FinalProject

2025-06-23

As I have told you at our first session, I really want to help people or planet through things that I can do. I believe that technology is a tool for solving global problems and improving the world.

One of my initiative is an affordable and convenient alternative to traditional electric wheelchairs made by combining a mechanical wheelchair with a gyroscooter(hoverboard) and lever systems. This innovation improves lifes of people with disabilities. Also I researched the accessibility of the environment and its effect on the quality of life of disabled people. I decided my short project to be meaningful and important to me, so I found a data set on kaggle which has different characteristics of vehicles and the column of their wheelchair accessibility

In this file, I will utilize all my current knowledge of R to get the most information out of this and be able to predict the accessibility ranking from predictors.

library(dplyr)

##   
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':  
##   
## filter, lag

## The following objects are masked from 'package:base':  
##   
## intersect, setdiff, setequal, union

library(caret)

## Loading required package: ggplot2

## Loading required package: lattice

library(ggplot2)  
library(repr)   
rm(list=ls())  
gc()

## used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)  
## Ncells 2222104 118.7 4515572 241.2 NA 2803201 149.8  
## Vcells 3786068 28.9 8388608 64.0 16384 6465893 49.4

mem.maxVSize()

## [1] 16384

Using these, I deleted previous variables and other stuff that I had from other files because there was no enough space for the new data set.

vehicles = read.csv("~/Downloads/algoritmika/python pro 1 year/Дасаева София PP/Learning\_R/Stanford/ShortProject/Vehicles.csv", na = c("", "NA"))  
dim(vehicles)

## [1] 99153 19

This is how many rows and columns the data set has. I decided to see how many of NA are there and what to do with them.

dim(vehicles[is.na(vehicles$Vehicle.Type),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Public.Vehicle.Number),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Status),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Vehicle.Make),])

## [1] 3286 19

dim(vehicles[is.na(vehicles$Vehicle.Model),])

## [1] 3422 19

dim(vehicles[is.na(vehicles$Vehicle.Model.Year),])

## [1] 3333 19

dim(vehicles[is.na(vehicles$Vehicle.Color),])

## [1] 3454 19

dim(vehicles[is.na(vehicles$Vehicle.Fuel.Source),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Wheelchair.Accessible),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Company.Name),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Address),])

## [1] 3266 19

dim(vehicles[is.na(vehicles$City),])

## [1] 3266 19

dim(vehicles[is.na(vehicles$State),])

## [1] 3266 19

dim(vehicles[is.na(vehicles$ZIP.Code),])

## [1] 3279 19

dim(vehicles[is.na(vehicles$Taxi.Affiliation),])

## [1] 34090 19

dim(vehicles[is.na(vehicles$Taxi.Medallion.License.Management),])

## [1] 40594 19

dim(vehicles[is.na(vehicles$Record.ID),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Last.Valid.Date),])

## [1] 0 19

dim(vehicles[is.na(vehicles$Record.Version.ID),])

## [1] 0 19

Vehicle.Make has 3286 NAs (out of 99153) - less than 3% Vehicle.Model Vehicle.Model.Year Vehicle.Color Address City State ZIP.Code Taxi.Medallion.License.Management Taxi.Affiliation

vehicles <- vehicles[ , -(15:16)]

columns like: Taxi.Medallion.License.Management Taxi.Affiliation have a lot of NAs (34k and 40k), so we can omit those columns entirely - 30%-40% of them are NAs

na\_counts <- rowSums(is.na(vehicles))  
rows\_4plus <- which(na\_counts >= 4)  
length(rows\_4plus)

## [1] 3444

vehicles <- vehicles[-rows\_4plus, ]

Then I also tried to find those rows that doesn’t possess a lot of information. For example, those rows that have 4+ NAs when only 8 columns might have NA, don’t have in fact a lot of valuable information. So we can delete those. It is 3444 rows.

There are still columns with NA’s, but the number of those NA’s decreased significantly.

dim(vehicles[is.na(vehicles$Vehicle.Make),])

## [1] 107 17

dim(vehicles[is.na(vehicles$Vehicle.Model),])

## [1] 241 17

dim(vehicles[is.na(vehicles$Vehicle.Model.Year),])

## [1] 154 17

dim(vehicles[is.na(vehicles$Vehicle.Color),])

## [1] 275 17

dim(vehicles[is.na(vehicles$Address),])

## [1] 0 17

dim(vehicles[is.na(vehicles$City),])

## [1] 0 17

dim(vehicles[is.na(vehicles$State),])

## [1] 0 17

dim(vehicles[is.na(vehicles$ZIP.Code),])

## [1] 13 17

Address, City and State don’t even have NAs anymore. It means that all those NA’s were actually rows with 4+ missing values. The number of those that still have missing value is too small. The biggest of them is just 0.29%. We can delete those too so they don’t create noise. Together, I will delete 0.82% of data which is not even a 1%.

na\_counts <- rowSums(is.na(vehicles))  
rows\_1plus <- which(na\_counts >= 1)  
length(rows\_1plus)

## [1] 563

vehicles <- vehicles[-rows\_1plus, ]

dim(vehicles[is.na(vehicles$Vehicle.Make),])

## [1] 0 17

dim(vehicles[is.na(vehicles$Vehicle.Model),])

## [1] 0 17

dim(vehicles[is.na(vehicles$Vehicle.Model.Year),])

## [1] 0 17

dim(vehicles[is.na(vehicles$Vehicle.Color),])

## [1] 0 17

dim(vehicles[is.na(vehicles$ZIP.Code),])

## [1] 0 17

There are no missing values now, therefore, we can make a good analysis.

summary(vehicles$Vehicle.Model.Year)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0 2012 2014 2010 2016 2911

The next thing I noticed is that Vehicle.Model.Year contains years like 0 and 2911.

dim(vehicles[(vehicles$Vehicle.Model.Year < 1885),])

## [1] 193 17

dim(vehicles[(vehicles$Vehicle.Model.Year > 2025),])

## [1] 5 17

It turned out that 198 rows have not appropriate years. It is about 0.2% of the data, so we can also drop them.

bad <- which(  
 vehicles$Vehicle.Model.Year < 1885 |  
 vehicles$Vehicle.Model.Year > as.integer(format(Sys.Date(), "%Y"))  
)  
  
length(bad)

## [1] 198

vehicles <- vehicles[-bad, ]

If we ran the summary again, there will be nothing odd:

summary(vehicles$Vehicle.Model.Year)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 1980 2012 2014 2014 2016 2025

It turned out that State column has only one value - IL, which will not affect the y in any way, so I will drop it.

vehicles = vehicles[, -13]

Next, there are a lot of company names, however they are repeated. Same with records, records.versions and dates

companies <- names(sort(table(vehicles$Company.Name), decreasing=TRUE))  
  
records <- names(sort(table(vehicles$Record.ID), decreasing=TRUE))  
records.versions <- names(sort(table(vehicles$Record.Version.ID), decreasing=TRUE))  
dates <- names(sort(table(vehicles$Last.Valid.Date), decreasing=TRUE))  
  
adresses <- names(sort(table(vehicles$Address), decreasing=TRUE))

There are 10591 unique names of companies. GLM will not be able to divide all of those names into separate predictors, therefore we need a new categorical value about companies. I will divide it into groups like top1000, top2000, top3000….

13445 unique of records 8131 unique of dates 4260 unique of addresses

companies <- sort(table(vehicles$Company.Name), decreasing = TRUE)  
company\_levels <- names(companies)  
ranks <- seq\_along(company\_levels)  
rank\_lookup <- setNames(ranks, company\_levels)  
  
breaks <- c(0, seq(1000, 10000, by = 1000), Inf)  
labels <- c(paste0("Top ", seq(1000, 10000, by = 1000)), "Other")  
  
vehicles$CompanyGroup <- sapply(vehicles$Company.Name, function(x) {  
 r <- rank\_lookup[x]  
 if (is.na(r)) return("Other")  
 bucket <- findInterval(r, vec = breaks, rightmost.closed = TRUE)  
 labels[bucket]  
})  
  
vehicles$CompanyGroup <- factor(  
 vehicles$CompanyGroup,  
 levels = labels  
)  
  
table(vehicles$CompanyGroup)

##   
## Top 1000 Top 2000 Top 3000 Top 4000 Top 5000 Top 6000 Top 7000 Top 8000   
## 45056 14057 8848 6662 5433 4438 3676 3000   
## Top 9000 Top 10000 Other   
## 2079 1107 592

Now there are only 11 variables which will be easier to process for the model. This means that those most frequent 1000 companies are grouped at Top1000, then Top2000 and etc

Same thing I will do for Record.ID

records <- sort(table(vehicles$Record.ID), decreasing = TRUE)  
record\_levels <- names(records)  
ranks <- seq\_along(record\_levels)  
rank\_lookup <- setNames(ranks, record\_levels)  
  
breaks <- c(0, seq(1000, 13000, by = 1000), Inf)  
labels <- c(paste0("Top ", seq(1000, 13000, by = 1000)), "Other")  
  
vehicles$RecordsGroup <- sapply(vehicles$Record.ID, function(x) {  
 r <- rank\_lookup[x]  
 if (is.na(r)) return("Other")  
 bucket <- findInterval(r, vec = breaks, rightmost.closed = TRUE)  
 labels[bucket]  
})  
  
vehicles$RecordsGroup <- factor(  
 vehicles$RecordsGroup,  
 levels = labels  
)  
  
table(vehicles$RecordsGroup)

##   
## Top 1000 Top 2000 Top 3000 Top 4000 Top 5000 Top 6000 Top 7000 Top 8000   
## 15215 12052 10542 9502 8789 8000 7037 6328   
## Top 9000 Top 10000 Top 11000 Top 12000 Top 13000 Other   
## 5543 4657 3368 2313 1156 446

And the same thing for Last.Valid.Date

dates <- sort(table(vehicles$Last.Valid.Date), decreasing = TRUE)  
dates\_levels <- names(dates)  
ranks <- seq\_along(dates\_levels)  
rank\_lookup <- setNames(ranks, dates\_levels)  
  
breaks <- c(0, seq(1000, 8000, by = 1000), Inf)  
labels <- c(paste0("Top ", seq(1000, 8000, by = 1000)), "Other")  
  
vehicles$DatesGroup <- sapply(vehicles$Last.Valid.Date, function(x) {  
 r <- rank\_lookup[x]  
 if (is.na(r)) return("Other")  
 bucket <- findInterval(r, vec = breaks, rightmost.closed = TRUE)  
 labels[bucket]  
})  
  
vehicles$DatesGroup <- factor(  
 vehicles$DatesGroup,  
 levels = labels  
)  
  
table(vehicles$DatesGroup)

##   
## Top 1000 Top 2000 Top 3000 Top 4000 Top 5000 Top 6000 Top 7000 Top 8000   
## 66279 12865 5498 3541 2535 2000 1098 1000   
## Other   
## 132

And same for Addresses

addresses <- sort(table(vehicles$Address), decreasing = TRUE)  
addresses\_levels <- names(addresses)  
ranks <- seq\_along(addresses\_levels)  
rank\_lookup <- setNames(ranks, addresses\_levels)  
  
breaks <- c(0, seq(1000, 4000, by = 1000), Inf)  
labels <- c(paste0("Top ", seq(1000, 4000, by = 1000)), "Other")  
  
vehicles$AddressesGroup <- sapply(vehicles$Address, function(x) {  
 r <- rank\_lookup[x]  
 if (is.na(r)) return("Other")  
 bucket <- findInterval(r, vec = breaks, rightmost.closed = TRUE)  
 labels[bucket]  
})  
  
vehicles$AddressesGroup <- factor(  
 vehicles$AddressesGroup,  
 levels = labels  
)  
  
table(vehicles$AddressesGroup)

##   
## Top 1000 Top 2000 Top 3000 Top 4000 Other   
## 85723 4881 2696 1387 261

After we have Group columns we don’t really need those original columns, so we can drop those. At the same time Records.Versions variable is unique for each row, so it will not give us valuable information in classification. Probably, it is connected to the Records.ID column. We can drop it, too.

vehicles = vehicles[, -(10:11)]  
vehicles = vehicles[, -(12:14)]

Let’s dive into real analysis after cleaning the data.

I decided to take just 80 percent of all the rows as train test and at the same time 20% of rows will be in test. I wanted to save the percantage of wheelchair accessible vehicles over all vehicles.

As Wheelchair.Accessible is a categorical value, linear regression is not the tool. I will try to use all models that are good at suprevised and supervised learning. Then I will compare which method was the most efficient one for this data set.

