Medical Appointment No Shows

Chris-FR

20 Feb. 2019

Introduction

For the second project, the EDX PH125.9x capstone course, we will try to predict when a patient makes a doctor appointment, if they will show up or not. The cost of heath care is expensive and continues to rise and securing an appointment with a specialist can take months, we need to look at the consequenses and impacts that a patient who does not show to an appointment has on the system. A high "no show" rate (20% in this dataset) is an important source of optimization. Furthermore, a no-show patient denies another patient an appointment who really needs it. In this report, we will:
- do a quick data exploratory analysis,
- split our data between a train and a validation datasets,
- build different models,
- test the accuracy model on our validation dataset.

The original dataset of this project can be found on kaggle's page: Medical Appointment No Shows.

Data exploration and visualization

Loading data

```
if(!require(tidyverse)) install.packages("tidyverse", repos = "http://cran.us.r-project.org")
if(!require(caret)) install.packages("caret", repos = "http://cran.us.r-project.org")
if(!require(gridExtra)) install.packages("gridExtra", repos = "http://cran.us.r-project.org")
if(!require(summarytools)) install.packages("summarytools", repos = "http://cran.us.r-project.org")

# load the ZIP file using the read_csv function of the names(aptds) package
aptds = read_csv("noshowappointments.zip")
# rename some of the columns
names(aptds)[names(aptds) == 'Hipertension'] <- 'Hypertension'
names(aptds)[names(aptds) == 'Handcap'] <- 'Disability'
names(aptds)[names(aptds) == 'Scholarship'] <- 'SocialAid'
names(aptds)[names(aptds) == 'SMS_received'] <- 'SMS'
names(aptds)[names(aptds) == 'No-show'] <- 'NoShow'
```

The dataset contains **110527** records of **14** features. It contains **0** NAs.

	Type	Label
PatientId	numeric	Identification of a patient
AppointmentID	integer	Identification of each appointment
Gender	character	Male or Female
ScheduledDay	POSIXct	The day someone called or registered the appointment
AppointmentDay	POSIXct	Appointment Day
Age	integer	Patient age
Neighbourhood	character	Where the appointment takes place
SocialAid	integer	True or False, Social Aid
Hypertension	integer	True or False
Diabetes	integer	True or False
Alcoholism	integer	True or False
Disability	integer	1 to 4

	Type	Label
SMS	integer	1 or more messages sent to the patient.
NoShow	character	True or False - The patient came to his appointment or not

Data quick check / validation

The data has been loaded with the default parameters. We will convert **factor** columns after the checks.

Patients / Appointments

There are 62299 unique patients and 110527 unique appointments (there are no duplicated appointements).

Patients go to *1* neighbourhood only : the PatientId may not be a *global* Ids like the SSN, ... but hospital / location Ids. This may be interesting as we may add Patients information in our model as it will be available when this patient schedules an appointement at the hospital / clinic / ...

Number of appointements by patient:

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   1.000   1.000   1.000   1.774   2.000  88.000
```

Only *248* patients had more than 10 appointments and most patients have 1 or 2 appointments.

Numeric column summary

```
##       Age            SocialAid        Hypertension       Diabetes
##  Min.   : -1.00   Min.   :0.00000   Min.   :0.0000   Min.   :0.00000
##  1st Qu.: 18.00   1st Qu.:0.00000   1st Qu.:0.0000   1st Qu.:0.00000
##  Median : 37.00   Median :0.00000   Median :0.0000   Median :0.00000
##  Mean   : 37.09   Mean   :0.09827   Mean   :0.1972   Mean   :0.07186
##  3rd Qu.: 55.00   3rd Qu.:0.00000   3rd Qu.:0.0000   3rd Qu.:0.00000
##  Max.   :115.00   Max.   :1.00000   Max.   :1.0000   Max.   :1.00000
##    Alcoholism       Disability           SMS
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.000
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:0.000
##  Median :0.0000   Median :0.00000   Median :0.000
##  Mean   :0.0304   Mean   :0.02225   Mean   :0.321
##  3rd Qu.:0.0000   3rd Qu.:0.00000   3rd Qu.:1.000
##  Max.   :1.0000   Max.   :4.00000   Max.   :1.000
```

We have some invalid values in the **Age** column : -1 and the max is high : 115.

PatientId	AppointmentID	Age	AppointmentDay	NoShow
4.659432e+14	5775010	-1	2016-06-06	No
2.342836e+11	5751563	102	2016-06-02	No
9.762948e+14	5651757	102	2016-05-03	No
3.196321e+13	5562812	115	2016-05-16	Yes
3.196321e+13	5700278	115	2016-05-19	Yes
3.196321e+13	5700279	115	2016-05-19	Yes
3.196321e+13	5744037	115	2016-05-30	No
7.482346e+14	5717451	115	2016-06-03	No

Using the `freq` function from the `summarytools` package, Hypertension, Diabetes, Alcoholis only contains 0 and 1.

SMS reminders are sent in 32% of the appointments:

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
0	75045	67.90	67.90	67.90	67.90
1	35482	32.10	100.00	32.10	100.00
<NA>	0			0.00	100.00

Disability contains 5 different values:

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
0	108286	97.97	97.97	97.97	97.97
1	2042	1.85	99.82	1.85	99.82
2	183	0.17	99.99	0.17	99.99
3	13	0.01	100.00	0.01	100.00
4	3	0.00	100.00	0.00	100.00
<NA>	0			0.00	100.00

Character columns

Gender contains 2 values. 65% of the appointments are for Female patients.

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
F	71840	65.00	65.00	65.00	65.00
M	38687	35.00	100.00	35.00	100.00
<NA>	0			0.00	100.00

NoShow contains 2 values. 20% of the appointements are 'No Show'.

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
No	88208	79.81	79.81	79.81	79.81
Yes	22319	20.19	100.00	20.19	100.00
<NA>	0			0.00	100.00

Date columns

There are 2 POSIXct columns:

```
##   ScheduledDay                  AppointmentDay
##  Min.   :2015-11-10 07:13:56   Min.   :2016-04-29 00:00:00
##  1st Qu.:2016-04-29 10:27:01   1st Qu.:2016-05-09 00:00:00
##  Median :2016-05-10 12:13:17   Median :2016-05-18 00:00:00
##  Mean   :2016-05-09 07:49:15   Mean   :2016-05-19 00:57:50
##  3rd Qu.:2016-05-20 11:18:37   3rd Qu.:2016-05-31 00:00:00
##  Max.   :2016-06-08 20:07:23   Max.   :2016-06-08 00:00:00
```

The dataset contains 27 appointments dates. The dataset contains 111 scheduled dates. The scheduled columns contains the time.

Monday to Wednesday are the main appointment days. Thursday and friday have a little bit less appointments. There are very few appointment on Saturdays.

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
Wednesday	25867	23.40	23.40	23.40	23.40
Tuesday	25640	23.20	46.60	23.20	46.60

	Freq	% Valid	% Valid Cum.	% Total	% Total Cum.
Monday	22715	20.55	67.15	20.55	67.15
Friday	19019	17.21	84.36	17.21	84.36
Thursday	17247	15.60	99.96	15.60	99.96
Saturday	39	0.04	100.00	0.04	100.00
<NA>	0			0.00	100.00

Delta between Sheduled and Appointment dates :

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   -6.00    0.00    4.00   10.18   15.00  179.00
```

We have **5** invalid records (delta < 0). The average delta is **10.2** days and the max delta is **179** days

Data conversion / clean up

We will :
- set the Age value -1 to 0,
- convert : Gender, Neighbourhood, SocialAid, Hypertension, Diabetes, Alcoholism, Disability, SMS and NoShow to factors,
- add the appointment week day (1 is Monday),
- add the delta between the scheduled and appointment days and set the negative deltas to 0,
- create an AgeBreak column (5 years intervals).

We will try to add 2 patients specific data.
I will consider that i have access to the previous Patient data when he takes en appointment.
We will add
- the Apt number (Apt rank : this is his first, second, third, ... appointment),
- if the patient miss his **last** appointment,
- the percentage of previously missed appointement.

