A BINARY LOGIT MODEL FOR CAR OWNERSHIP IN DVRPC 9-COUNTY REGION

Summary

Using the household travel data from the DVRPC 2012-13 Household travel survey, two binary logit models are fitted in R. The model with higher prediction accuracy suggests that, given all the other conditions fixed,

1) households with annual income between $75,000 and $99,999 are the most likely to have car(s) among all the income groups;
2) larger households are more likely to own car(s);
3) suburban and rural households are more likely to own car(s);
4) households with higher share of motorized trips are more likely to have car(s);
5) households maintaining tolled road accounts are more likely to have car(s),

while

6) households living in apartment buildings are less likely to own car(s);
7) households with more trips or longer travel time are less likely to have car(s);
8) households maintaining car share service membership are less likely to have car(s).

Model Development

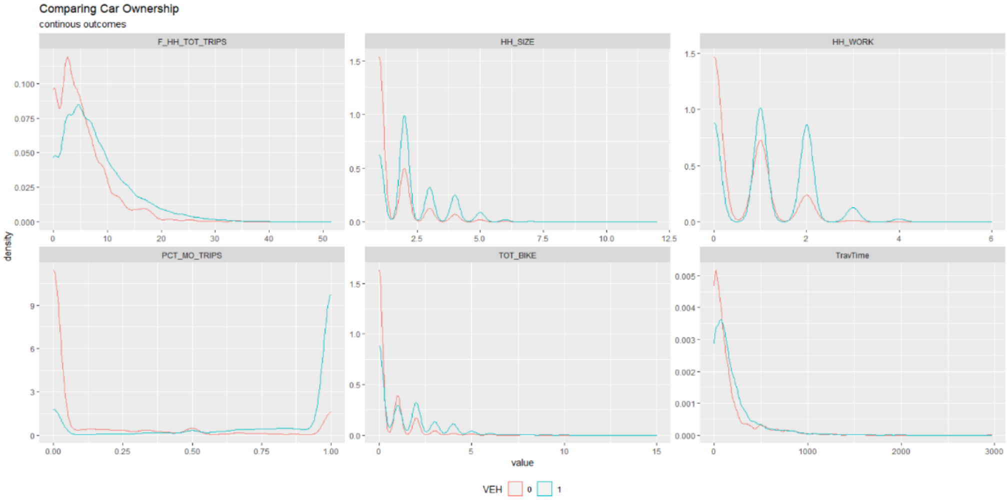
The household travel data contains 38 variables and 9235 observation. To predict whether a household own any car(s), a binary variable named “VEH” is created based on the total number of vehicles owned by the household. VEH = 0 refers to the household not owning any car, and VEH = 1 refers to the household owning at least one car. By reviewing the dataset, following independent variables are considered to be included in the logit model predicting household car ownership.

Variable Names	Type	What the Variables are	Reasons for being Included
H_COUNTY	Categorical	County of the Household	To understand if geographic location affects household car ownership. *Note: geographic variables with finer spatial resolution, such as census tracts or transportation analysis zones, are used not used due to too many categories.
A_TYPE RES_TYPE	Categorical Categorical	Area Type Residence Type	To understand if the area type and building type influence household car ownership.
F_HH_TOT_TRIPS F_HH_MO_TRIPS	Continuous Continuous	Total Household Trips Total Motorized Household Trips	Whether the number of trips made is related to the car ownership; whether number of motorized trips influence the car ownership.
HH_SIZE HH_WORK INCOME	Count Count Categorical	Household Size Number of Household Workers Annual House Income	How demographics of a household are related to car ownership.
TOT_BIKE TOLL_ACCNT CAR_SHARE	Count Categorical Categorical	Total Household Bikes Tolled Road Account Car Share Services Membership	If variables which indirectly suggest traveling behaviors relates to car ownership.
RETRIEVE SURV_LANG	Categorical Categorical	Retrieval Mode of The Survey Language Used in Survey	Whether households owning cars have preferences on the survey method and language.

Besides the variables above, two more variables are created. Since there is no information about traveling time for each household, the traveling time for each trip (from the trip dataset) are summarized by household ID and then merged to this dataset. Another variable is the ratio of the motorized trips to the total number of trips.

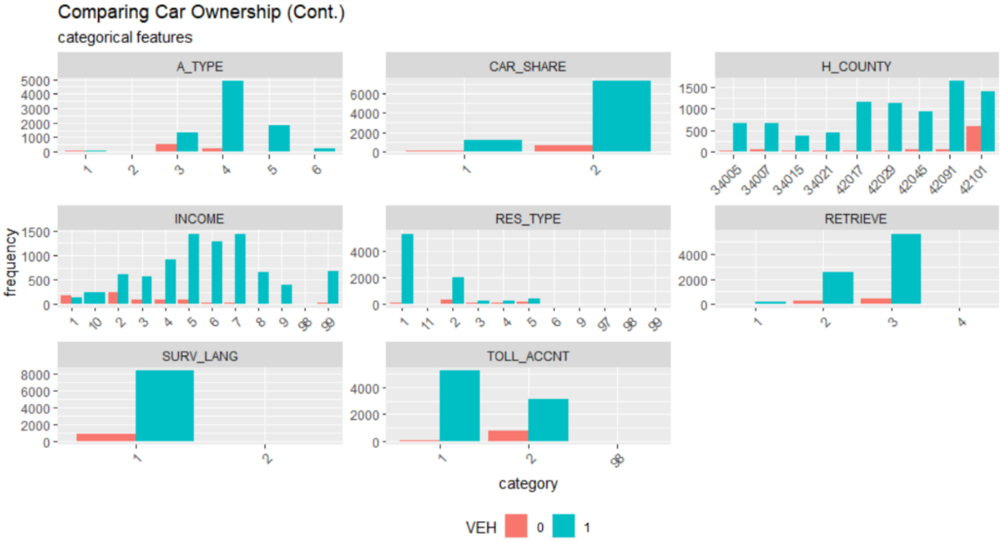
Variable Names	Type	What the Variables are	Reasons for being Included
TravTime	Continuous	Total Travel Time of the Household	To understand if travel time affects the household car ownership.
PCT_MO_TRIPS	Continuous	Ratio of motorized trips to total trips	Whether the share of motorized trips related to car ownership.

