R Notebook

Code

This is an R Markdown (http://rmarkdown.rstudio.com) Notebook. When you execute code within the notebook, the results appear beneath the code.

Try executing this chunk by clicking the *Run* button within the chunk or by placing your cursor inside it and pressing *Ctrl+Shift+Enter*.

Hide

```
# Data visualization
library(readr) # CSV file I/O, e.g. the read_csv function
library(needs)
needs(lubridate,
      dplyr,
      tidyr,
      Boruta,
      ggplot2,
      gridExtra,
      caret,
      rpart.plot,
      caTools,
      )
library(data.table)
#registerDoMC(cores = 3)
suppressMessages(library(pscl))
suppressMessages(library(data.table))
suppressMessages(library(FeatureHashing))
suppressMessages(library(Matrix))
suppressMessages(library(xgboost))
suppressMessages(require(caret))
suppressMessages(library(pROC))
suppressMessages(library(dummies))
suppressMessages(library(Metrics))
suppressMessages(library(kernlab))
suppressMessages(library(mlbench))
suppressMessages(library(tidyverse))
```

Hide

```
#### Step 1: Import Data.
setwd("C:/Users/Daniela Orovwiroro/Downloads")
data <- fread("UPHFinal.csv", header=T)
str(data)
```

```
Classes ‘data.table’ and 'data.frame':  95221 obs. of  40 variables:
 $ V1                   : int  1 2 3 4 5 6 7 8 9 10 ...
 $ AppointmentID        : int  21725 11206 12548 12727 86882 95113 56930 19704 29351 9
2214 ...
 $ PatientID            : int  1 2 2 2 2 3 3 4 5 6 ...
 $ ClinicNM             : chr  "E" "A" "A" "A" ...
 $ AppointmentDTS       : chr  "4/10/2018" "2/7/2018" "2/8/2018" "3/8/2018" ...
 $ AppointmentMonthNBR  : int  4 2 2 3 11 12 8 3 4 12 ...
 $ AppointmentWeekdayNBR: int  3 4 5 5 6 6 6 2 5 6 ...
 $ AppointmentHourNBR   : int  17 10 16 15 16 15 15 16 10 14 ...
 $ AgeNBR               : int  50 80 80 80 80 32 32 37 23 33 ...
 $ SexFLG               : chr  "F" "M" "M" "M" ...
 $ HispanicFLG          : int  0 0 0 0 0 0 0 0 0 1 ...
 $ SingleFLG            : int  1 0 0 0 0 1 1 1 0 1 ...
 $ LivesInApartmentFLG  : int  1 0 0 0 0 0 0 0 1 0 ...
 $ EmailFLG             : int  0 0 0 0 0 1 1 0 0 0 ...
 $ ApptLagNBR           : int  32 2 0 28 0 1 18 0 0 2 ...
 $ InsuranceDSC         : chr  "Commercial" "Medicare" "Medicare" "Medicare" ...
 $ HypertensionFLG      : int  0 1 1 1 1 0 0 0 0 0 ...
 $ AsthmaFLG            : int  0 0 0 0 0 0 0 0 0 0 ...
 $ HeartDiseaseFLG      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ ObeseFLG             : int  0 1 1 1 1 0 0 0 0 0 ...
 $ DiabetesFLG          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Noshow24NBR          : int  1 1 1 1 1 1 1 2 1 1 ...
 $ CancellationsNBR     : int  1 1 2 2 2 1 1 2 25 1 ...
 $ Latearrivals24NBR    : int  2 2 2 2 1 2 2 2 42 1 ...
 $ CheckintoCheckoutNBR : int  789 792 792 801 811 787 482 805 826 702 ...
 $ AppttoCheckoutNBR    : int  865 859 859 852 846 836 503 824 901 731 ...
 $ CheckintoApptNBR     : int  1 267 267 374 291 311 127 385 268 291 ...
 $ Arrived24NBR         : int  86 86 86 114 114 114 86 125 109 1 ...
 $ Providers24CNT       : int  13 2 2 2 2 24 13 35 10 1 ...
 $ ThatProvider24NBR    : int  1 218 229 251 36 14 36 14 1 1 ...
 $ NoshowRate24NBR      : int  1 1 1 1 1 1 1 657 1 1 ...
 $ EdVisitsNBR          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ IpVisitsNBR          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ NoShowFLG            : int  1 0 0 0 0 0 0 0 0 0 ...
 $ CancelledLateFLG     : int  0 1 0 0 0 0 0 0 0 0 ...
 $ NewPatient           : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Cost                 : int  75 75 75 75 75 75 75 75 75 75 ...
 $ Rand                 : num  0 0 0 0 0 ...
 $ Revenue              : int  125 95 95 95 95 125 125 125 80 30 ...
 $ Profit               : int  50 20 20 20 20 50 50 50 5 -45 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

Hide

```
#Exploratory analysis
data$SexFLG  <- factor(data$SexFLG, levels = c("M", "F"))
#data$AppointmentDTS= gsub("/","-",data$AppointmentDTS)
# some models don't like levels with character "-", so we apply make.names
data$InsuranceDSC = factor(make.names(data$InsuranceDSC))
data$ClinicNM = factor(make.names(data$ClinicNM))
#make all continous values int
#data$ThatProvider24NBR=as.numeric(data$ThatProvider24NBR)
#data$Providers24CNT=as.numeric(data$Providers24CNT)
#data$Arrived24NBR=as.numeric(data$Arrived24NBR)
#data$CheckintoApptNBR=as.numeric(data$CheckintoApptNBR)
#data$AppttoCheckoutNBR=as.numeric(data$AppttoCheckoutNBR)
#data$CheckintoCheckoutNBR=as.numeric(data$CheckintoCheckoutNBR)
str(data)
```

```
Classes ‘data.table’ and 'data.frame':  95221 obs. of  40 variables:
 $ V1                   : int  1 2 3 4 5 6 7 8 9 10 ...
 $ AppointmentID        : int  21725 11206 12548 12727 86882 95113 56930 19704 29351 9
2214 ...
 $ PatientID            : int  1 2 2 2 2 3 3 4 5 6 ...
 $ ClinicNM             : Factor w/ 5 levels "A","B","C","D",..: 5 1 1 1 1 1 1 1 3 2
...
 $ AppointmentDTS       : chr  "4/10/2018" "2/7/2018" "2/8/2018" "3/8/2018" ...
 $ AppointmentMonthNBR  : int  4 2 2 3 11 12 8 3 4 12 ...
 $ AppointmentWeekdayNBR: int  3 4 5 5 6 6 6 2 5 6 ...
 $ AppointmentHourNBR   : int  17 10 16 15 16 15 15 16 10 14 ...
 $ AgeNBR               : int  50 80 80 80 80 32 32 37 23 33 ...
 $ SexFLG               : Factor w/ 2 levels "M","F": 2 1 1 1 1 2 2 2 2 1 ...
 $ HispanicFLG          : int  0 0 0 0 0 0 0 0 0 1 ...
 $ SingleFLG            : int  1 0 0 0 0 1 1 1 0 1 ...
 $ LivesInApartmentFLG  : int  1 0 0 0 0 0 0 0 1 0 ...
 $ EmailFLG             : int  0 0 0 0 0 1 1 0 0 0 ...
 $ ApptLagNBR           : int  32 2 0 28 0 1 18 0 0 2 ...
 $ InsuranceDSC         : Factor w/ 4 levels "Commercial","Medicaid",..: 1 3 3 3 3 1 1
1 2 4 ...
 $ HypertensionFLG      : int  0 1 1 1 1 0 0 0 0 0 ...
 $ AsthmaFLG            : int  0 0 0 0 0 0 0 0 0 0 ...
 $ HeartDiseaseFLG      : int  0 0 0 0 0 0 0 0 0 0 ...
 $ ObeseFLG             : int  0 1 1 1 1 0 0 0 0 0 ...
 $ DiabetesFLG          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Noshow24NBR          : int  1 1 1 1 1 1 1 2 1 1 ...
 $ CancellationsNBR     : int  1 1 2 2 2 1 1 2 25 1 ...
 $ Latearrivals24NBR    : int  2 2 2 2 1 2 2 2 42 1 ...
 $ CheckintoCheckoutNBR : int  789 792 792 801 811 787 482 805 826 702 ...
 $ AppttoCheckoutNBR    : int  865 859 859 852 846 836 503 824 901 731 ...
 $ CheckintoApptNBR     : int  1 267 267 374 291 311 127 385 268 291 ...
 $ Arrived24NBR         : int  86 86 86 114 114 114 86 125 109 1 ...
 $ Providers24CNT       : int  13 2 2 2 2 24 13 35 10 1 ...
 $ ThatProvider24NBR    : int  1 218 229 251 36 14 36 14 1 1 ...
 $ NoshowRate24NBR      : int  1 1 1 1 1 1 1 657 1 1 ...
 $ EdVisitsNBR          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ IpVisitsNBR          : int  0 0 0 0 0 0 0 0 0 0 ...
 $ NoShowFLG            : int  1 0 0 0 0 0 0 0 0 0 ...
 $ CancelledLateFLG     : int  0 1 0 0 0 0 0 0 0 0 ...
 $ NewPatient           : int  0 0 0 0 0 0 0 0 0 0 ...
 $ Cost                 : int  75 75 75 75 75 75 75 75 75 75 ...
 $ Rand                 : num  0 0 0 0 0 ...
 $ Revenue              : int  125 95 95 95 95 125 125 125 80 30 ...
 $ Profit               : int  50 20 20 20 20 50 50 50 5 -45 ...
 - attr(*, ".internal.selfref")=<externalptr>
```

Hide