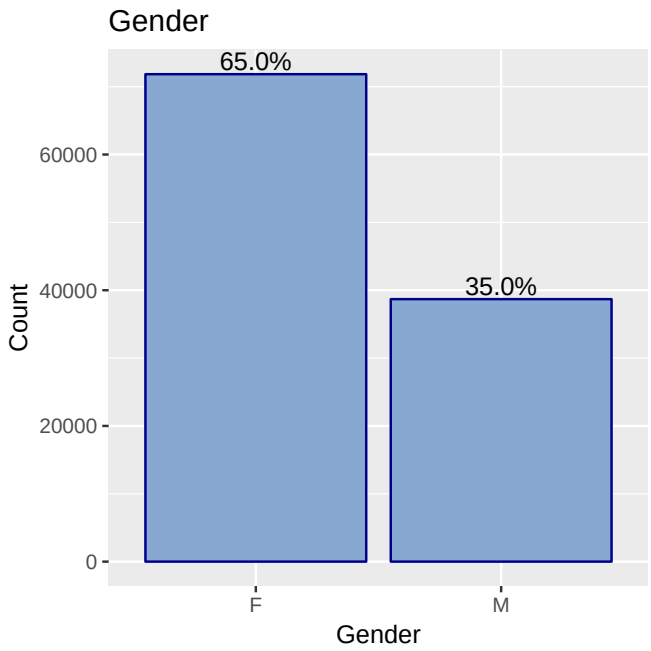
Here is an exemple (patient 762753796133238) of these new columns:

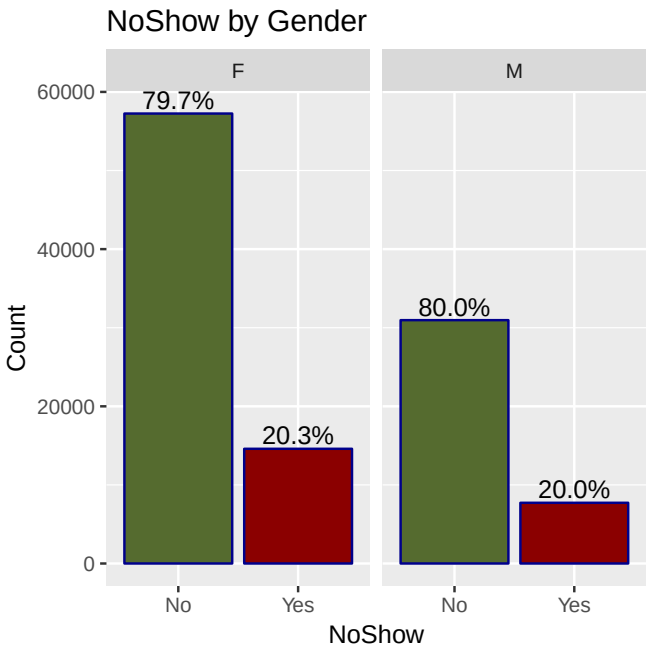
AppointmentDay	Gender	NoShow	AptNumber	PreviousNoShow	PreviousNoShowCount	PreviousPercentNoShow
2016-05-10	M	Yes	1	No	0	0.00
2016-05-10	M	Yes	2	Yes	1	1.00
2016-05-10	M	Yes	3	Yes	2	1.00
2016-05-20	M	No	4	Yes	3	1.00
2016-05-20	M	No	5	No	3	0.75
2016-05-20	M	No	6	No	3	0.60

The first record have no history (Previous NoShow = 0). The second ahe 0 or 1 (0% or 100%).

Data exploration

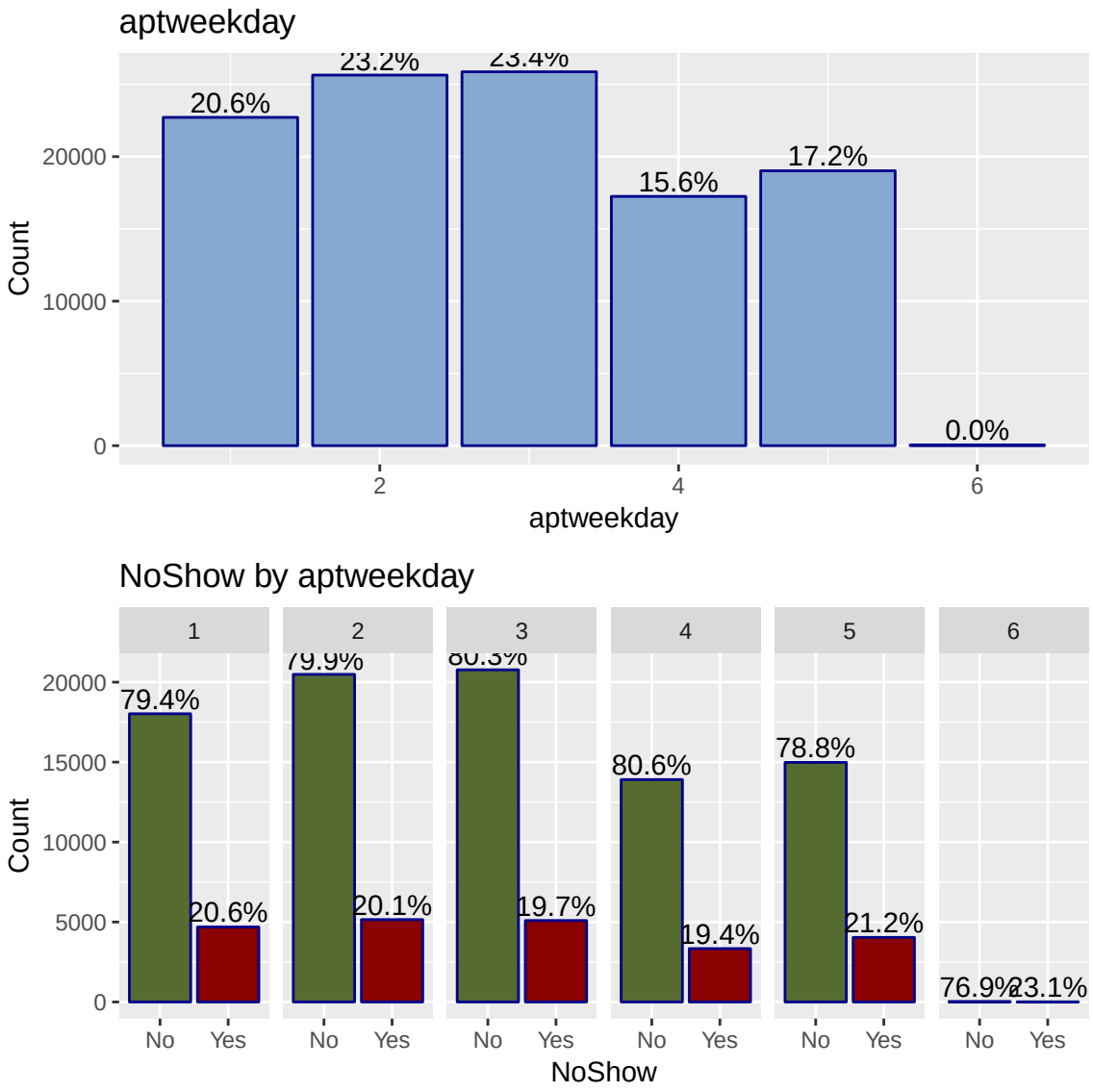
Gender





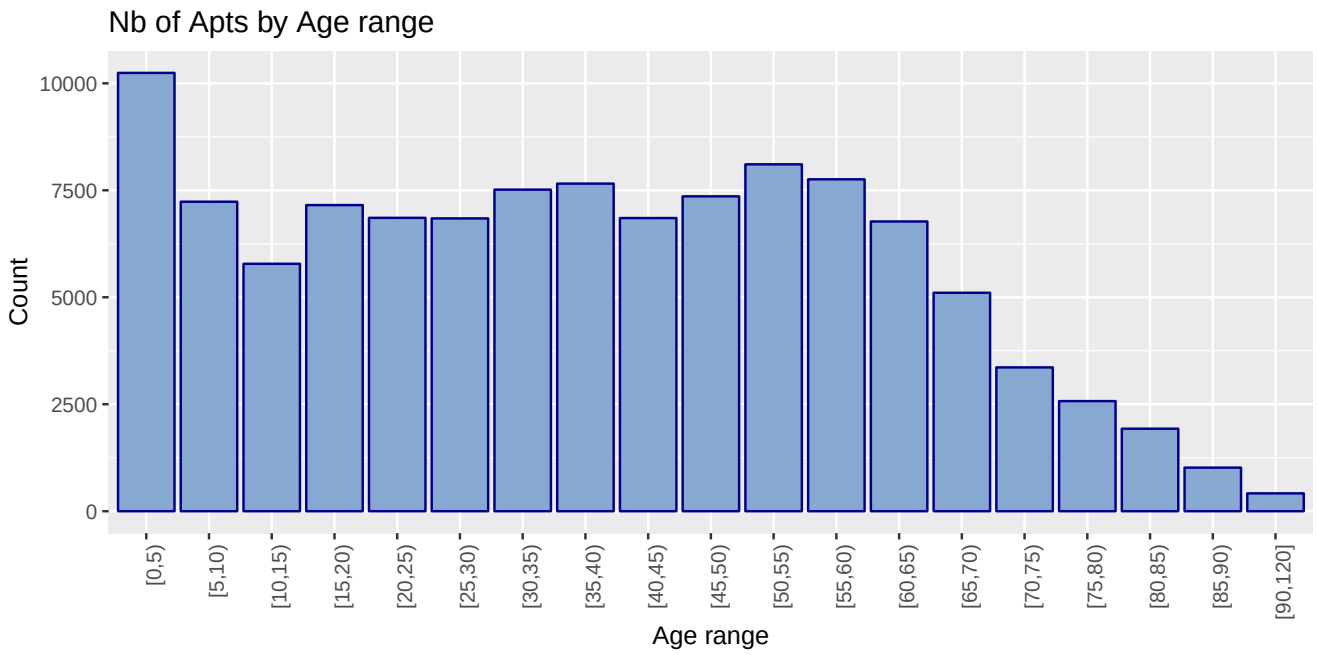
Gender does not seem to have an impact on the No-Show rate. Both Male and Female patients have a 20% No-Show rate (the average).

