Data 621 Group 2 HW 4: Insurance

Members: Omar Pineda, Jeff Littlejohn, Sergio Ortega Cruz, Chester Poon, Simon Ustoyev

11/15/2019

R Markdown

```
library(knitr)
library(kableExtra)
library(tidyverse)
library(corrplot)
```

```
url <- './insurance_training_data.csv'
df <- read.csv(url, header = TRUE, row.names = 'INDEX')
```

It is helpful to get a glimpse of the data in a table format

```
kable(df[1:15,]) %>% kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

	TARGET_FLAG	TARGET_AMT	KIDSDRIV	AGE	HOMEKIDS	YOJ	INCOME	PARENT1	HOME_VAL	MSTATUS	SEX	EDUCATION	JOB	T
1	0	0.000	0	60	0	11	$67,349	No	$0	z_No	M	PhD	Professional	
2	0	0.000	0	43	0	11	$91,449	No	$257,252	z_No	M	z_High School	z_Blue Collar	
4	0	0.000	0	35	1	10	$16,039	No	$124,191	Yes	z_F	z_High School	Clerical	
5	0	0.000	0	51	0	14		No	$306,251	Yes	M	<High School	z_Blue Collar	
6	0	0.000	0	50	0	NA	$114,986	No	$243,925	Yes	z_F	PhD	Doctor	
7	1	2946.000	0	34	1	12	$125,301	Yes	$0	z_No	z_F	Bachelors	z_Blue Collar	
8	0	0.000	0	54	0	NA	$18,755	No		Yes	z_F	<High School	z_Blue Collar	
11	1	4021.000	1	37	2	NA	$107,961	No	$333,680	Yes	M	Bachelors	z_Blue Collar	
12	1	2501.000	0	34	0	10	$62,978	No	$0	z_No	z_F	Bachelors	Clerical	
13	0	0.000	0	50	0	7	$106,952	No	$0	z_No	M	Bachelors	Professional	
14	1	6077.000	0	53	0	14	$77,100	No	$0	z_No	z_F	Masters	Lawyer	
15	0	0.000	0	43	0	5	$52,642	No	$209,970	Yes	z_F	Masters	Professional	
16	0	0.000	0	55	0	11	$59,162	No	$180,232	Yes	M	Bachelors	Manager	
17	1	1267.000	0	53	0	11	$130,795	No	$0	z_No	M	PhD		
19	1	2920.167	0	45	0	0	$0	No	$106,859	Yes	z_F	<High School	Home Maker	