```
summary(data)
```

```
       V1        AppointmentID     PatientID     ClinicNM  AppointmentDTS     Appointm
entMonthNBR
 Min.   :    1   Min.   :    1   Min.   :    1   A:26370   Length:95221       Min.   :
1.000
 1st Qu.:23806   1st Qu.:23806   1st Qu.: 8386   B:29990   Class :character   1st Qu.:
4.000
 Median :47611   Median :47611   Median :16640   C:14480   Mode  :character   Median :
6.000
 Mean   :47611   Mean   :47611   Mean   :16752   D: 9297                      Mean   :
6.561
 3rd Qu.:71416   3rd Qu.:71416   3rd Qu.:25323   E:15084                      3rd Qu.:
10.000
 Max.   :95221   Max.   :95221   Max.   :33473                                Max.   :
12.000
 AppointmentWeekdayNBR AppointmentHourNBR     AgeNBR       SexFLG     HispanicFLG
SingleFLG
 Min.   :1.000         Min.   : 7.00      Min.   :  0.00   M:38151   Min.   :0.0000
Min.   :0.0000
 1st Qu.:3.000         1st Qu.:10.00      1st Qu.: 27.00   F:57070   1st Qu.:0.0000
1st Qu.:0.0000
 Median :4.000         Median :11.00      Median : 46.00             Median :0.0000
Median :1.0000
 Mean   :3.988         Mean   :12.01      Mean   : 44.53             Mean   :0.1295
Mean   :0.5536
 3rd Qu.:5.000         3rd Qu.:14.00      3rd Qu.: 62.00             3rd Qu.:0.0000
3rd Qu.:1.0000
 Max.   :7.000         Max.   :19.00      Max.   :118.00             Max.   :1.0000
Max.   :1.0000
 LivesInApartmentFLG    EmailFLG        ApptLagNBR         InsuranceDSC   Hypertension
FLG
 Min.   :0.0000      Min.   :0.0000   Min.   :  0.00   Commercial:42768   Min.   :0.00
00
 1st Qu.:0.0000      1st Qu.:0.0000   1st Qu.:  1.00   Medicaid  :21900   1st Qu.:0.00
00
 Median :0.0000      Median :0.0000   Median :  7.00   Medicare  :22848   Median :0.00
00
 Mean   :0.1963      Mean   :0.4905   Mean   : 19.78   Self.Pay  : 7705   Mean   :0.35
79
 3rd Qu.:0.0000      3rd Qu.:1.0000   3rd Qu.: 23.00                      3rd Qu.:1.00
00
 Max.   :1.0000      Max.   :1.0000   Max.   :546.00                      Max.   :1.00
00
   AsthmaFLG       HeartDiseaseFLG     ObeseFLG       DiabetesFLG        Noshow24NBR
CancellationsNBR
 Min.   :0.00000   Min.   :0.0000   Min.   :0.0000   Min.   :0.000000   Min.   : 0.000
Min.   :  0.00
 1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.0000   1st Qu.:0.000000   1st Qu.: 1.000
1st Qu.:  1.00
```

```
 Median :0.00000   Median :0.0000   Median :0.0000   Median :0.000000   Median : 1.000
Median : 57.00
 Mean   :0.09416   Mean   :0.1091   Mean   :0.2591   Mean   :0.005492   Mean   : 5.727
Mean   : 53.36
 3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:1.0000   3rd Qu.:0.000000   3rd Qu.: 2.000
3rd Qu.: 92.00
 Max.   :1.00000   Max.   :1.0000   Max.   :1.0000   Max.   :1.000000   Max.   :38.000
Max.   :174.00
 Latearrivals24NBR CheckintoCheckoutNBR AppttoCheckoutNBR CheckintoApptNBR  Arrived24N
BR    Providers24CNT
 Min.   :  0.00    Min.   :  0.0        Min.   :  0.0     Min.   :  0.0    Min.   :
0.00   Min.   : 0.00
 1st Qu.:  2.00    1st Qu.:667.0        1st Qu.:630.0     1st Qu.:268.0    1st Qu.: 4
4.00   1st Qu.: 6.00
 Median : 17.00    Median :787.0        Median :803.0     Median :306.0    Median :11
2.00   Median :24.00
 Mean   : 21.81    Mean   :690.9        Mean   :720.1     Mean   :286.8    Mean   : 9
4.38   Mean   :24.95
 3rd Qu.: 31.00    3rd Qu.:814.0        3rd Qu.:891.0     3rd Qu.:339.0    3rd Qu.:13
6.00   3rd Qu.:45.00
 Max.   :101.00    Max.   :852.0        Max.   :940.0     Max.   :406.0    Max.   :19
0.00   Max.   :55.00
 ThatProvider24NBR NoshowRate24NBR   EdVisitsNBR        IpVisitsNBR        NoShowFLG
CancelledLateFLG
 Min.   :  0.0     Min.   :   0.0   Min.   :  0.0000   Min.   :0.00000   Min.   :0.000
00   Min.   :0.0000
 1st Qu.:  1.0     1st Qu.:   1.0   1st Qu.:  0.0000   1st Qu.:0.00000   1st Qu.:0.000
00   1st Qu.:0.0000
 Median :110.0     Median :   1.0   Median :  0.0000   Median :0.00000   Median :0.000
00   Median :0.0000
 Mean   :106.2     Mean   : 206.7   Mean   :  0.6494   Mean   :0.02111   Mean   :0.095
63   Mean   :0.1598
 3rd Qu.:207.0     3rd Qu.: 406.0   3rd Qu.:  1.0000   3rd Qu.:0.00000   3rd Qu.:0.000
00   3rd Qu.:0.0000
 Max.   :272.0     Max.   :1054.0   Max.   :155.0000   Max.   :6.00000   Max.   :1.000
00   Max.   :1.0000
   NewPatient           Cost         Rand            Revenue          Profit
 Min.   :0.00000   Min.   :75   Min.   :0.00000   Min.   :  0.0   Min.   :-75.00
 1st Qu.:0.00000   1st Qu.:75   1st Qu.:0.00000   1st Qu.: 80.0   1st Qu.:  5.00
 Median :0.00000   Median :75   Median :0.00000   Median : 95.0   Median : 20.00
 Mean   :0.01536   Mean   :75   Mean   :0.04047   Mean   :102.2   Mean   : 27.24
 3rd Qu.:0.00000   3rd Qu.:75   3rd Qu.:0.00000   3rd Qu.:125.0   3rd Qu.: 50.00
 Max.   :1.00000   Max.   :75   Max.   :0.99983   Max.   :140.0   Max.   : 65.00
```

Hide

```
#Let's check if there are duplicated data
dup_rows <- duplicated(data)
dup_rows_num <- sum(dup_rows)
dup_rows_num
```

```
[1] 0
```

Hide

```
#Step 2 : Visualization
status_table <- table(data$NoShowFLG)
status_table
```

```
    0     1
86115  9106
```

Hide

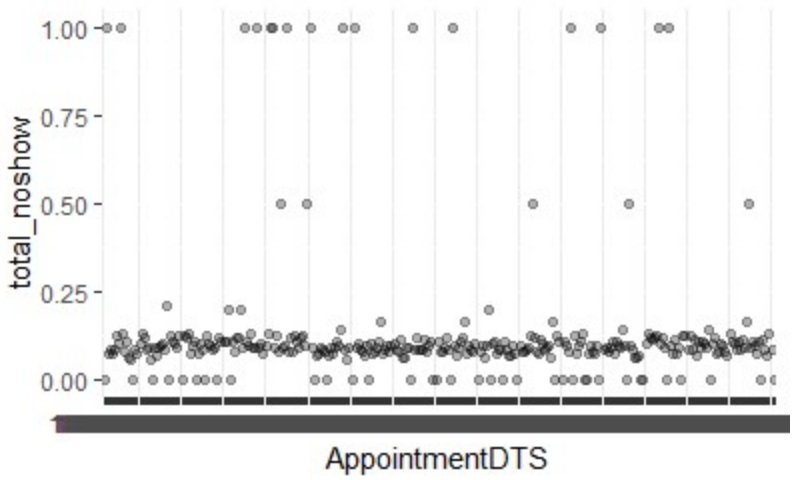
```
#the percentage of people who do not show up
(status_table["1"]/status_table["0"])*100
```

