OBJECTIVE

Although we think of college as a time when young adults experiment with alcohol, the college years are rarely the first time students have faced decisions about alcohol. According to the nationally representative “Monitoring the Future Study”, in 2012, 42 percent of high school seniors reported having had alcohol (more than just a few sips) within 30 days prior to the survey, and 24 percent reported binge drinking within the previous two weeks. During childhood and teenage years, the brain is still developing. Alcohol consumption showed negative associations with motivation for and subjectively achieved academic performance. Drinking could aﬀect child’s performance at school and prevent them from reaching their full potential.

University alcohol prevention activities might have positive impact on students’ academic success.( [Walid El Ansari](http://www.ncbi.nlm.nih.gov/pubmed/?term=El%20Ansari%20W%5BAuthor%5D&cauthor=true&cauthor_uid=24319558), [Christiane Stock](http://www.ncbi.nlm.nih.gov/pubmed/?term=Stock%20C%5BAuthor%5D&cauthor=true&cauthor_uid=24319558) ,[Int J Prev Med](http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3843305/). 2013 Oct; 4(10): 1175–1188. Is Alcohol Consumption Associated with Poor Academic Achievement in University Students? ). Modeling student alcohol consumption is an important tool for both educators and students, since it can help a better understanding of this problem and improve it. For instance, school professionals could perform corrective measures for the students.

The present work intends to approach student alcohol consumption in secondary education using regression models with “R”.

DATABASE

Students Alcohol Consumption

<https://archive.ics.uci.edu/ml/datasets/STUDENT+ALCOHOL+CONSUMPTION>

Variables :

1 school - student's school (binary: "GP" - Gabriel Pereira or "MS" - Mousinho da Silveira)

2 sex - student's sex (binary: "F" - female or "M" - male)

3 age - student's age (numeric: from 15 to 22)

4 address - student's home address type (binary: "U" - urban or "R" - rural)

5 famsize - family size (binary: "LE3" - less or equal to 3 or "GT3" - greater than 3)

6 Pstatus - parent's cohabitation status (binary: "T" - living together or "A" - apart)

7 Medu - mother's education (numeric: 0 - none,  1 - primary education (4th grade), 2 â 5th to 9th grade, 3 â secondary education or 4 â higher education)

8 Fedu - father's education (numeric: 0 - none,  1 - primary education (4th grade), 2 â 5th to 9th grade, 3 â secondary education or 4 â higher education)

9 Mjob - mother's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at\_home" or "other")

10 Fjob - father's job (nominal: "teacher", "health" care related, civil "services" (e.g. administrative or police), "at\_home" or "other")

11 reason - reason to choose this school (nominal: close to "home", school "reputation", "course" preference or "other")

12 guardian - student's guardian (nominal: "mother", "father" or "other")

13 travel+time - home to school travel time (numeric: 1 - <15 min., 2 - 15 to 30 min., 3 - 30 min. to 1 hour, or 4 - >1 hour)

14 studytime - weekly study time (numeric: 1 - <2 hours, 2 - 2 to 5 hours, 3 - 5 to 10 hours, or 4 - >10 hours)

15 failures - number of past class failures (numeric: n if 1<=n<3, else 4)

16 schoolsup - extra educational support (binary: yes or no)

17 famsup - family educational support (binary: yes or no)

18 paid - extra paid classes within the course subject (Math or Portuguese) (binary: yes or no)

19 activities - extra-curricular activities (binary: yes or no)

20 nursery - attended nursery school (binary: yes or no)

21 higher - wants to take higher education (binary: yes or no)

22 internet - Internet access at home (binary: yes or no)

23 romantic - with a romantic relationship (binary: yes or no)

24 famrel - quality of family relationships (numeric: from 1 - very bad to 5 - excellent)

25 freetime - free time after school (numeric: from 1 - very low to 5 - very high)

26 goout - going out with friends (numeric: from 1 - very low to 5 - very high)

27 Dalc - workday alcohol consumption (numeric: from 1 - very low to 5 - very high)

28 Walc - weekend alcohol consumption (numeric: from 1 - very low to 5 - very high)

29 health - current health status (numeric: from 1 - very bad to 5 - very good)

30 absences - number of school absences (numeric: from 0 to 93)

31 G1 - first period grade (numeric: from 0 to 20)

31 G2 - second period grade (numeric: from 0 to 20)

32 G3 - final grade (numeric: from 0 to 20, output target)

There are several  students that belong to both datasets .