```
summary(df)
```

```
##   TARGET_FLAG       TARGET_AMT        KIDSDRIV           AGE
##  Min.   :0.0000   Min.   :     0   Min.   :0.0000   Min.   :16.00
##  1st Qu.:0.0000   1st Qu.:     0   1st Qu.:0.0000   1st Qu.:39.00
##  Median :0.0000   Median :     0   Median :0.0000   Median :45.00
##  Mean   :0.2638   Mean   :  1504   Mean   :0.1711   Mean   :44.79
##  3rd Qu.:1.0000   3rd Qu.:  1036   3rd Qu.:0.0000   3rd Qu.:51.00
##  Max.   :1.0000   Max.   :107586   Max.   :4.0000   Max.   :81.00
##                                                     NA's   :6
##     HOMEKIDS           YOJ            INCOME     PARENT1
##  Min.   :0.0000   Min.   : 0.0   $0      : 615   No :7084
##  1st Qu.:0.0000   1st Qu.: 9.0           : 445   Yes:1077
##  Median :0.0000   Median :11.0   $26,840 :   4
##  Mean   :0.7212   Mean   :10.5   $48,509 :   4
##  3rd Qu.:1.0000   3rd Qu.:13.0   $61,790 :   4
##  Max.   :5.0000   Max.   :23.0   $107,375:   3
##                   NA's   :454    (Other) :7086
##      HOME_VAL    MSTATUS      SEX               EDUCATION
##  $0      :2294   Yes :4894   M  :3786   <High School :1203
##          : 464   z_No:3267   z_F:4375   Bachelors    :2242
##  $111,129:   3                          Masters      :1658
##  $115,249:   3                          PhD          : 728
##  $123,109:   3                          z_High School:2330
##  $153,061:   3
##  (Other) :5391
##             JOB          TRAVTIME            CAR_USE        BLUEBOOK
##  z_Blue Collar:1825   Min.   :  5.00   Commercial:3029   $1,500 : 157
##  Clerical     :1271   1st Qu.: 22.00   Private   :5132   $6,000 :  34
##  Professional :1117   Median : 33.00                     $5,800 :  33
##  Manager      : 988   Mean   : 33.49                     $6,200 :  33
##  Lawyer       : 835   3rd Qu.: 44.00                     $6,400 :  31
##  Student      : 712   Max.   :142.00                     $5,900 :  30
##  (Other)      :1413                                      (Other):7843
##       TIF              CAR_TYPE    RED_CAR       OLDCLAIM
##  Min.   : 1.000   Minivan    :2145   no :5783   $0     :5009
##  1st Qu.: 1.000   Panel Truck: 676   yes:2378   $1,310 :   4
##  Median : 4.000   Pickup     :1389              $1,391 :   4
##  Mean   : 5.351   Sports Car : 907              $4,263 :   4
##  3rd Qu.: 7.000   Van        : 750              $1,105 :   3
##  Max.   :25.000   z_SUV      :2294              $1,332 :   3
##                                                 (Other):3134
##     CLM_FREQ      REVOKED       MVR_PTS          CAR_AGE
##  Min.   :0.0000   No :7161   Min.   : 0.000   Min.   :-3.000
##  1st Qu.:0.0000   Yes:1000   1st Qu.: 0.000   1st Qu.: 1.000
##  Median :0.0000              Median : 1.000   Median : 8.000
##  Mean   :0.7986              Mean   : 1.696   Mean   : 8.328
##  3rd Qu.:2.0000              3rd Qu.: 3.000   3rd Qu.:12.000
##  Max.   :5.0000              Max.   :13.000   Max.   :28.000
##                                               NA's   :510
##                  URBANICITY
##  Highly Urban/ Urban  :6492
##  z_Highly Rural/ Rural:1669
##
##
##
##
##
```

Data Clean-up and Missing values

The summary on the data identified the following variables with missing values
1. Age (6)
2. YOJ (454)
3. CAR_AGE (510)
4. INCOME(445)
5. HOME_VAL(464)

AGE Missing Values

Assigning a medium age would be appropriate given that there are only 6 records with missing values and those records either indicates having kids at home or being married.

```
df %>% filter(is.na(AGE)) %>% select(MSTATUS, HOMEKIDS)
```

```
##   MSTATUS HOMEKIDS
## 1    z_No        2
## 2    z_No        3
## 3    z_No        2
## 4    z_No        2
## 5     Yes        3
## 6     Yes        0
```

```
summary(df$AGE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##   16.00   39.00   45.00   44.79   51.00   81.00       6
```

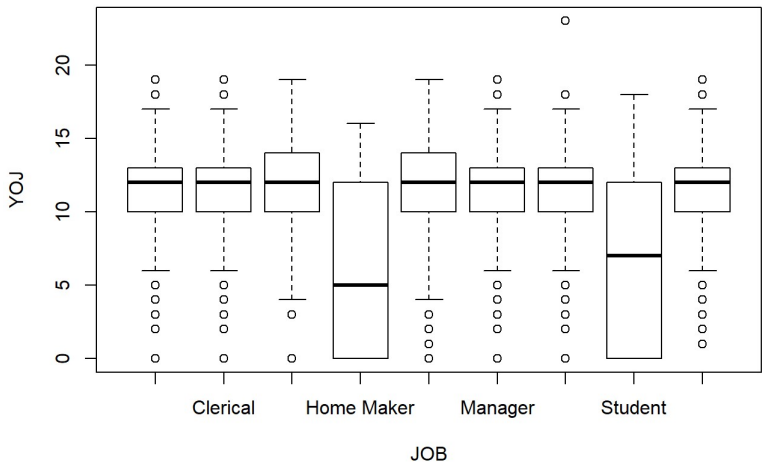
```
median_age <- summary(df$AGE)[['Median']]
df[is.na(df$AGE),]['AGE'] <- median_age
summary(df$AGE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   16.00   39.00   45.00   44.79   51.00   81.00
```

YOJ (Years on Job) Missing values

For the `YOJ` variable it would make sense to assign median values per Job type rather than just the overall median value. As you can see from the plot below, not all job types have similar median values.