After adding the ratio of motorized trips to total trips, the total motorized trip variable is then removed due to collinearity. Twenty-four observations with response values 98 (don't know) and 99 (refused) in continuous or count variables are also removed. The ready-to-use dataset includes all 15 variables mentioned above with 9211 observations. Exploratory data analysis (EDA) is then performed to identify which variables can distinguish the ownership pattern well. A density graph for continuous/count data is generated, and a bar chart of frequency for categorical data is also generated.



Density of Countinous Variables Distinguishing Household Car Ownership

Among the continuous/count variables, total household workers, total household bikes and total household travel time seem not to make clear distinctions between household with car(s) and household without car(s).



Counts of Categorical Variables Distinguishing Household Car Ownership

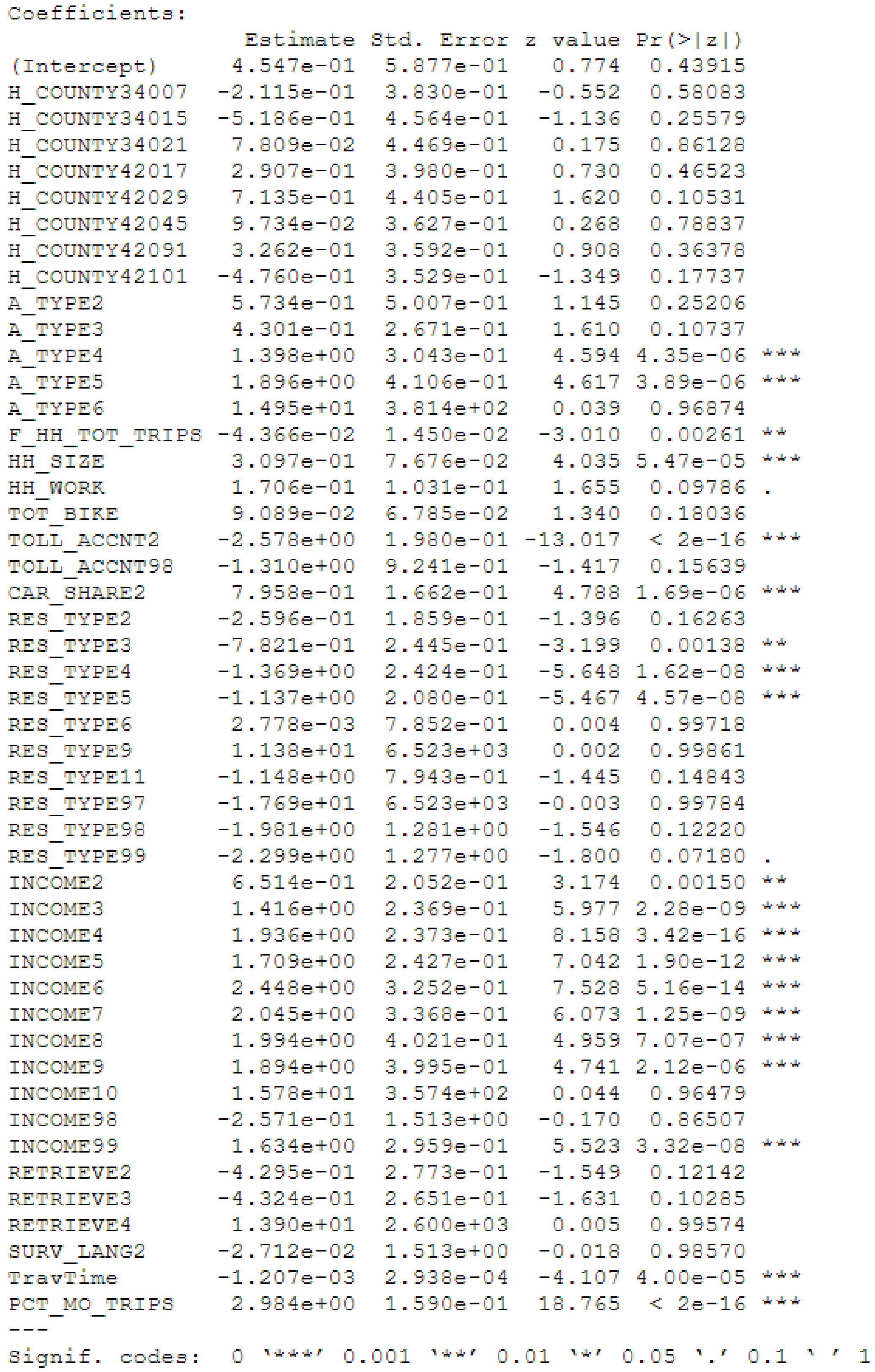
All the categorical variables seem to distinguish the ownership with certain categories, so significance of coefficients and prediction accuracy will be compared for model selection. 1000 observations are randomly selected as test data, and the 8211 observations left are used to train the model.

Using the glm() function in R, a model fitted with all 15 variables above is generated. Please refer to the Key Findings section or the corresponding section in the Appendix for the detailed regression summary. The AIC of the model fitted is 2153.1. After testing this model using the test set left out, the accuracy of the model is 93.7%. The area under the receiver operating characteristic (ROC) curve is 0.9528.

