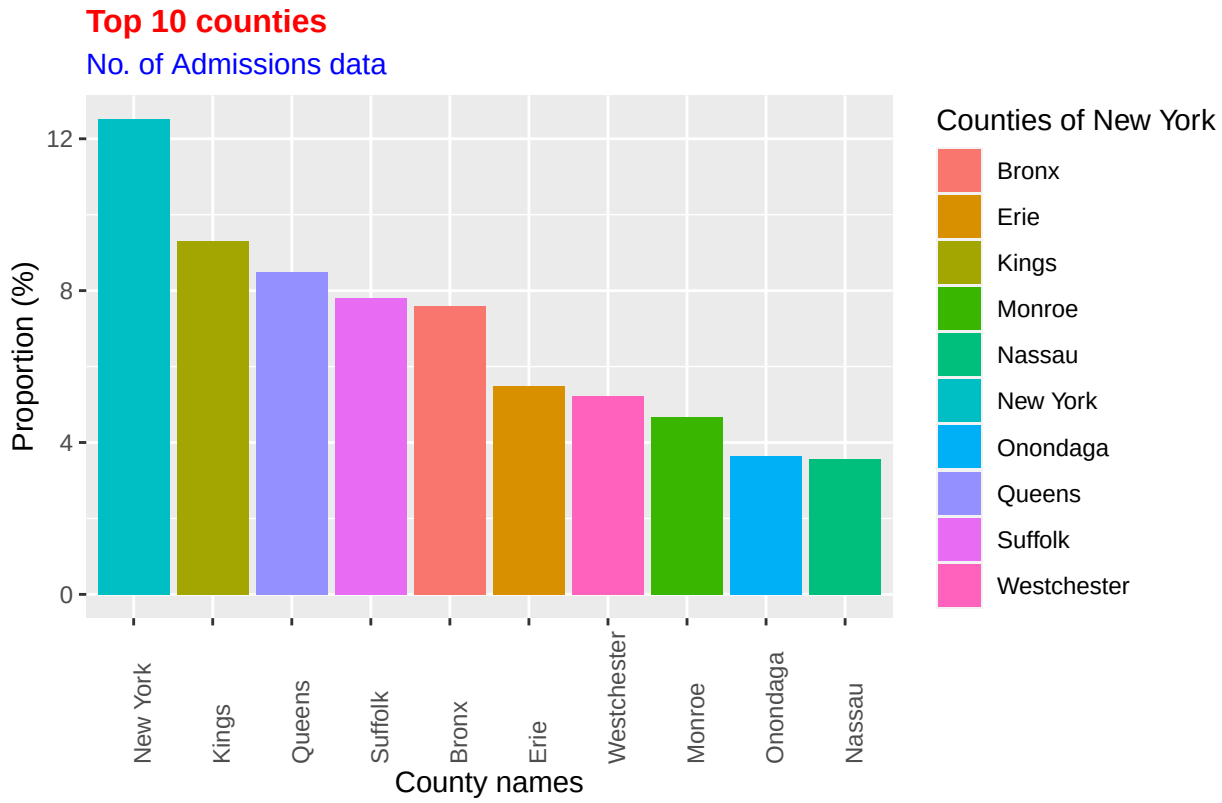
```
## Warning in system2("/usr/bin/otool", c("-L", shQuote(DSO)), stdout = TRUE):
```

```
## running command ''/usr/bin/otool' -L '/Library/Frameworks/R.framework/Resources/
## modules/R_de.so'' had status 1
```

```
#Comment: The maximum proportion of cases are in New York accounting for 14.87% (561853) while the mini

#To find top 10
top10 <- proportion %>% head(10)
top10
```

```
## # A tibble: 10 x 3
##    County.of.Program.Location  Total Proportion
##    <chr>                        <int>      <dbl>
##  1 New York                    521320      12.5
##  2 Kings                       387856       9.31
##  3 Queens                      353322       8.48
##  4 Suffolk                     325150       7.81
##  5 Bronx                       316706       7.60
##  6 Erie                        228610       5.49
##  7 Westchester                 217889       5.23
##  8 Monroe                      194765       4.68
##  9 Onondaga                    151995       3.65
## 10 Nassau                      148455       3.56
```

```
ggplot(data= top10, aes(x= reorder(County.of.Program.Location, -Proportion), y= Proportion, fill= County
  geom_bar(stat= "identity") +
   theme(axis.text.x = element_text(angle= 90, hjust=0))+
   labs(title= "Top 10 counties",
        subtitle= "No. of Admissions data",
        caption = "Bar chart to visulaize the no of admissions",
        fill= "Counties of New York") +
  theme(
  plot.title = element_text(color = "red", size = 12, face = "bold"),
  plot.subtitle = element_text(color = "blue"),
  plot.caption = element_text(color = "green", size= 12, face = "italic")
) +
  xlab("County names") + ylab("Proportion (%)")
```