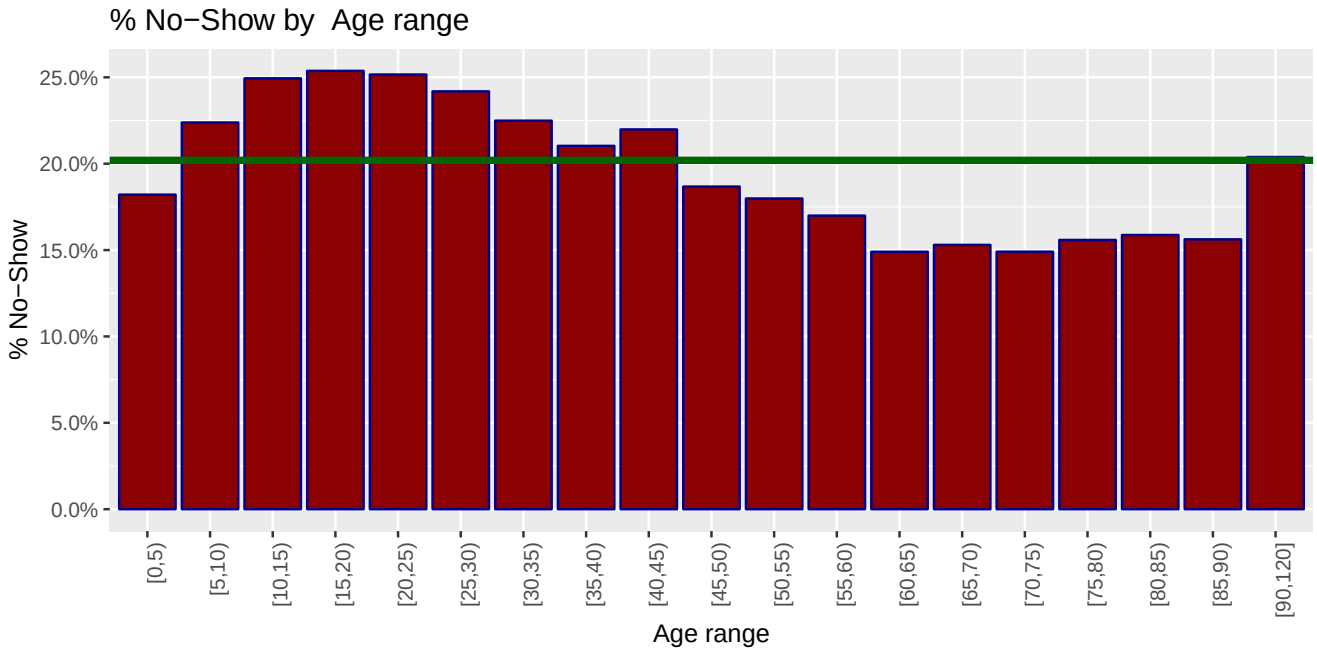
Appointment weekday



Even if there are less appointements on Thursday and Friday, the evrage No-Show does not vary a lot (less than 1% compare to the average).

Age

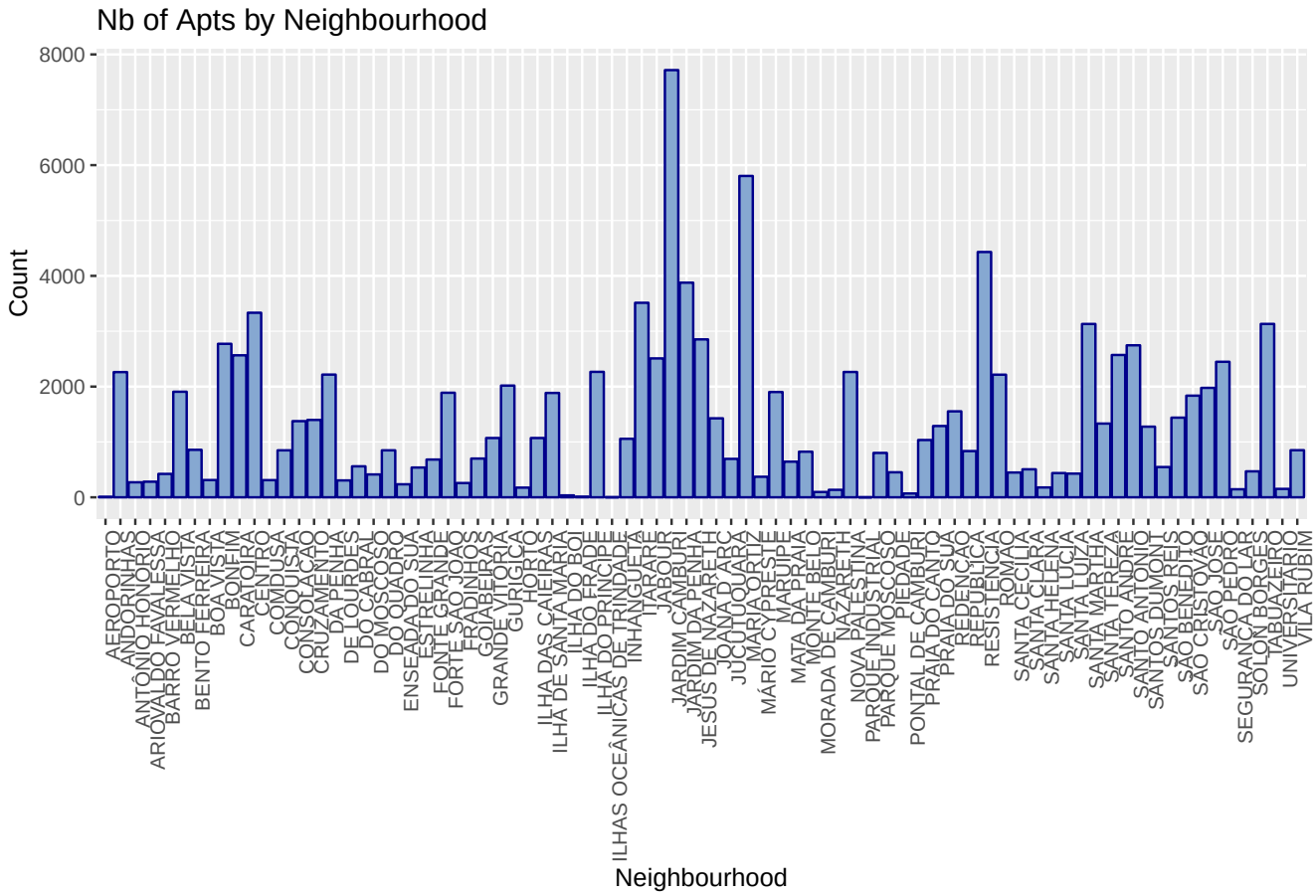


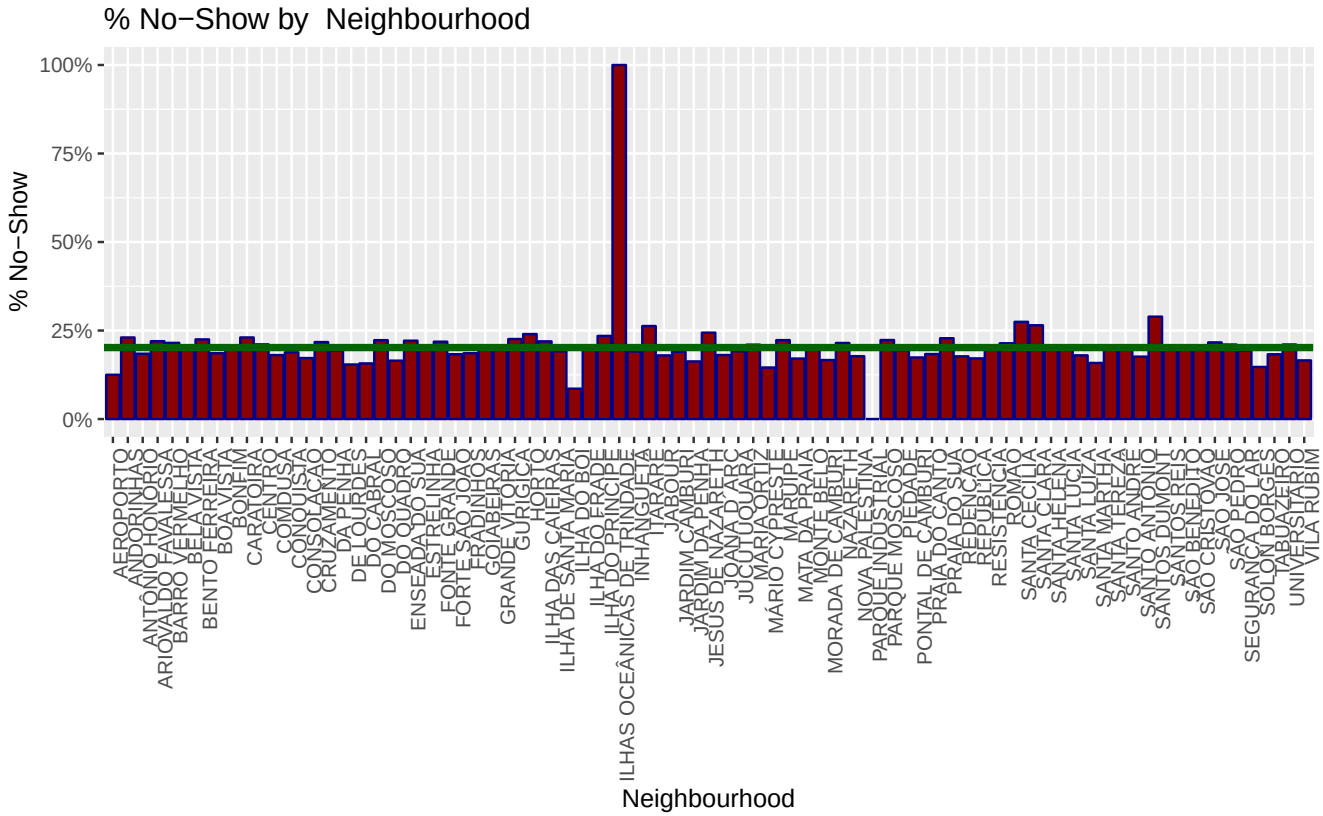


There seems to be 3 differents groups :
- young children (less than 5 years) have a lower No-Show rate than average. Their parents bring them to the appointments,
- patients from 5 to 45 years : have a higher No-Show rate than the average,
- patients older than 45 years have a lower No-Show rate.