Since not all the variables are significant at the significance level of 0.05, a second model is then built without insignificant variables (categorical variables are eliminated if all categories are insignificant). Variable excluded are county of the household, total household workers, total household bikes, retrieve modes of the survey, and language of the survey. Note that although in EDA, total household travel time seems not to distinguish the ownership pattern, it is significant at the significance level of 0.05, so it is kept in the second model. The second model is fitted with the same training data using the glm() function in R. Please refer to "The Second Model (Reduced Model)" section in the Appendix for the detailed regression results. The AIC of the model is 2156.3. The prediction accuracy using the same test set as above is 93%, and the area under the ROC curve is 0.9519.

Comparing the two models, although the second (reduced) model contains more significant independent variables, the first model works better because it has higher accuracy, lower AIC, and greater area under the ROC curve. Therefore, the first model fitted with all 15 variables is selected for the further analysis in the next section.

Key Findings



```
Coefficients:
                  Estimate Std. Error z value Pr(>|z|)    
(Intercept)      4.547e-01  5.877e-01   0.774  0.43915    
H_COUNTY34007   -2.115e-01  3.830e-01  -0.552  0.58083    
H_COUNTY34015   -5.186e-01  4.564e-01  -1.136  0.25579    
H_COUNTY34021    7.809e-02  4.469e-01   0.175  0.86128    
H_COUNTY42017    2.907e-01  3.980e-01   0.730  0.46523    
H_COUNTY42029    7.135e-01  4.405e-01   1.620  0.10531    
H_COUNTY42045    9.734e-02  3.627e-01   0.268  0.78837    
H_COUNTY42091    3.262e-01  3.592e-01   0.908  0.36378    
H_COUNTY42101   -4.760e-01  3.529e-01  -1.349  0.17737    
A_TYPE2          5.734e-01  5.007e-01   1.145  0.25206    
A_TYPE3          4.301e-01  2.671e-01   1.610  0.10737    
A_TYPE4          1.398e+00  3.043e-01   4.594 4.35e-06 ***
A_TYPE5          1.896e+00  4.106e-01   4.617 3.89e-06 ***
A_TYPE6          1.495e+01  3.814e+02   0.039  0.96874    
F_HH_TOT_TRIPS  -4.366e-02  1.450e-02  -3.010  0.00261 ** 
HH_SIZE          3.097e-01  7.676e-02   4.035 5.47e-05 ***
HH_WORK          1.706e-01  1.031e-01   1.655  0.09786 .  
TOT_BIKE         9.089e-02  6.785e-02   1.340  0.18036    
TOLL_ACCNT2     -2.578e+00  1.980e-01 -13.017  < 2e-16 ***
TOLL_ACCNT98    -1.310e+00  9.241e-01  -1.417  0.15639    
CAR_SHARE2       7.958e-01  1.662e-01   4.788 1.69e-06 ***
RES_TYPE2       -2.596e-01  1.859e-01  -1.396  0.16263    
RES_TYPE3       -7.821e-01  2.445e-01  -3.199  0.00138 ** 
RES_TYPE4       -1.369e+00  2.424e-01  -5.648 1.62e-08 ***
RES_TYPE5       -1.137e+00  2.080e-01  -5.467 4.57e-08 ***
RES_TYPE6        2.778e-03  7.852e-01   0.004  0.99718    
RES_TYPE9        1.138e+01  6.523e+03   0.002  0.99861    
RES_TYPE11      -1.148e+00  7.943e-01  -1.445  0.14843    
RES_TYPE97      -1.769e+01  6.523e+03  -0.003  0.99784    
RES_TYPE98      -1.981e+00  1.281e+00  -1.546  0.12220    
RES_TYPE99      -2.299e+00  1.277e+00  -1.800  0.07180 .  
INCOME2          6.514e-01  2.052e-01   3.174  0.00150 ** 
INCOME3          1.416e+00  2.369e-01   5.977 2.28e-09 ***
INCOME4          1.936e+00  2.373e-01   8.158 3.42e-16 ***
INCOME5          1.709e+00  2.427e-01   7.042 1.90e-12 ***
INCOME6          2.448e+00  3.252e-01   7.528 5.16e-14 ***
INCOME7          2.045e+00  3.368e-01   6.073 1.25e-09 ***
INCOME8          1.994e+00  4.021e-01   4.959 7.07e-07 ***
INCOME9          1.894e+00  3.995e-01   4.741 2.12e-06 ***
INCOME10         1.578e+01  3.574e+02   0.044  0.96479    
INCOME98        -2.571e-01  1.513e+00  -0.170  0.86507    
INCOME99         1.634e+00  2.959e-01   5.523 3.32e-08 ***
RETRIEVE2       -4.295e-01  2.773e-01  -1.549  0.12142    
RETRIEVE3       -4.324e-01  2.651e-01  -1.631  0.10285    
RETRIEVE4        1.390e+01  2.600e+03   0.005  0.99574    
SURV_LANG2      -2.712e-02  1.513e+00  -0.018  0.98570    
TravTime        -1.207e-03  2.938e-04  -4.107 4.00e-05 ***
PCT_MO_TRIPS     2.984e+00  1.590e-01  18.765  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1
```

Although the model fitted with all the variables selected generated more accurate predictions, not all the variables used are found significant at the significance level of 0.05. Below, significant variables are summarized into three categories to provide more insights about household private vehicle ownership.

Demographics

Larger households are more likely to own car(s) than smaller households. When all the other conditions hold constant, a one-person increase in the household size will lead to (e$^{3.097*10^\wedge(-1)}$ – 1)*100% = 36.3% increase in the odds for this household to own at least one car. Therefore, more people in the household, more likely the household owning car(s).