I tried to iterate the code on different predictors, because it was to much for the model to proccess if do it everything all at once

set.seed(1)  
lines <- createDataPartition(vehicles$Wheelchair.Accessible, p = 0.8,  
 list = FALSE,   
 times = 1)  
  
vehicles$Wheelchair.Accessible<- ifelse(  
 vehicles$Wheelchair.Accessible == "Y", 1, 0  
)  
  
vehicles.train = vehicles[lines, ]  
vehicles.test = vehicles[-lines, ]  
  
vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Type, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Type, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -7.2766 1.0003 -7.274 3.49e-13 \*\*\*  
## Vehicle.TypeCharter Sightseeing 1.2570 1.0960 1.147 0.251420   
## Vehicle.TypeHorse Drawn Carriage -7.2895 214.0991 -0.034 0.972839   
## Vehicle.TypeJitney -7.2895 99.3214 -0.073 0.941493   
## Vehicle.TypeLivery -0.3024 1.0448 -0.289 0.772235   
## Vehicle.TypeLow Speed Electric -7.2895 254.8280 -0.029 0.977179   
## Vehicle.TypeMedicar 3.8409 1.0135 3.790 0.000151 \*\*\*  
## Vehicle.TypePedicab 0.9639 1.2254 0.787 0.431508   
## Vehicle.TypeTaxi 4.5719 1.0005 4.570 4.89e-06 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 23360 on 75951 degrees of freedom  
## AIC: 23378  
##   
## Number of Fisher Scoring iterations: 13

If type is Charter Sightseeing, Medicar or Taxi, it has statistically significant effect on the y

vehicles.glm = glm(Wheelchair.Accessible ~ Public.Vehicle.Number, data = vehicles.train,   
 family = binomial)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Public.Vehicle.Number,   
## family = binomial, data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.335e+00 2.998e-02 -77.89 <2e-16 \*\*\*  
## Public.Vehicle.Number -1.830e-04 6.570e-06 -27.86 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 24824 on 75958 degrees of freedom  
## AIC: 24828  
##   
## Number of Fisher Scoring iterations: 10

Public.Vehicle.Number is considered statistically important, but it needs to be checked

vehicles.glm = glm(Wheelchair.Accessible ~ Status, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Status, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.26880 0.02805 -116.520 < 2e-16 \*\*\*  
## StatusFORECLOSURE 0.74824 0.06536 11.448 < 2e-16 \*\*\*  
## StatusHOLD 1.43622 0.53925 2.663 0.00774 \*\*   
## StatusINACTIVE -3.63856 0.44828 -8.117 4.79e-16 \*\*\*  
## StatusRESERVED -4.33831 0.70552 -6.149 7.79e-10 \*\*\*  
## StatusREVOKED -0.59193 0.35836 -1.652 0.09858 .   
## StatusSURRENDER 0.60105 0.04712 12.755 < 2e-16 \*\*\*  
## StatusVIOLATION 0.26385 0.04777 5.523 3.33e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 24912 on 75952 degrees of freedom  
## AIC: 24928  
##   
## Number of Fisher Scoring iterations: 9

If status is FORECLOSURE, INACTIVE, RESERVED, SURRENDER, VIOLATION, it does have a significant effect on the y. Status of HOLD is considered moderately significant

vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Make, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Make, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -1.857e+01 6.522e+03 -0.003 0.998  
## Vehicle.MakeACURA -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeALEXANDER DENNIS LTD -3.779e-04 6.618e+03 0.000 1.000  
## Vehicle.MakeALEXDER DENNIS LTD -3.779e-04 6.918e+03 0.000 1.000  
## Vehicle.MakeANDERSON -3.779e-04 7.988e+03 0.000 1.000  
## Vehicle.MakeANKAI -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeAUDI -3.779e-04 6.545e+03 0.000 1.000  
## Vehicle.MakeBARTH -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeBLUEBIRD -3.779e-04 6.918e+03 0.000 1.000  
## Vehicle.MakeBMW -3.779e-04 6.535e+03 0.000 1.000  
## Vehicle.MakeBMX -3.779e-04 7.292e+03 0.000 1.000  
## Vehicle.MakeBOYERTOWN -3.779e-04 6.973e+03 0.000 1.000  
## Vehicle.MakeBURRITO BROTHERS -3.779e-04 7.988e+03 0.000 1.000  
## Vehicle.MakeCADILLAC 1.046e+01 6.522e+03 0.002 0.999  
## Vehicle.MakeCHAMPION -3.779e-04 6.601e+03 0.000 1.000  
## Vehicle.MakeCHARLESTON -3.779e-04 7.145e+03 0.000 1.000  
## Vehicle.MakeCHEVROLET 1.132e+01 6.522e+03 0.002 0.999  
## Vehicle.MakeCHRYSLER 1.549e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeCOASTER -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeCUSTOM -3.779e-04 6.546e+03 0.000 1.000  
## Vehicle.MakeCYCLESTONE -3.779e-04 7.988e+03 0.000 1.000  
## Vehicle.MakeDodge -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeDODGE 1.852e+01 6.522e+03 0.003 0.998  
## Vehicle.MakeDOUBLE SHUFLE -3.779e-04 6.973e+03 0.000 1.000  
## Vehicle.MakeFord -3.779e-04 7.988e+03 0.000 1.000  
## Vehicle.MakeFORD 1.274e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeFREIGHLINE 1.330e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeFREIGHTLINER 1.396e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeGENISIS -3.779e-04 7.145e+03 0.000 1.000  
## Vehicle.MakeGILLIG -3.779e-04 6.647e+03 0.000 1.000  
## Vehicle.MakeGMC 1.129e+01 6.522e+03 0.002 0.999  
## Vehicle.MakeHONDA 1.448e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeHUMMER -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeHYUNDAI -3.779e-04 6.531e+03 0.000 1.000  
## Vehicle.MakeIC -3.779e-04 6.606e+03 0.000 1.000  
## Vehicle.MakeINFINITI -3.779e-04 6.543e+03 0.000 1.000  
## Vehicle.MakeINTL -3.779e-04 6.606e+03 0.000 1.000  
## Vehicle.MakeISUZU -3.779e-04 6.973e+03 0.000 1.000  
## Vehicle.MakeJEEP -3.779e-04 6.769e+03 0.000 1.000  
## Vehicle.MakeKIA 1.354e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeKSIR -3.779e-04 9.224e+03 0.000 1.000  
## Vehicle.MakeLAND ROVER -3.779e-04 6.841e+03 0.000 1.000  
## Vehicle.MakeLEXUS -3.779e-04 6.551e+03 0.000 1.000  
## Vehicle.MakeLEYLAND -3.779e-04 6.751e+03 0.000 1.000  
## Vehicle.MakeLINCOLN 1.123e+01 6.522e+03 0.002 0.999  
## Vehicle.MakeLUCID -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeLUCKY -3.779e-04 7.988e+03 0.000 1.000  
## Vehicle.MakeMAINSTREET -3.779e-04 6.530e+03 0.000 1.000  
## Vehicle.MakeMANMADE -3.779e-04 6.537e+03 0.000 1.000  
## Vehicle.MakeMARTIN -3.779e-04 6.841e+03 0.000 1.000  
## Vehicle.MakeMASERATI -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeMAZDA -3.779e-04 6.751e+03 0.000 1.000  
## Vehicle.MakeMCI -3.779e-04 6.583e+03 0.000 1.000  
## Vehicle.MakeMERCEDES 1.147e+01 6.522e+03 0.002 0.999  
## Vehicle.MakeMERCURY -3.779e-04 6.603e+03 0.000 1.000  
## Vehicle.MakeMITSUBISHI -3.779e-04 6.723e+03 0.000 1.000  
## Vehicle.MakeMOBILITY VENTURE -3.779e-04 9.224e+03 0.000 1.000  
## Vehicle.MakeMODIFIED -3.779e-04 7.988e+03 0.000 1.000  
## Vehicle.MakeMOVITRON -3.779e-04 7.292e+03 0.000 1.000  
## Vehicle.MakeNEOPLAN -3.779e-04 6.669e+03 0.000 1.000  
## Vehicle.MakeNISSAN 1.123e+01 6.522e+03 0.002 0.999  
## Vehicle.MakeORION -3.779e-04 6.973e+03 0.000 1.000  
## Vehicle.MakeOSHKOSH -3.779e-04 7.292e+03 0.000 1.000  
## Vehicle.MakePOLARIS -3.779e-04 6.642e+03 0.000 1.000  
## Vehicle.MakePONTIAC -3.779e-04 7.292e+03 0.000 1.000  
## Vehicle.MakePRECISION -3.779e-04 6.601e+03 0.000 1.000  
## Vehicle.MakeROBERTS -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeSCHWARTZ -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeSCHWINN -3.779e-04 6.692e+03 0.000 1.000  
## Vehicle.MakeSCION -3.779e-04 6.712e+03 0.000 1.000  
## Vehicle.MakeSMASH CAB -3.779e-04 9.224e+03 0.000 1.000  
## Vehicle.MakeSPEC-CONSTED -3.779e-04 7.045e+03 0.000 1.000  
## Vehicle.MakeSTEEL -3.779e-04 7.531e+03 0.000 1.000  
## Vehicle.MakeSUBARU -3.779e-04 6.769e+03 0.000 1.000  
## Vehicle.MakeTESLA -3.779e-04 6.544e+03 0.000 1.000  
## Vehicle.MakeTHOMAS -3.779e-04 6.736e+03 0.000 1.000  
## Vehicle.MakeTIPKE 1.557e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeTOYOTA 1.523e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeTROYER -3.779e-04 9.224e+03 0.000 1.000  
## Vehicle.MakeTUK TUK -3.779e-04 6.789e+03 0.000 1.000  
## Vehicle.MakeTURTLE TOP -3.779e-04 7.045e+03 0.000 1.000  
## Vehicle.MakeVAN HOOL -3.779e-04 6.663e+03 0.000 1.000  
## Vehicle.MakeVIP 1.311e+01 6.522e+03 0.002 0.998  
## Vehicle.MakeVOLKSWAGEN -3.779e-04 6.769e+03 0.000 1.000  
## Vehicle.MakeVOLVO -3.779e-04 6.556e+03 0.000 1.000  
## Vehicle.MakeVPG 2.073e+01 6.522e+03 0.003 0.997  
## Vehicle.MakeYODER -3.779e-04 9.224e+03 0.000 1.000  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 14549 on 75873 degrees of freedom  
## AIC: 14723  
##   
## Number of Fisher Scoring iterations: 17