These students can be identified by searching for identical attributes

that characterize each student.

First of all necessary packages needs to be installed:

#installing packages

wants <- c("mlogit","mgcv", "nnet","e1071" ,"VGAM","nnet","rpart.plot","ROCR","randomForest",

"caret","lift","nnet","ggplot2","reshape2","caTools","mlbench","SDMTools","pROC")

has <- wants %in% rownames(installed.packages())

if(any(!has)) install.packages(wants[!has])

Loading data from two .csv files:

#loading data

setwd("C:/Users/111/Desktop/Alcohol-master")

d1=read.table("student-mat.csv",sep=";",header=TRUE)

d2=read.table("student-por.csv",sep=";",header=TRUE)

#there are severalstudents that belong to both datasets .

#These students can be identified by searching for identical attributes

#that characterize each student.

#binding datasets

df=rbind(d1, d2)

creating the unique index using “mgcv”, and getting the final data set d3:

library(mgcv)

unique=uniquecombs(df[1:13]) #columnes to identify the unique subjects by.

uniqueIndex<-attributes(unique)

d3=df[uniqueIndex$row.names,]

Structure d3:

'data.frame': 666 obs. of 33 variables:

$ school : Factor w/ 2 levels "GP","MS": 1 1 1 1 1 1 1 1 1 1 ...

$ sex : Factor w/ 2 levels "F","M": 1 1 1 1 1 2 2 1 2 2 ...

$ age : int 18 17 15 15 16 16 16 17 15 15 ...

$ address : Factor w/ 2 levels "R","U": 2 2 2 2 2 2 2 2 2 2 ...

$ famsize : Factor w/ 2 levels "GT3","LE3": 1 1 2 1 1 2 2 1 2 1 ...

$ Pstatus : Factor w/ 2 levels "A","T": 1 2 2 2 2 2 2 1 1 2 ...

$ Medu : int 4 1 1 4 3 4 2 4 3 3 ...

$ Fedu : int 4 1 1 2 3 3 2 4 2 4 ...

$ Mjob : Factor w/ 5 levels "at\_home","health",..: 1 1 1 2 3 4 3 3 4 3 ...

$ Fjob : Factor w/ 5 levels "at\_home","health",..: 5 3 3 4 3 3 3 5 3 3 ...

$ reason : Factor w/ 4 levels "course","home",..: 1 1 3 2 2 4 2 2 2 2 ...

$ guardian : Factor w/ 3 levels "father","mother",..: 2 1 2 2 1 2 2 2 2 2 ...

$ traveltime: int 2 1 1 1 1 1 1 2 1 1 ...

$ studytime : int 2 2 2 3 2 2 2 2 2 2 ...

$ failures : int 0 0 3 0 0 0 0 0 0 0 ...

$ schoolsup : Factor w/ 2 levels "no","yes": 2 1 2 1 1 1 1 2 1 1 ...

$ famsup : Factor w/ 2 levels "no","yes": 1 2 1 2 2 2 1 2 2 2 ...

$ paid : Factor w/ 2 levels "no","yes": 1 1 2 2 2 2 1 1 2 2 ...

$ activities: Factor w/ 2 levels "no","yes": 1 1 1 2 1 2 1 1 1 2 ...

$ nursery : Factor w/ 2 levels "no","yes": 2 1 2 2 2 2 2 2 2 2 ...

$ higher : Factor w/ 2 levels "no","yes": 2 2 2 2 2 2 2 2 2 2 ...

$ internet : Factor w/ 2 levels "no","yes": 1 2 2 2 1 2 2 1 2 2 ...

$ romantic : Factor w/ 2 levels "no","yes": 1 1 1 2 1 1 1 1 1 1 ...