Households with annual income less than $10,000 are the least likely to own car(s) among all income groups, and households with annual income between $75,000 and $99,999 are the most likely to own car(s). With all other conditions fixed, the odds for a household that makes between $75,000 and $99,999 per year to own car(s) is 1056.5% higher than the odds for a household that makes less than $10,000 per year. In general, for groups making less than $100,000 per year, the likelihood of owning car(s) increases as their income increase. However, that is not always true: households making $50,000 to $74,999 are less likely to own car(s) than those making $35,000 to $49,999 are. Furthermore, the likelihoods for groups making greater than $100,000 to own car(s) decrease as the income increases.

Percentage of the odds of owning car(s) ***higher*** than the odds for households making less than $10,000							
$10,000- $24,999	$25,000- $34,999	$35,000- $49,999	$50,000- $74,999	$75,000- $99,999	$100,000- $149,999	$150,000- $199,999	$200,000- $249,999
91.8%	312.1%	593.1%	452.3%	1056.5%	672.9%	634.5%	564.6%

Area and Residence Type

Comparing to the households in CBD, households in suburban and rural areas are more likely to own at least one car. Specifically, with all other condition fixed, the odds of a suburban household owning car(s) is 304.7% higher than the odds of a household in CBD, and the odds of a rural household owning car(s) is 565.9% higher than the odds of a household in CBD.

Households living in apartment buildings are less likely to own car(s) in general. With other variables fixed, the odds of owning car(s) for households living in buildings with 2 to 4, 5 to 19 and over 20 apartments are 54.3%, 75.6% and 67.9% lower than the odds of households living in single family house not attracted to other houses, respectively. This finding corresponds to the relationship between car ownerships and area types discussed above because apartment buildings are more often seen in areas with higher urban density, such as CBDs. Limited and higher cost parking space, congestions, and easier access to transit might be factors that keep apartment building households from owning car(s).

Travel Behavior

Households with more total trips and longer travel time are less likely to own car(s). With all other conditions fixed, a one-unit increase in total trip numbers and in travel time will lead to a 4.3% decrease and a 0.1% decrease, respectively, in the likelihood for a household owning car(s). Such findings might imply poor accessibility provided by public transit in the region. For households with no cars commonly relied on public transportation, they make more trips due to transferring among different lines or modes, and they need longer time to get destinations than those who drive their own cars. However, such conclusions will need further research and evidence to support because the changes of odds are small.

Although increasing total trips reduces the odds for households to own car(s), increase motorized trips increases this odds. Holding all other conditions, a one-unit increase in the ratio of motorized trips to total trips will lead to 1876.7% increase in the odds to own car(s). Therefore, among all the trips generated by motorized vehicle, trips made by private vehicles have a large share.

Households renting cars or maintaining carshare services memberships and households not using tolled roads or bridges are less likely to own car(s). Comparing to the odds of owning car(s) for households rent cars or use carshare services, the odds for households not doing so is 121.6% higher with all other conditions fixed. The odds of owning car(s) for households not having toll road/bridge accounts is 92.4% lower than the odds for household maintained those account. These findings obey common sense that there is no need for car owners to rent cars for long time, and there is also no need for most of whom not owning a car to maintain toll road accounts.

Appendix

```
# load packages needed
library(tidyverse)

## -- Attaching packages ----------------------------------------------------
--------- tidyverse 1.3.0 --

## v ggplot2 3.3.2     v purrr   0.3.4
## v tibble  3.0.3     v dplyr   1.0.2
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.3.1     v forcats 0.5.0

## -- Conflicts -------------------------------------------------------------
--- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(ROCR)

## Warning: package 'ROCR' was built under R version 4.0.3
```

Data Preparation

```
# import data
hh <- read.csv("1_Household_Public.csv")
trip <- read.csv("4_Trip_Public.csv")

# add travel time to the dataset
trip1 <- trip %>%
  group_by(HH_ID) %>%
  summarise(TravTime = sum(Survey_TravTime, na.rm = T))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
hh <- merge(hh, trip1, by = "HH_ID")

# create a new binary column of whether the household own any vehicle(s)
hh$VEH <- ifelse(hh$TOT_VEH >= 1, 1, 0)

# variables for use only
# state, county, etc. are not considered
hh <- hh %>%
  dplyr::select(VEH, # dependent variable
                H_COUNTY,
                A_TYPE, # area type
                F_HH_TOT_TRIPS, F_HH_MO_TRIPS, # number of trips by categorie
s
                HH_SIZE, HH_WORK, # hh demographics
                TOT_BIKE, TOLL_ACCNT, CAR_SHARE, # some helpful info
                RES_TYPE, INCOME, # hh demographics
                RETRIEVE, SURV_LANG,
                TravTime) # how the survey was done

# how many 98 and 99s
length(which(hh$TOT_BIKE == 98|hh$TOT_BIKE == 99))
```

```
## [1] 13
```

```
length(which(hh$CAR_SHARE == 98|hh$CAR_SHARE == 99))
```