```
plot(YOJ ~ JOB, df)
```




```
aggregate(YOJ ~ JOB, df, median)
```

```
##             JOB YOJ
## 1                12
## 2      Clerical  12
## 3        Doctor  12
## 4    Home Maker   5
## 5        Lawyer  12
## 6       Manager  12
## 7  Professional  12
## 8       Student   7
## 9 z_Blue Collar  12
```

```
summary(df$JOB)
```

```
##                  Clerical        Doctor    Home Maker        Lawyer
##           526          1271           246           641           835
##       Manager  Professional       Student z_Blue Collar
##           988          1117           712          1825
```

```
df$JOB <- fct_recode(df$JOB, 'UNKNOWN' = '')
summary(df$JOB)
```

```
##       UNKNOWN      Clerical        Doctor    Home Maker        Lawyer
##           526          1271           246           641           835
##       Manager  Professional       Student z_Blue Collar
##           988          1117           712          1825
```

```
df_tmp <- df %>% group_by(JOB) %>%
  mutate(NEW_YOJ = median(YOJ, na.rm = TRUE)) %>%
  select(JOB, YOJ, NEW_YOJ)

df[is.na(df$YOJ),]$YOJ <- df_tmp[is.na(df_tmp$YOJ),]$NEW_YOJ
summary(df$YOJ)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    0.00    9.00   12.00   10.53   13.00   23.00
```

Car Age Missing Values

Car age also have some invalid negative values. We can assign them to `NA` and then deal with them as missing values.

```
df$CAR_AGE[which(df$CAR_AGE < 0)] <- NA
summary(df$CAR_AGE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##    0.00    1.00    8.00    8.33   12.00   28.00     511
```

To deal with missing values of `CAR_AGE` it may be a good idea to find a correlation with `BLUEBOOK` value and derive approximate values for the age. However, for this we would require knowing the make and model of the cars. Given that this information is not available to us and that it is considerable number of rows with the missing values, it may be best to simply asign median age.

```
median_car_age <- summary(df$CAR_AGE)[['Median']]
df[is.na(df$CAR_AGE),]['CAR_AGE'] <- median_car_age
summary(df$CAR_AGE)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   4.000   8.000   8.309  12.000  28.000
```

INCOME Missing values

By default the income data was read as factors because of `$` currency symbol. We need to convert it to numerical values. By looking at the structure of the data frame `str(df)` it turns out that there are other similar columns which need to be converted as well.