Vehicle.Make doesn’t affect the Wheelchair Accessibility

vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Model, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Model, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -2.057e+01 4.918e+03 -0.004 0.997  
## Vehicle.Model102DL3 6.890e-07 7.098e+03 0.000 1.000  
## Vehicle.Model3 SEATER 6.773e-07 1.840e+04 0.000 1.000  
## Vehicle.Model300 6.948e-07 5.030e+03 0.000 1.000  
## Vehicle.Model3000 SERIES 6.950e-07 1.840e+04 0.000 1.000  
## Vehicle.Model300SER 6.948e-07 8.312e+03 0.000 1.000  
## Vehicle.Model350 6.971e-07 5.387e+03 0.000 1.000  
## Vehicle.Model3500 6.959e-07 6.620e+03 0.000 1.000  
## Vehicle.Model4500 6.885e-07 8.312e+03 0.000 1.000  
## Vehicle.Model4505 6.998e-07 7.688e+03 0.000 1.000  
## Vehicle.Model450H 6.949e-07 1.840e+04 0.000 1.000  
## Vehicle.Model5 SEATER 6.997e-07 6.453e+03 0.000 1.000  
## Vehicle.Model528XI 7.003e-07 1.136e+04 0.000 1.000  
## Vehicle.Model530E 6.856e-07 1.136e+04 0.000 1.000  
## Vehicle.Model530I 7.007e-07 6.719e+03 0.000 1.000  
## Vehicle.Model530IX 7.002e-07 8.751e+03 0.000 1.000  
## Vehicle.Model530XE 6.906e-07 1.840e+04 0.000 1.000  
## Vehicle.Model540 XI 6.921e-07 9.330e+03 0.000 1.000  
## Vehicle.Model540I 6.962e-07 1.347e+04 0.000 1.000  
## Vehicle.Model6 SEATER 7.060e-07 7.264e+03 0.000 1.000  
## Vehicle.Model6 TOURING PLUS 6.379e-07 1.840e+04 0.000 1.000  
## Vehicle.Model650 6.977e-07 7.967e+03 0.000 1.000  
## Vehicle.Model740 I 6.937e-07 5.327e+03 0.000 1.000  
## Vehicle.Model740 LXI 6.948e-07 6.453e+03 0.000 1.000  
## Vehicle.Model740LI 6.949e-07 5.509e+03 0.000 1.000  
## Vehicle.Model740XI 6.932e-07 7.098e+03 0.000 1.000  
## Vehicle.Model750 I 6.968e-07 1.014e+04 0.000 1.000  
## Vehicle.Model750LI 6.948e-07 6.954e+03 0.000 1.000  
## Vehicle.Model750XI 7.006e-07 6.317e+03 0.000 1.000  
## Vehicle.Model8071 6.947e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelA6 6.948e-07 5.951e+03 0.000 1.000  
## Vehicle.ModelA8 6.918e-07 5.409e+03 0.000 1.000  
## Vehicle.ModelACADIA 6.948e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelACCORD 6.854e-07 6.063e+03 0.000 1.000  
## Vehicle.ModelAIR TOURING 6.946e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelALTIMA 1.336e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelAMBULANCE 7.051e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelARMADA 6.924e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelAVALANCHE K1500 LT 6.793e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelAVALON 6.948e-07 5.498e+03 0.000 1.000  
## Vehicle.ModelAVIATOR 6.944e-07 4.999e+03 0.000 1.000  
## Vehicle.ModelBACK TO BACK 6.951e-07 6.954e+03 0.000 1.000  
## Vehicle.ModelBB36 6.949e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelBENZ 6.948e-07 5.167e+03 0.000 1.000  
## Vehicle.ModelBOARDWALK 7.057e-07 5.240e+03 0.000 1.000  
## Vehicle.ModelBRAUN 7.077e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelBROADWAY 6.947e-07 5.132e+03 0.000 1.000  
## Vehicle.ModelBURRITO 6.948e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelBZ4X 6.826e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelC-MAX 6.948e-07 4.968e+03 0.000 1.000  
## Vehicle.ModelC300 6.958e-07 8.751e+03 0.000 1.000  
## Vehicle.ModelC5500 6.949e-07 5.806e+03 0.000 1.000  
## Vehicle.ModelCAMRY 1.400e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelCAPRICE 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelCARAVAN 2.048e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelCARRYALL 6.948e-07 5.401e+03 0.000 1.000  
## Vehicle.ModelCG33503 6.974e-07 6.620e+03 0.000 1.000  
## Vehicle.ModelCHALLENGER 6.948e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelCHAMPION 7.002e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelCHASSIS 6.949e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelCHR 6.938e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelCIVIC 6.948e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelCLASSIC 6.951e-07 5.050e+03 0.000 1.000  
## Vehicle.ModelCOACH 6.992e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelCOMPASS 6.948e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelCONCORDE 6.947e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelCONTINENTAL 1.399e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelCOROLLA 6.945e-07 4.972e+03 0.000 1.000  
## Vehicle.ModelCORSAIR 6.943e-07 5.052e+03 0.000 1.000  
## Vehicle.ModelCROWN 7.006e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelCROWN VICTORIA 6.949e-07 4.988e+03 0.000 1.000  
## Vehicle.ModelCRUISER 6.946e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelCRV 1.590e+01 4.918e+03 0.003 0.997  
## Vehicle.ModelCSM STYLE 6.948e-07 5.151e+03 0.000 1.000  
## Vehicle.ModelCT6 6.948e-07 5.678e+03 0.000 1.000  
## Vehicle.ModelCTS 6.948e-07 7.458e+03 0.000 1.000  
## Vehicle.ModelCTS WAGON 6.948e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelCUTAWAY 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelCX5 7.015e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelD4505 6.947e-07 6.063e+03 0.000 1.000  
## Vehicle.ModelDEVILLE 6.948e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelDOUBLE DECKER 6.986e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelDTS 6.948e-07 5.380e+03 0.000 1.000  
## Vehicle.ModelE-TRON 6.948e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelE150 1.703e+01 4.918e+03 0.003 0.997  
## Vehicle.ModelE250 1.752e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelE300 6.932e-07 6.203e+03 0.000 1.000  
## Vehicle.ModelE350 1.458e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelE400H 6.997e-07 6.719e+03 0.000 1.000  
## Vehicle.ModelE450 6.922e-07 5.123e+03 0.000 1.000  
## Vehicle.ModelECONOLINE 1.708e+01 4.918e+03 0.003 0.997  
## Vehicle.ModelELANTRA 6.950e-07 5.117e+03 0.000 1.000  
## Vehicle.ModelELDORADO BUS 6.948e-07 7.967e+03 0.000 1.000  
## Vehicle.ModelENTOURAGE 6.937e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelENVIPO400 7.069e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelENVIRO 400 6.947e-07 5.985e+03 0.000 1.000  
## Vehicle.ModelEQUINOX 6.966e-07 6.317e+03 0.000 1.000  
## Vehicle.ModelEQUUS 6.961e-07 6.533e+03 0.000 1.000  
## Vehicle.ModelES 250 7.051e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelES 300 6.973e-07 6.954e+03 0.000 1.000  
## Vehicle.ModelES 350 6.901e-07 6.620e+03 0.000 1.000  
## Vehicle.ModelESCALADE 6.941e-07 4.953e+03 0.000 1.000  
## Vehicle.ModelESCALADE ESV 1.408e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelESCAPE 1.368e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelEXCURSION 6.949e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelEXPEDITION 1.324e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelEXPEDITION PLATINUM 6.947e-07 5.488e+03 0.000 1.000  
## Vehicle.ModelEXPLORER 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelEXPRESS 6.948e-07 5.327e+03 0.000 1.000  
## Vehicle.ModelF150 6.948e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelF350 7.015e-07 7.458e+03 0.000 1.000  
## Vehicle.ModelF450 6.948e-07 6.203e+03 0.000 1.000  
## Vehicle.ModelF53 6.948e-07 5.985e+03 0.000 1.000  
## Vehicle.ModelF550 6.955e-07 5.293e+03 0.000 1.000  
## Vehicle.ModelF630 6.952e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelF650 1.611e+01 4.918e+03 0.003 0.997  
## Vehicle.ModelFACE TO FACE 6.948e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelFB65 1.718e+01 4.918e+03 0.003 0.997  
## Vehicle.ModelFESTIVAL 6.949e-07 8.751e+03 0.000 1.000  
## Vehicle.ModelFLEX 6.931e-07 6.203e+03 0.000 1.000  
## Vehicle.ModelFOCUS 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelFORESTER 6.950e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelFORTE LX 6.952e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelFREESTAR 6.948e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelFREESTYLE 6.948e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelFUSION 1.506e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelFX 6.948e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelFX35 1.987e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelG3500 6.948e-07 7.688e+03 0.000 1.000  
## Vehicle.ModelG40M2 6.944e-07 5.583e+03 0.000 1.000  
## Vehicle.ModelG90 6.967e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelGEM E6 6.979e-07 5.985e+03 0.000 1.000  
## Vehicle.ModelGENESIS 7.017e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelGL320 6.948e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelGL350 6.966e-07 5.887e+03 0.000 1.000  
## Vehicle.ModelGL450 6.960e-07 5.373e+03 0.000 1.000  
## Vehicle.ModelGLAVAL LEGACY 6.884e-07 6.257e+03 0.000 1.000  
## Vehicle.ModelGLAVAL SYNERGY 6.901e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelGLC300W4 6.929e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelGLE350 6.976e-07 7.967e+03 0.000 1.000  
## Vehicle.ModelGLS 450 6.946e-07 7.688e+03 0.000 1.000  
## Vehicle.ModelGOLDEN GATE 6.948e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelGRAND CARAVAN 2.066e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelGRAND MARQUIS 6.949e-07 5.737e+03 0.000 1.000  
## Vehicle.ModelGS 200T 6.948e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelGS350 6.996e-07 7.458e+03 0.000 1.000  
## Vehicle.ModelGS460 6.984e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelGX460 6.948e-07 6.453e+03 0.000 1.000  
## Vehicle.ModelH3 6.941e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelHHR 6.972e-07 5.394e+03 0.000 1.000  
## Vehicle.ModelHIGHLANDER 6.949e-07 5.188e+03 0.000 1.000  
## Vehicle.ModelHORSESHOE 6.944e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelHR-V 6.948e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelIMPALA 6.948e-07 5.104e+03 0.000 1.000  
## Vehicle.ModelINSIGHT 6.947e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelINTREPID 7.011e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelIONIQ 6.909e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelJ4500 6.962e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelJOURNEY 7.004e-07 6.829e+03 0.000 1.000  
## Vehicle.ModelKICKS 7.049e-07 7.688e+03 0.000 1.000  
## Vehicle.ModelKRYSTAL KK28 BUS 6.990e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelKSIR290 6.846e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelL5 LIMO GT 6.949e-07 7.098e+03 0.000 1.000  
## Vehicle.ModelLIMO 6.985e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelLS 460L 6.950e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelLS500 6.829e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelLX570 6.951e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelLYRIQ 6.950e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelM 37X 6.949e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelM LINE SHUTTLE BUS 6.977e-07 5.312e+03 0.000 1.000  
## Vehicle.ModelM1000 6.967e-07 6.829e+03 0.000 1.000  
## Vehicle.ModelM1235 6.949e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelM2 6.945e-07 5.859e+03 0.000 1.000  
## Vehicle.ModelM2 106 6.928e-07 5.951e+03 0.000 1.000  
## Vehicle.ModelM2106 6.948e-07 6.317e+03 0.000 1.000  
## Vehicle.ModelM5 6.955e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelM8 6.816e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelMAGNUM 6.948e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelMALIBU 6.948e-07 7.458e+03 0.000 1.000  
## Vehicle.ModelMAZDA 3 6.810e-07 8.751e+03 0.000 1.000  
## Vehicle.ModelMC 6.948e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelMDX 6.928e-07 7.688e+03 0.000 1.000  
## Vehicle.ModelMETRIS 6.949e-07 6.023e+03 0.000 1.000  
## Vehicle.ModelMINI COACH 6.948e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelMIRAGE G4 ES 6.954e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelMKC 7.051e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelMKS 6.948e-07 4.941e+03 0.000 1.000  
## Vehicle.ModelMKT 1.287e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelMKX 6.948e-07 5.509e+03 0.000 1.000  
## Vehicle.ModelMKZ 6.948e-07 5.349e+03 0.000 1.000  
## Vehicle.ModelMKZ HYBRID 6.859e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelML 250 6.939e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelML350 6.956e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelMODEL 3 6.949e-07 7.458e+03 0.000 1.000  
## Vehicle.ModelMODEL Y 6.948e-07 6.317e+03 0.000 1.000  
## Vehicle.ModelMOUNTAINEER 6.754e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelMUSTANG MACH-E 7.114e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelMV-1 2.274e+01 4.918e+03 0.005 0.996  
## Vehicle.ModelMV1 6.940e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelNAUTILUS 1.514e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelNAVIGATOR 1.300e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelNAVIGATOR L 6.952e-07 4.970e+03 0.000 1.000  
## Vehicle.ModelNEWPORT 6.946e-07 6.453e+03 0.000 1.000  
## Vehicle.ModelNIRO 6.814e-07 7.967e+03 0.000 1.000  
## Vehicle.ModelNONE 1.505e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelNV200 6.948e-07 5.297e+03 0.000 1.000  
## Vehicle.ModelODYSSEY 1.803e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelOLYMPIAN 6.950e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelOPTIMA 6.951e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelORION-6 6.954e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelOUTBACK 6.948e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelOUTLANDER 6.951e-07 7.688e+03 0.000 1.000  
## Vehicle.ModelPACIFICA 6.948e-07 6.533e+03 0.000 1.000  
## Vehicle.ModelPASSAT 6.948e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelPATHFINDER 6.973e-07 6.533e+03 0.000 1.000  
## Vehicle.ModelPB20500 6.920e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelPHANTOM 6.944e-07 7.688e+03 0.000 1.000  
## Vehicle.ModelPILOT 6.942e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelPP6 6.920e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelPRECISION 6.953e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelPRIUS 1.357e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelQ5 6.916e-07 6.620e+03 0.000 1.000  
## Vehicle.ModelQ7 6.951e-07 7.688e+03 0.000 1.000  
## Vehicle.ModelQ70 6.955e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelQUATTROPORTE S 6.934e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelQUEST 6.946e-07 6.719e+03 0.000 1.000  
## Vehicle.ModelQX56 6.946e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelQX60 6.948e-07 5.138e+03 0.000 1.000  
## Vehicle.ModelQX80 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelR320 6.943e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelR350 6.948e-07 6.023e+03 0.000 1.000  
## Vehicle.ModelRANGE ROVER 6.948e-07 7.967e+03 0.000 1.000  
## Vehicle.ModelRAV4 1.486e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelRENEGADE 6.961e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelRIO 6.886e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelROGUE 6.946e-07 5.416e+03 0.000 1.000  
## Vehicle.ModelROUTAN 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelRX 450H 7.211e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelRX350 6.947e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelS 6.950e-07 5.343e+03 0.000 1.000  
## Vehicle.ModelS-350 6.948e-07 5.806e+03 0.000 1.000  
## Vehicle.ModelS2 106 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelS341 6.931e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelS34G 6.948e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelS400 6.939e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelS430 6.948e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelS450 1.768e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelS500 6.968e-07 8.751e+03 0.000 1.000  
## Vehicle.ModelS550 6.948e-07 5.096e+03 0.000 1.000  
## Vehicle.ModelS560 6.948e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelS580 6.944e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelS600 6.948e-07 8.751e+03 0.000 1.000  
## Vehicle.ModelS63 7.007e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelS90 6.947e-07 5.380e+03 0.000 1.000  
## Vehicle.ModelSABLE 6.949e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelSANTA FE 6.948e-07 6.829e+03 0.000 1.000  
## Vehicle.ModelSAVANA 6.949e-07 7.967e+03 0.000 1.000  
## Vehicle.ModelSCION 1.413e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelSEDONA 1.768e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelSENTRA 6.948e-07 5.104e+03 0.000 1.000  
## Vehicle.ModelSEQUOIA 6.941e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelSHUTTLE 6.950e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelSIENNA 2.103e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelSINGLE BENCH 6.947e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelSMASHCAB 6.948e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelSONATA 6.948e-07 5.083e+03 0.000 1.000  
## Vehicle.ModelSONIC 6.959e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelSORENTO 6.949e-07 8.312e+03 0.000 1.000  
## Vehicle.ModelSOUL 6.948e-07 5.272e+03 0.000 1.000  
## Vehicle.ModelSPEC-CONSTD 6.953e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelSPORTAGE LX 6.948e-07 7.098e+03 0.000 1.000  
## Vehicle.ModelSPRINTER 6.927e-07 4.982e+03 0.000 1.000  
## Vehicle.ModelSRX 6.898e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelSTARCRAFT 6.945e-07 5.644e+03 0.000 1.000  
## Vehicle.ModelSUBURBAN 1.235e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelSUPER 6.948e-07 8.751e+03 0.000 1.000  
## Vehicle.ModelT150 6.950e-07 6.023e+03 0.000 1.000  
## Vehicle.ModelT250 6.947e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelTAHOE 6.948e-07 5.401e+03 0.000 1.000  
## Vehicle.ModelTAURUS 6.948e-07 6.203e+03 0.000 1.000  
## Vehicle.ModelTC2000 6.948e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelTD925 6.949e-07 6.152e+03 0.000 1.000  
## Vehicle.ModelTELLURIDE 6.944e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelTENNER 6.945e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelTERRA 6.948e-07 7.458e+03 0.000 1.000  
## Vehicle.ModelTERRAIN 6.947e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelTIPKE 6.954e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelTOWN AND COUNTRY 1.874e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelTOWN CAR 1.401e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelTRANSIT 6.948e-07 5.011e+03 0.000 1.000  
## Vehicle.ModelTRANSIT 350 6.582e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelTRANSIT BUS 6.948e-07 5.343e+03 0.000 1.000  
## Vehicle.ModelTRANSIT CONNECT 6.947e-07 5.678e+03 0.000 1.000  
## Vehicle.ModelTRAVERSE 6.948e-07 7.264e+03 0.000 1.000  
## Vehicle.ModelTRAX 6.948e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelTROLLEY 1.500e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelTUCSON 6.944e-07 8.751e+03 0.000 1.000  
## Vehicle.ModelTX350 6.979e-07 1.014e+04 0.000 1.000  
## Vehicle.ModelUPLANDER 1.786e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelUTILITY 6.945e-07 6.954e+03 0.000 1.000  
## Vehicle.ModelVAN 6.948e-07 5.660e+03 0.000 1.000  
## Vehicle.ModelVENTURE 1.987e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelVENZA 6.948e-07 5.478e+03 0.000 1.000  
## Vehicle.ModelVIP 1.581e+01 4.918e+03 0.003 0.997  
## Vehicle.ModelVIS-A-VIS 6.947e-07 6.533e+03 0.000 1.000  
## Vehicle.ModelVOLT 6.950e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelVOYAGER 2.126e+01 4.918e+03 0.004 0.997  
## Vehicle.ModelWAGON 6.948e-07 5.737e+03 0.000 1.000  
## Vehicle.ModelWAGONEER 6.949e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelWINDSTAR 6.948e-07 1.136e+04 0.000 1.000  
## Vehicle.ModelX 6.948e-07 7.458e+03 0.000 1.000  
## Vehicle.ModelX LINE SHUTTLE 6.949e-07 5.951e+03 0.000 1.000  
## Vehicle.ModelX5 6.948e-07 5.918e+03 0.000 1.000  
## Vehicle.ModelX6 6.961e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelX6M 6.948e-07 1.840e+04 0.000 1.000  
## Vehicle.ModelX7 6.948e-07 9.330e+03 0.000 1.000  
## Vehicle.ModelXB 6.942e-07 4.968e+03 0.000 1.000  
## Vehicle.ModelXBS 6.951e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelXC90 6.948e-07 5.831e+03 0.000 1.000  
## Vehicle.ModelXT5 6.943e-07 1.347e+04 0.000 1.000  
## Vehicle.ModelXT6 6.952e-07 6.533e+03 0.000 1.000  
## Vehicle.ModelXTS 6.949e-07 4.937e+03 0.000 1.000  
## Vehicle.ModelY 6.949e-07 5.478e+03 0.000 1.000  
## Vehicle.ModelYUKON 1.347e+01 4.918e+03 0.003 0.998  
## Vehicle.ModelYUKON XL 6.948e-07 5.070e+03 0.000 1.000  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879.1 on 75959 degrees of freedom  
## Residual deviance: 7118.2 on 75635 degrees of freedom  
## AIC: 7768.2  
##   
## Number of Fisher Scoring iterations: 19