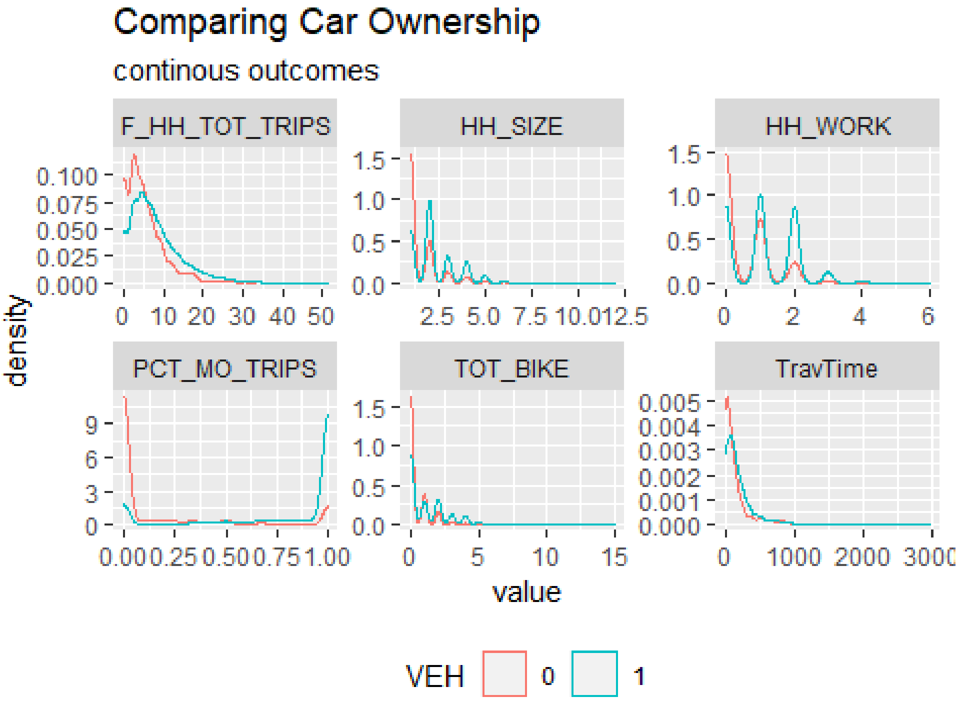
```
## [1] 11
```

```
# removing 98 and 99 for continuous variables
# factorized categorical variables
hh$VEH <- factor(hh$VEH)
hh$A_TYPE <- factor(hh$A_TYPE)
hh <- hh[hh$TOT_BIKE != 98&hh$TOT_BIKE != 99, ]
hh <- hh[hh$CAR_SHARE != 98&hh$CAR_SHARE != 99, ]
hh$TOLL_ACCNT <- factor(hh$TOLL_ACCNT)
hh$CAR_SHARE <- factor(hh$CAR_SHARE)
hh$RES_TYPE <- factor(hh$RES_TYPE)
hh$INCOME <- factor(hh$INCOME)
hh$RETRIEVE <- factor(hh$RETRIEVE)
hh$SURV_LANG <- factor(hh$SURV_LANG)
hh$H_COUNTY <- factor(hh$H_COUNTY)

# get the percentage of trip by motor vehicle by total trip number
hh$PCT_MO_TRIPS <- hh$F_HH_MO_TRIPS/(hh$F_HH_TOT_TRIPS + 0.00000000001)
# get rid of F_HH_MO_TRIPS
hh <- hh %>%
  dplyr::select(-F_HH_MO_TRIPS)
# check if data contains any missing values
any(is.na(hh))
```

```
## [1] FALSE
```

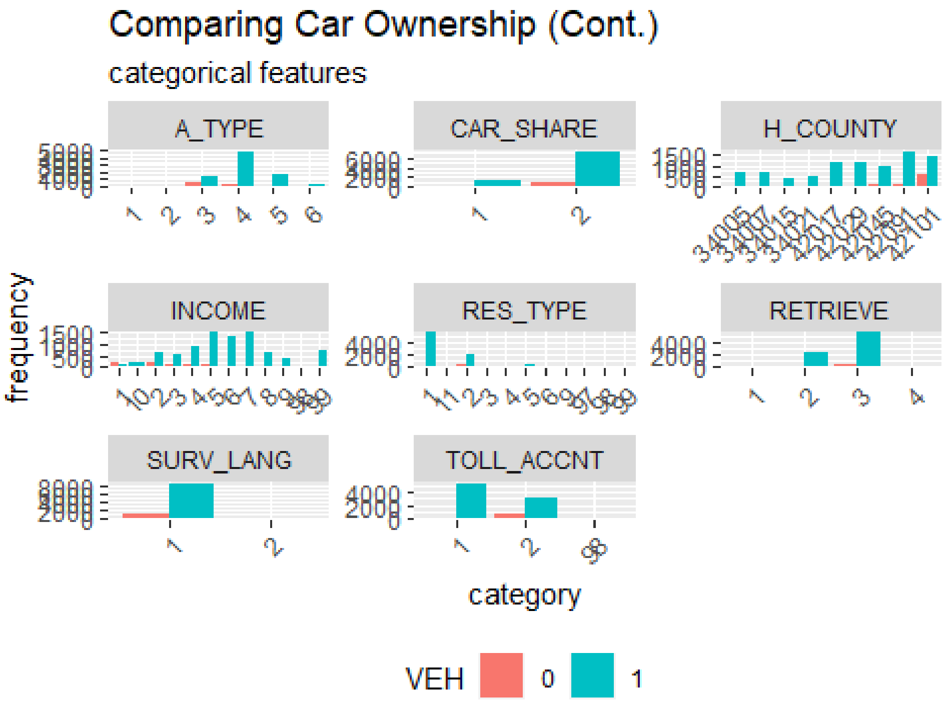
Exploratory data analysis (EDA)

```
# EDA
# codes from MUSA 508 class
# density of continuous variables
hh %>%
    dplyr::select(VEH, PCT_MO_TRIPS, HH_SIZE, HH_WORK, TOT_BIKE, F_HH_TOT_TRI
PS, TravTime) %>%
    gather(Variable, value, -VEH) %>%
    ggplot() +
    geom_density(aes(value, color=VEH), fill = "transparent") +
    facet_wrap(~Variable, scales = "free") +
    labs(title = "Comparing Car Ownership",
         subtitle = "continous outcomes",
         caption = "Density of Countinous Variables Distinguishing Household
Car Ownership") +
    theme(legend.position = "bottom")
```