Bar chart to visulaize the no of admissions

```
#Among the top 10 counties, New York has maximum cases, 14.87% (561853) as seen earlier from the total
```

6.(15 pts) Filter the data, using a regular expression, and extract all admissions to the various "Rehab" facilities; i.e. your regex should match all facilities that include the word rehab, rehabilitation, etc. Using the filtered data, identify which substance is the most prominent among each age group. Visualize and explain the results.

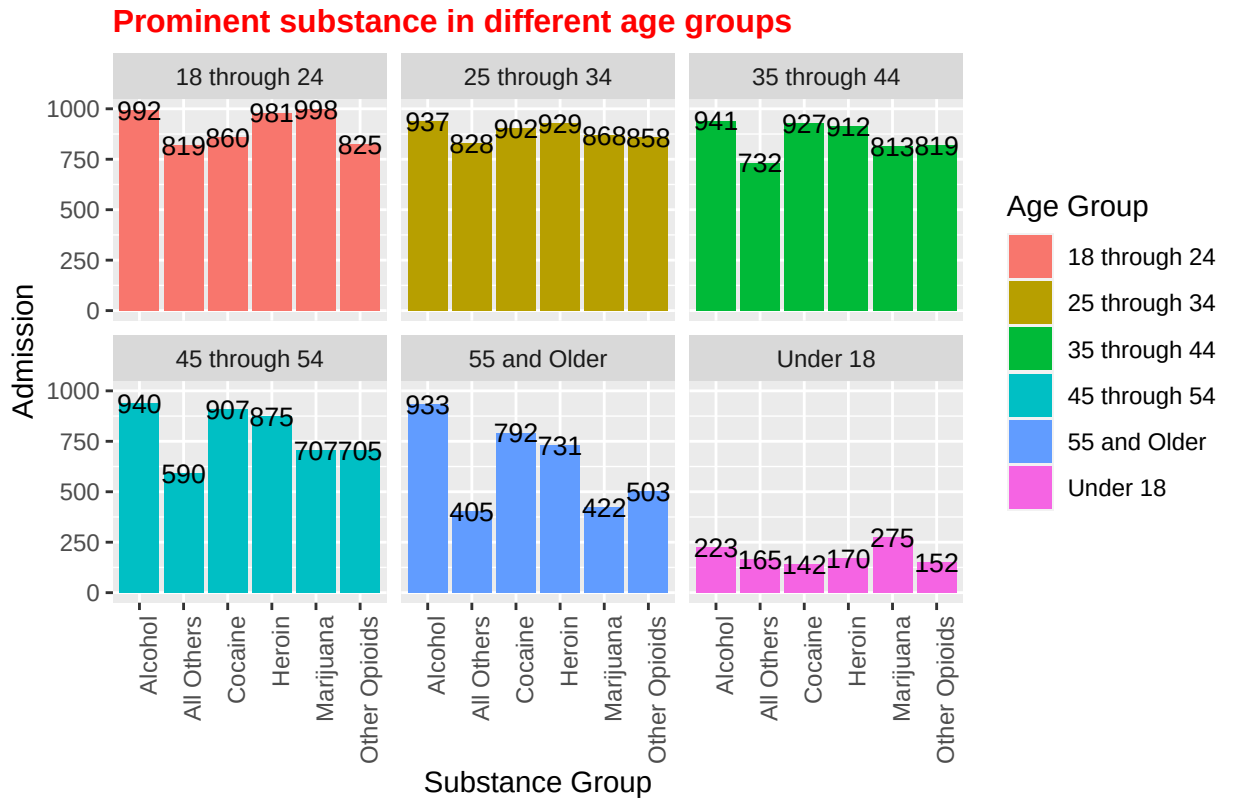
```
# Filter the data, using a regular expression
Rehab <- Admissions_data %>%
  filter(str_detect(Service.Type, "Rehab")) # Using str_detect to select elements matching the patter "
# To check that all the service facilities contain the word Rehab
Rehab %>%
  distinct(Service.Type)
```

```
## # A tibble: 8 x 1
##   Service.Type
##   <chr>
## 1 Inpatient Rehabilitation
## 2 Outpatient Rehabilitation
## 3 Residential Rehab for Youth
## 4 Specialized Services Outpt Rehab
## 5 Stabilization Rehab Reintegration
## 6 Rehab and Reintegration
## 7 Stabilization and Rehab
## 8 Residential Rehabilitation
```

```r
# To check the count of the substance group according to the service facilities in different Age groups
Rehab %>%
  count(Primary_Substance_Group,Age.Group)
```

```
## # A tibble: 36 x 3
##    Primary_Substance_Group Age.Group         n
##    <chr>                   <chr>         <int>
##  1 A                       18 through 24   819
##  2 A                       25 through 34   828
##  3 A                       35 through 44   