```
str(df)
```

```
## 'data.frame':    8161 obs. of  25 variables:
##  $ TARGET_FLAG: int  0 0 0 0 0 1 0 1 1 0 ...
##  $ TARGET_AMT : num  0 0 0 0 0 0 ...
##  $ KIDSDRIV   : int  0 0 0 0 0 0 0 1 0 0 ...
##  $ AGE        : num  60 43 35 51 50 34 54 37 34 50 ...
##  $ HOMEKIDS   : int  0 0 1 0 0 1 0 2 0 0 ...
##  $ YOJ        : num  11 11 10 14 12 12 12 12 10 7 ...
##  $ INCOME     : Factor w/ 6613 levels "","$0","$1,007",..: 5033 6292 1250 1 509 746 1488 315 4765 282 ...
##  $ PARENT1    : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 2 1 1 1 1 ...
##  $ HOME_VAL   : Factor w/ 5107 levels "","$0","$100,093",..: 2 3259 348 3917 3034 2 1 4167 2 2 ...
##  $ MSTATUS    : Factor w/ 2 levels "Yes","z_No": 2 2 1 1 1 2 1 1 2 2 ...
##  $ SEX        : Factor w/ 2 levels "M","z_F": 1 1 2 1 2 2 2 1 2 1 ...
##  $ EDUCATION  : Factor w/ 5 levels "<High School",..: 4 5 5 1 4 2 1 2 2 2 ...
##  $ JOB        : Factor w/ 9 levels "UNKNOWN","Clerical",..: 7 9 2 9 3 9 9 9 2 7 ...
##  $ TRAVTIME   : int  14 22 5 32 36 46 33 44 34 48 ...
##  $ CAR_USE    : Factor w/ 2 levels "Commercial","Private": 2 1 2 2 2 1 2 1 2 1 ...
##  $ BLUEBOOK   : Factor w/ 2789 levels "$1,500","$1,520",..: 434 503 2212 553 802 746 2672 701 135 852 ...
##  $ TIF        : int  11 1 4 7 1 1 1 1 1 7 ...
##  $ CAR_TYPE   : Factor w/ 6 levels "Minivan","Panel Truck",..: 1 1 6 1 6 4 6 5 6 5 ...
##  $ RED_CAR    : Factor w/ 2 levels "no","yes": 2 2 1 2 1 1 1 2 1 1 ...
##  $ OLDCLAIM   : Factor w/ 2857 levels "$0","$1,000",..: 1449 1 1311 1 432 1 1 510 1 1 ...
##  $ CLM_FREQ   : int  2 0 2 0 2 0 0 1 0 0 ...
##  $ REVOKED    : Factor w/ 2 levels "No","Yes": 1 1 1 1 2 1 1 2 1 1 ...
##  $ MVR_PTS    : int  3 0 3 0 3 0 0 10 0 1 ...
##  $ CAR_AGE    : num  18 1 10 6 17 7 1 7 1 17 ...
##  $ URBANICITY : Factor w/ 2 levels "Highly Urban/ Urban",..: 1 1 1 1 1 1 1 1 1 2 ...
```

```
# INCOME
class(df$INCOME)
```

```
## [1] "factor"
```

```
df$INCOME <- parse_number(as.character(df$INCOME))
summary(df$INCOME)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0   28097   54028   61898   85986  367030     445
```

```
# HOME_VAL
df$HOME_VAL <- parse_number(as.character(df$HOME_VAL))
summary(df$HOME_VAL)
```

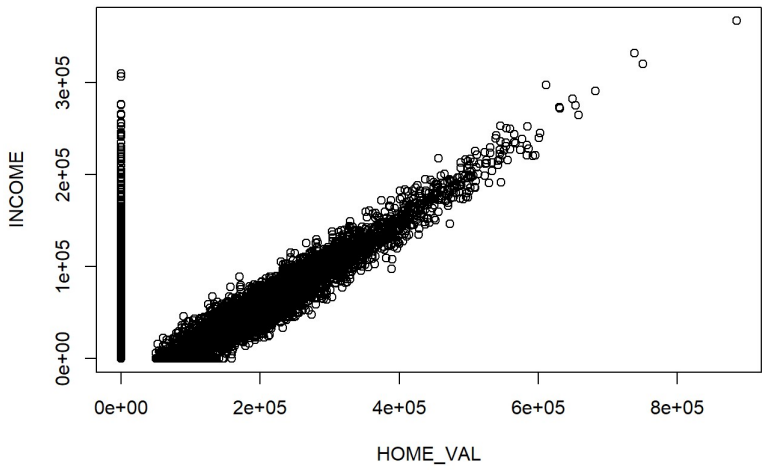
```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's
##       0       0  161160  154867  238724  885282     464
```

```
# BLUEBOOK
df$BLUEBOOK <- parse_number(as.character(df$BLUEBOOK))