Density of Countinous Variables Distinguishing Household Car Ownership

```
# codes from MUSA 508 class
# categorical data EDA
hh %>%
    dplyr::select(VEH,
                  H_COUNTY, A_TYPE, # area type
                  TOLL_ACCNT, CAR_SHARE, # some helpful info
                  RES_TYPE, INCOME, # hh demographics
                  RETRIEVE, SURV_LANG) %>%
    gather(Variable, value, -VEH) %>%
    count(Variable, value, VEH) %>%
      ggplot(., aes(value, n, fill = VEH)) +
        geom_bar(position = "dodge", stat="identity") +
        facet_wrap(~Variable, scales="free") +
        labs(x="category", y="frequency",
             title = "Comparing Car Ownership (Cont.)",
             subtitle = "categorical features",
             caption = "Counts of Categorical Variables Distinguishing Househ
old Car Ownership") +
        theme(axis.text.x = element_text(angle = 45, hjust = 1), legend.posit
ion = "bottom")

## Warning: attributes are not identical across measure variables;
## they will be dropped
```



Counts of Categorical Variables Distinguishing Household Car Ownership

# Training and Test Sets

```
# split training and test sets
# randomly select 1000 observations for testing
set.seed(3)
test.indices = sample(1:nrow(hh), 1000)
hh.train=hh[-test.indices,]
hh.test=hh[test.indices,]
```

Model Fitted with All Variables

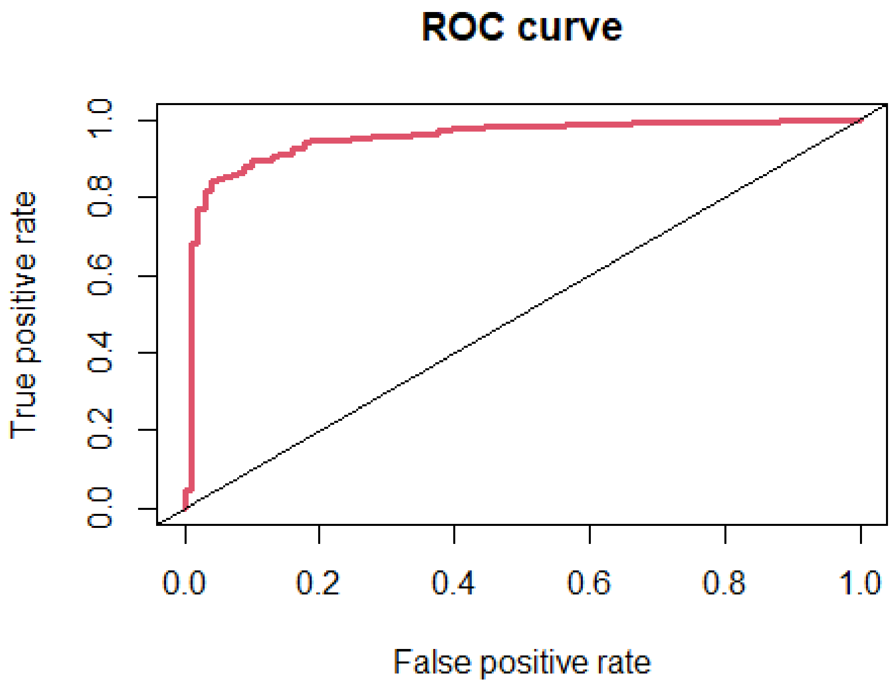
```
# model with all 15 variables
glm.fit.all <- glm(VEH ~ ., data = hh.train, family = "binomial")

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

summary(glm.fit.all)

##
## Call:
## glm(formula = VEH ~ ., family = "binomial", data = hh.train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.7108   0.0233   0.0527   0.1592   2.8030
##
## Coefficients:
##                  Estimate Std. Error z value Pr(>|z|)
## (Intercept)     4.547e-01  5.877e-01   0.774  0.43915
## H_COUNTY34007  -2.115e-01  3.830e-01  -0.552  0.58083
## H_COUNTY34015  -5.186e-01  4.564e-01  -1.136  0.25579
## H_COUNTY34021   7.809e-02  4.469e-01   0.175  0.86128
## H_COUNTY42017   2.907e-01  3.980e-01   0.730  0.46523
## H_COUNTY42029   7.135e-01  4.405e-01   1.620  0.10531
## H_COUNTY42045   9.734e-02  3.627e-01   0.268  0.78837
## H_COUNTY42091   3.262e-01  3.592e-01   0.908  0.36378
## H_COUNTY42101  -4.760e-01  3.529e-01  -1.349  0.17737
## A_TYPE2         5.734e-01  5.007e-01   1.145  0.25206
## A_TYPE3         4.301e-01  2.671e-01   1.610  0.10737
## A_TYPE4         1.398e+00  3.043e-01   4.594 4.35e-06 ***
## A_TYPE5         1.896e+00  4.106e-01   4.617 3.89e-06 ***
## A_TYPE6         1.495e+01  3.814e+02   0.039  0.96874
## F_HH_TOT_TRIPS -4.366e-02  1.450e-02  -3.010  0.00261 **
## HH_SIZE         3.097e-01  7.676e-02   4.035 5.47e-05 ***
## HH_WORK         1.706e-01  1.031e-01   1.655  0.09786 .
## TOT_BIKE        9.089e-02  6.785e-02   1.340  0.18036
## TOLL_ACCNT2    -2.578e+00  1.980e-01 -13.017  < 2e-16 ***
## TOLL_ACCNT98   -1.310e+00  9.241e-01  -1.417  0.15639
## CAR_SHARE2      7.958e-01  1.662e-01   4.788 1.69e-06 ***
## RES_TYPE2      -2.596e-01  1.859e-01  -1.396  0.16263
## RES_TYPE3      -7.821e-01  2.445e-01  -3.199  0.00138 **
## RES_TYPE4      -1.369e+00  2.424e-01  -5.648 1.62e-08 ***
## RES_TYPE5      -1.137e+00  2.080e-01  -5.467 4.57e-08 ***
## RES_TYPE6       2.778e-03  7.852e-01   0.004  0.99718
## RES_TYPE9       1.138e+01  6.523e+03   0.002  0.99861
## RES_TYPE11     -1.148e+00  7.943e-01  -1.445  0.14843
## RES_TYPE97     -1.769e+01  6.523e+03  -0.003  0.99784
## RES_TYPE98     -1.981e+00  1.281e+00  -1.546  0.12220
## RES_TYPE99     -2.299e+00  1.277e+00  -1.800  0.07180 .
## INCOME2         6.514e-01  2.052e-01   3.174  0.00150 **
## INCOME3          1.416e+00  2.369e-01   5.977 2.28e-09 ***
## INCOME4          1.936e+00  2.373e-01   8.158 3.42e-16 ***
## INCOME5          1.709e+00  2.427e-01   7.042 1.90e-12 ***
## INCOME6          2.448e+00  3.252e-01   7.528 5.16e-14 ***
## INCOME7          2.045e+00  3.368e-01   6.073 1.25e-09 ***
## INCOME8          1.994e+00  4.021e-01   4.959 7.07e-07 ***
## INCOME9          1.894e+00  3.995e-01   4.741 2.12e-06 ***
## INCOME10         1.578e+01  3.574e+02   0.044  0.96479    
## INCOME98        -2.571e-01  1.513e+00  -0.170  0.86507    
## INCOME99         1.634e+00  2.959e-01   5.523 3.32e-08 ***
## RETRIEVE2       -4.295e-01  2.773e-01  -1.549  0.12142    
## RETRIEVE3       -4.324e-01  2.651e-01  -1.631  0.10285    
## RETRIEVE4        1.390e+01  2.600e+03   0.005  0.99574    
## SURV_LANG2      -2.712e-02  1.513e+00  -0.018  0.98570    
## TravTime        -1.207e-03  2.938e-04  -4.107 4.00e-05 ***
## PCT_MO_TRIPS     2.984e+00  1.590e-01  18.765  < 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: 4926.6  on 8210  degrees of freedom
## Residual deviance: 2057.1  on 8163  degrees of freedom
## AIC: 2153.1
##
## Number of Fisher Scoring iterations: 17
```