For the patients older than 90 years have an average no-show rate but the number of associated appointments is low.

Neighbourhood



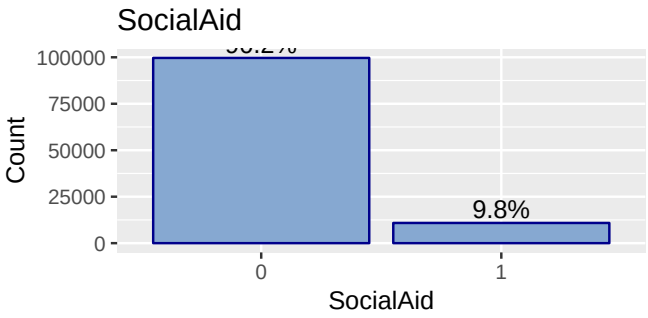


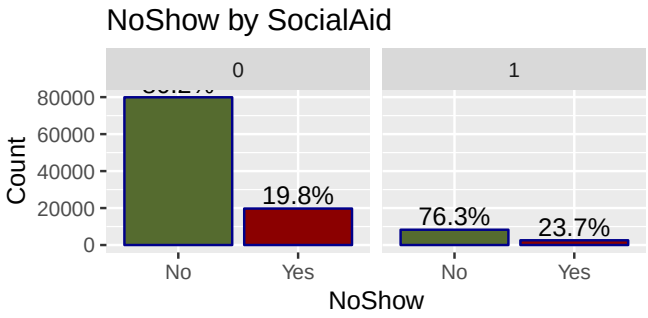
There are a few variations depending of the neighbourhood. There are 2 outlier (100%, 0% NoShow) but the number of appointments is very low in both cases.

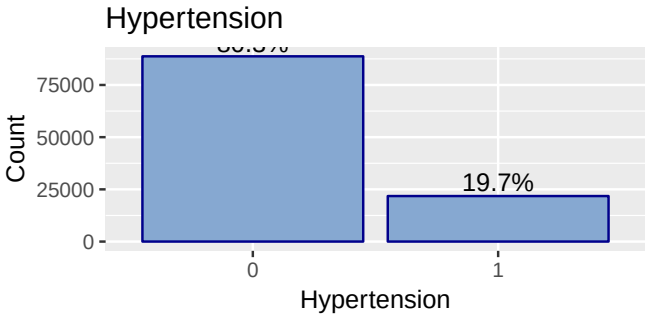
Neighbourhood	Total	PercentNoShow
PARQUE INDUSTRIAL	1	0.000000
ILHA DO BOI	35	8.571429

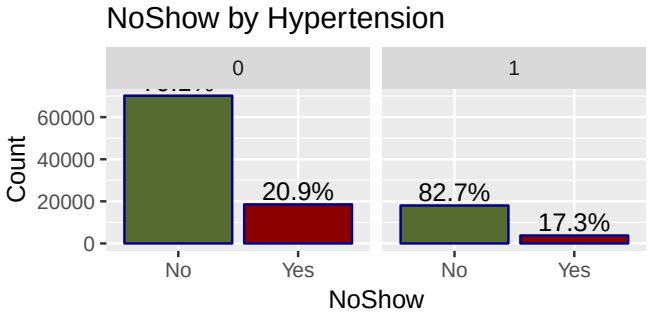
Neighbourhood	Total	PercentNoShow
ILHAS OCEÂNICAS DE TRINDADE	2	100.0000
SANTOS DUMONT	1276	28.9185

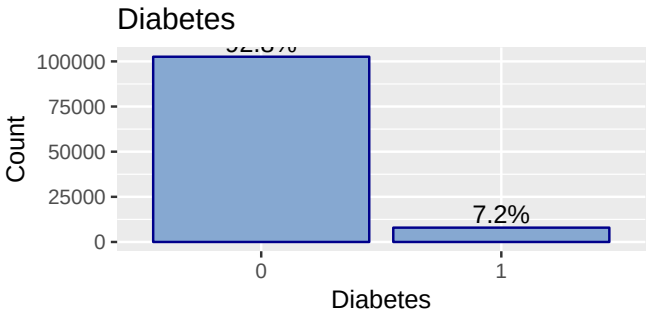
SocialAid / Hypertension / Diabetes / Alcoholism / Disability