```
       1
10.57423
```

From the data we can see 10% of the populatio don’t show up

Hide

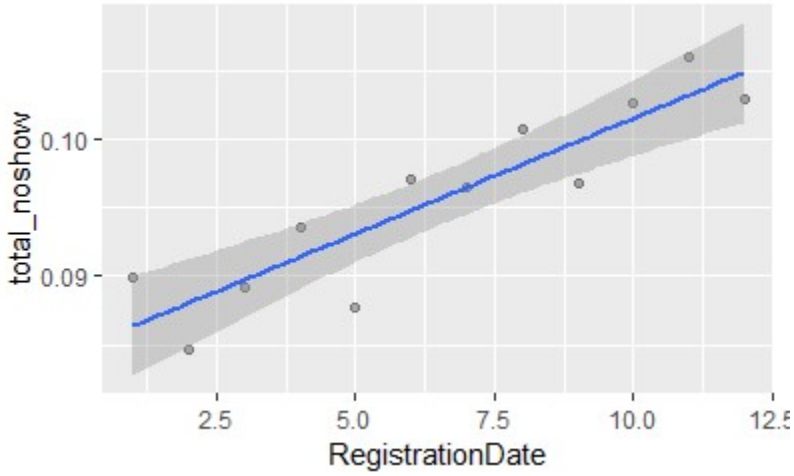
```
#visualize no show based on date
data %>% group_by(AppointmentDTS) %>% summarise(total_noshow=sum(NoShowFLG=="1")/n())
%>% ggplot(aes(x=AppointmentDTS, y=total_noshow)) +
    geom_point(alpha=0.3) + geom_smooth(method = "lm")
```



From the chart above we can see that the proportion of “No-show” each day stays approximately constant through the time period

Hide

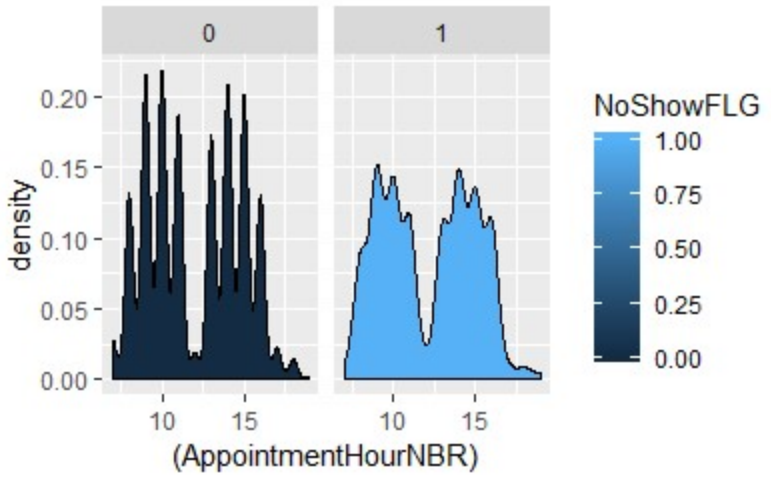
```
#Visualize month of appointment wrt noshow
data %>% group_by(RegistrationDate=(AppointmentMonthNBR)) %>% summarise(total_noshow=s
um(NoShowFLG=="1")/n()) %>% ggplot(aes(x=RegistrationDate, y=total_noshow)) + geom_poi
nt(alpha=0.3) + geom_smooth(method = "lm")
```



Appointment registration month is not significant because there isn’t a clear increase or decrease trend in the proportion of “No-Show”

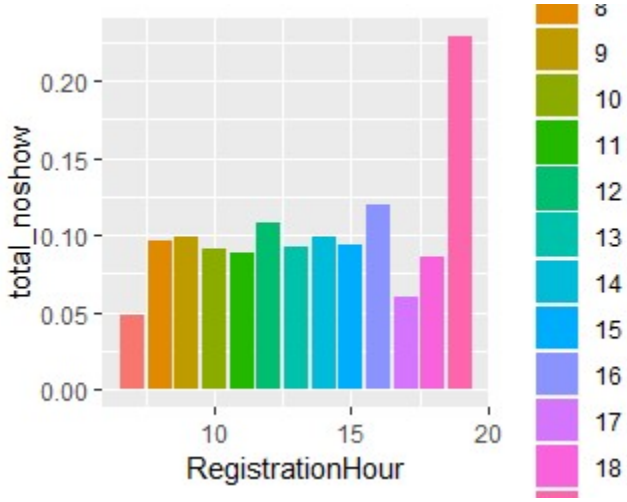
Hide

```
#Let's take a look to the hour of the registration appointment:
ggplot(data, aes(x=(AppointmentHourNBR), fill=NoShowFLG)) + geom_density() + facet_gri
d(.~NoShowFLG)
```



Hide

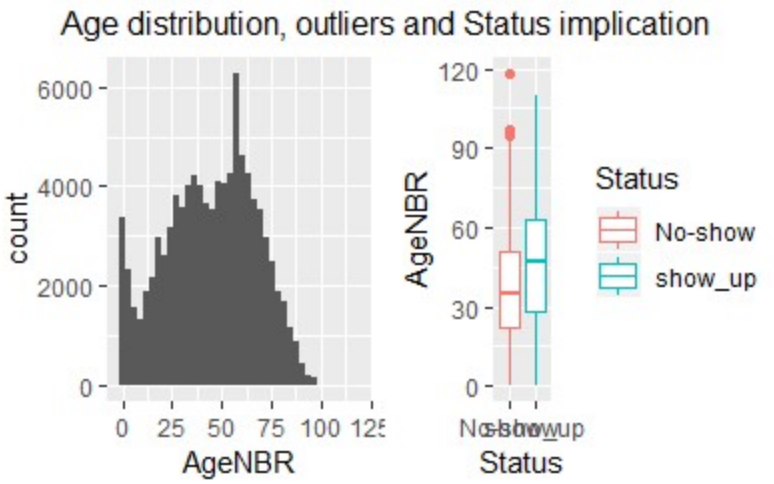
```
data %>% group_by(RegistrationHour=(AppointmentHourNBR)) %>% summarise(total_noshow=su
m(NoShowFLG=="1")/n()) %>% ggplot(aes(x=RegistrationHour, y=total_noshow, fill=as.fact
or(RegistrationHour))) + geom_bar(stat="identity") + scale_fill_discrete("Registration
Hour")
```



From our barchart we see that people scheduled from 7pm lead always to "No-Show

Hide

```
data$Status= ifelse(data$NoShowFLG == "1","No-show","show_up")
g_Age_1 <- ggplot(data, aes(x=AgeNBR )) + geom_histogram(bins=40)
g_Age_2 <- ggplot(data, aes(x=Status, y=AgeNBR , col=Status)) + geom_boxplot()
grid.arrange(g_Age_1, g_Age_2,ncol=2, top='Age distribution, outliers and Status impli
cation')
```



From the boxplot it can be assumed that the younger people no-show more than older ones.

Hide

```
#Let's see if Gender is important:
tab_Gender <- table(data$SexFLG, data$Status)
addmargins(tab_Gender)
```

```
     No-show show_up   Sum
  M      3563   34588 38151
  F      5543   51527 57070
  Sum    9106   86115 95221
```

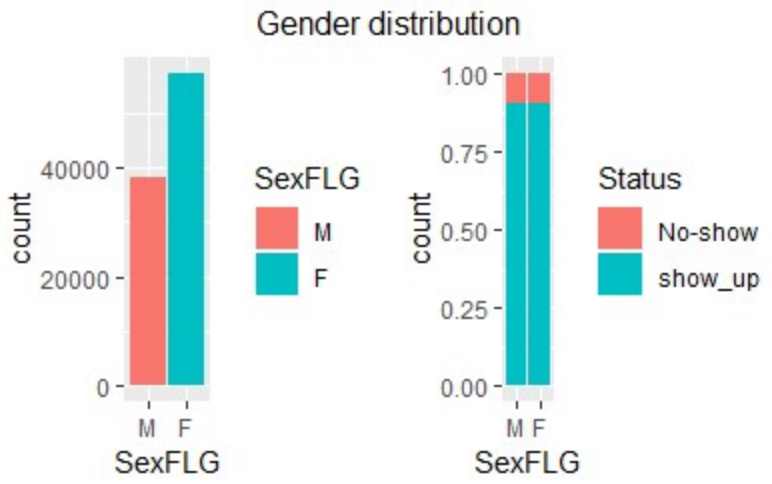
Hide

```
prop.table(tab_Gender,2)
```

```
     No-show   show_up
  M 0.3912805 0.4016490
  F 0.6087195 0.5983510
```

Hide

```
g_Gender_1 <- ggplot(data, aes(x=SexFLG, fill=SexFLG)) + geom_bar(position="dodge")
g_Gender_2 <- ggplot(data, aes(x=SexFLG, fill=Status)) + geom_bar(position="fill")
grid.arrange(g_Gender_1, g_Gender_2,ncol=2, top='Gender distribution')
```

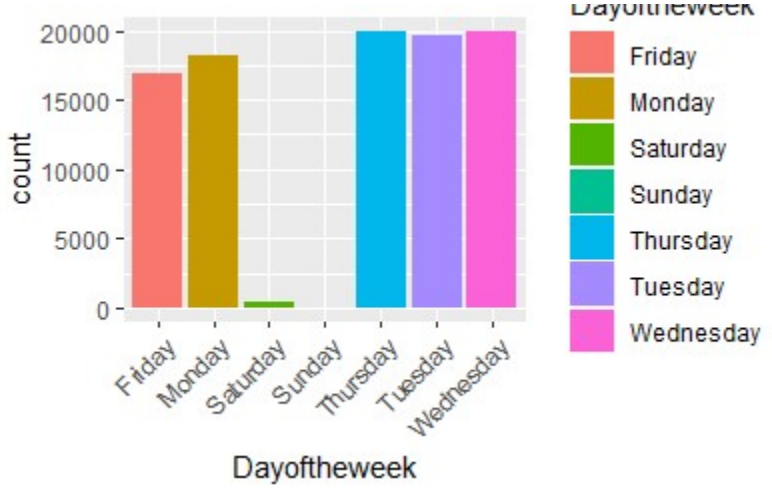


Hide

```
days<-c("Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday")
for(i in 1:7)
  data$Dayoftheweek[data$AppointmentWeekdayNBR==i]<-days[[i]]
```

Hide

```
ggplot(data, aes(x=Dayoftheweek, fill=Dayoftheweek )) + geom_bar() + theme(axis.text.x
= element_text(angle = 45, hjust = 1))
```



It can be seen that most people choose the weekday monday to friday for their appointment

Hide

```
#Perhaps some days of the week have more "No-Show". Let's check:
tab_DayOfTheWeek <- table(data$Status, data$Dayoftheweek)
addmargins(tab_DayOfTheWeek)
```