Vehicle.Model also doesn’t affect the Wheelchair Accessibility

vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Model.Year, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Model.Year, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.788e+02 9.710e+00 -28.71 <2e-16 \*\*\*  
## Vehicle.Model.Year 1.368e-01 4.817e-03 28.40 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 25101 on 75958 degrees of freedom  
## AIC: 25105  
##   
## Number of Fisher Scoring iterations: 6

Wheelchair.Model.Year is statistically significant

vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Color, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Color, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value  
## (Intercept) -1.757e+01 1.495e+03 -0.012  
## Vehicle.ColorBEIGE -6.502e-06 1.727e+03 0.000  
## Vehicle.ColorBLACK 1.094e+01 1.495e+03 0.007  
## Vehicle.ColorBLACK/GREEN -6.502e-06 3.172e+03 0.000  
## Vehicle.ColorBLACK/RED -6.507e-06 1.777e+03 0.000  
## Vehicle.ColorBLACK/RED/WHITE/BLUE -6.504e-06 3.172e+03 0.000  
## Vehicle.ColorBLACK/SILVER STRIPE -6.504e-06 2.115e+03 0.000  
## Vehicle.ColorBLACK/WHITE 1.553e+01 1.495e+03 0.010  
## Vehicle.ColorBLACK/WHITE/RED/BLUE -6.498e-06 1.913e+03 0.000  
## Vehicle.ColorBLACK/WHITE/STRIPES -6.502e-06 3.172e+03 0.000  
## Vehicle.ColorBLACK/YELLOW -6.492e-06 2.317e+03 0.000  
## Vehicle.ColorBLUE 1.442e+01 1.495e+03 0.010  
## Vehicle.ColorBLUE/BLACK -6.502e-06 1.882e+03 0.000  
## Vehicle.ColorBLUE/GREEN -6.502e-06 2.480e+03 0.000  
## Vehicle.ColorBLUE/ORANGE -6.502e-06 2.730e+03 0.000  
## Vehicle.ColorBLUE/RED -6.502e-06 2.201e+03 0.000  
## Vehicle.ColorBLUE/WHITE 1.474e+01 1.495e+03 0.010  
## Vehicle.ColorBLUE/WHITE/BLACK -6.502e-06 2.048e+03 0.000  
## Vehicle.ColorBROWN -6.502e-06 1.708e+03 0.000  
## Vehicle.ColorBURGUNDY 1.339e+01 1.495e+03 0.009  
## Vehicle.ColorCREAM -6.481e-06 3.172e+03 0.000  
## Vehicle.ColorCREAM/BLACK 1.423e+01 1.495e+03 0.010  
## Vehicle.ColorCREAM/BLACK/RED STRIPES -6.501e-06 1.634e+03 0.000  
## Vehicle.ColorCREAM/BLUE 1.467e+01 1.495e+03 0.010  
## Vehicle.ColorCREAM/GREEN 1.298e+01 1.495e+03 0.009  
## Vehicle.ColorDARK BLUE -6.502e-06 1.692e+03 0.000  
## Vehicle.ColorDARK GRAY 1.562e+01 1.495e+03 0.010  
## Vehicle.ColorDARK GREEN -6.502e-06 2.317e+03 0.000  
## Vehicle.ColorFUSCIA -6.506e-06 1.994e+03 0.000  
## Vehicle.ColorGOLD 1.198e+01 1.495e+03 0.008  
## Vehicle.ColorGRAY 1.493e+01 1.495e+03 0.010  
## Vehicle.ColorGRAY/BLACK -6.499e-06 2.317e+03 0.000  
## Vehicle.ColorGRAY/BLACK/WHITE -6.502e-06 2.730e+03 0.000  
## Vehicle.ColorGRAY/BLUE/STRIPS -6.506e-06 3.172e+03 0.000  
## Vehicle.ColorGRAY/WHITE 1.369e+01 1.495e+03 0.009  
## Vehicle.ColorGREEN 1.544e+01 1.495e+03 0.010  
## Vehicle.ColorGREEN WITH RED STRIPE -6.502e-06 3.172e+03 0.000  
## Vehicle.ColorGREEN/WHITE/BLACK -6.492e-06 2.048e+03 0.000  
## Vehicle.ColorLAVENDER -6.511e-06 1.607e+03 0.000  
## Vehicle.ColorMAROON 1.351e+01 1.495e+03 0.009  
## Vehicle.ColorMAROON WITH WHITE STRIPES 1.445e+01 1.495e+03 0.010  
## Vehicle.ColorMAROON/BLUE/STRIPES 1.452e+01 1.495e+03 0.010  
## Vehicle.ColorMAROON/GOLD/STRIPES 1.504e+01 1.495e+03 0.010  
## Vehicle.ColorMAROON/GREEN/STRIPES -6.500e-06 1.692e+03 0.000  
## Vehicle.ColorNAVY -6.498e-06 2.730e+03 0.000  
## Vehicle.ColorNAVY BLUE -6.505e-06 2.317e+03 0.000  
## Vehicle.ColorNONE -6.521e-06 2.480e+03 0.000  
## Vehicle.ColorORANGE 1.536e+01 1.495e+03 0.010  
## Vehicle.ColorORANGE/BLUE 1.448e+01 1.495e+03 0.010  
## Vehicle.ColorORANGE/WHITE -6.501e-06 2.115e+03 0.000  
## Vehicle.ColorPINK/WHITE -6.499e-06 1.855e+03 0.000  
## Vehicle.ColorPURPLE -6.501e-06 1.638e+03 0.000  
## Vehicle.ColorRED 1.411e+01 1.495e+03 0.009  
## Vehicle.ColorRED/BLACK -6.500e-06 1.950e+03 0.000  
## Vehicle.ColorRED/BLACK/YELLOW STRIPE -6.502e-06 3.172e+03 0.000  
## Vehicle.ColorRED/BLUE -6.502e-06 1.913e+03 0.000  
## Vehicle.ColorRED/BLUE/WHITE -6.513e-06 1.610e+03 0.000  
## Vehicle.ColorRED/CREAM 1.381e+01 1.495e+03 0.009  
## Vehicle.ColorRED/GREEN -6.504e-06 1.526e+03 0.000  
## Vehicle.ColorRED/ORANGE -6.491e-06 3.172e+03 0.000  
## Vehicle.ColorRED/WHITE -6.502e-06 1.549e+03 0.000  
## Vehicle.ColorRED/WHITE/BLUE -6.502e-06 1.737e+03 0.000  
## Vehicle.ColorRED/WHITE/CHECKER -6.502e-06 3.172e+03 0.000  
## Vehicle.ColorSAGE -6.501e-06 3.172e+03 0.000  
## Vehicle.ColorSILVER -6.502e-06 1.506e+03 0.000  
## Vehicle.ColorSILVER/BLACK -6.502e-06 1.793e+03 0.000  
## Vehicle.ColorSILVER/BLACK STRIPE -6.509e-06 2.480e+03 0.000  
## Vehicle.ColorSILVER/GRAY -6.500e-06 1.913e+03 0.000  
## Vehicle.ColorTAN -6.499e-06 2.115e+03 0.000  
## Vehicle.ColorTAN/RED -6.502e-06 2.730e+03 0.000  
## Vehicle.ColorTAN/RED/BLACK -6.486e-06 2.730e+03 0.000  
## Vehicle.ColorTEAL -6.511e-06 3.172e+03 0.000  
## Vehicle.ColorTURQUOISE -6.530e-06 4.229e+03 0.000  
## Vehicle.ColorWHITE 1.484e+01 1.495e+03 0.010  
## Vehicle.ColorWHITE/BLACK 1.586e+01 1.495e+03 0.011  
## Vehicle.ColorWHITE/BLACK/GOLD -6.502e-06 4.229e+03 0.000  
## Vehicle.ColorWHITE/BLACK/RED -6.500e-06 3.172e+03 0.000  
## Vehicle.ColorWHITE/BLACK/STRIPES 1.579e+01 1.495e+03 0.011  
## Vehicle.ColorWHITE/BLUE 1.527e+01 1.495e+03 0.010  
## Vehicle.ColorWHITE/BLUE LETTERS -6.499e-06 1.762e+03 0.000  
## Vehicle.ColorWHITE/BLUE/RED -6.501e-06 2.317e+03 0.000  
## Vehicle.ColorWHITE/BLUE/STRIPES 1.457e+01 1.495e+03 0.010  
## Vehicle.ColorWHITE/GRAY -6.502e-06 1.994e+03 0.000  
## Vehicle.ColorWHITE/GREEN -6.502e-06 2.201e+03 0.000  
## Vehicle.ColorWHITE/ORANGE -6.488e-06 1.913e+03 0.000  
## Vehicle.ColorWHITE/ORANGE STRIPE/BLUE LETTERS -6.502e-06 2.730e+03 0.000  
## Vehicle.ColorWHITE/PURPLE -6.502e-06 2.201e+03 0.000  
## Vehicle.ColorWHITE/RED 1.397e+01 1.495e+03 0.009  
## Vehicle.ColorWHITE/RED STRIPES 1.566e+01 1.495e+03 0.010  
## Vehicle.ColorWHITE/RED/BLACK -6.501e-06 1.627e+03 0.000  
## Vehicle.ColorWHITE/RED/BLUE -6.502e-06 1.950e+03 0.000  
## Vehicle.ColorWHITE/RED/STRIPES 1.480e+01 1.495e+03 0.010  
## Vehicle.ColorWHITE/SILVER -6.501e-06 1.727e+03 0.000  
## Vehicle.ColorWHITE/SILVER/BLUE STRIPES -6.505e-06 2.317e+03 0.000  
## Vehicle.ColorWHITE/STARS/STRIPES 1.484e+01 1.495e+03 0.010  
## Vehicle.ColorWHITE/YELLOW -6.502e-06 1.913e+03 0.000  
## Vehicle.ColorYELLOW 1.479e+01 1.495e+03 0.010  
## Vehicle.ColorYELLOW/BLACK -6.501e-06 1.950e+03 0.000  
## Vehicle.ColorYELLOW/BLUE -6.501e-06 1.777e+03 0.000  
## Pr(>|z|)  
## (Intercept) 0.991  
## Vehicle.ColorBEIGE 1.000  
## Vehicle.ColorBLACK 0.994  
## Vehicle.ColorBLACK/GREEN 1.000  
## Vehicle.ColorBLACK/RED 1.000  
## Vehicle.ColorBLACK/RED/WHITE/BLUE 1.000  
## Vehicle.ColorBLACK/SILVER STRIPE 1.000  
## Vehicle.ColorBLACK/WHITE 0.992  
## Vehicle.ColorBLACK/WHITE/RED/BLUE 1.000  
## Vehicle.ColorBLACK/WHITE/STRIPES 1.000  
## Vehicle.ColorBLACK/YELLOW 1.000  
## Vehicle.ColorBLUE 0.992  
## Vehicle.ColorBLUE/BLACK 1.000  
## Vehicle.ColorBLUE/GREEN 1.000  
## Vehicle.ColorBLUE/ORANGE 1.000  
## Vehicle.ColorBLUE/RED 1.000  
## Vehicle.ColorBLUE/WHITE 0.992  
## Vehicle.ColorBLUE/WHITE/BLACK 1.000  
## Vehicle.ColorBROWN 1.000  
## Vehicle.ColorBURGUNDY 0.993  
## Vehicle.ColorCREAM 1.000  
## Vehicle.ColorCREAM/BLACK 0.992  
## Vehicle.ColorCREAM/BLACK/RED STRIPES 1.000  
## Vehicle.ColorCREAM/BLUE 0.992  
## Vehicle.ColorCREAM/GREEN 0.993  
## Vehicle.ColorDARK BLUE 1.000  
## Vehicle.ColorDARK GRAY 0.992  
## Vehicle.ColorDARK GREEN 1.000  
## Vehicle.ColorFUSCIA 1.000  
## Vehicle.ColorGOLD 0.994  
## Vehicle.ColorGRAY 0.992  
## Vehicle.ColorGRAY/BLACK 1.000  
## Vehicle.ColorGRAY/BLACK/WHITE 1.000  
## Vehicle.ColorGRAY/BLUE/STRIPS 1.000  
## Vehicle.ColorGRAY/WHITE 0.993  
## Vehicle.ColorGREEN 0.992  
## Vehicle.ColorGREEN WITH RED STRIPE 1.000  
## Vehicle.ColorGREEN/WHITE/BLACK 1.000  
## Vehicle.ColorLAVENDER 1.000  
## Vehicle.ColorMAROON 0.993  
## Vehicle.ColorMAROON WITH WHITE STRIPES 0.992  
## Vehicle.ColorMAROON/BLUE/STRIPES 0.992  
## Vehicle.ColorMAROON/GOLD/STRIPES 0.992  
## Vehicle.ColorMAROON/GREEN/STRIPES 1.000  
## Vehicle.ColorNAVY 1.000  
## Vehicle.ColorNAVY BLUE 1.000  
## Vehicle.ColorNONE 1.000  
## Vehicle.ColorORANGE 0.992  
## Vehicle.ColorORANGE/BLUE 0.992  
## Vehicle.ColorORANGE/WHITE 1.000  
## Vehicle.ColorPINK/WHITE 1.000  
## Vehicle.ColorPURPLE 1.000  
## Vehicle.ColorRED 0.992  
## Vehicle.ColorRED/BLACK 1.000  
## Vehicle.ColorRED/BLACK/YELLOW STRIPE 1.000  
## Vehicle.ColorRED/BLUE 1.000  
## Vehicle.ColorRED/BLUE/WHITE 1.000  
## Vehicle.ColorRED/CREAM 0.993  
## Vehicle.ColorRED/GREEN 1.000  
## Vehicle.ColorRED/ORANGE 1.000  
## Vehicle.ColorRED/WHITE 1.000  
## Vehicle.ColorRED/WHITE/BLUE 1.000  
## Vehicle.ColorRED/WHITE/CHECKER 1.000  
## Vehicle.ColorSAGE 1.000  
## Vehicle.ColorSILVER 1.000  
## Vehicle.ColorSILVER/BLACK 1.000  
## Vehicle.ColorSILVER/BLACK STRIPE 1.000  
## Vehicle.ColorSILVER/GRAY 1.000  
## Vehicle.ColorTAN 1.000  
## Vehicle.ColorTAN/RED 1.000  
## Vehicle.ColorTAN/RED/BLACK 1.000  
## Vehicle.ColorTEAL 1.000  
## Vehicle.ColorTURQUOISE 1.000  
## Vehicle.ColorWHITE 0.992  
## Vehicle.ColorWHITE/BLACK 0.992  
## Vehicle.ColorWHITE/BLACK/GOLD 1.000  
## Vehicle.ColorWHITE/BLACK/RED 1.000  
## Vehicle.ColorWHITE/BLACK/STRIPES 0.992  
## Vehicle.ColorWHITE/BLUE 0.992  
## Vehicle.ColorWHITE/BLUE LETTERS 1.000  
## Vehicle.ColorWHITE/BLUE/RED 1.000  
## Vehicle.ColorWHITE/BLUE/STRIPES 0.992  
## Vehicle.ColorWHITE/GRAY 1.000  
## Vehicle.ColorWHITE/GREEN 1.000  
## Vehicle.ColorWHITE/ORANGE 1.000  
## Vehicle.ColorWHITE/ORANGE STRIPE/BLUE LETTERS 1.000  
## Vehicle.ColorWHITE/PURPLE 1.000  
## Vehicle.ColorWHITE/RED 0.993  
## Vehicle.ColorWHITE/RED STRIPES 0.992  
## Vehicle.ColorWHITE/RED/BLACK 1.000  
## Vehicle.ColorWHITE/RED/BLUE 1.000  
## Vehicle.ColorWHITE/RED/STRIPES 0.992  
## Vehicle.ColorWHITE/SILVER 1.000  
## Vehicle.ColorWHITE/SILVER/BLUE STRIPES 1.000  
## Vehicle.ColorWHITE/STARS/STRIPES 0.992  
## Vehicle.ColorWHITE/YELLOW 1.000  
## Vehicle.ColorYELLOW 0.992  
## Vehicle.ColorYELLOW/BLACK 1.000  
## Vehicle.ColorYELLOW/BLUE 1.000  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 23330 on 75861 degrees of freedom  
## AIC: 23528  
##   
## Number of Fisher Scoring iterations: 16