# OLDCLAIM
df$OLDCLAIM <- parse_number(as.character(df$OLDCLAIM))
```

```
#n_na <- df %>% filter(is.na(INCOME) & is.na(HOME_VAL)) %>% count()
nrow_na <- nrow(df[is.na(df$INCOME) & is.na(df$HOME_VAL),])
```

Both the `Income (INCOME)` and the `Home Value (HOME_VAL)` variables have missing values. However only 33 instances where both are missing. On their own these variables have over 400 missing values. However, this is not a surprise that the two variables are positively correlated, because the higher the income, the more expesive a home value can be. The plot below does show this correlation:

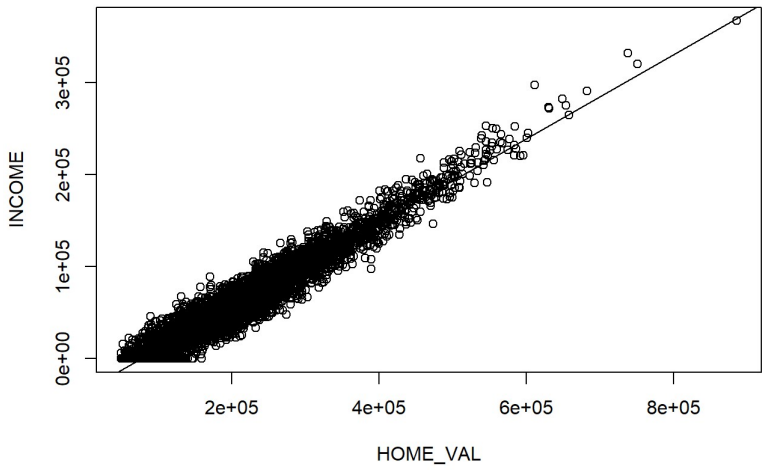
```
plot(INCOME~HOME_VAL, df)
```



Given this correlation it may be possible to come up with an impute strategy where the two variables can help each other. The `Home Value` variable with value of 0 is considered to indicate that someone is not a home owner. Therefore, we've decided to execute the following strategy for imputing these two variables:

1. For the 33 instances where both are missing, randomly assign a value to `HOME_VAL` variable choosing between 0 and median home value.
2. Build a simple linear model to predict income based on home value (i.e. where home value > 0). Any negative predicted amounts should be changed to 0.
3. Use median income for the remaining missing income values.
4. Finally transform the `HOME_VAL` variable to a 0 or 1 binary indicator (0=*not a home owner*). Any missing values are to be randomly assigned 0 or 1.

```
# 1
median_home_val <- summary(df$HOME_VAL)[['Median']]
df[is.na(df$INCOME) & is.na(df$HOME_VAL),]$HOME_VAL <- sample(c(0, median_home_val), size=nrow_na, replace = T)

# 2
lm_data <- df[df$HOME_VAL > 0,]
lm1 <- lm(INCOME~HOME_VAL, data = lm_data)
plot(INCOME~HOME_VAL, data = lm_data)
abline(lm1)
```




```
summary(lm1)
```

```
##
## Call:
## lm(formula = INCOME ~ HOME_VAL, data = lm_data)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -44742  -8370    -80   8223  53303
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -3.587e+04  4.362e+02  -82.24   <2e-16 ***
## HOME_VAL     4.580e-01  1.808e-03  253.36   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12430 on 5107 degrees of freedom
##   (743 observations deleted due to missingness)
## Multiple R-squared:  0.9263, Adjusted R-squared:  0.9263
## F-statistic: 6.419e+04 on 1 and 5107 DF,  p-value: < 2.2e-16
```

```
coef(lm1)
```

```
##   (Intercept)      HOME_VAL
## -3.587234e+04  4.580265e-01
```

```
  # qqnorm(resid(lm1))
  # qqline(resid(lm1))
lm1.predict <- predict(lm1, newdata = df[is.na(df$INCOME) & df$HOME_VAL > 0,]['HOME_VAL'])
df[is.na(df$INCOME) & df$HOME_VAL > 0,]$INCOME <- lm1.predict
# deal with negative values
df[!is.na(df$INCOME) & df$INCOME < 0,]$INCOME <- 0

# 3
median_income <- summary(df$INCOME)[['Median']]
df[is.na(df$INCOME),]$INCOME <- median_income
summary(df$INCOME)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##       0   28629   53677   61642   85056  367030
```

```
# 4
df$HOME_OWN <- ifelse(df$HOME_VAL > 0, 1, 0)
# deal with missing values
nrow_na <- nrow(df[is.na(df$HOME_OWN),])
df[is.na(df$HOME_OWN),]$HOME_OWN <- sample(c(0, 1), size=nrow_na, replace = T)
summary(df$HOME_OWN)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0000  0.0000  1.0000  0.6888  1.0000  1.0000
```

Before moving on, it would make sense to transform the Income variable as well from being a continuous numeric variable into a categorical 3 level (Low, Mid, High) variable. Having numerical values would not make sense as a predictor for the kind of responses we want to predict. Also, it would help us deal with cases where Income is entered as 0 value.