```
         Friday Monday Saturday Sunday Thursday Tuesday Wednesday   Sum
No-show    1699   1839       33     20     1820    1846      1849  9106
show_up   15230  16357      396     17    18116   17865     18134 86115
Sum       16929  18196      429     37    19936   19711     19983 95221
```

Hide

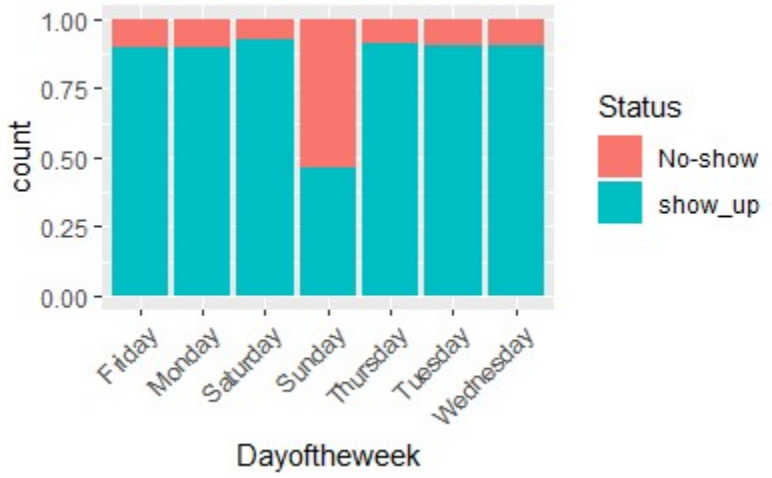
```
#probability table
prop.table(tab_DayOfTheWeek,2)
```

```
             Friday     Monday   Saturday     Sunday   Thursday    Tuesday  Wednesday
No-show 0.10036033 0.10106617 0.07692308 0.54054054 0.09129213 0.09365329 0.09252865
show_up 0.89963967 0.89893383 0.92307692 0.45945946 0.90870787 0.90634671 0.90747135
```

Most people with appointment day sunday do not show up

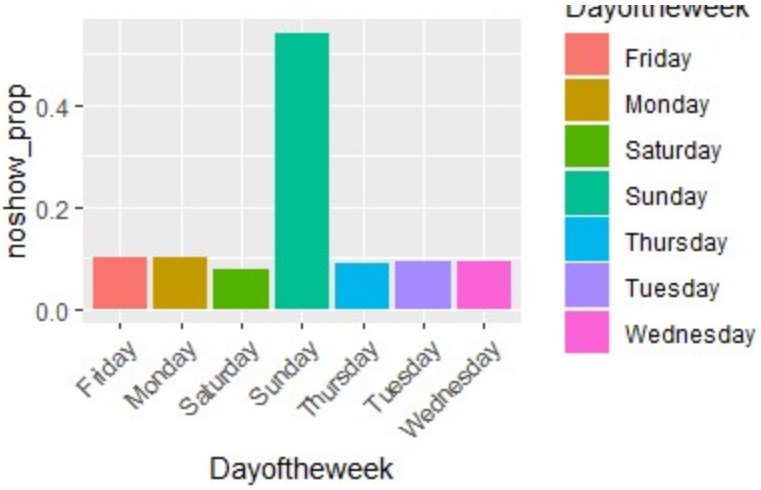
Hide

```
ggplot(data, aes(x=Dayoftheweek, fill=Status )) + geom_bar(position="fill") + theme(ax
is.text.x = element_text(angle = 45, hjust = 1))
```



Hide

```
#Let's see the proportion of "No.Show" per each day of the week:
data %>% group_by(Dayoftheweek) %>%
    summarise(noshow_prop=sum(Status=="No-show")/n()) %>%
    ggplot(aes(x=Dayoftheweek, y=noshow_prop, fill=Dayoftheweek)) +
        geom_bar(stat="identity") +
        theme(axis.text.x = element_text(angle = 45, hjust = 1))
```



Sunday is the day less people show up

Hide

```
#Let's check if Whether or not patient is Hispanic affects the no show
tab_hispanic <- table(data$HispanicFLG, data$Status)
addmargins(tab_hispanic)
```

```
     No-show show_up   Sum
  0      7478   75410 82888
  1      1628   10705 12333
  Sum    9106   86115 95221
```

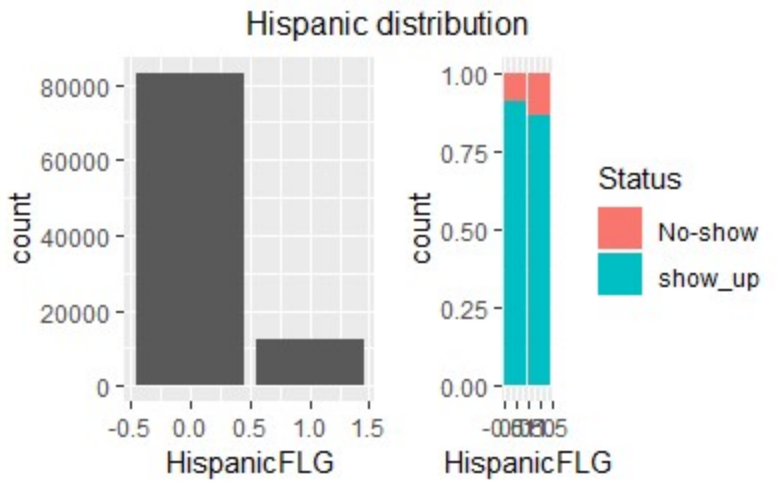
Hide

```
#probabability table
prop.table(tab_hispanic,2)
```

```
     No-show   show_up
  0 0.8212168 0.8756895
  1 0.1787832 0.1243105
```

Hide

```
#visualize
g_hispanic_1 <- ggplot(data, aes(x=HispanicFLG, fill=HispanicFLG)) + geom_bar(position
="dodge")
g_hispanic_2 <- ggplot(data, aes(x=HispanicFLG, fill=Status)) + geom_bar(position="fil
l")
grid.arrange(g_hispanic_1, g_hispanic_2,ncol=2, top='Hispanic distribution')
```



17% of the no show are hispanic

Hide

```
#Let's check if Whether or not patient is single or not affects the no show
tab_single <- table(data$SingleFLG, data$Status)
addmargins(tab_single)
```

```
     No-show show_up   Sum
  0      2951   39551 42502
  1      6155   46564 52719
  Sum    9106   86115 95221
```

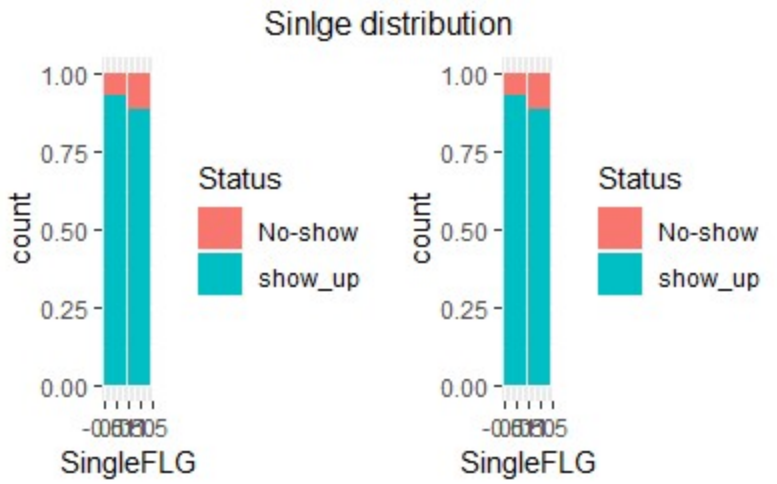
Hide

```
#probabability table
prop.table(tab_single,2)
```

```
     No-show   show_up
  0 0.3240720 0.4592812
  1 0.6759280 0.5407188
```

Hide

```
#visualize
g_single_1 <- ggplot(data, aes(x=SingleFLG, fill=SingleFLG)) + geom_bar(position="dodg
e")
g_single_2 <- ggplot(data, aes(x=SingleFLG, fill=Status)) + geom_bar(position="fill")
grid.arrange(g_single_2, g_single_2,ncol=2, top='Sinlge distribution')
```



from our analysis we see that 67% of the patients who live alone do not show up

Hide

```
#Let's check if Whether or not patient lives in an apartment affects the no show
tab_LIA <- table(data$LivesInApartmentFLG, data$Status)
addmargins(tab_LIA)
```

```
     No-show show_up   Sum
  0      6884   69641 76525
  1      2222   16474 18696
  Sum    9106   86115 95221
```

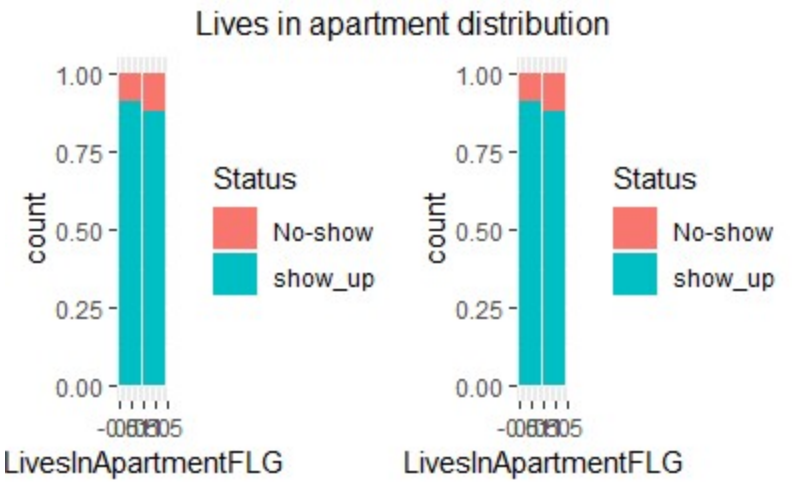
Hide