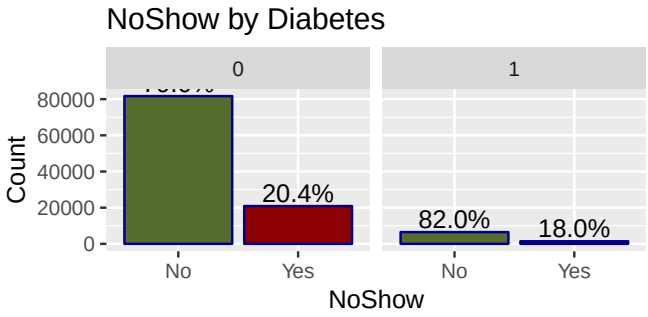


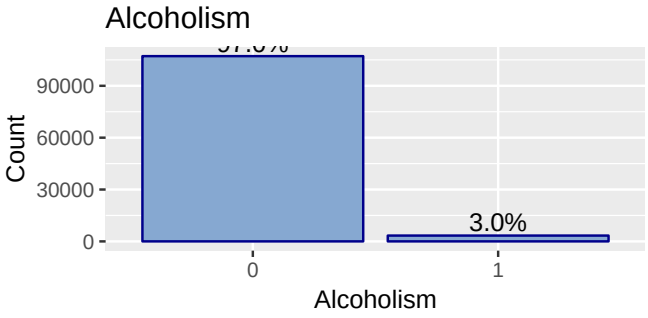


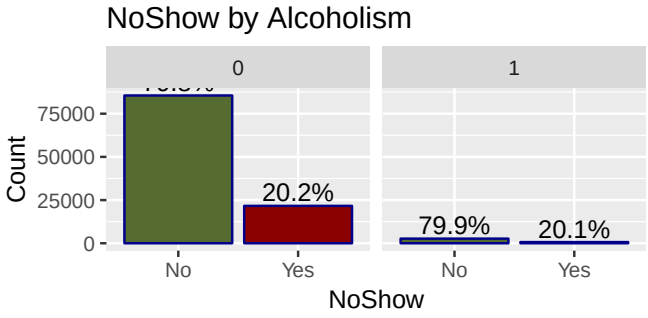




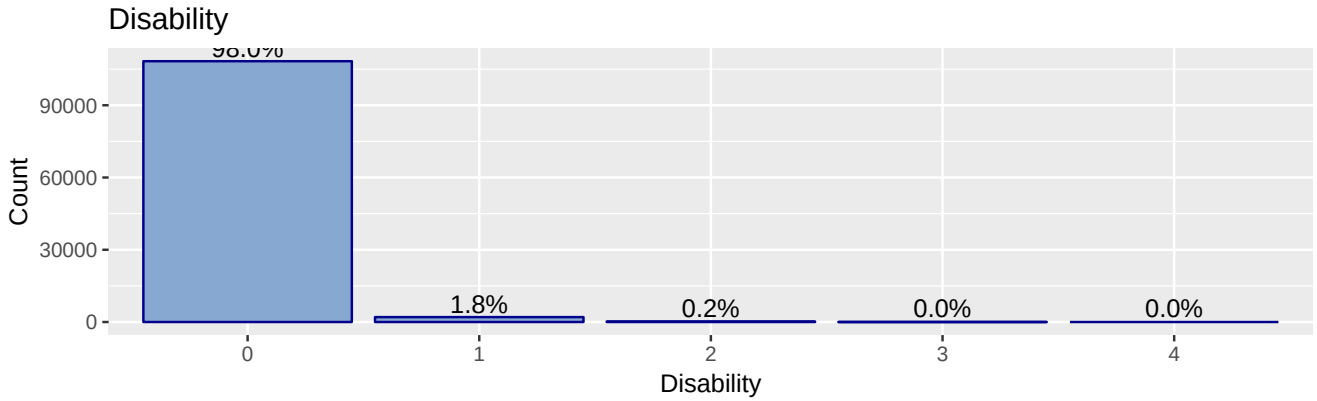


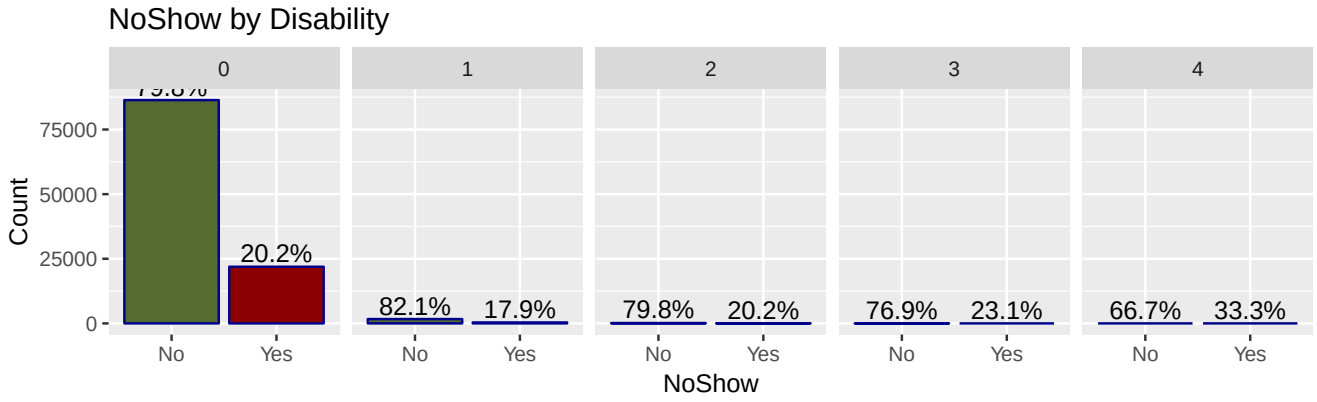






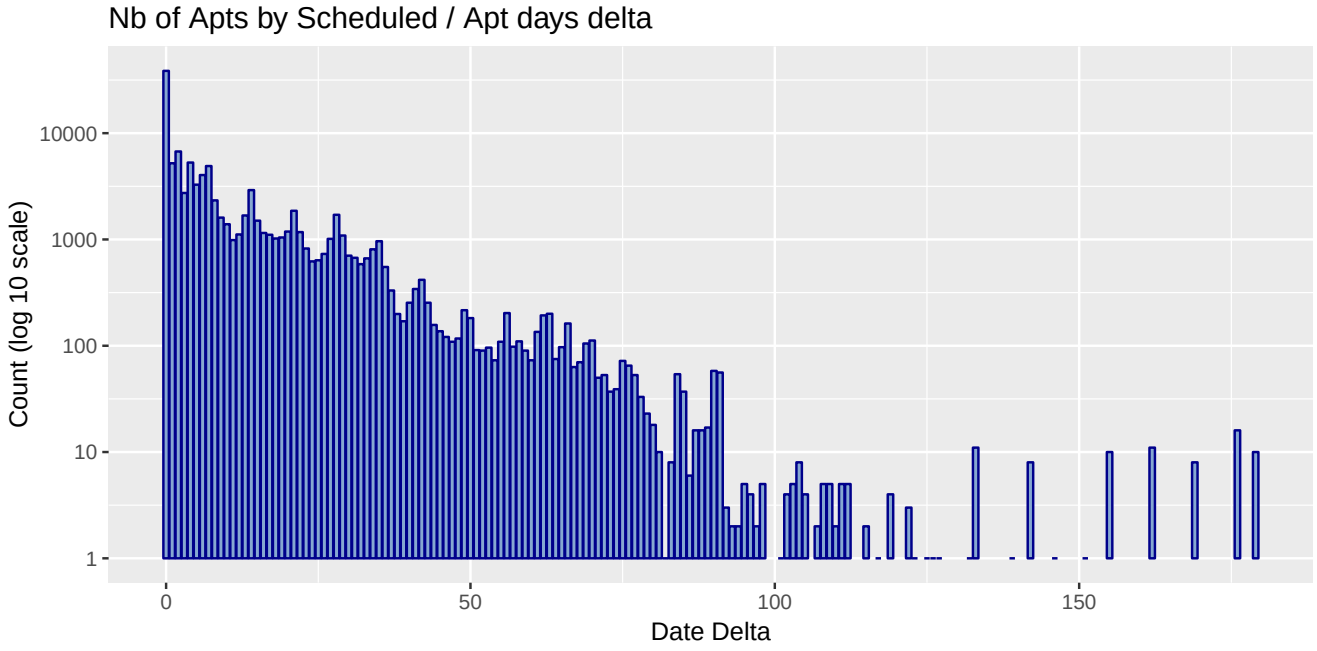
Patients with Hypertension or Diabetes tends to have a lower No-Show percentage than average.
Alcoholism does not seem to have an impact.
patients with social aid have a higher No-Show percentage.

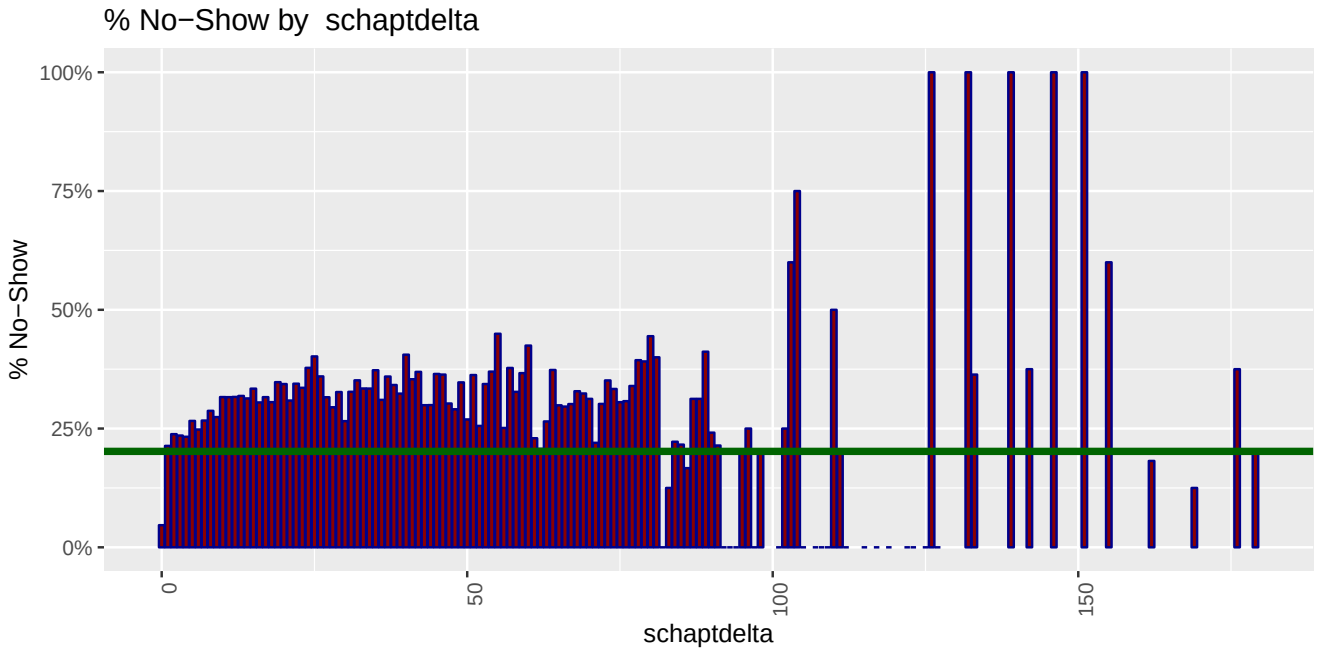




Patients with Disability type 1 have a lower No-Show percentage than average.

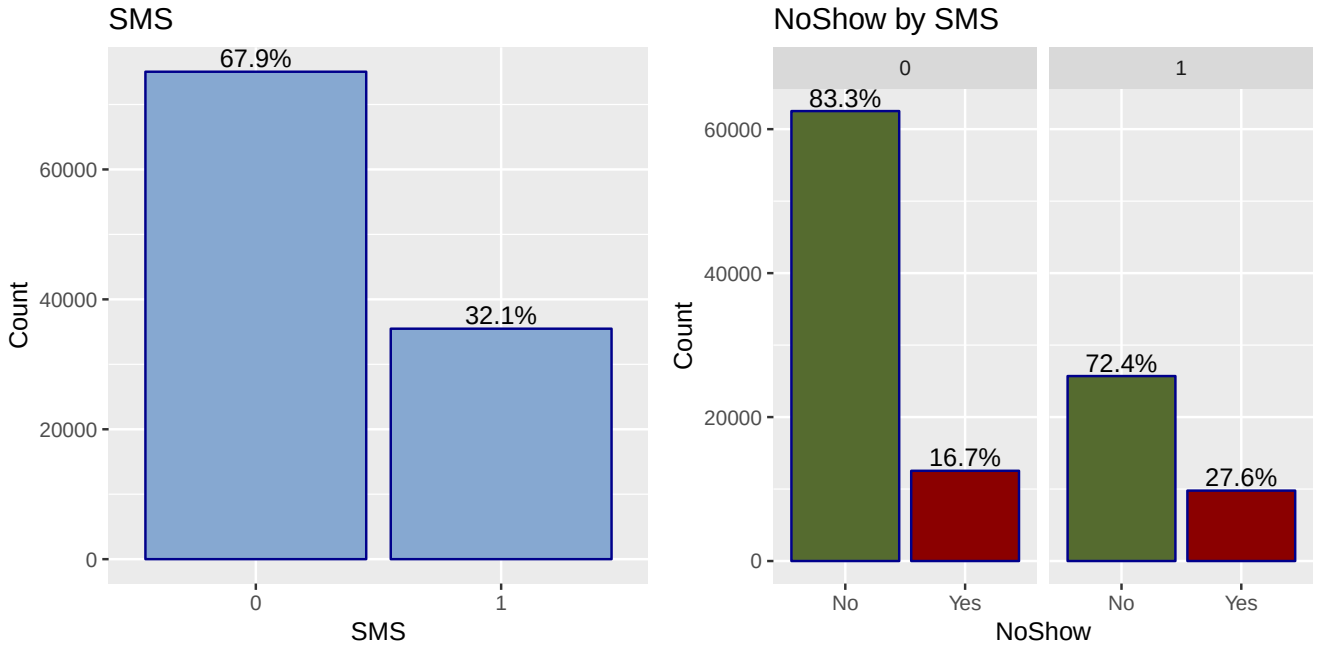
Delta between scheduled and appointment days