```
sum_income <- summary(df$INCOME)
low_income_ub <- sum_income[['1st Qu.']]
high_income_lb <- sum_income[['3rd Qu.']]
df$INCOME_CLASS <- as.factor(case_when(
  df$INCOME < low_income_ub ~ 'LOW',
  df$INCOME > high_income_lb ~ 'HIGH',
  TRUE ~ 'MID'))
summary(df$INCOME_CLASS)
```

```
## HIGH  LOW  MID
## 2040 2040 4081
```

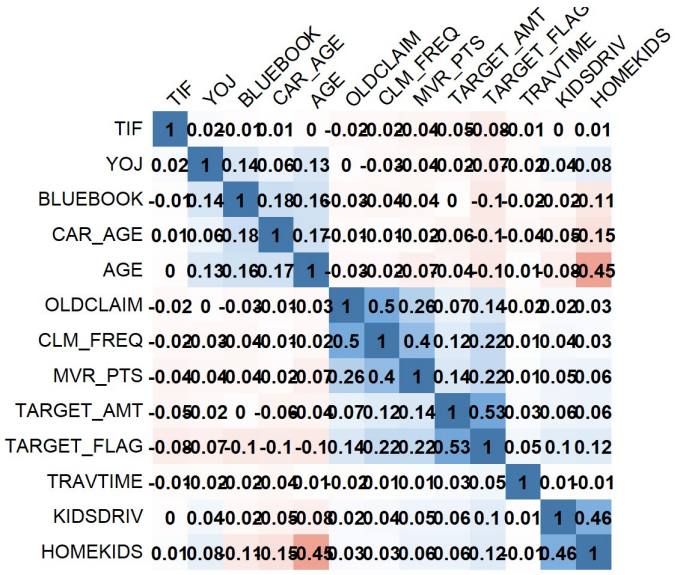
To create the 3 category levels, we used Inter-Quartile ranges, where below 25% would rank as Low, above 75% would rank as High and the rest is Mid.

Now let's explore correlation between the numeric variables:

```
# df_train <- select(df, -'INCOME', -'HOME_VAL')

df_train <- select(df, TARGET_FLAG,TARGET_AMT,KIDSDRIV,AGE,HOMEKIDS,YOJ,TRAVTIME,BLUEBOOK,TIF,OLDCLAIM,CLM_FREQ,MVR_PTS,CAR_
AGE)

# str(df_train)

cor_train <- cor(df_train)