```
#probabability table
prop.table(tab_LIA,2)
```

```
     No-show   show_up
  0 0.7559851 0.8086977
  1 0.2440149 0.1913023
```

Hide

```
#visualize
g_LIA_1 <- ggplot(data, aes(x=LivesInApartmentFLG, fill=LivesInApartmentFLG)) + geom_b
ar(position="dodge")
g_LIA_2 <- ggplot(data, aes(x=LivesInApartmentFLG, fill=Status)) + geom_bar(position
="fill")
grid.arrange(g_LIA_2, g_LIA_2,ncol=2, top='Lives in apartment distribution')
```



From the analysis above we can see that 75% of people who do not live in apartment do not show up

Hide

```
#Let's check if Whether or not patient receives EmailFLG or not affects the no show
tab_EmailFLG <- table(data$EmailFLG, data$Status)
addmargins(tab_EmailFLG)
```

```
     No-show show_up   Sum
  0      5475   43038 48513
  1      3631   43077 46708
  Sum    9106   86115 95221
```

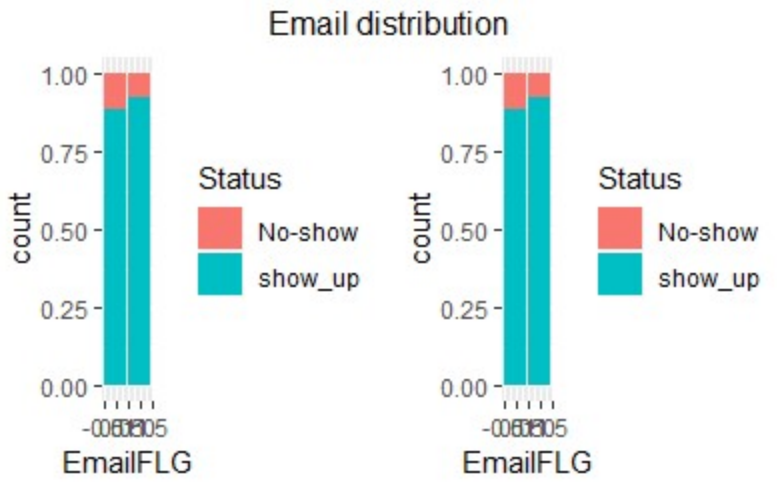
Hide

```
#probabability table
prop.table(tab_EmailFLG,2)
```

```
     No-show   show_up
  0 0.6012519 0.4997736
  1 0.3987481 0.5002264
```

Hide

```
#visualize
g_EmailFLG_1 <- ggplot(data, aes(x=EmailFLG, fill=EmailFLG)) + geom_bar(position="dodg
e")
g_EmailFLG_2 <- ggplot(data, aes(x=EmailFLG, fill=Status)) + geom_bar(position="fill")
grid.arrange(g_EmailFLG_2, g_EmailFLG_2,ncol=2, top='Email distribution')
```



60% of the patiens that did not provide email are no show

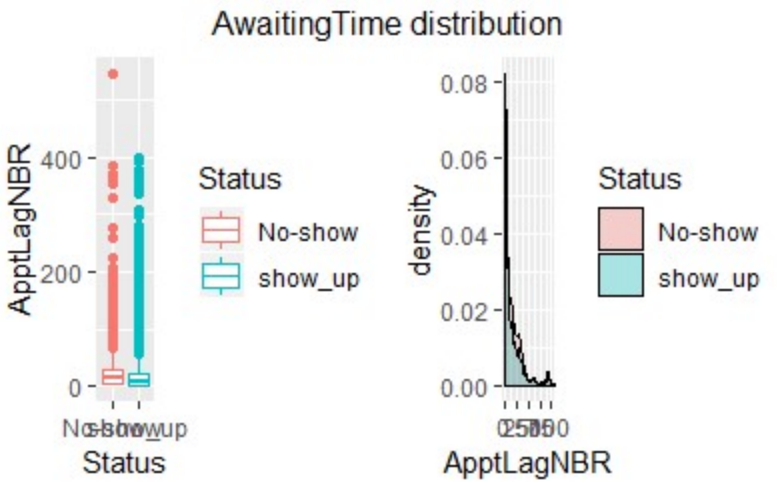
Hide

```
#Let's see if near Number of days between date appointment made and actual appointment
date have less "No-Show":
summary(data[data$Status=="No-Show","ApptLagNBR"])
```

```
   ApptLagNBR
 Min.   : NA
 1st Qu.: NA
 Median : NA
 Mean   :NaN
 3rd Qu.: NA
 Max.   : NA
```

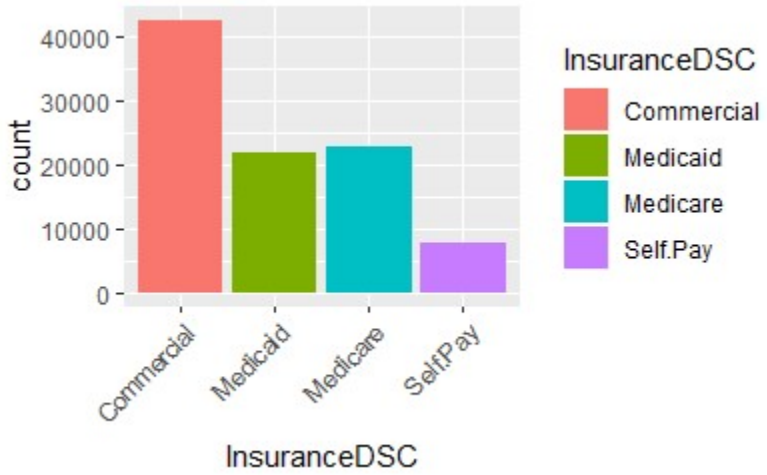
Hide

```
g_AwaitingTime_1 <- ggplot(data, aes(x=Status, y=ApptLagNBR, col=Status)) + geom_boxpl
ot()
g_AwaitingTime_2 <- ggplot(data, aes(x=ApptLagNBR, fill=Status)) +
                                geom_density(alpha=0.30) +
                                coord_cartesian(xlim=c(0, 100))
grid.arrange(g_AwaitingTime_1, g_AwaitingTime_2,ncol=2, top='AwaitingTime distributio
n')
```



Hide

```
#Let's look at the insurance of the patient to determine the no show
ggplot(data, aes(x=InsuranceDSC, fill=InsuranceDSC )) + geom_bar() + theme(axis.text.x
= element_text(angle = 45, hjust = 1))
```



Hide

```
#Perhaps some insurance plan have more "No-Show". Let's check:
tab_InsuranceDSC <- table(data$Status, data$InsuranceDSC)
addmargins(tab_InsuranceDSC)
```

```
         Commercial Medicaid Medicare Self.Pay   Sum
  No-show       2869     3534     1277     1426  9106
  show_up      39899    18366    21571     6279 86115
  Sum          42768    21900    22848     7705 95221
```

We can see that patients using medicaid make up 1/3rd of the no show

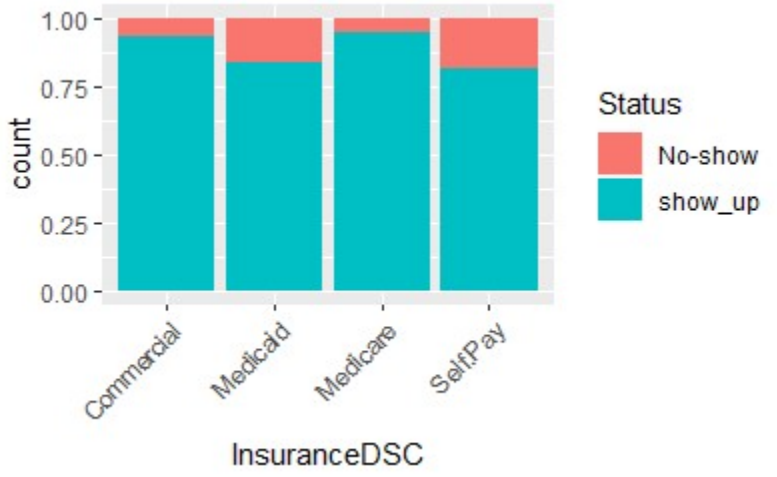
Hide

```
#probability table
prop.table(tab_InsuranceDSC,2)
```

```
         Commercial   Medicaid   Medicare   Self.Pay
  No-show 0.06708287 0.16136986 0.05589111 0.18507463
  show_up 0.93291713 0.83863014 0.94410889 0.81492537
```

Hide

```
ggplot(data, aes(x=InsuranceDSC, fill=Status )) + geom_bar(position="fill") + theme(ax
is.text.x = element_text(angle = 45, hjust = 1))
```



Hide

```
#Let's check if Whether or not patient with a known history of hypertension affects th
e no show
tab_HypertensionFLG <- table(data$HypertensionFLG, data$Status)
addmargins(tab_HypertensionFLG)
```