$ famrel : int 4 5 4 3 4 5 4 4 4 5 ...

$ freetime : int 3 3 3 2 3 4 4 1 2 5 ...

$ goout : int 4 3 2 2 2 2 4 4 2 1 ...

$ Dalc : int 1 1 2 1 1 1 1 1 1 1 ...

$ Walc : int 1 1 3 1 2 2 1 1 1 1 ...

$ health : int 3 3 3 5 5 5 3 1 1 5 ...

$ absences : int 6 4 10 2 4 10 0 6 0 0 ...

$ G1 : int 5 5 7 15 6 15 12 6 16 14 ...

$ G2 : int 6 5 8 14 10 15 12 5 18 15 ...

$ G3 : int 6 6 10 15 10 15 11 6 19 15 ...

ANALIZING THE DATA

#storing my themes

library(ggplot2)

mytheme1=theme\_bw(base\_size = 12, base\_family = "")

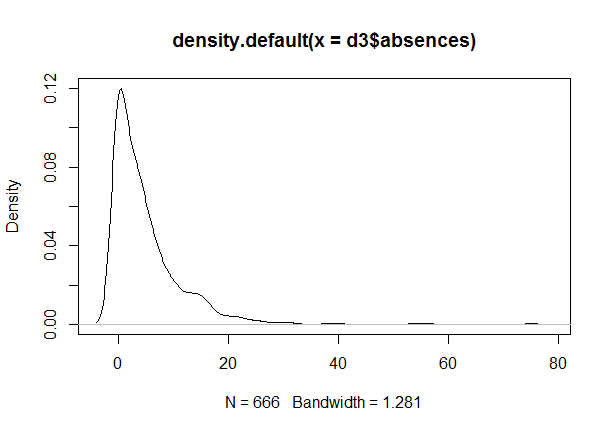
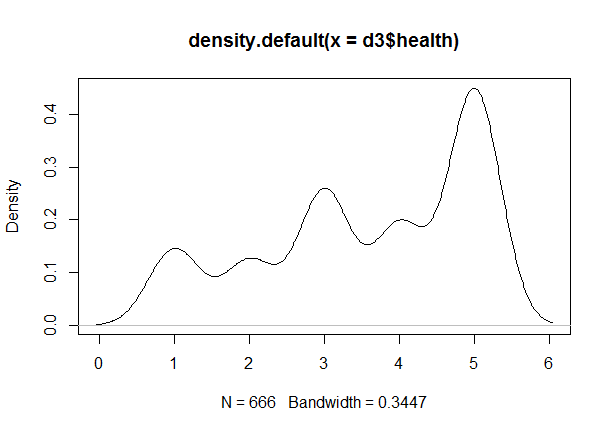
mytheme2=theme(panel.grid.major = element\_line(colour = "white")) +

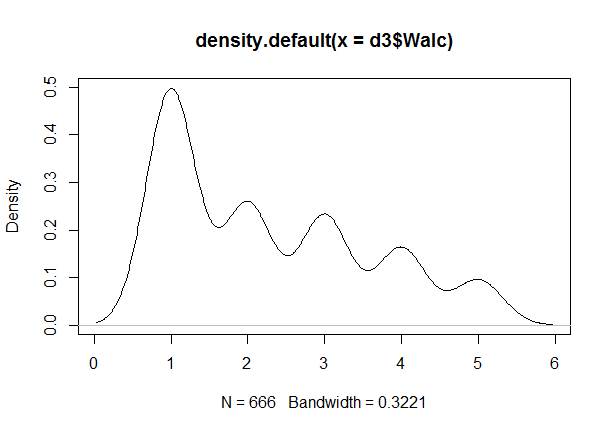
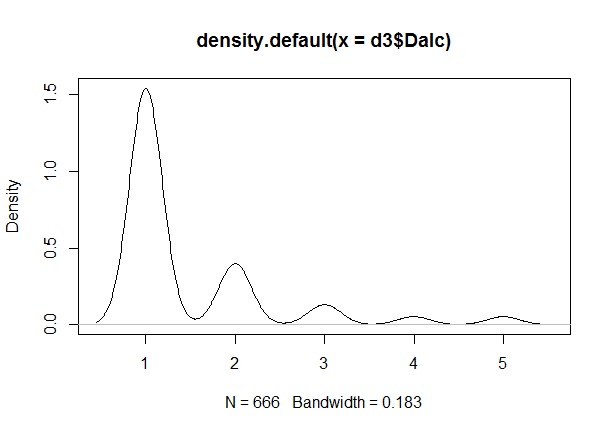
theme(panel.border =