Vehicle.Color is not statistically significant for y

vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Fuel.Source, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Fuel.Source, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value  
## (Intercept) -1.557e+01 2.393e+02 -0.065  
## Vehicle.Fuel.SourceCompressed Natural Gas 1.539e+01 2.393e+02 0.064  
## Vehicle.Fuel.SourceDiesel 1.041e+01 2.393e+02 0.044  
## Vehicle.Fuel.SourceElectric 2.476e-07 2.833e+02 0.000  
## Vehicle.Fuel.SourceFlex Fuel 1.497e+01 2.393e+02 0.063  
## Vehicle.Fuel.SourceGasoline 1.266e+01 2.393e+02 0.053  
## Vehicle.Fuel.SourceHorse 2.472e-07 4.264e+02 0.000  
## Vehicle.Fuel.SourceHybrid 1.106e+01 2.393e+02 0.046  
## Vehicle.Fuel.SourcePedal 9.359e+00 2.393e+02 0.039  
## Pr(>|z|)  
## (Intercept) 0.948  
## Vehicle.Fuel.SourceCompressed Natural Gas 0.949  
## Vehicle.Fuel.SourceDiesel 0.965  
## Vehicle.Fuel.SourceElectric 1.000  
## Vehicle.Fuel.SourceFlex Fuel 0.950  
## Vehicle.Fuel.SourceGasoline 0.958  
## Vehicle.Fuel.SourceHorse 1.000  
## Vehicle.Fuel.SourceHybrid 0.963  
## Vehicle.Fuel.SourcePedal 0.969  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 21551 on 75951 degrees of freedom  
## AIC: 21569  
##   
## Number of Fisher Scoring iterations: 14

Vehicle.Fuel.Source is also considered not important for the vehicle.

For the following columns, training data set was too large, therefore, I decided to make a very little data subset just to check whether those predictors are significant or not.

vehicles.glm = glm(Wheelchair.Accessible ~ CompanyGroup, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ CompanyGroup, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.13576 0.02638 -118.891 < 2e-16 \*\*\*  
## CompanyGroupTop 2000 0.22354 0.05010 4.461 8.14e-06 \*\*\*  
## CompanyGroupTop 3000 0.12749 0.06202 2.055 0.03983 \*   
## CompanyGroupTop 4000 0.10578 0.07065 1.497 0.13436   
## CompanyGroupTop 5000 -0.14615 0.08522 -1.715 0.08634 .   
## CompanyGroupTop 6000 -0.30437 0.10039 -3.032 0.00243 \*\*   
## CompanyGroupTop 7000 -0.32091 0.11026 -2.911 0.00361 \*\*   
## CompanyGroupTop 8000 -0.58865 0.13778 -4.272 1.93e-05 \*\*\*  
## CompanyGroupTop 9000 -1.34278 0.23222 -5.782 7.36e-09 \*\*\*  
## CompanyGroupTop 10000 -2.27142 0.50181 -4.526 6.00e-06 \*\*\*  
## CompanyGroupOther -3.06475 1.00086 -3.062 0.00220 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 25661 on 75949 degrees of freedom  
## AIC: 25683  
##   
## Number of Fisher Scoring iterations: 8

CompanyGroup is a significant predictor

vehicles.glm = glm(Wheelchair.Accessible ~ AddressesGroup, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ AddressesGroup, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.09739 0.01878 -164.918 < 2e-16 \*\*\*  
## AddressesGroupTop 2000 -0.69102 0.11065 -6.245 4.24e-10 \*\*\*  
## AddressesGroupTop 3000 -1.01629 0.17144 -5.928 3.07e-09 \*\*\*  
## AddressesGroupTop 4000 -1.52491 0.30357 -5.023 5.08e-07 \*\*\*  
## AddressesGroupOther -2.21582 1.00178 -2.212 0.027 \*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 25733 on 75955 degrees of freedom  
## AIC: 25743  
##   
## Number of Fisher Scoring iterations: 7