```
      No-show show_up   Sum
  0      6734   54404 61138
  1      2372   31711 34083
  Sum    9106   86115 95221
```

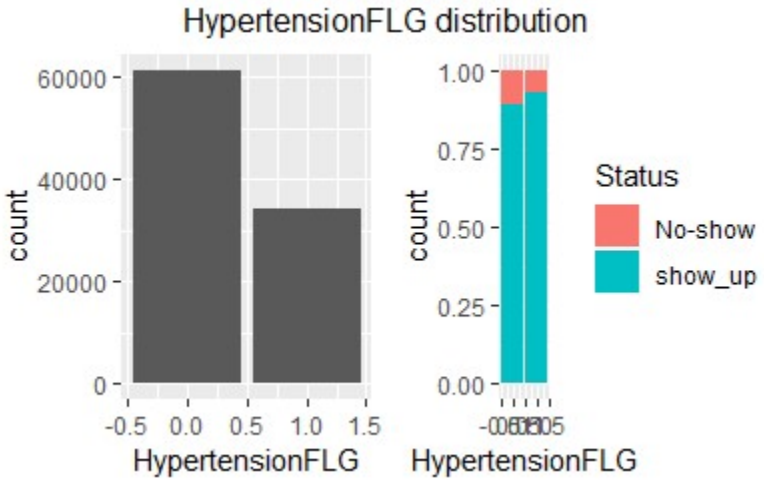
Hide

```
#probabability table
prop.table(tab_HypertensionFLG,2)
```

```
     No-show   show_up
  0 0.7395124 0.6317599
  1 0.2604876 0.3682401
```

Hide

```
#visualize
g_HypertensionFLG_1 <- ggplot(data, aes(x=HypertensionFLG, fill=HypertensionFLG)) + ge
om_bar(position="dodge")
g_HypertensionFLG_2 <- ggplot(data, aes(x=HypertensionFLG, fill=Status)) + geom_bar(po
sition="fill")
grid.arrange(g_HypertensionFLG_1, g_HypertensionFLG_2,ncol=2, top='HypertensionFLG dis
tribution')
```



74% of patients that do not have a history of hypertension are no show

Hide

```
#Let's check if Whether or not patient with a known history of asthma affects the no s
how
tab_AsthmaFLG <- table(data$AsthmaFLG, data$Status)
addmargins(tab_AsthmaFLG)
```

```
     No-show show_up   Sum
  0      8198   78057 86255
  1       908    8058  8966
  Sum    9106   86115 95221
```

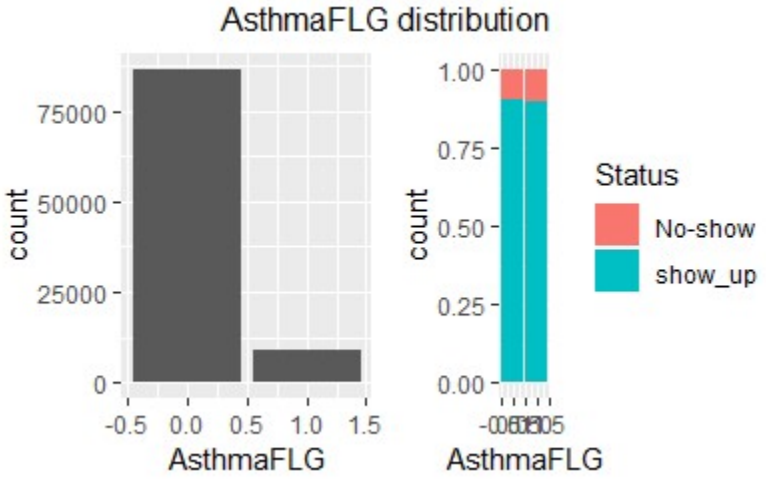
Hide

```
#probabability table
prop.table(tab_AsthmaFLG,2)
```

```
      No-show    show_up
  0 0.90028553 0.90642745
  1 0.09971447 0.09357255
```

Hide

```
#visualize
g_AsthmaFLG_1 <- ggplot(data, aes(x=AsthmaFLG, fill=AsthmaFLG)) + geom_bar(position="d
odge")
g_AsthmaFLG_2 <- ggplot(data, aes(x=AsthmaFLG, fill=Status)) + geom_bar(position="fil
l")
grid.arrange(g_AsthmaFLG_1, g_AsthmaFLG_2,ncol=2, top='AsthmaFLG distribution')
```



90% of patients that do not have a history of hypertension are no show

Hide

```
#Let's check if Whether or not patient with a known history of asthma affects the no s
how
tab_HeartDiseaseFLG <- table(data$HeartDiseaseFLG, data$Status)
addmargins(tab_HeartDiseaseFLG)
```

```
     No-show show_up   Sum
  0      8359   76473 84832
  1       747    9642 10389
  Sum    9106   86115 95221
```

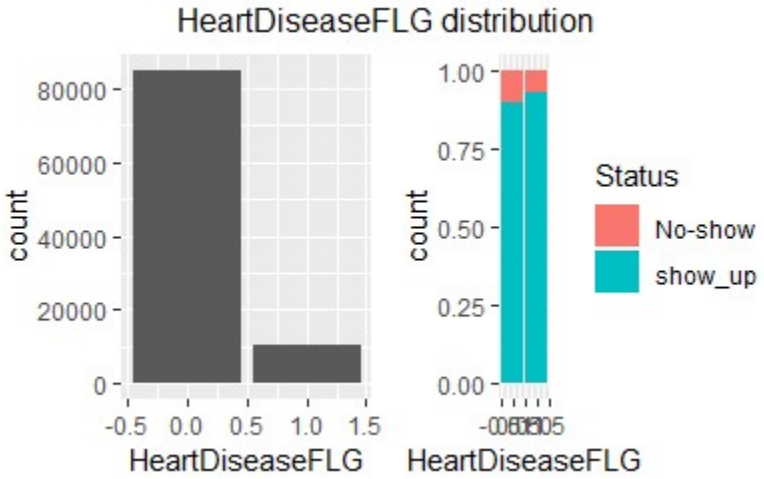
Hide

```
#probabability table
prop.table(tab_HeartDiseaseFLG,2)
```

```
      No-show    show_up
  0 0.91796618 0.88803344
  1 0.08203382 0.11196656
```

Hide

```
#visualize
g_HeartDiseaseFLG_1 <- ggplot(data, aes(x=HeartDiseaseFLG, fill=HeartDiseaseFLG)) + ge
om_bar(position="dodge")
g_HeartDiseaseFLG_2 <- ggplot(data, aes(x=HeartDiseaseFLG, fill=Status)) + geom_bar(po
sition="fill")
grid.arrange(g_HeartDiseaseFLG_1, g_HeartDiseaseFLG_2,ncol=2, top='HeartDiseaseFLG dis
tribution')
```



92% of patients that do not have a history of heart diseases are no show

Hide

```
#Let's check if Whether or not patient with a known history of Obese affects the no sh
ow
tab_ObeseFLG <- table(data$ObeseFLG, data$Status)
addmargins(tab_ObeseFLG)
```

```
     No-show show_up   Sum
  0      6892   63660 70552
  1      2214   22455 24669
  Sum    9106   86115 95221
```

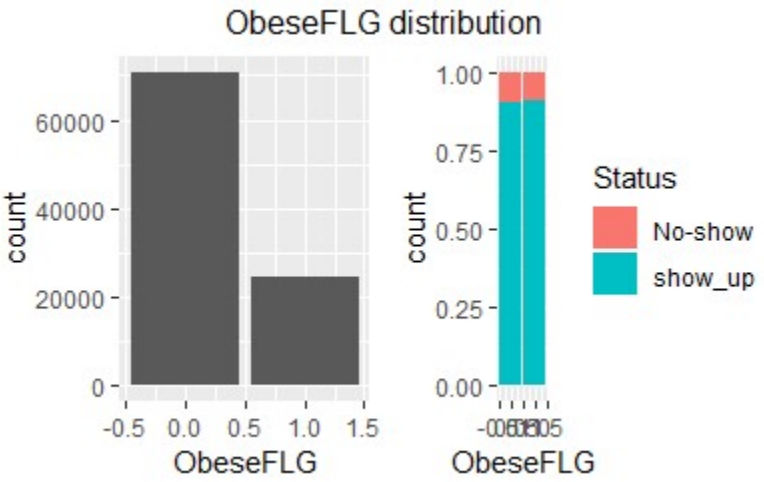
Hide

```
#probabability table
prop.table(tab_ObeseFLG,2)
```

```
     No-show   show_up
  0 0.7568636 0.7392440
  1 0.2431364 0.2607560
```

Hide

```
#visualize
g_ObeseFLG_1 <- ggplot(data, aes(x=ObeseFLG, fill=ObeseFLG)) + geom_bar(position="dodg
e")
g_ObeseFLG_2 <- ggplot(data, aes(x=ObeseFLG, fill=Status)) + geom_bar(position="fill")
grid.arrange(g_ObeseFLG_1, g_ObeseFLG_2,ncol=2, top='ObeseFLG distribution')
```



76% of patients that do not have a history of Obese are no show

Hide

```
#Let's check if Whether or not patient with a known history of Obese affects the no sh
ow
tab_DiabetesFLG <- table(data$DiabetesFLG, data$Status)
addmargins(tab_DiabetesFLG)
```

```
     No-show show_up   Sum
  0      9057   85641 94698
  1        49     474   523
  Sum    9106   86115 95221
```

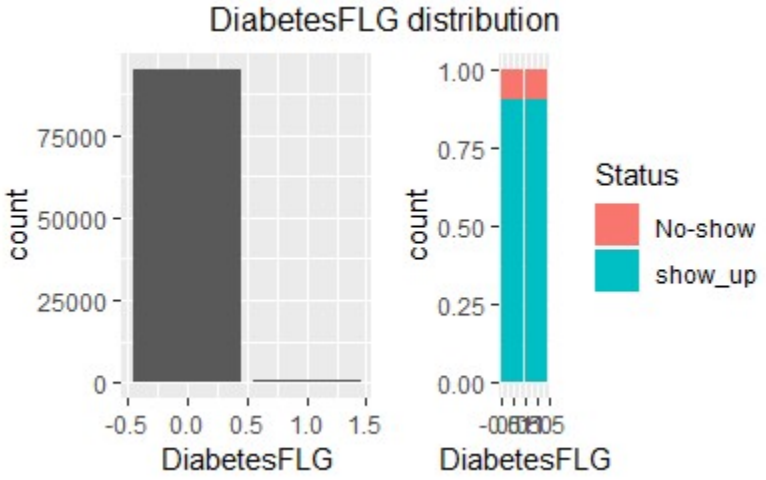
Hide