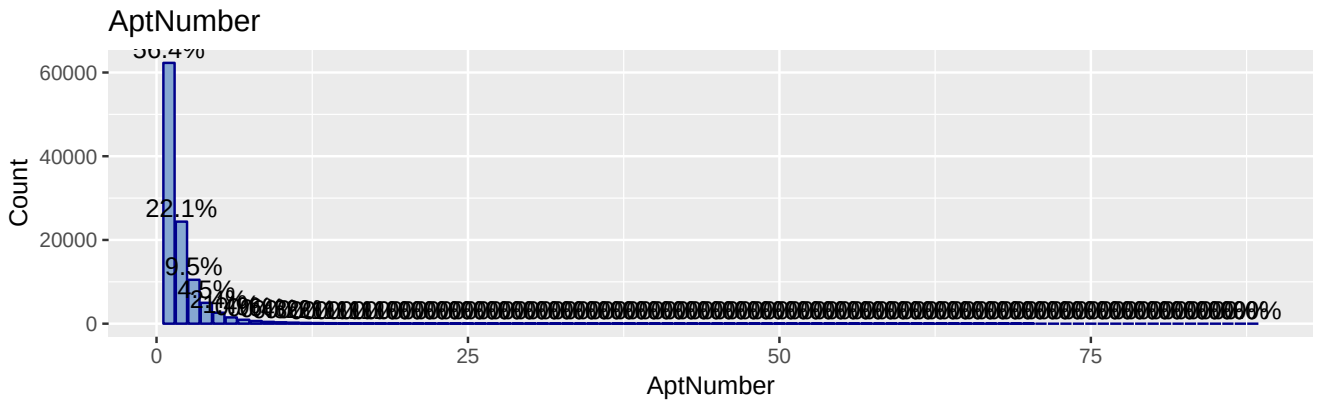
38568 appointments (**34.9**%) have been scheduled the same day. These appointments have a very low No-Show rate : **4.7**%.

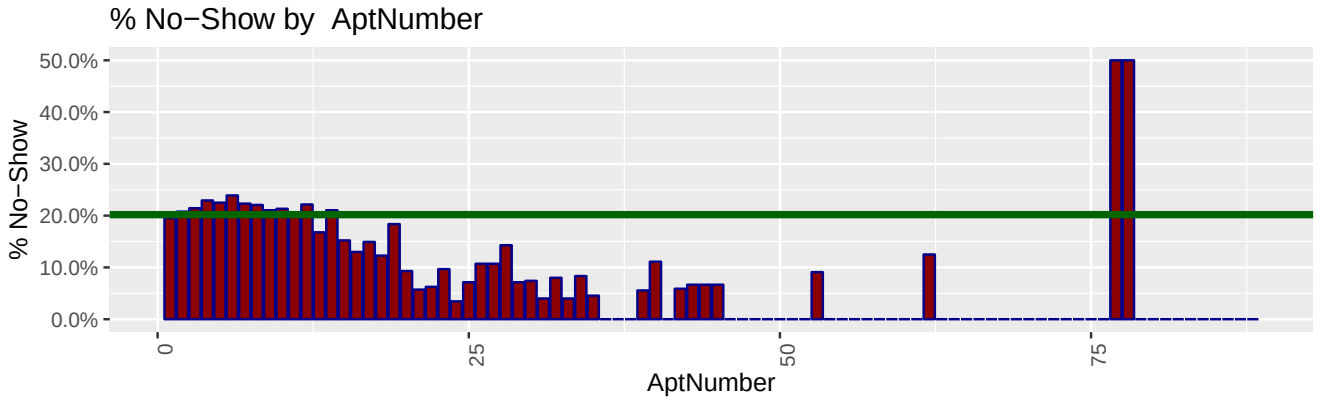
SMS



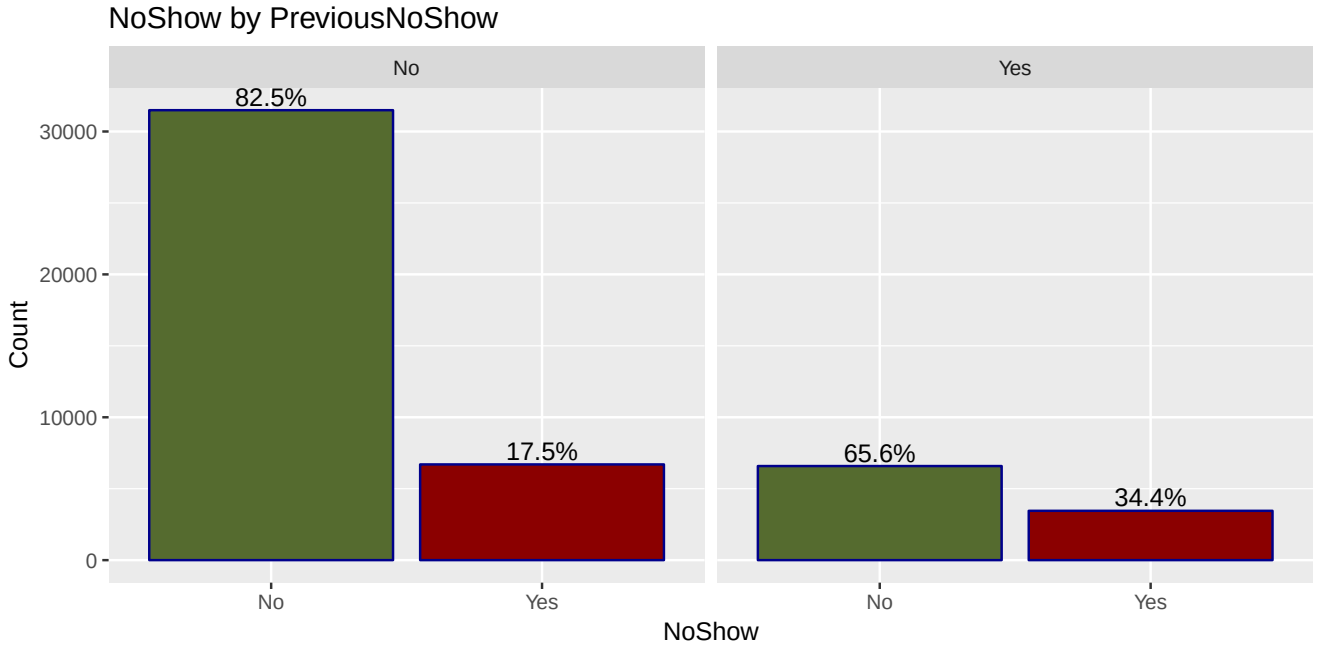
Patients who received an SMS reminder tends to have a higher No-Show percentage. I would have think that reminders would have lower it.

Patients





2nd to 12th apointements seems to have a higher No-Show rate. After that, the rate is lower but as the number of appointments is low, it is not really a usefull.



If the patient missed his last appointment, he has twice more chances to miss this new one.

Observations

- Gender does not seem to have an effect on No-Show,
- Younger children and people above 45 years have a lower No-Show,
- Patients with social aid have a higher No-Show rate,
- Patients with Hypertension and Diabetes have a lower rate, Alcoholism does not seem to have an effect,
- Delta between scheduled and appointment days have an effect(the lower, the better) : if the appointment is taken the same day, the No-Show rate is lower,
- Patients who received an SMS reminder tends to have a higher No-Show percentage,
- If the patient missed his last appointment, he has twice more chances to miss this new one.

Creating models

To facilitate the confusion matrix reading, we change the NoShow in Show (to set the NoShow as Positive)

```
# save
aptds.sav <- aptds

# transform NoShow in Show to have the NoShow in Positive column
aptds$Show <- ifelse(aptds$NoShow=='Yes', "No", "Yes")
aptds$Show <- as.factor(aptds$Show )
aptds$NoShow<- NULL
```

Using CARET and setting Sensitivity as metric

As the classes are unbalanced (80-20), we choose Sensitivity over Accuracy. When we pick accuracy, the models tend to the class with the highest number of records, and we never predict any NoShow.