AddressesGroup is also a significant predictor

vehicles.glm = glm(Wheelchair.Accessible ~ City, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ City, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -1.657e+01 1.385e+03 -0.012 0.990  
## CityADDISON 1.772e-05 1.833e+03 0.000 1.000  
## CityARLINGTON HEIGHTS 1.772e-05 2.771e+03 0.000 1.000  
## CityBARRINGTON 1.772e-05 1.481e+03 0.000 1.000  
## CityBARTLETT 1.772e-05 1.424e+03 0.000 1.000  
## CityBENSENVILLE 1.772e-05 2.190e+03 0.000 1.000  
## CityBOILINGBROOK 1.772e-05 2.190e+03 0.000 1.000  
## CityBOLINGBROOK 1.772e-05 2.190e+03 0.000 1.000  
## CityBRIDGEVIEW 1.771e-05 1.959e+03 0.000 1.000  
## CityBROADVIEW 1.772e-05 2.771e+03 0.000 1.000  
## CityBROOKFIELD 1.772e-05 1.624e+03 0.000 1.000  
## CityBUFFALO GROVE 1.772e-05 2.771e+03 0.000 1.000  
## CityBURBANK 1.772e-05 1.833e+03 0.000 1.000  
## CityCHCAGO 1.772e-05 1.563e+03 0.000 1.000  
## CityCHGO 1.887e+01 1.385e+03 0.014 0.989  
## CityCHICAGO 1.343e+01 1.385e+03 0.010 0.992  
## CityCHICAGO RIDGE 1.772e-05 1.656e+03 0.000 1.000  
## CityCHIICAGO 1.772e-05 1.537e+03 0.000 1.000  
## CityCREST HILL 1.772e-05 2.771e+03 0.000 1.000  
## CityCRYSTAL LAKE 1.772e-05 1.503e+03 0.000 1.000  
## CityDEERFIELD 1.772e-05 2.771e+03 0.000 1.000  
## CityDES PLAINES 1.772e-05 1.442e+03 0.000 1.000  
## CityDOLTON 1.772e-05 1.428e+03 0.000 1.000  
## CityDOWNERS GROVE 1.772e-05 2.771e+03 0.000 1.000  
## CityEARLVILLE 1.772e-05 1.510e+03 0.000 1.000  
## CityELGIN 1.772e-05 1.549e+03 0.000 1.000  
## CityELK GROVE VILLAGE 1.772e-05 1.580e+03 0.000 1.000  
## CityELMHURST 1.020e+01 1.385e+03 0.007 0.994  
## CityEVERGREEN PARK 1.772e-05 1.959e+03 0.000 1.000  
## CityFOREST PARK 1.772e-05 1.624e+03 0.000 1.000  
## CityFRANKFORT 1.772e-05 1.697e+03 0.000 1.000  
## CityGLENVIEW 1.772e-05 1.833e+03 0.000 1.000  
## CityGRAYSLAKE 1.771e-05 2.771e+03 0.000 1.000  
## CityGURNEE 1.771e-05 2.771e+03 0.000 1.000  
## CityHOFFMAN ESTATES 1.771e-05 2.771e+03 0.000 1.000  
## CityITASCA 1.772e-05 1.549e+03 0.000 1.000  
## CityJOHNSBURG 1.771e-05 2.771e+03 0.000 1.000  
## CityJUSTICE 1.772e-05 2.190e+03 0.000 1.000  
## CityLAGRANGE PARK 1.771e-05 2.771e+03 0.000 1.000  
## CityLIBERTYVILLE 1.772e-05 2.190e+03 0.000 1.000  
## CityLISLE 1.772e-05 1.549e+03 0.000 1.000  
## CityLOMBARD 1.771e-05 2.771e+03 0.000 1.000  
## CityLONG GROVE 1.771e-05 2.771e+03 0.000 1.000  
## CityLOVES PARK 1.772e-05 1.656e+03 0.000 1.000  
## CityLYNWOOD 1.772e-05 1.656e+03 0.000 1.000  
## CityMENOMONEE FALLS 1.771e-05 2.771e+03 0.000 1.000  
## CityMETTAWA 1.772e-05 2.190e+03 0.000 1.000  
## CityMOKENA 1.772e-05 1.402e+03 0.000 1.000  
## CityMORTON GROVE 1.772e-05 1.959e+03 0.000 1.000  
## CityMOUNT PROSPECT 1.772e-05 2.190e+03 0.000 1.000  
## CityMT PROSPECT 1.772e-05 1.959e+03 0.000 1.000  
## CityMUNDELEIN 1.772e-05 2.190e+03 0.000 1.000  
## CityNAPERVILLE 1.772e-05 1.477e+03 0.000 1.000  
## CityNEW LENOX 1.772e-05 1.439e+03 0.000 1.000  
## CityNILES 1.772e-05 1.697e+03 0.000 1.000  
## CityNORRIDGE 1.772e-05 2.771e+03 0.000 1.000  
## CityNORTHBROOK 1.772e-05 1.697e+03 0.000 1.000  
## CityOAKBROOK 1.772e-05 1.697e+03 0.000 1.000  
## CityORLAND PARK 1.772e-05 1.405e+03 0.000 1.000  
## CityOSWEGO 1.772e-05 2.190e+03 0.000 1.000  
## CityPARK 1.771e-05 2.771e+03 0.000 1.000  
## CityRIVER GROVE 1.772e-05 2.190e+03 0.000 1.000  
## CityROLLING MEADOWS 1.771e-05 2.771e+03 0.000 1.000  
## CityROMEOVILLE 1.772e-05 2.190e+03 0.000 1.000  
## CityROSELLE 1.772e-05 2.190e+03 0.000 1.000  
## CityROUND LAKE 1.771e-05 2.771e+03 0.000 1.000  
## CityS CHICAGO HEIGHT 1.771e-05 2.771e+03 0.000 1.000  
## CitySCHILLER PARK 1.772e-05 1.833e+03 0.000 1.000  
## CitySHOREWOOD 1.772e-05 1.518e+03 0.000 1.000  
## CitySKOKIE 1.053e+01 1.385e+03 0.008 0.994  
## CityST. CHARLES 1.772e-05 1.959e+03 0.000 1.000  
## CitySTREAMWOOD 1.772e-05 2.771e+03 0.000 1.000  
## CityTINLEY PARK 1.772e-05 1.393e+03 0.000 1.000  
## CityWEST CHICAGO 1.772e-05 2.771e+03 0.000 1.000  
## CityWESTMONT 1.772e-05 2.771e+03 0.000 1.000  
## CityWHEELING 1.772e-05 2.771e+03 0.000 1.000  
## CityWOODDALE 1.772e-05 2.190e+03 0.000 1.000  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 25616 on 75883 degrees of freedom  
## AIC: 25770  
##   
## Number of Fisher Scoring iterations: 15

City is not statistically significant

vehicles.glm = glm(Wheelchair.Accessible ~ RecordsGroup, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ RecordsGroup, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.44679 0.03352 -72.993 < 2e-16 \*\*\*  
## RecordsGroupTop 2000 -0.43804 0.05641 -7.766 8.12e-15 \*\*\*  
## RecordsGroupTop 3000 -0.54154 0.06123 -8.844 < 2e-16 \*\*\*  
## RecordsGroupTop 4000 -0.93099 0.07230 -12.877 < 2e-16 \*\*\*  
## RecordsGroupTop 5000 -1.13542 0.08065 -14.078 < 2e-16 \*\*\*  
## RecordsGroupTop 6000 -1.07989 0.08215 -13.145 < 2e-16 \*\*\*  
## RecordsGroupTop 7000 -0.77098 0.07692 -10.023 < 2e-16 \*\*\*  
## RecordsGroupTop 8000 -1.22468 0.09624 -12.725 < 2e-16 \*\*\*  
## RecordsGroupTop 9000 -1.33026 0.10703 -12.429 < 2e-16 \*\*\*  
## RecordsGroupTop 10000 -1.28866 0.11414 -11.290 < 2e-16 \*\*\*  
## RecordsGroupTop 11000 -2.01093 0.18373 -10.945 < 2e-16 \*\*\*  
## RecordsGroupTop 12000 -1.67387 0.18708 -8.947 < 2e-16 \*\*\*  
## RecordsGroupTop 13000 -4.38517 1.00110 -4.380 1.18e-05 \*\*\*  
## RecordsGroupOther -13.11928 78.58409 -0.167 0.867   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 25051 on 75946 degrees of freedom  
## AIC: 25079  
##   
## Number of Fisher Scoring iterations: 14

RecordsGroup also appears as statistically significant

vehicles.glm = glm(Wheelchair.Accessible ~ ZIP.Code, data = vehicles.train,   
 family = binomial)

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ ZIP.Code, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -5.236e+02 4.870e+01 -10.75 <2e-16 \*\*\*  
## ZIP.Code 8.584e-03 8.032e-04 10.69 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 25669 on 75958 degrees of freedom  
## AIC: 25673  
##   
## Number of Fisher Scoring iterations: 7

ZIP code appears to be statistically significant.

vehicles.glm = glm(Wheelchair.Accessible ~ DatesGroup, data = vehicles.train,   
 family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ DatesGroup, family = binomial,   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.27131 0.02313 -141.427 < 2e-16 \*\*\*  
## DatesGroupTop 2000 -0.02292 0.05784 -0.396 0.691942   
## DatesGroupTop 3000 0.43255 0.06994 6.185 6.21e-10 \*\*\*  
## DatesGroupTop 4000 0.55227 0.08140 6.785 1.16e-11 \*\*\*  
## DatesGroupTop 5000 0.43757 0.10037 4.360 1.30e-05 \*\*\*  
## DatesGroupTop 6000 0.90784 0.09231 9.835 < 2e-16 \*\*\*  
## DatesGroupTop 7000 0.54856 0.14237 3.853 0.000117 \*\*\*  
## DatesGroupTop 8000 0.79722 0.13425 5.938 2.88e-09 \*\*\*  
## DatesGroupOther 0.31640 0.45929 0.689 0.490895   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 25693 on 75951 degrees of freedom  
## AIC: 25711  
##   
## Number of Fisher Scoring iterations: 6

DatesGroup also is statistically significant

Now, as we undesrtood statistically significant predictors, we can make a real model.

vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Type + Status + Vehicle.Model.Year   
 + Public.Vehicle.Number + ZIP.Code + AddressesGroup + CompanyGroup  
 + RecordsGroup + DatesGroup, data = vehicles.train, family = "binomial")

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Type + Status +   
## Vehicle.Model.Year + Public.Vehicle.Number + ZIP.Code + AddressesGroup +   
## CompanyGroup + RecordsGroup + DatesGroup, family = "binomial",   
## data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.021e+03 5.984e+01 -17.069 < 2e-16 \*\*\*  
## Vehicle.TypeCharter Sightseeing 3.148e-01 1.128e+00 0.279 0.780201   
## Vehicle.TypeHorse Drawn Carriage -9.219e+00 4.559e+02 -0.020 0.983867   
## Vehicle.TypeJitney -8.915e+00 2.424e+02 -0.037 0.970663   
## Vehicle.TypeLivery -3.024e+00 1.072e+00 -2.822 0.004772 \*\*   
## Vehicle.TypeLow Speed Electric -9.729e+00 6.926e+02 -0.014 0.988792   
## Vehicle.TypeMedicar 3.574e+00 1.046e+00 3.417 0.000633 \*\*\*  
## Vehicle.TypePedicab -1.461e+00 1.251e+00 -1.169 0.242563   
## Vehicle.TypeTaxi 2.694e+00 1.031e+00 2.612 0.009003 \*\*   
## StatusFORECLOSURE 4.023e-01 6.954e-02 5.786 7.23e-09 \*\*\*  
## StatusHOLD 1.573e+00 5.599e-01 2.809 0.004970 \*\*   
## StatusINACTIVE -1.401e+00 4.879e-01 -2.872 0.004077 \*\*   
## StatusRESERVED -1.849e+00 7.417e-01 -2.493 0.012680 \*   
## StatusREVOKED -9.932e-01 3.607e-01 -2.754 0.005890 \*\*   
## StatusSURRENDER -1.320e-01 5.165e-02 -2.556 0.010585 \*   
## StatusVIOLATION -6.689e-02 5.138e-02 -1.302 0.192976   
## Vehicle.Model.Year 3.497e-01 7.326e-03 47.736 < 2e-16 \*\*\*  
## Public.Vehicle.Number -1.530e-05 8.628e-06 -1.774 0.076107 .   
## ZIP.Code 5.147e-03 9.605e-04 5.359 8.39e-08 \*\*\*  
## AddressesGroupTop 2000 8.504e-02 1.276e-01 0.667 0.505090   
## AddressesGroupTop 3000 1.107e-01 1.916e-01 0.578 0.563257   
## AddressesGroupTop 4000 -9.146e-02 3.383e-01 -0.270 0.786903   
## AddressesGroupOther -2.363e-01 1.058e+00 -0.223 0.823300   
## CompanyGroupTop 2000 -1.360e-01 5.488e-02 -2.478 0.013222 \*   
## CompanyGroupTop 3000 -5.736e-01 7.037e-02 -8.151 3.60e-16 \*\*\*  
## CompanyGroupTop 4000 -6.802e-01 8.305e-02 -8.190 2.62e-16 \*\*\*  
## CompanyGroupTop 5000 -1.073e+00 9.866e-02 -10.876 < 2e-16 \*\*\*  
## CompanyGroupTop 6000 -1.211e+00 1.150e-01 -10.531 < 2e-16 \*\*\*  
## CompanyGroupTop 7000 -1.240e+00 1.263e-01 -9.818 < 2e-16 \*\*\*  
## CompanyGroupTop 8000 -1.525e+00 1.584e-01 -9.626 < 2e-16 \*\*\*  
## CompanyGroupTop 9000 -1.758e+00 2.502e-01 -7.024 2.15e-12 \*\*\*  
## CompanyGroupTop 10000 -1.486e+00 5.566e-01 -2.669 0.007599 \*\*   
## CompanyGroupOther -1.705e+00 1.068e+00 -1.597 0.110278   
## RecordsGroupTop 2000 -1.330e-01 6.051e-02 -2.198 0.027939 \*   
## RecordsGroupTop 3000 -2.132e-01 6.513e-02 -3.274 0.001061 \*\*   
## RecordsGroupTop 4000 -2.339e-01 7.676e-02 -3.048 0.002306 \*\*   
## RecordsGroupTop 5000 -1.552e-01 8.651e-02 -1.794 0.072849 .   
## RecordsGroupTop 6000 -2.064e-01 8.827e-02 -2.338 0.019392 \*   
## RecordsGroupTop 7000 -2.443e-01 8.296e-02 -2.944 0.003239 \*\*   
## RecordsGroupTop 8000 -3.022e-01 1.022e-01 -2.956 0.003121 \*\*   
## RecordsGroupTop 9000 -1.096e-01 1.128e-01 -0.971 0.331392   
## RecordsGroupTop 10000 -9.947e-02 1.214e-01 -0.819 0.412527   
## RecordsGroupTop 11000 -3.873e-01 2.199e-01 -1.761 0.078211 .   
## RecordsGroupTop 12000 -1.798e-01 2.559e-01 -0.703 0.482143   
## RecordsGroupTop 13000 -2.994e+00 1.043e+00 -2.869 0.004114 \*\*   
## RecordsGroupOther -1.137e+01 1.006e+02 -0.113 0.910007   
## DatesGroupTop 2000 4.445e-02 6.143e-02 0.723 0.469377   
## DatesGroupTop 3000 6.171e-02 7.844e-02 0.787 0.431490   
## DatesGroupTop 4000 9.612e-02 9.200e-02 1.045 0.296099   
## DatesGroupTop 5000 -4.679e-02 1.118e-01 -0.418 0.675622   
## DatesGroupTop 6000 4.742e-01 1.055e-01 4.496 6.92e-06 \*\*\*  
## DatesGroupTop 7000 8.848e-02 1.573e-01 0.562 0.573830   
## DatesGroupTop 8000 2.411e-01 1.529e-01 1.577 0.114814   
## DatesGroupOther 2.891e-01 5.049e-01 0.573 0.566885   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 20308 on 75906 degrees of freedom  
## AIC: 20416  
##   
## Number of Fisher Scoring iterations: 15