```
#probabability table
prop.table(tab_DiabetesFLG,2)
```

```
       No-show     show_up
  0 0.994618933 0.994495732
  1 0.005381067 0.005504268
```

Hide

```
#visualize
g_DiabetesFLG_1 <- ggplot(data, aes(x=DiabetesFLG, fill=DiabetesFLG)) + geom_bar(posit
ion="dodge")
g_DiabetesFLG_2 <- ggplot(data, aes(x=DiabetesFLG, fill=Status)) + geom_bar(position
="fill")
grid.arrange(g_DiabetesFLG_1, g_DiabetesFLG_2,ncol=2, top='DiabetesFLG distribution')
```



99.5% of patients that do not have a history of diabetes are no show

Hide

```
#Let's check if Whether or not patient with a known history of cancellation affects th
e no show
tab_CancelledLateFLG <- table(data$CancelledLateFLG, data$Status)
addmargins(tab_CancelledLateFLG)
```

```
     No-show show_up   Sum
  0      9106   70894 80000
  1         0   15221 15221
  Sum    9106   86115 95221
```

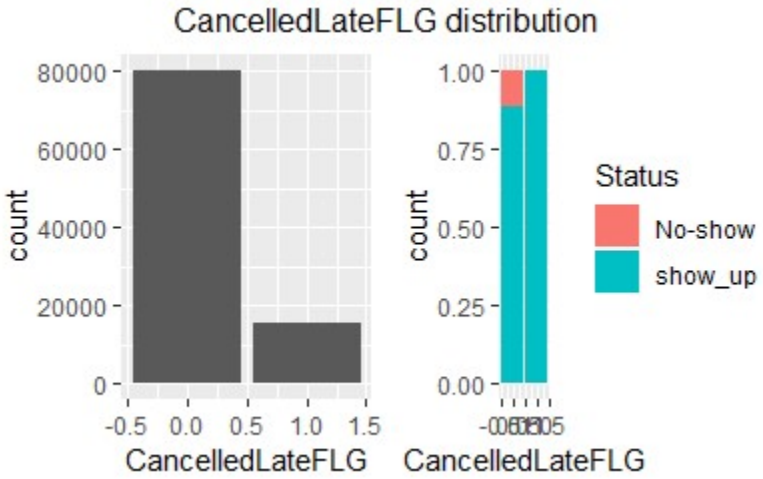
Hide

```
#probabability table
prop.table(tab_CancelledLateFLG,2)
```

```
    No-show  show_up
  0 1.000000 0.823248
  1 0.000000 0.176752
```

Hide

```
#visualize
g_CancelledLateFLG_1 <- ggplot(data, aes(x=CancelledLateFLG, fill=CancelledLateFLG)) +
geom_bar(position="dodge")
g_CancelledLateFLG_2 <- ggplot(data, aes(x=CancelledLateFLG, fill=Status)) + geom_bar
(position="fill")
grid.arrange(g_CancelledLateFLG_1, g_CancelledLateFLG_2,ncol=2, top='CancelledLateFLG
distribution')
```



All the no shows dis not cancel within 24 hours of this appointment

Hide

```
#Let's check if Whether or not patient with a known history of in patient visits affec
ts the no show
tab_IpVisitsNBR <- table(data$IpVisitsNBR, data$Status)
addmargins(tab_IpVisitsNBR)
```

```
     No-show show_up   Sum
  0      8918   84592 93510
  1       169    1328  1497
  2        16     142   158
  3         1      35    36
  4         1      13    14
  5         1       2     3
  6         0       3     3
  Sum    9106   86115 95221
```

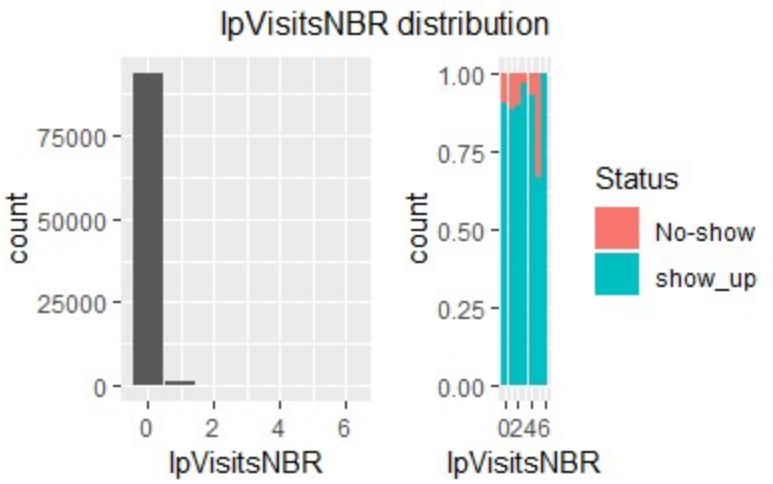
Hide

```
#probabability table
prop.table(tab_IpVisitsNBR,2)
```

```
        No-show      show_up
  0 9.793543e-01 9.823143e-01
  1 1.855919e-02 1.542124e-02
  2 1.757083e-03 1.648958e-03
  3 1.098177e-04 4.064333e-04
  4 1.098177e-04 1.509609e-04
  5 1.098177e-04 2.322476e-05
  6 0.000000e+00 3.483714e-05
```

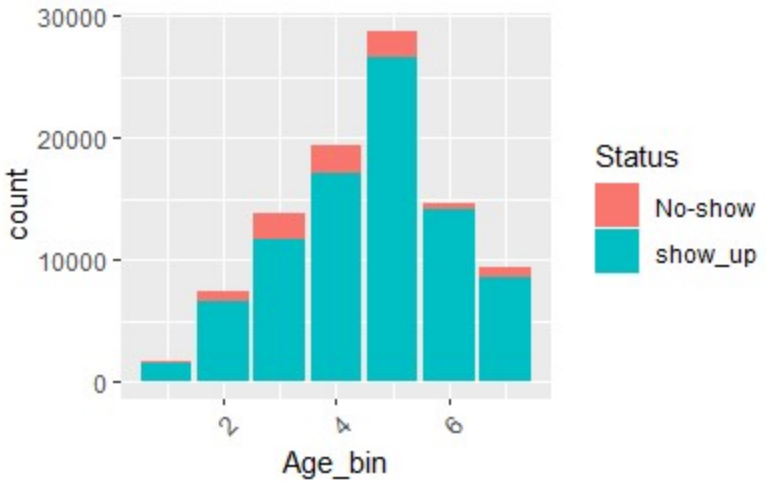
Hide

```
#visualize
g_IpVisitsNBR_1 <- ggplot(data, aes(x=IpVisitsNBR, fill=IpVisitsNBR)) + geom_bar(posit
ion="dodge")
g_IpVisitsNBR_2 <- ggplot(data, aes(x=IpVisitsNBR, fill=Status)) + geom_bar(position
="fill")
grid.arrange(g_IpVisitsNBR_1, g_IpVisitsNBR_2,ncol=2, top='IpVisitsNBR distribution')
```



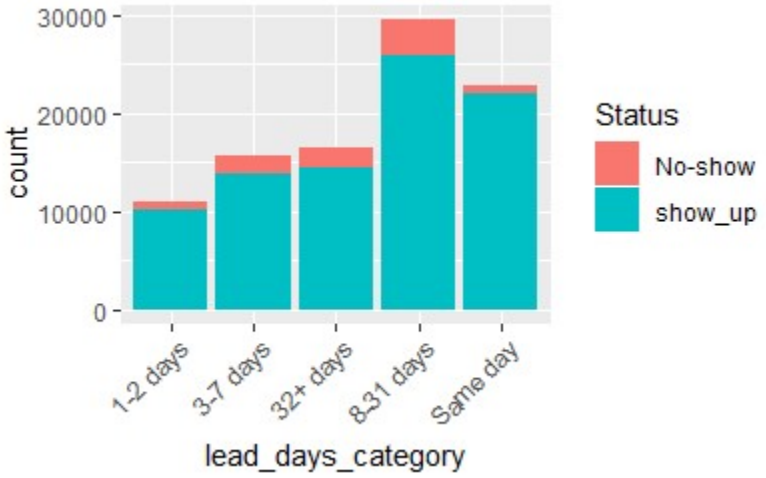
Hide