kable(cor_train, "html") %>% kable_styling(bootstrap_options = c("striped", "hover", "condensed", "responsive"))
```

	TARGET_FLAG	TARGET_AMT	KIDSDRIV	AGE	HOMEKIDS	YOJ	TRAVTIME	BLUEBOOK	TIF	OLDCLAIM	CL
TARGET_FLAG	1.0000000	0.5342461	0.1036683	-0.1031030	0.1156210	-0.0703184	0.0483683	-0.1033832	-0.0823700	0.1380838	0
TARGET_AMT	0.5342461	1.0000000	0.0553942	-0.0417152	0.0619880	-0.0207650	0.0279870	-0.0046995	-0.0464808	0.0709533	0
KIDSDRIV	0.1036683	0.0553942	1.0000000	-0.0751817	0.4640152	0.0420904	0.0084473	-0.0215493	-0.0019887	0.0204027	0
AGE	-0.1031030	-0.0417152	-0.0751817	1.0000000	-0.4450739	0.1341494	0.0052607	0.1649057	-0.0000592	-0.0292718	-0
HOMEKIDS	0.1156210	0.0619880	0.4640152	-0.4450739	1.0000000	0.0800700	-0.0072456	-0.1078936	0.0118133	0.0299110	0
YOJ	-0.0703184	-0.0207650	0.0420904	0.1341494	0.0800700	1.0000000	-0.0196966	0.1444239	0.0223028	-0.0037974	-0
TRAVTIME	0.0483683	0.0279870	0.0084473	0.0052607	-0.0072456	-0.0196966	1.0000000	-0.0170013	-0.0116046	-0.0192672	0
BLUEBOOK	-0.1033832	-0.0046995	-0.0215493	0.1649057	-0.1078936	0.1444239	-0.0170013	1.0000000	-0.0054246	-0.0295176	-0
TIF	-0.0823700	-0.0464808	-0.0019887	-0.0000592	0.0118133	0.0223028	-0.0116046	-0.0054246	1.0000000	-0.0219582	-0
OLDCLAIM	0.1380838	0.0709533	0.0204027	-0.0292718	0.0299110	-0.0037974	-0.0192672	-0.0295176	-0.0219582	1.0000000	0
CLM_FREQ	0.2161961	0.1164192	0.0370629	-0.0240716	0.0293493	-0.0250507	0.0065602	-0.0363415	-0.0230230	0.4951308	1
MVR_PTS	0.2191971	0.1378655	0.0535664	-0.0715052	0.0606013	-0.0351706	0.0105985	-0.0391308	-0.0410457	0.2644850	0
CAR_AGE	-0.0970694	-0.0576010	-0.0520731	0.1709784	-0.1473703	0.0630084	-0.0364222	0.1834542	0.0075595	-0.0121662	-0

```
col <- colorRampPalette(c("#BB4444", "#EE9988", "#FFFFFF", "#77AADD", "#4477AA"))

corrplot(cor_train, method = "shade", shade.col = NA, tl.col = "black", tl.srt = 45, col = col(200), addCoef.col = "black",
cl.pos = "n", order = "AOE")
```


TIF YOJ BLUEBOOK CAR_AGE AGE OLDCLAIM CLM_FREQ MVR_PTS TARGET_AMT TARGET_FLAG TRAVTIME KIDSDRIV HOMEKIDS
TIF 1 0.02 -0.01 0.01 0 -0.02 0.02 0.04 0.05 0.08 0.01 0 0.01
YOJ 0.02 1 0.14 0.06 0.13 0 -0.03 0.04 0.02 0.07 0.02 0.04 0.08
BLUEBOOK -0.01 0.14 1 0.18 0.16 0.03 0.04 0.04 0 -0.1 -0.02 0.02 0.11
CAR_AGE 0.01 0.06 0.18 1 0.17 0.01 0.01 0.02 0.06 -0.1 0.04 0.05 0.15
AGE 0 0.13 0.16 0.17 1 -0.03 0.02 0.07 0.04 -0.1 0.01 0.08 0.45
OLDCLAIM -0.02 0 -0.03 0.01 0.03 1 0.5 0.26 0.07 0.14 0.02 0.02 0.03
CLM_FREQ -0.02 0.03 0.04 0.01 0.02 0.5 1 0.4 0.12 0.22 0.01 0.04 0.03
MVR_PTS -0.04 0.04 0.04 0.02 0.07 0.26 0.4 1 0.14 0.22 0.01 0.05 0.06
TARGET_AMT -0.05 0.02 0 -0.06 0.04 0.07 0.12 0.14 1 0.53 0.03 0.06 0.06
TARGET_FLAG -0.08 0.07 -0.1 -0.1 -0.1 0.14 0.22 0.22 0.53 1 0.05 0.1 0.12
TRAVTIME -0.01 0.02 0.02 0.04 0.01 0.02 0.01 0.01 0.03 0.05 1 0.01 0.01
KIDSDRIV 0 0.04 0.02 0.05 0.08 0.02 0.04 0.05 0.06 0.1 0.01 1 0.46
HOMEKIDS 0.01 0.08 0.11 0.15 0.45 0.03 0.03 0.06 0.06 0.12 0.01 0.46 1