From this summary we can understand that Vehicle.Type, Status, Vehicle.Model.Year, Public.Vehicle.Numer, ZIP.Code and Groups variables are indeed statistically significant predictors.

Now we need to see how accurately the model predicts the outcomes (whether the vehicle will be accessible or not)

accessibility.train = vehicles.train$Wheelchair.Accessible  
model.probs = predict(vehicles.glm, vehicles.train, type="response")  
  
summary(model.probs)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000000 0.0004551 0.0225003 0.0407583 0.0506859 0.8069750

model.preds = rep(0, 75960)  
model.preds[model.probs > 0.5] = 1  
  
table(model.preds, accessibility.train)

## accessibility.train  
## model.preds 0 1  
## 0 72745 3029  
## 1 119 67

mean(model.preds == accessibility.train)

## [1] 0.9585571

We can see that the model predicts the accessibility of places with a 96% accuracy on a training set. Let’s see how’s that going in a test set.

accessibility.test = vehicles.test$Wheelchair.Accessible  
model.probs = predict(vehicles.glm, vehicles.test, type="response")  
  
summary(model.probs)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000000 0.0004588 0.0221105 0.0401468 0.0511021 0.7678319

model.preds = rep(0, 18988)  
model.preds[model.probs > 0.5] = 1  
  
table(model.preds, accessibility.test)

## accessibility.test  
## model.preds 0 1  
## 0 18183 762  
## 1 32 11

mean(model.preds == accessibility.test)

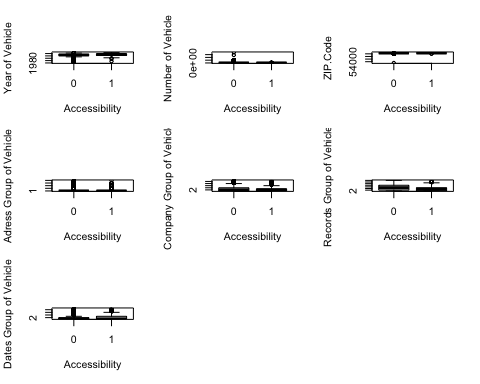
## [1] 0.9581841

The accuracy of prediction on the test set is the same - 96%. It means that the model does really well to identify which vehicles will be accessible and which not.

Now we will plot graphs to visualize our data.

We can make some quick boxplots: Vehicle.Type + Status + Vehicle.Model.Year + Public.Vehicle.Number + ZIP.Code + AddressesGroup + CompanyGroup + RecordsGroup + DatesGroup

attach(vehicles)  
par(mfrow=c(3,3))  
boxplot(Vehicle.Model.Year~Wheelchair.Accessible, xlab="Accessibility", ylab="Year of Vehicle")  
boxplot(Public.Vehicle.Number~Wheelchair.Accessible, xlab="Accessibility", ylab="Number of Vehicle")  
boxplot(ZIP.Code~Wheelchair.Accessible, xlab="Accessibility", ylab="ZIP.Code")  
boxplot(AddressesGroup~Wheelchair.Accessible, xlab="Accessibility", ylab="Adress Group of Vehicle")  
boxplot(CompanyGroup~Wheelchair.Accessible, xlab="Accessibility", ylab="Company Group of Vehicle")  
boxplot(RecordsGroup~Wheelchair.Accessible, xlab="Accessibility", ylab="Records Group of Vehicle")  
boxplot(DatesGroup~Wheelchair.Accessible, xlab="Accessibility", ylab="Dates Group of Vehicle")

 From these simple boxplots we can see that there’s no real relationship between AdressGroup and Accessibility. We can try not to include it into our glm, because also it might be higly correlated with ZIP.Code. Too much data creates overfitting.

vehicles.glm = glm(Wheelchair.Accessible ~ Vehicle.Type + Status + Vehicle.Model.Year   
 + Public.Vehicle.Number + ZIP.Code + CompanyGroup  
 + RecordsGroup + DatesGroup, data = vehicles.train, family = "binomial")

## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Type + Status +   
## Vehicle.Model.Year + Public.Vehicle.Number + ZIP.Code + CompanyGroup +   
## RecordsGroup + DatesGroup, family = "binomial", data = vehicles.train)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.022e+03 5.980e+01 -17.083 < 2e-16 \*\*\*  
## Vehicle.TypeCharter Sightseeing 3.154e-01 1.128e+00 0.280 0.779771   
## Vehicle.TypeHorse Drawn Carriage -9.219e+00 4.560e+02 -0.020 0.983869   
## Vehicle.TypeJitney -8.916e+00 2.424e+02 -0.037 0.970664   
## Vehicle.TypeLivery -3.019e+00 1.071e+00 -2.818 0.004834 \*\*   
## Vehicle.TypeLow Speed Electric -9.721e+00 6.926e+02 -0.014 0.988801   
## Vehicle.TypeMedicar 3.581e+00 1.046e+00 3.424 0.000617 \*\*\*  
## Vehicle.TypePedicab -1.456e+00 1.250e+00 -1.164 0.244347   
## Vehicle.TypeTaxi 2.695e+00 1.031e+00 2.613 0.008976 \*\*   
## StatusFORECLOSURE 4.020e-01 6.952e-02 5.782 7.39e-09 \*\*\*  
## StatusHOLD 1.572e+00 5.599e-01 2.807 0.005000 \*\*   
## StatusINACTIVE -1.401e+00 4.881e-01 -2.871 0.004091 \*\*   
## StatusRESERVED -1.853e+00 7.417e-01 -2.498 0.012497 \*   
## StatusREVOKED -9.930e-01 3.607e-01 -2.753 0.005902 \*\*   
## StatusSURRENDER -1.316e-01 5.164e-02 -2.548 0.010840 \*   
## StatusVIOLATION -6.721e-02 5.138e-02 -1.308 0.190850   
## Vehicle.Model.Year 3.498e-01 7.324e-03 47.766 < 2e-16 \*\*\*  
## Public.Vehicle.Number -1.522e-05 8.624e-06 -1.764 0.077669 .   
## ZIP.Code 5.146e-03 9.600e-04 5.361 8.29e-08 \*\*\*  
## CompanyGroupTop 2000 -1.345e-01 5.484e-02 -2.453 0.014151 \*   
## CompanyGroupTop 3000 -5.708e-01 7.021e-02 -8.129 4.33e-16 \*\*\*  
## CompanyGroupTop 4000 -6.728e-01 8.230e-02 -8.175 2.96e-16 \*\*\*  
## CompanyGroupTop 5000 -1.062e+00 9.718e-02 -10.929 < 2e-16 \*\*\*  
## CompanyGroupTop 6000 -1.201e+00 1.140e-01 -10.539 < 2e-16 \*\*\*  
## CompanyGroupTop 7000 -1.229e+00 1.255e-01 -9.794 < 2e-16 \*\*\*  
## CompanyGroupTop 8000 -1.513e+00 1.564e-01 -9.673 < 2e-16 \*\*\*  
## CompanyGroupTop 9000 -1.751e+00 2.492e-01 -7.023 2.16e-12 \*\*\*  
## CompanyGroupTop 10000 -1.493e+00 5.531e-01 -2.700 0.006936 \*\*   
## CompanyGroupOther -1.712e+00 1.067e+00 -1.605 0.108514   
## RecordsGroupTop 2000 -1.333e-01 6.050e-02 -2.203 0.027595 \*   
## RecordsGroupTop 3000 -2.139e-01 6.512e-02 -3.285 0.001021 \*\*   
## RecordsGroupTop 4000 -2.341e-01 7.675e-02 -3.050 0.002285 \*\*   
## RecordsGroupTop 5000 -1.564e-01 8.647e-02 -1.809 0.070429 .   
## RecordsGroupTop 6000 -2.065e-01 8.825e-02 -2.341 0.019257 \*   
## RecordsGroupTop 7000 -2.449e-01 8.294e-02 -2.953 0.003147 \*\*   
## RecordsGroupTop 8000 -3.027e-01 1.022e-01 -2.961 0.003069 \*\*   
## RecordsGroupTop 9000 -1.105e-01 1.128e-01 -0.980 0.327250   
## RecordsGroupTop 10000 -9.991e-02 1.214e-01 -0.823 0.410362   
## RecordsGroupTop 11000 -3.851e-01 2.199e-01 -1.751 0.079885 .   
## RecordsGroupTop 12000 -1.794e-01 2.557e-01 -0.702 0.482891   
## RecordsGroupTop 13000 -3.003e+00 1.043e+00 -2.879 0.003994 \*\*   
## RecordsGroupOther -1.137e+01 1.007e+02 -0.113 0.910108   
## DatesGroupTop 2000 4.450e-02 6.143e-02 0.724 0.468861   
## DatesGroupTop 3000 6.257e-02 7.843e-02 0.798 0.424993   
## DatesGroupTop 4000 9.772e-02 9.197e-02 1.063 0.287987   
## DatesGroupTop 5000 -4.427e-02 1.118e-01 -0.396 0.692090   
## DatesGroupTop 6000 4.756e-01 1.055e-01 4.510 6.47e-06 \*\*\*  
## DatesGroupTop 7000 8.890e-02 1.574e-01 0.565 0.572105   
## DatesGroupTop 8000 2.425e-01 1.529e-01 1.586 0.112683   
## DatesGroupOther 2.926e-01 5.048e-01 0.580 0.562236   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 25879 on 75959 degrees of freedom  
## Residual deviance: 20309 on 75910 degrees of freedom  
## AIC: 20409  
##   
## Number of Fisher Scoring iterations: 15

accessibility.train = vehicles.train$Wheelchair.Accessible  
model.probs = predict(vehicles.glm, vehicles.train, type="response")  
  
summary(model.probs)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000000 0.0004536 0.0225096 0.0407583 0.0507496 0.8070354

model.preds = rep(0, 75960)  
model.preds[model.probs > 0.5] = 1  
  
table(model.preds, accessibility.train)

## accessibility.train  
## model.preds 0 1  
## 0 72742 3027  
## 1 122 69

mean(model.preds == accessibility.train)

## [1] 0.958544

accessibility.test = vehicles.test$Wheelchair.Accessible  
model.probs = predict(vehicles.glm, vehicles.test, type="response")  
  
summary(model.probs)

## Min. 1st Qu. Median Mean 3rd Qu. Max.   
## 0.0000000 0.0004531 0.0221117 0.0401456 0.0511191 0.7679744

model.preds = rep(0, 18988)  
model.preds[model.probs > 0.5] = 1  
  
table(model.preds, accessibility.test)

## accessibility.test  
## model.preds 0 1  
## 0 18184 762  
## 1 31 11

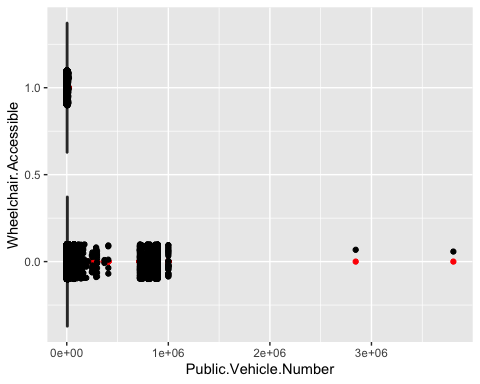
mean(model.preds == accessibility.test)

## [1] 0.9582368

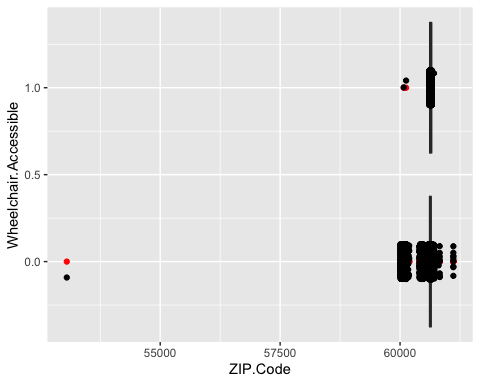
Overall, it barely changed. Address didn’t play huge role it predicting the accessibility.