The Second Model (Reduced Model)

```
# model 2
glm.fit.2 <- glm(VEH ~ PCT_MO_TRIPS + HH_SIZE + F_HH_TOT_TRIPS + TravTime + A
_TYPE + TOLL_ACCNT + CAR_SHARE + RES_TYPE + INCOME, data = hh.train, family =
 "binomial")

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

summary(glm.fit.2)

##
## Call:
## glm(formula = VEH ~ PCT_MO_TRIPS + HH_SIZE + F_HH_TOT_TRIPS +
##     TravTime + A_TYPE + TOLL_ACCNT + CAR_SHARE + RES_TYPE + INCOME,
##     family = "binomial", data = hh.train)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.6932   0.0256   0.0562   0.1633   2.7947
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)    -3.477e-01  4.171e-01  -0.834 0.404513
## PCT_MO_TRIPS    3.038e+00  1.568e-01  19.371  < 2e-16 ***
## HH_SIZE         3.554e-01  7.168e-02   4.958 7.12e-07 ***
## F_HH_TOT_TRIPS -3.977e-02  1.384e-02  -2.874 0.004050 **
## TravTime       -1.235e-03  2.890e-04  -4.272 1.93e-05 ***
## A_TYPE2         5.731e-01  5.016e-01   1.143 0.253157
## A_TYPE3         5.630e-01  2.624e-01   2.145 0.031924 *
## A_TYPE4         1.852e+00  2.705e-01   6.847 7.55e-12 ***
## A_TYPE5         2.512e+00  3.654e-01   6.874 6.26e-12 ***
## A_TYPE6         1.552e+01  3.878e+02   0.040 0.968067
## TOLL_ACCNT2    -2.591e+00  1.960e-01 -13.219  < 2e-16 ***
## TOLL_ACCNT98   -1.396e+00  9.016e-01  -1.548 0.121665
## CAR_SHARE2      7.660e-01  1.615e-01   4.743 2.11e-06 ***
## RES_TYPE2      -3.837e-01  1.767e-01  -2.172 0.029893 *
## RES_TYPE3      -7.933e-01  2.375e-01  -3.340 0.000838 ***
## RES_TYPE4      -1.376e+00  2.347e-01  -5.860 4.62e-09 ***
## RES_TYPE5      -1.182e+00  2.028e-01  -5.832 5.49e-09 ***
## RES_TYPE6      -7.338e-02  7.622e-01  -0.096 0.923301
## RES_TYPE9       1.138e+01  6.523e+03   0.002 0.998608
## RES_TYPE11     -1.185e+00  7.761e-01  -1.527 0.126754
## RES_TYPE97     -1.771e+01  6.523e+03  -0.003 0.997834
## RES_TYPE98     -2.038e+00  1.187e+00  -1.717 0.086020 .
## RES_TYPE99     -2.315e+00  1.260e+00  -1.837 0.066163 .
## INCOME2         6.854e-01  2.032e-01   3.372 0.000745 ***
## INCOME3         1.454e+00  2.331e-01   6.240 4.38e-10 ***
## INCOME4         1.988e+00  2.328e-01   8.540  < 2e-16 ***
## INCOME5         1.809e+00  2.327e-01   7.773 7.68e-15 ***
## INCOME6         2.610e+00  3.117e-01   8.374  < 2e-16 ***
## INCOME7         2.238e+00  3.184e-01   7.028 2.09e-12 ***
## INCOME8         2.165e+00  3.900e-01   5.550 2.85e-08 ***
## INCOME9         2.023e+00  3.919e-01   5.161 2.45e-07 ***
## INCOME10        1.593e+01  3.630e+02   0.044 0.965005
## INCOME98       -2.698e-01  1.492e+00  -0.181 0.856486
## INCOME99        1.702e+00  2.880e-01   5.910 3.42e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4926.6  on 8210  degrees of freedom
## Residual deviance: 2088.3  on 8177  degrees of freedom
## AIC: 2156.3
##
## Number of Fisher Scoring iterations: 17
```