We remove the keys and the dates. We also removed Neighbourhood but for a performance issue (> 8 hours).

```
aptds <- aptds %>% select(-PatientId,-AppointmentID,
                          -ScheduledDay,-AppointmentDay,
                          -Age, -Neighbourhood)

names(aptds)
```

```
##  [1] "Gender"                "SocialAid"
##  [3] "Hypertension"          "Diabetes"
##  [5] "Alcoholism"            "Disability"
##  [7] "SMS"                   "aptweekday"
##  [9] "schaptdelta"           "AgeBreak"
## [11] "AptNumber"             "PreviousNoShow"
## [13] "PreviousNoShowCount"   "PreviousPercentNoShow"
## [15] "Show"
```

We split the model in Train and Test datasets.

```
# split data in Train / test
set.seed(123)
test_index <- createDataPartition(y = aptds$Show, times = 1, p = 0.2, list = FALSE)
training <- aptds[-test_index,]
testing <- aptds[test_index,]
```

We will use 5-folds cross validation and Sensitivity as metric.

```
## Sensitivity metric, 5 folds validations
trControl <- trainControl(method="cv", number=5,
                          classProbs = TRUE,
                          savePredictions=TRUE,
                          summaryFunction = twoClassSummary,
                          allowParallel=TRUE)
metric <- "Sens"
```

Building the models

We train 4 different models:

```
# Tree
set.seed(123)
fit.rpart <- train(Show~., data=training, method="rpart",
                   metric=metric, trControl=trControl,
                   preProcess = c("center", "scale"))

# Random Forest ***
set.seed(123)
fit.rf <- train(Show~., data=training, method="rf", metric=metric,
                trControl=trControl, preProcess = c("center", "scale"))

# Logistic Regression
set.seed(123)
fit.glm <- train(Show~., data=training, method="glm", metric=metric,
                 trControl=trControl,
                 preProcess = c("center", "scale"))

# gradient boosting machine
set.seed(123)
fit.gbm <- train(Show~., data=training, method="gbm", metric=metric,
                     trControl=trControl, verbose=FALSE,
                     preProcess = c("center", "scale"))
```

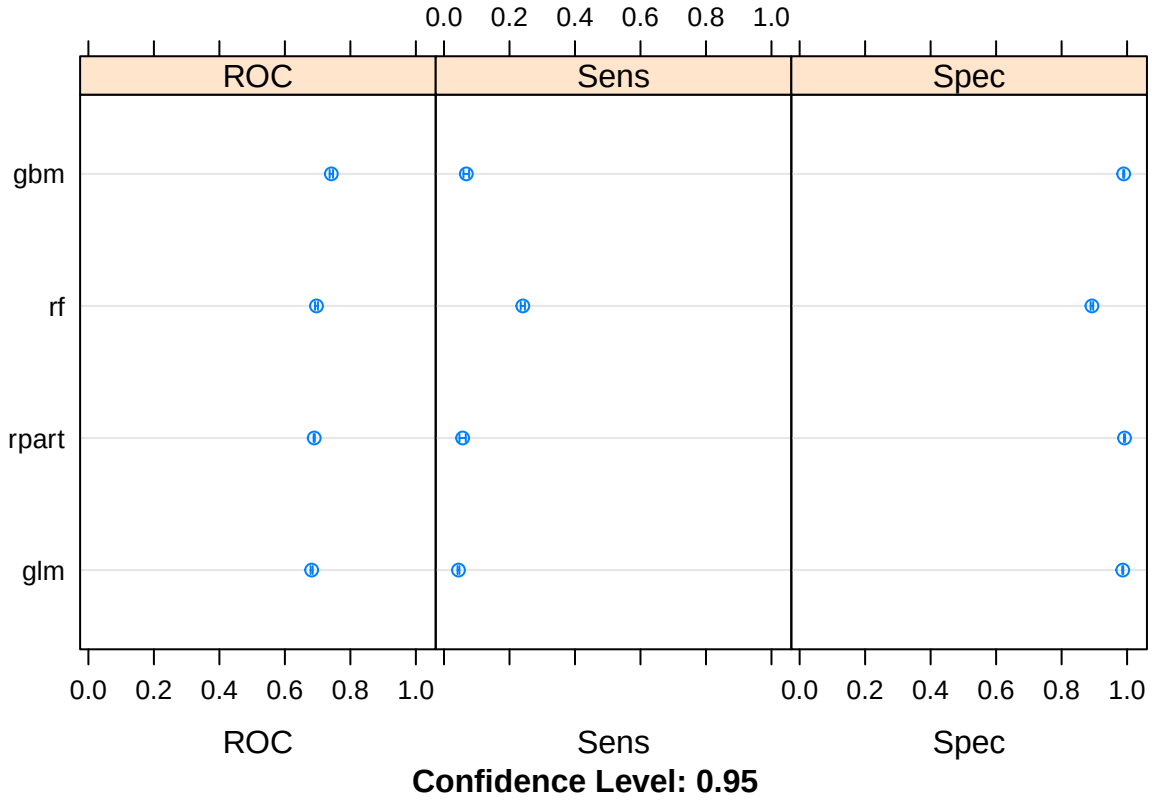
Comparing the models

To compare the models, we use the `resamples` function.

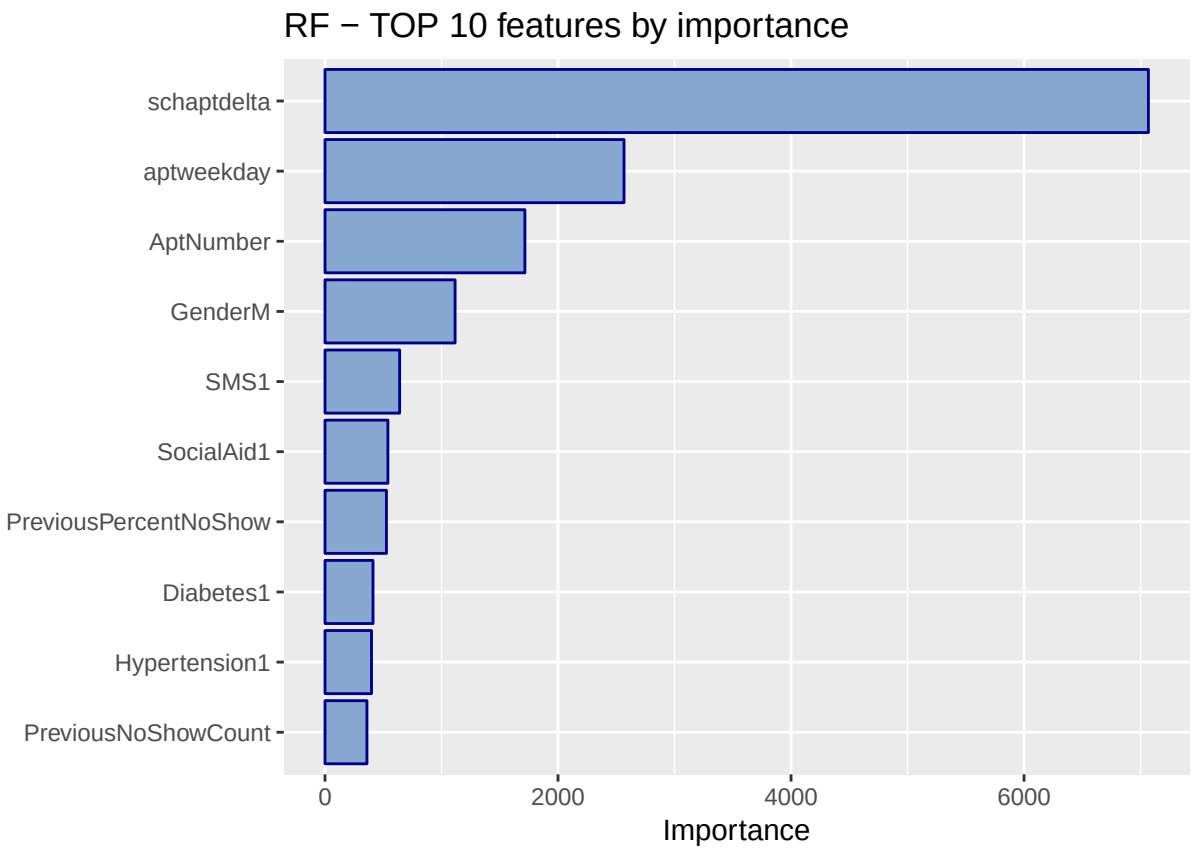
```
# Compare algorithms using the resamples caret function
results <- resamples(list(rpart=fit.rpart,
                          rf=fit.rf,
                          glm=fit.glm,
                          gbm=fit.gbm))
summary(results)
```

```
##
## Call:
```

```
## summary.resamples(object = results)
##
## Models: rpart, rf, glm, gbm
## Number of resamples: 5
##
## ROC
##            Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## rpart 0.6878225 0.6885108 0.6899610 0.6895910 0.6905407 0.6911201    0
## rf    0.6918996 0.6945512 0.6964356 0.6965353 0.6988898 0.7009000    0
## glm   0.6784501 0.6806798 0.6818332 0.6817235 0.6820652 0.6855894    0
## gbm   0.7353002 0.7410824 0.7429515 0.7419036 0.7438987 0.7462852    0
##
## Sens
##             Min.    1st Qu.     Median       Mean    3rd Qu.       Max.
## rpart 0.04844581 0.04984598 0.05796696 0.05684682 0.06104733 0.06692803
## rf    0.23298796 0.23942873 0.24194903 0.24066088 0.24194903 0.24698964
## glm   0.04116494 0.04256511 0.04508541 0.04418930 0.04536544 0.04676561
## gbm   0.05684682 0.06776813 0.06804817 0.06804817 0.07364884 0.07392887
##       NA's
## rpart    0
## rf       0
## glm      0
## gbm      0
##
## Spec
##            Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## rpart 0.9905761 0.9910018 0.9924183 0.9921776 0.9927726 0.9941189    0
## rf    0.8890385 0.8908807 0.8914553 0.8926112 0.8951321 0.8965493    0
## glm   0.9849784 0.9863247 0.9871050 0.9867642 0.9872458 0.9881669    0
## gbm   0.9880961 0.9890172 0.9892298 0.9896267 0.9904350 0.9913555    0
```



Of the 4 model tested, only the Random Forest model was able to achieve a sentitivity around 24%.
The delta between the scheduled day and the appointment dau seems to be the more important feature.