732
##  4 A                       45 through 54   590
##  5 A                       55 and Older    405
##  6 A                       Under 18        165
##  7 AO                      18 through 24   992
##  8 AO                      25 through 34   937
##  9 AO                      35 through 44   941
## 10 AO                      45 through 54   940
## # ... with 26 more rows
```

```r
# Visualizing the data using a bar chart
Rehab %>%
  rename("Substance_code" = "Primary_Substance_Group") %>% # Renaming the substance code to join the ti
  left_join(Primary_substance_group, by = "Substance_code") %>% # Joining substance group tibble to get
  ggplot(aes(x = Primary.Substance.Group)) +
  geom_bar(aes(fill = Age.Group)) + # to filter data of different services
  facet_wrap(~Age.Group,nrow = 2) +
  theme(axis.text.x = element_text(angle = 90, hjust = 1)) +
  geom_text(stat = "count", aes(label = ..count..),vjust = 0.5,size = 3.5) +
  # Naming the x and y axes, title, caption and legend
  labs(x = "Substance Group", y = "Admission", title = "Prominent substance in different age groups",
       caption = "Prominent substances in different groups of people along with Rehab services",
       fill = "Age Group") +
  theme(
  plot.title = element_text(color = "red", size = 12, face = "bold"),
  plot.caption = element_text(color = "green", size= 12, face = "italic")
)
```



t substances in different groups of people along with Rehab services

Answer 6

The above bar graph is showing results of the prominent substance group in different age groups in the New York state:

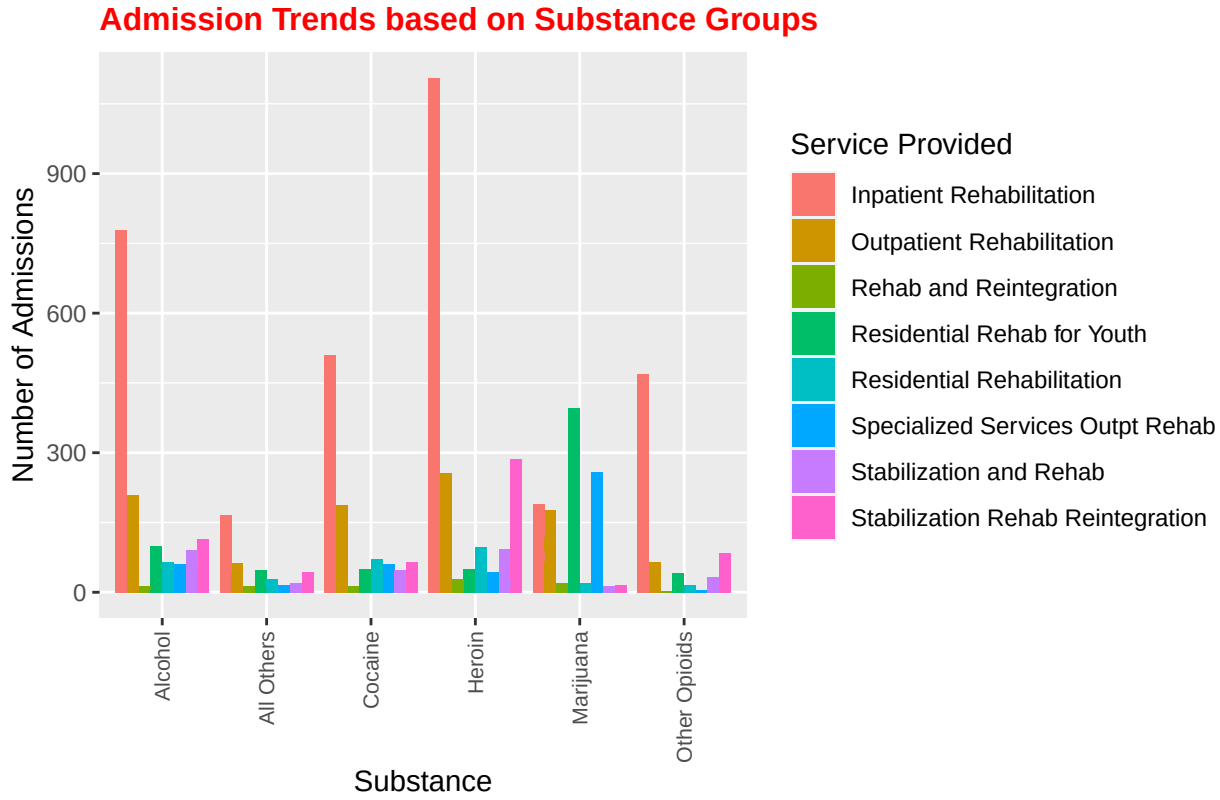
1. Under 18: The prominent substance is Marijuana including Hashish(250), The least common is Cocaine including crack(133)
2. Age group 18 - 24: The prominent substance is Marijuana including Hashish(863).
3. Age group 25 - 34: The prominent substance is Alcohol(772).
4. Age group 35 - 44: The prominent substance is Alcohol(776).
5. Age group 45 - 54: The prominent substance is Alcohol(775).
6. Age group 55 and older: The prominent substance is Alcohol(766).

To conclude, The prominent substance group in people younger than 25 years is Marijuana including Hashish. And, Alcohol in people older than 24 years

7.(20 pts) Using the "rehab" data from question 6 above, perform a detailed analysis to identify any patterns or trends with respect to the admission to rehab facilities in certain counties and substance groups. Explain your observations. Note: ensure that you join any related dataframes/tibbles.

```
# Using Rehab data from Q6 to visualize number of admissions based on County and type of Substance.
Rehab %>%
  rename("Substance_code" = "Primary_Substance_Group") %>%
  left_join(Primary_substance_group, by = "Substance_code") %>%
```