```
#data$appointment_time= strftime(data$AppointmentDTS, format="%H:%M:%S")
#data$appointment_date=strftime(data$AppointmentDTS, format = "%Y-%m-%d")
#data$scheduled_day=ymd(data$appointment_date) - data$ApptLagNBR
# Create lead_days_category column
data$lead_days_category=data$ApptLagNBR
data$lead_days_category=ifelse(data$ApptLagNBR<=0,"Same day",
                               ifelse(data$ApptLagNBR>0 & data$ApptLagNBR<=2,"1-2 d
ays",
                                      ifelse(data$ApptLagNBR>2 & data$ApptLagNBR<=
7,"3-7 days",
                                             ifelse(data$ApptLagNBR>7 & data$ApptLa
gNBR<=31,"8-31 days","32+ days"))))
data$Age_bin=ifelse(data$AgeNBR<=0 & data$AgeNBR<=5,1,
                               ifelse(data$AgeNBR>5 & data$AgeNBR<=17,2,
                                      ifelse(data$AgeNBR>17 & data$AgeNBR<=30,3,
                                             ifelse(data$AgeNBR>30 & data$AgeNBR<=4
5,4,
                                                    ifelse(data$AgeNBR>45 & data$AgeNBR<
=65,5,
                                             ifelse(data$AgeNBR>65 & data$AgeNBR<=8
0,6,7))))))
#age distribution
ggplot(data, aes(x=Age_bin, fill=Status )) + geom_bar() + theme(axis.text.x = element_
text(angle = 45, hjust = 1))
```



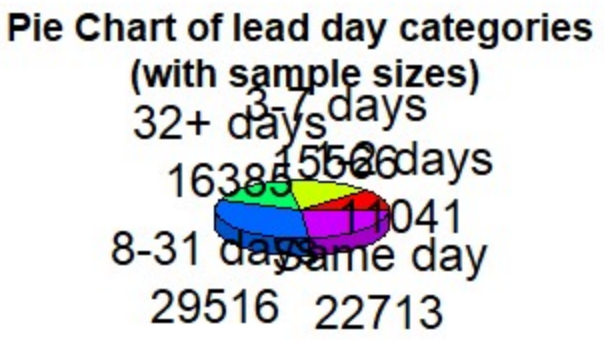
Hide

```
ggplot(data, aes(x=lead_days_category, fill=Status )) + geom_bar() + theme(axis.text.x
= element_text(angle = 45, hjust = 1))
```



Hide

```
# Pie Chart from data frame with Appended Sample Sizes
library(plotrix)
mytable <- table(data$lead_days_category)
lbls <- paste(names(mytable), "\n", mytable, sep="")
pie3D(mytable, labels = lbls,
   main="Pie Chart of lead day categories\n (with sample sizes)")
```



Predictive Modeling With new features ready to go and missing values taken care of, let’s apply machine learning. We’ll start by preparing the data, followed by splitting it into testing and training set, modeling and finally, evaluating our results.

Hide

```
library(dplyr)
data %>%
  select_if(function(x) any(is.na(x))) %>%
  summarise_each(funs(sum(is.na(.)))) -> extra_NA
```

```
`summarise_each()` is deprecated.
Use `summarise_all()`, `summarise_at()` or `summarise_if()` instead.
To map `funs` over all variables, use `summarise_all()`
```

Hide

```
#==================================
# [2] Predictive Modeling
#==================================
data$V1=NULL
```

```
Adding new column 'V1' then assigning NULL (deleting it).
```

Hide

```
data$lead_days_category=as.factor(data$lead_days_category)
data$Dayoftheweek=as.factor(data$Dayoftheweek)
#data$AppointmentID=NULL
#data$PatientID=NULL
data$Rand=NULL
```

```
Adding new column 'Rand' then assigning NULL (deleting it).
```

Hide

```
#data$ClinicNM=NULL
#data$AppointmentDTS=NULL
#data$AppointmentMonthNBR=as.numeric(data$AppointmentMonthNBR)
data$Age_bin=as.factor(data$Age_bin)
set.seed(1234)
split_data <- createDataPartition(data$Status, p = 0.7, list = FALSE)
train_data <- data[split_data,]
test_data <- data[-split_data,]
# Logistic Regression Model
no_showLog = glm(NoShowFLG ~ AppointmentMonthNBR + ClinicNM + AppointmentWeekdayNBR+ A
ppointmentHourNBR + Age_bin + AgeNBR + EdVisitsNBR, data=train_data, family=binomial)
summary(no_showLog)
```

```
Call:
glm(formula = NoShowFLG ~ AppointmentMonthNBR + ClinicNM + AppointmentWeekdayNBR +
    AppointmentHourNBR + Age_bin + AgeNBR + EdVisitsNBR, family = binomial,
    data = train_data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.9407  -0.5273  -0.3970  -0.2917   2.9284

Coefficients:
                      Estimate Std. Error z value Pr(>|z|)
(Intercept)          -3.059175   0.138508 -22.087  < 2e-16 ***
AppointmentMonthNBR   0.020692   0.004032   5.131 2.88e-07 ***
ClinicNMB             0.919064   0.041813  21.980  < 2e-16 ***
ClinicNMC             0.944908   0.046858  20.165  < 2e-16 ***
ClinicNMD             0.809493   0.063162  12.816  < 2e-16 ***
ClinicNME             1.055576   0.045765  23.065  < 2e-16 ***
AppointmentWeekdayNBR -0.029217   0.009645  -3.029 0.002451 **
AppointmentHourNBR    -0.020204   0.004938  -4.092 4.28e-05 ***
Age_bin2               0.692206   0.118902   5.822 5.83e-09 ***
Age_bin3               1.203013   0.116883  10.292  < 2e-16 ***
Age_bin4               1.120602   0.121471   9.225  < 2e-16 ***
Age_bin5               0.853088   0.131019   6.511 7.46e-11 ***
Age_bin6               0.533290   0.147392   3.618 0.000297 ***
Age_bin7               0.516990   0.122037   4.236 2.27e-05 ***
AgeNBR                -0.015357   0.001227 -12.517  < 2e-16 ***
EdVisitsNBR            0.037659   0.006280   5.997 2.01e-09 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 42046  on 66655  degrees of freedom
Residual deviance: 39933  on 66640  degrees of freedom
AIC: 39965

Number of Fisher Scoring iterations: 6
```

Hide

```
# Make predictions on training set
predictTrain = predict(no_showLog, type="response")
# Analyze predictions
summary(predictTrain)
```

```
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
0.01104 0.05205 0.08584 0.09564 0.13814 0.98675 
```

Hide

```
tapply(predictTrain, train_data$NoShowFLG, mean)
```

```
         0          1 
0.09259411 0.12444460 
```

Hide

```
# Confusion matrix for threshold of 0.5
table(train_data$NoShowFLG, predictTrain > 0.5)
```

```
   FALSE  TRUE 
  0 60275     6 
  1  6375     0 
```

Hide

```
#==================================
# [2] Predictive Modeling
#==================================
# [2.1] Remove unecessary columns, change var types to factor
data.prep = data[,c(-1,-2,-3,-5,-37:-44)]
data.prep$ClinicNM = as.factor(data.prep$ClinicNM)
data.prep$Target = as.factor(data.prep$NoShowFLG)
data.prep$SexFLG = NULL
data.prep$AgeNBR=NULL
data.prep$InsuranceDSC = NULL
data.prep$NoShowFLG = NULL
```

Hide

```
data.prep$NoshowRate24NBR=as.factor(data.prep$NoshowRate24NBR)


data.prep$CancelledLateFLG=as.factor(data.prep$CancelledLateFLG)
data.prep$NewPatient=as.factor(data.prep$NewPatient)
data.prep$Noshow24NBR=as.factor(data.prep$Noshow24NBR)
data.prep$DiabetesFLG=as.factor(data.prep$DiabetesFLG)
data.prep$ObeseFLG=as.factor(data.prep$ObeseFLG)
data.prep$HeartDiseaseFLG=as.factor(data.prep$HeartDiseaseFLG)
data.prep$AsthmaFLG=as.factor(data.prep$AsthmaFLG)
data.prep$HypertensionFLG=as.factor(data.prep$HypertensionFLG)
data.prep$EmailFLG=as.factor(data.prep$EmailFLG)
data.prep$LivesInApartmentFLG=as.factor(data.prep$LivesInApartmentFLG)
data.prep$SingleFLG=as.factor(data.prep$SingleFLG)
data.prep$HispanicFLG=as.factor(data.prep$HispanicFLG)
data.prep$Age_bin=as.factor(data.prep$Age_bin)
str(data.prep)
```

Hide

```
#Classification tree (rpart)
set.seed(1234)
split_data <- createDataPartition(data$Status, p = 0.7, list = FALSE)
train_data <- data[split_data,]
test_data <- data[-split_data,]

fitControl <- trainControl(method = "cv",
                           number = 5,
                           #savePredictions="final",
                           summaryFunction = twoClassSummary,
                           classProbs = TRUE
                           )
#We are going to upsample the "No.Show" class, so there will be the same number of cla
sses of each type:

##https://topepo.github.io/caret/subsampling-for-class-imbalances.html
train_data <- upSample(train_data[, setdiff(names(data), 'Status')], train_data$Statu
s, yname="Status")
table(train_data$Status)
```

Hide

```
# [4.2] Split into training and testing set
set.seed(41)
splitr = sample.split(data.prep$Target, SplitRatio = 0.7)
train  = subset(data.prep,splitr == TRUE)
test   = subset(data.prep,splitr == FALSE)
# [4.3] Set cross validation parameters
fit.control = trainControl(method="cv",number=3,
                           classProbs = TRUE,summaryFunction = twoClassSummary)
```

Hide

```
##Step 4: Classification
library(mlbench)
library(partykit)
library(rpart.plot)
library(RWeka)


##(a)using classification with Decision Tree.
# Build a decision tree model.
library(rpart) ## recursive partitioning
m =  rpart(Target~ ., data = train, cp=0)
pfit= prune(m, cp=m$cptable[9,"CP"])

prp(pfit,type=1,extra=100,fallen.leaves=F,shadow.col="darkgray",box.col=rgb(0.8,0.9,0.
8))
```

Hide

```
set.seed(38)
xgb.model = train(Target ~ .,data=train,method="xgbTree",metric="ROC",
                  tuneGrid=xgb.grid, trControl=fit.control)
```