Checking RF on the test dataset

```
pred.rf <- predict(fit.rf, newdata=testing)
confusionMatrix(pred.rf, testing$Show)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    No   Yes
##        No   1062  1834
##        Yes  3402 15808
##
##                Accuracy : 0.7631
##                  95% CI : (0.7575, 0.7687)
##     No Information Rate : 0.7981
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.1542
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.23790
##             Specificity : 0.89604
##          Pos Pred Value : 0.36671
##          Neg Pred Value : 0.82290
##              Prevalence : 0.20194
##          Detection Rate : 0.04804
```

```
##    Detection Prevalence : 0.13101
##       Balanced Accuracy : 0.56697
##
##        'Positive' Class : No
##
```

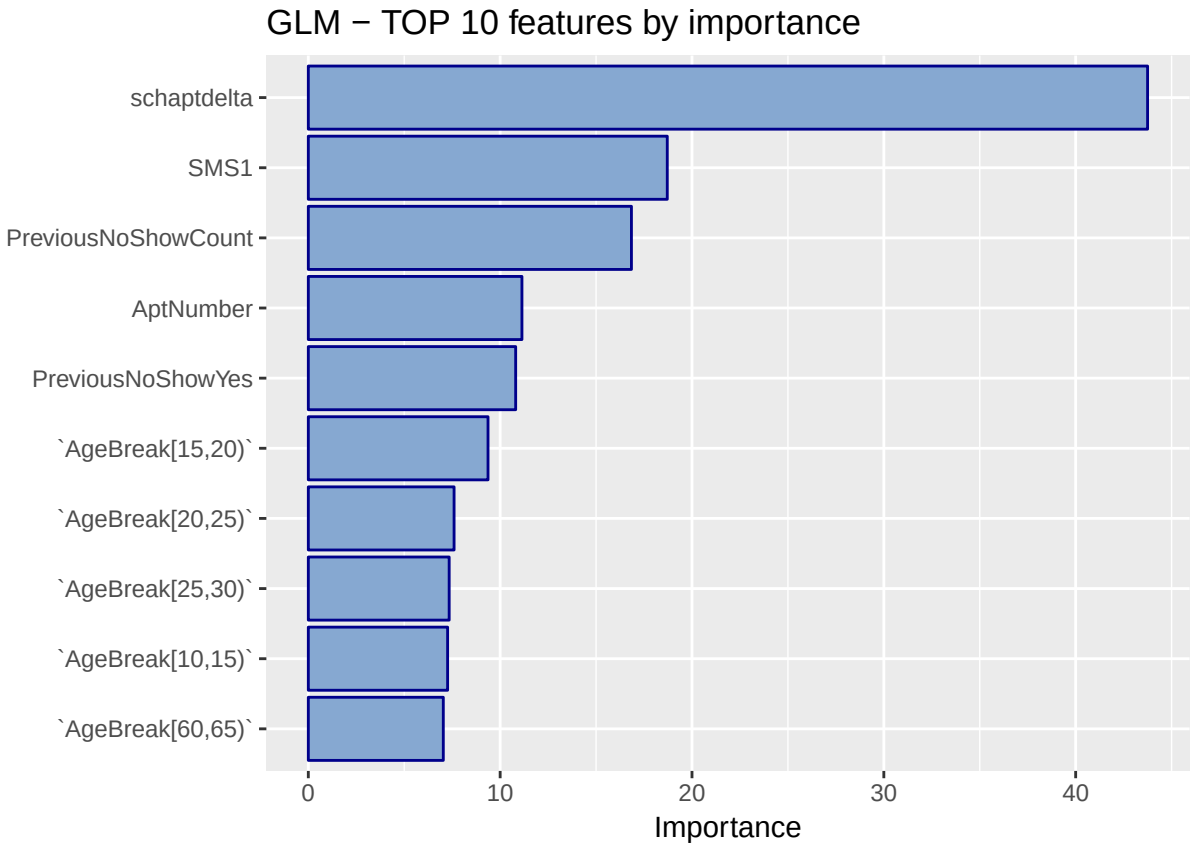
On the testing dataset, the Sensitivity is around 24% and the Positive Prediction Value around : 37%.
The accuracy is around 76% (if we predicted only "show"" it wouyld have been 80%) so there is no feature with a lot of predictive power in this model to account for the imbalance.

Modifying the cutoff

In this section, we will compute a GLM model and try the find the best treshold to optimize the accuracy and sensitivity.

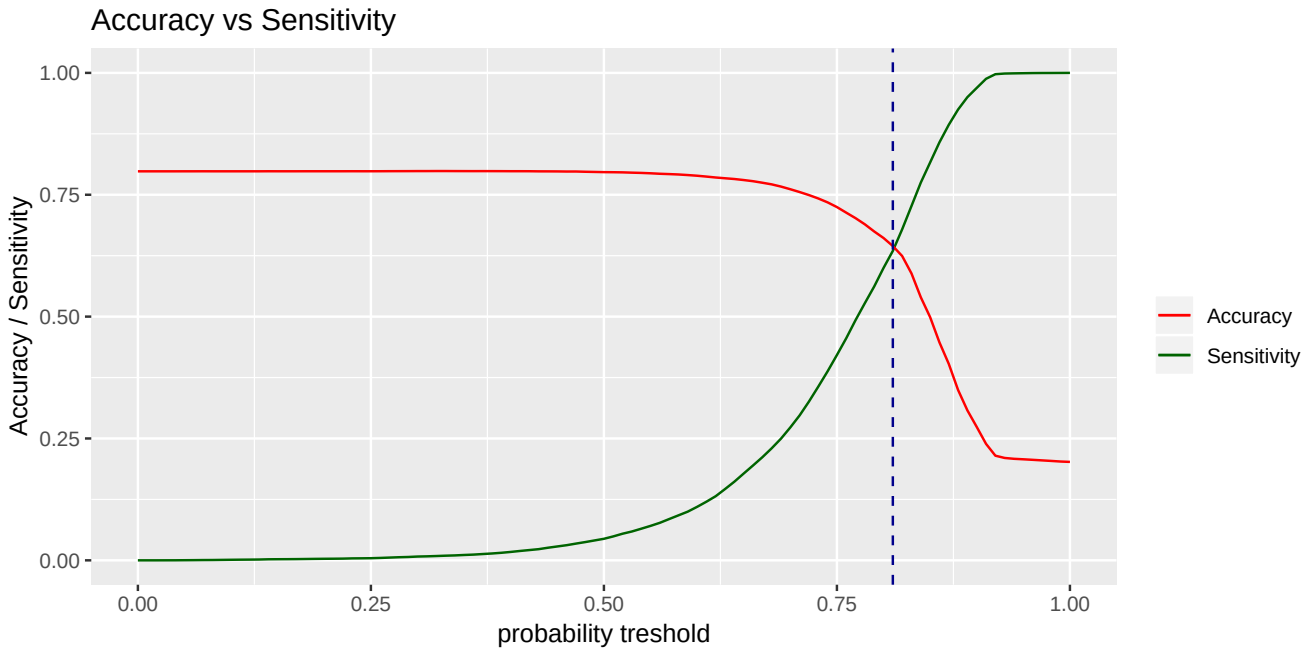
```
# Boosting Logistic Regression
set.seed(123)
fit.glm2 <- train(Show~., data=training, method="glm", metric="ROC",
                  trControl=trControl,
                  preProcess = c("center", "scale"))
```

In this model, the delta between the cheduled and appointment days is still the main feature:



Computing the treshold

To compute the treshold, we will use the model predicted values and not cross validation (I want to keep the Test dataset for the final validation).



The treshold optimizing accuracy and sensitivity is **0.81** (expected sensitivity : **63.5%**).

Checking GML and threshold on the test dataset

```
# compute the probality of each class for the Test dataset
prob.glm2 <- predict(fit.glm2, newdata=testing, type='prob')
# apply the treshold
pred.glm2 <- sapply(prob.glm2$Yes, function(x) if(x>=treshold.opt$treshold){'Yes'}else{'No'})
pred.glm2 <-factor(pred.glm2, levels = levels(testing$Show))
# display the confusion matrix
conf.glm2<-confusionMatrix(pred.glm2, testing$Show)
conf.glm2
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction    No   Yes
##        No   2818  6174
##        Yes  1646 11468
##
##                Accuracy : 0.6462
##                  95% CI : (0.6399, 0.6526)
##     No Information Rate : 0.7981
##     P-Value [Acc > NIR] : 1
##
##                   Kappa : 0.204
##  Mcnemar's Test P-Value : <2e-16
##
##             Sensitivity : 0.6313
##             Specificity : 0.6500
##          Pos Pred Value : 0.3134
##          Neg Pred Value : 0.8745
##              Prevalence : 0.2019
##          Detection Rate : 0.1275
##    Detection Prevalence : 0.4068
```

```
##       Balanced Accuracy : 0.6407
##
##        'Positive' Class : No
##
```

On the testing dataset, the Sensitivity is around **63.1%** and the Positive Prediction Value around : **31.3%**. The accuracy is around **64.6%**

Conclusion

None of the models gave excellent results. Even if the 2nd model gives a higher sensitivity of 63%, it does not bring any precision. The choice may depend of the satff ability to call or send a reminder to the predicted No-Show patients number.
The available and computed features does not seem to contain enought information to have a better accuracy and to overcome the imbalance (apointment type, specialist, price, . . .).
I did not check the techniques to try to resolving the No-Show class imbalance (weight, up sample, down sample, smote) but this may be a way to improve the accuracy or at least the precision.