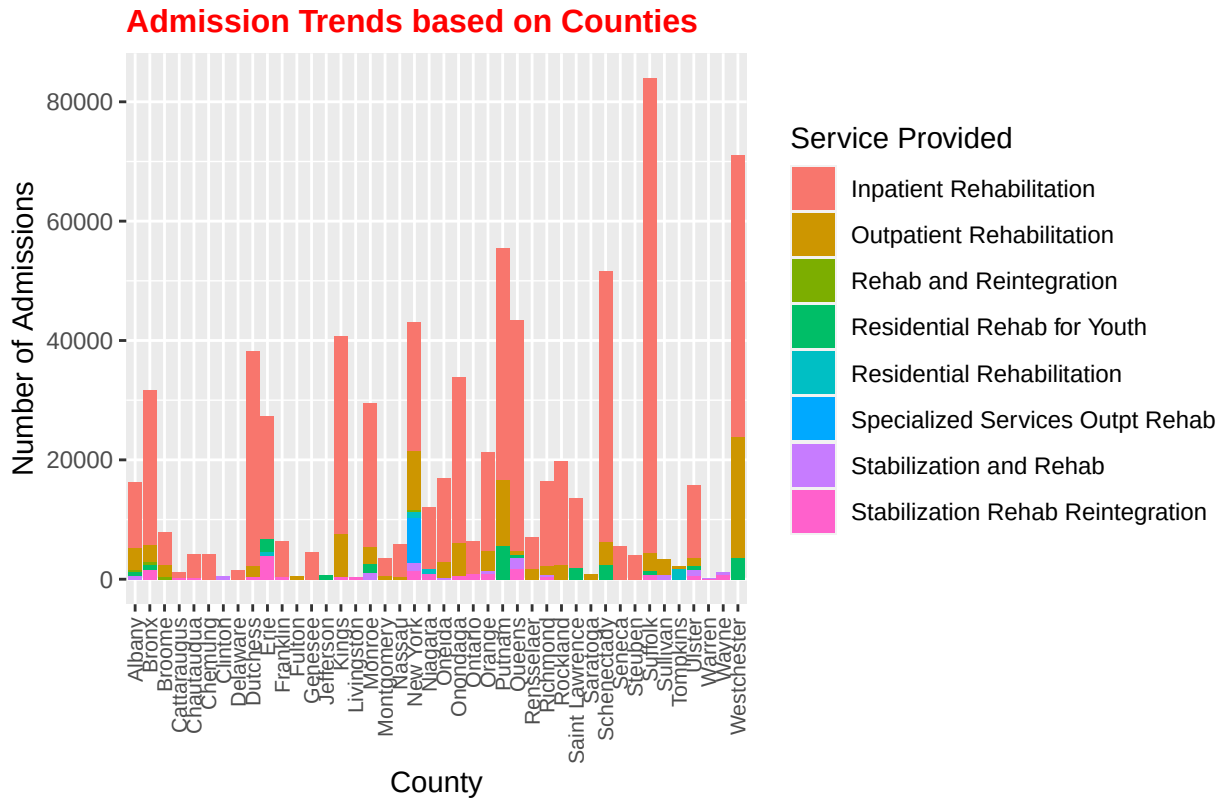
```
ggplot(aes(x = Primary.Substance.Group, y = Admissions, fill = Service.Type)) +
  geom_bar(stat = "Identity",position = "dodge") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5, size = 8)) +
  labs(x = "Substance", y = "Number of Admissions",
       title = "Admission Trends based on Substance Groups",
       caption = "Number of admissions in Rehab facilities based on the Primary Substance Groups.",
       fill = "Service Provided")+
  theme(
  plot.title = element_text(color = "red", size = 12, face = "bold"),
  plot.caption = element_text(color = "green", size= 12, face = "italic")
)
```



ab facilities based on the Primary Substance Groups.

```
Rehab %>%
  rename("County_code" = County_of_Program_Location) %>%
  left_join(county, by = "County_code") %>%
  ggplot(aes(x = County.of.Program.Location, y = Admissions, fill = Service.Type)) +
  geom_bar(stat = "Identity") +
  theme(axis.text.x = element_text(angle = 90, hjust = 1, vjust = 0.5, size = 8)) +
  labs(x = "County", y = "Number of Admissions",
       title = "Admission Trends based on Counties",
       caption = "Number of admissions in Rehab facilities based on the counties",
       fill = "Service Provided")+
  theme(
  plot.title = element_text(color = "red", size = 12, face = "bold"),
  plot.caption = element_text(color = "green", size= 12, face = "italic")
```

```
)
```



admissions in Rehab facilities based on the counties