We can create fancy boxplots:

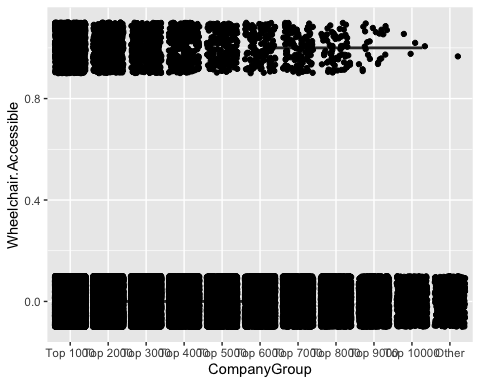
binaryboxplot <- function(data, xlab=NULL, ylab=NULL){  
 # A fancier boxplot   
 # Inputs: A data frame data,   
 # xlab to be used as group  
 # ylab to be used as regressor  
 # Output: A boxplot with data points  
   
 #data[xlab] makes a list   
 #so we need to extract just the values as a vector with unlist()   
 group <- unlist(data[xlab])   
 regressor <- unlist(data[ylab])  
  
 options(repr.plot.width=7, repr.plot.height=3)  
 ggplot(data, aes(x=group, y=regressor, group=group)) +   
 geom\_boxplot(outlier.colour="red") +  
 geom\_jitter(width=0.1) +  
 coord\_flip() +  
 xlab(xlab) + ylab(ylab)  
}  
  
par(mfrow=c(5,3))  
binaryboxplot(vehicles, "Wheelchair.Accessible", "Public.Vehicle.Number")



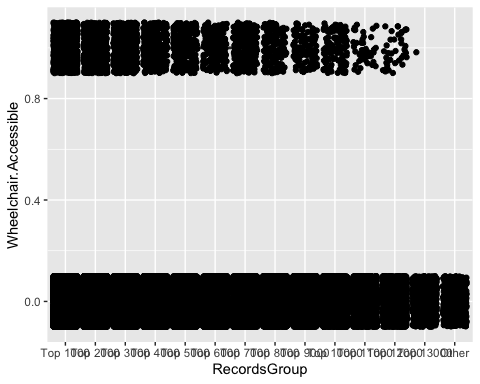
binaryboxplot(vehicles, "Wheelchair.Accessible", "ZIP.Code")



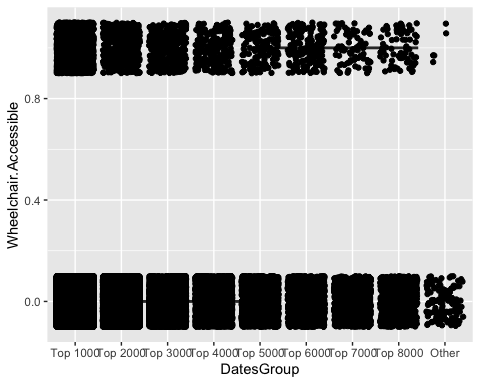
binaryboxplot(vehicles, "Wheelchair.Accessible", "CompanyGroup")



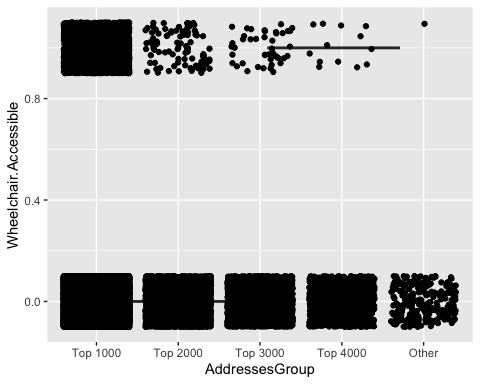
binaryboxplot(vehicles, "Wheelchair.Accessible", "RecordsGroup")



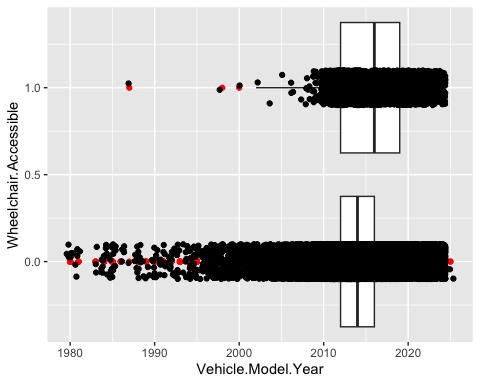
binaryboxplot(vehicles, "Wheelchair.Accessible", "DatesGroup")



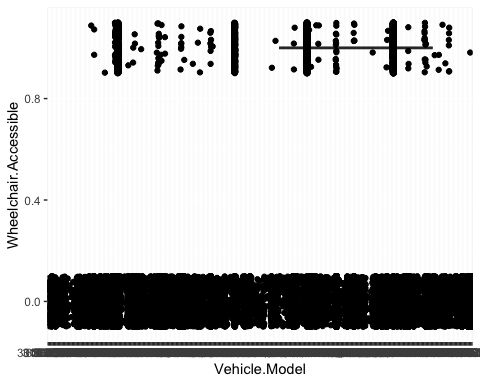
binaryboxplot(vehicles, "Wheelchair.Accessible", "AddressesGroup")



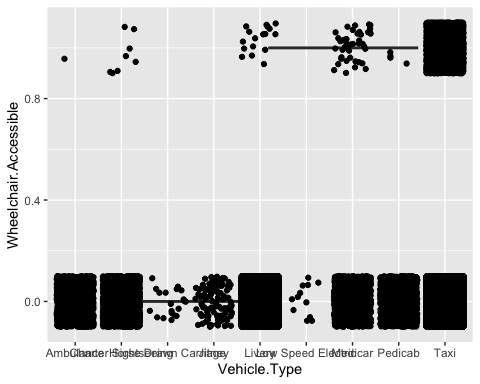
binaryboxplot(vehicles, "Wheelchair.Accessible", "Vehicle.Model.Year")



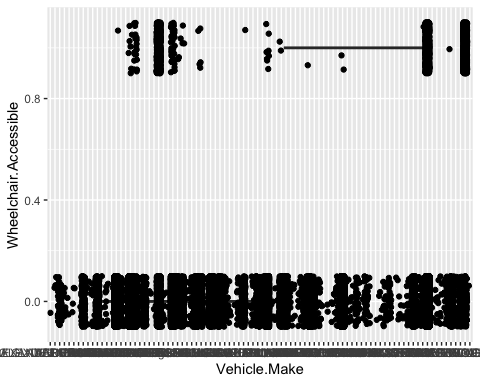
binaryboxplot(vehicles, "Wheelchair.Accessible", "Vehicle.Model")



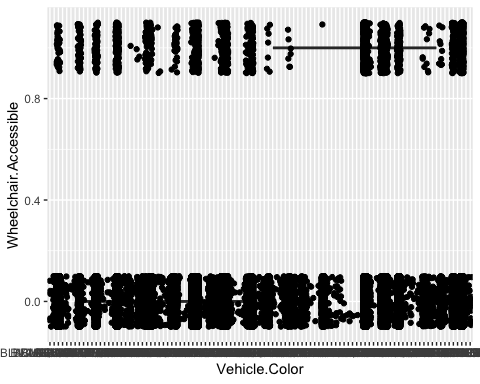
binaryboxplot(vehicles, "Wheelchair.Accessible", "Vehicle.Type")



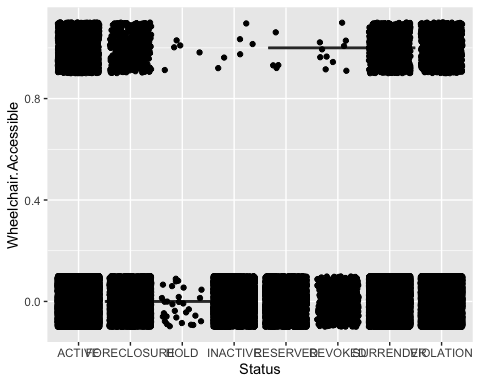
binaryboxplot(vehicles, "Wheelchair.Accessible", "Vehicle.Make")



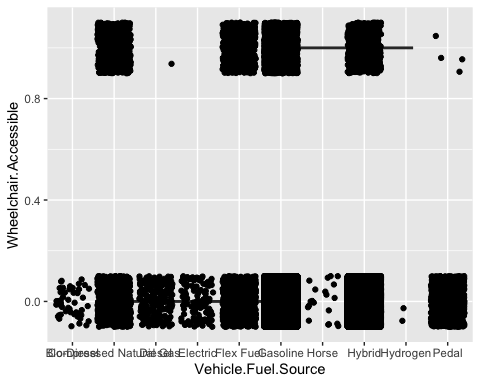
binaryboxplot(vehicles, "Wheelchair.Accessible", "Vehicle.Color")



binaryboxplot(vehicles, "Wheelchair.Accessible", "Status")



binaryboxplot(vehicles, "Wheelchair.Accessible", "Vehicle.Fuel.Source")

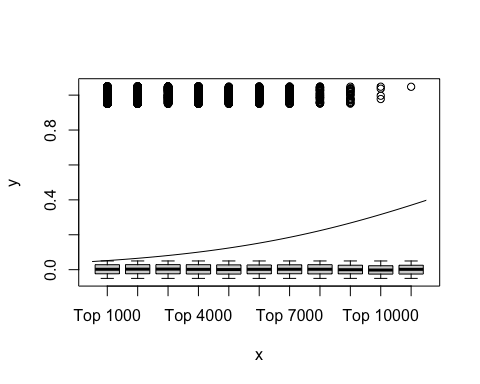
 Out of these fancy boxplots, we can understand that vehicles with Gasoline are more likely to be accessible. Those with Active/Surrender/Violation Status are more likely to be accessible. There is no relationship between color, manufacturer, model and accessibility. Taxi is more likely to be accessible, though.

I will also plot the logistic regression curve.

vehicles.glm <- glm(Wheelchair.Accessible ~ CompanyGroup, family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ CompanyGroup, family = binomial)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -3.13679 0.02359 -132.971 < 2e-16 \*\*\*  
## CompanyGroupTop 2000 0.23662 0.04467 5.296 1.18e-07 \*\*\*  
## CompanyGroupTop 3000 0.13773 0.05528 2.491 0.012723 \*   
## CompanyGroupTop 4000 0.08596 0.06353 1.353 0.176024   
## CompanyGroupTop 5000 -0.16998 0.07717 -2.203 0.027623 \*   
## CompanyGroupTop 6000 -0.34817 0.09146 -3.807 0.000141 \*\*\*  
## CompanyGroupTop 7000 -0.31418 0.09842 -3.192 0.001412 \*\*   
## CompanyGroupTop 8000 -0.54053 0.12005 -4.503 6.71e-06 \*\*\*  
## CompanyGroupTop 9000 -1.31319 0.20666 -6.354 2.09e-10 \*\*\*  
## CompanyGroupTop 10000 -2.48270 0.50146 -4.951 7.38e-07 \*\*\*  
## CompanyGroupOther -3.24502 0.99990 -3.245 0.001173 \*\*   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 32342 on 94947 degrees of freedom  
## Residual deviance: 32065 on 94937 degrees of freedom  
## AIC: 32087  
##   
## Number of Fisher Scoring iterations: 8

b0 <- coef(vehicles.glm)[1]  
b1 <- coef(vehicles.glm)[2]  
plot(CompanyGroup, jitter(Wheelchair.Accessible, amount = 0.05))  
 curve(exp(b0+b1\*x)/(1+exp(b0+b1\*x)),add=TRUE)



vehicles.glm <- glm(Wheelchair.Accessible ~ Vehicle.Model.Year, family = binomial)  
summary(vehicles.glm)

##   
## Call:  
## glm(formula = Wheelchair.Accessible ~ Vehicle.Model.Year, family = binomial)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -2.765e+02 8.691e+00 -31.82 <2e-16 \*\*\*  
## Vehicle.Model.Year 1.357e-01 4.312e-03 31.47 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 32342 on 94947 degrees of freedom  
## Residual deviance: 31388 on 94946 degrees of freedom  
## AIC: 31392  
##   
## Number of Fisher Scoring iterations: 6

b0 <- coef(vehicles.glm)[1]  
b1 <- coef(vehicles.glm)[2]  
plot(Vehicle.Model.Year, jitter(Wheelchair.Accessible, amount = 0.05))  
 curve(exp(b0+b1\*x)/(1+exp(b0+b1\*x)),add=TRUE)