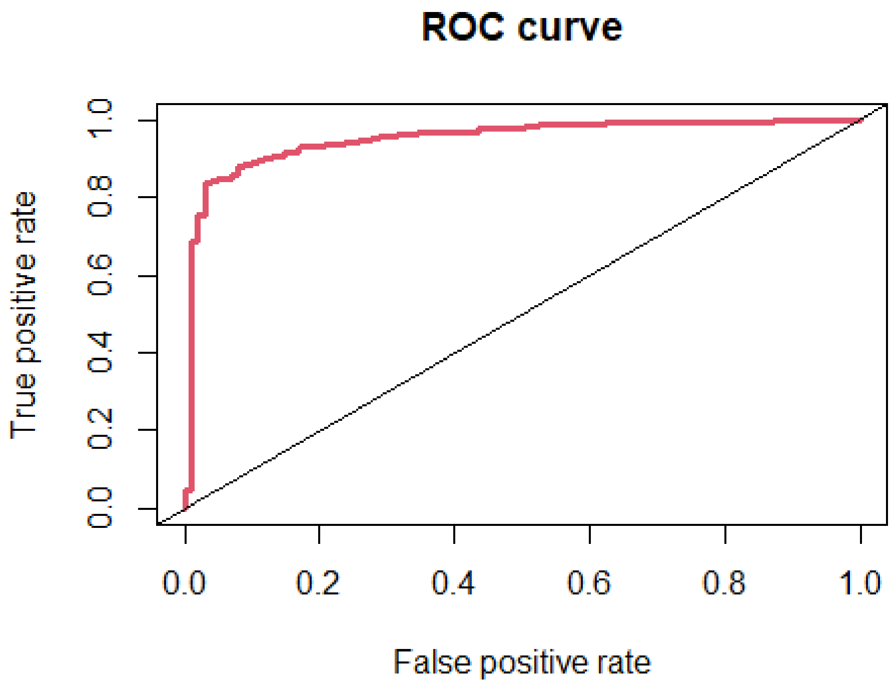
```
# codes from MUSA 508 class
# prediction of test set using all model
predAll <- data.frame(Outcome = hh.test$VEH,
                      Probs = predict(glm.fit.all, hh.test, type= "response
"))
```

```
predAll <-
  predAll %>%
  mutate(predOutcome  = as.factor(ifelse(predAll$Probs > 0.5, 1, 0)))

# get the confusion matrix and accuracy for the all model
cmatAll <-
  caret::confusionMatrix(predAll$predOutcome,
                         predAll$Outcome,
                         positive = "1")
cmatAll

## Confusion Matrix and Statistics
##
##            Reference
## Prediction   0   1
##          0  61  23
##          1  40 876
##
##                Accuracy : 0.937
##                  95% CI : (0.9201, 0.9513)
##     No Information Rate : 0.899
##     P-Value [Acc > NIR] : 1.454e-05
##
##                   Kappa : 0.6251
##
##  Mcnemar's Test P-Value : 0.04382
##
##             Sensitivity : 0.9744
##             Specificity : 0.6040
##          Pos Pred Value : 0.9563
##          Neg Pred Value : 0.7262
##              Prevalence : 0.8990
##          Detection Rate : 0.8760
##    Detection Prevalence : 0.9160
##       Balanced Accuracy : 0.7892
##
##        'Positive' Class : 1
##

# plot the ROC curve for the all model
predAll <- prediction(predAll$Probs, hh.test$VEH)
perfAll <- performance(predAll, measure="tpr", x.measure="fpr")

plot(perfAll, col=2, lwd=3, main="ROC curve")
abline(0,1)
```



```
# calculate auc of roc curve
auc = performance(predAll, "auc")@y.values
auc

## [[1]]
## [1] 0.9527913

# codes from MUSA 508 class
# prediction of test set using model 2
pred2 <- data.frame(Outcome = hh.test$VEH,
                      Probs = predict(glm.fit.2, hh.test, type= "response"))

pred2 <-
  pred2 %>%
  mutate(predOutcome  = as.factor(ifelse(pred2$Probs > 0.5, 1, 0)))

# get the confusion matrix and accuracy for model 2
cmat2 <-
  caret::confusionMatrix(pred2$predOutcome,
                         pred2$Outcome,
                         positive = "1")
cmat2

## Confusion Matrix and Statistics
##
##           Reference
## Prediction   0   1
##          0  57  26
##          1  44 873
##                                           
##                Accuracy : 0.93            
##                  95% CI : (0.9124, 0.945)
##     No Information Rate : 0.899           
##     P-Value [Acc > NIR] : 0.000402        
##                                           
##                   Kappa : 0.5814          
##                                           
##  Mcnemar's Test P-Value : 0.042165        
##                                           
##             Sensitivity : 0.9711          
##             Specificity : 0.5644          
##          Pos Pred Value : 0.9520          
##          Neg Pred Value : 0.6867          
##              Prevalence : 0.8990          
##          Detection Rate : 0.8730          
##    Detection Prevalence : 0.9170          
##       Balanced Accuracy : 0.7677          
##                                           
##        'Positive' Class : 1               
##
```

```
# plot the ROC curve for model 2
pred2 <- prediction(pred2$Probs, hh.test$VEH)
perfAll <- performance(pred2, measure="tpr", x.measure="fpr")

plot(perfAll, col=2, lwd=3, main="ROC curve")
abline(0,1)
```



```
# calculate auc of the roc curve
auc = performance(pred2, "auc")@y.values
auc

## [[1]]
## [1] 0.9519433
```