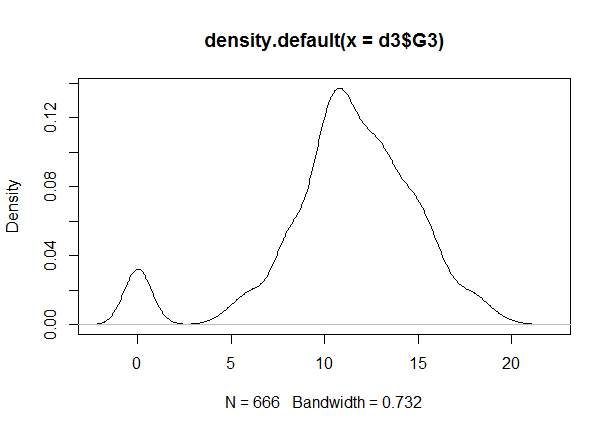
element\_rect(linetype = "solid", colour = "white"))

Simple density plots of the dependent variables will be plotted:

#plotting dependent variables







As we see there are 2 different alcohol consumption variables: Dalc and Walc, daily and weekends prospectively.

table(d2$Dalc) #weekday alcohol consumption 1-5 score

1 2 3 4 5

451 121 43 17 17 =34

table(d2$Walc) #weekend alcohol consumption 1-5 score

1 2 3 4 5

247 150 120 87 45 = 132

As observed high level drinking (4-5) is greater on the weekends and it is not so significant for everyday performance in the schools as daily drinking. Hence “Dalc” variable will be used for the next models as a dependent variable.

To find out the most influential variables the linear logistic regression will be build.

#building linear regression model

linear<-lm(Dalc ~ ., d3)

summary(linear)

Coefficients:

Estimate Std. Error t value Pr(>|t|)

(Intercept) -0.170797 0.565495 -0.302 0.762729

schoolMS 0.041222 0.116969 0.352 0.724646

sexM 0.183566 0.059005 3.111 0.001950 \*\*

age 0.019120 0.027873 0.686 0.492989

addressU -0.013894 0.073109 -0.190 0.849337

famsizeLE3 0.072926 0.060021 1.215 0.224828

PstatusT -0.139556 0.084718 -1.647 0.100001

Medu 0.127691 0.039199 3.258 0.001185 \*\*

Fedu -0.065954 0.033547 -1.966 0.049737 \*

Mjobhealth -0.409413 0.136727 -2.994 0.002859 \*\*

Mjobother 0.056958 0.087009 0.655 0.512950

Mjobservices -0.067699 0.098598 -0.687 0.492576

Mjobteacher -0.124710 0.126670 -0.985 0.325240

Fjobhealth -0.066378 0.178882 -0.371 0.710711

Fjobother -0.327085 0.130334 -2.510 0.012339 \*

Fjobservices -0.050507 0.135782 -0.372 0.710039

Fjobteacher -0.113211 0.162857 -0.695 0.487217

reasonhome 0.047930 0.066920 0.716 0.474123

reasonother 0.345565 0.102068 3.386 0.000755 \*\*\*

reasonreputation -0.045160 0.070091 -0.644 0.519611

guardianmother -0.032562 0.066193 -0.492 0.622950

guardianother 0.176591 0.129999 1.358 0.174826

traveltime 0.063482 0.040474 1.568 0.117286

studytime 0.008953 0.035211 0.254 0.799380

failures 0.017393 0.044753 0.389 0.697667

schoolsupyes 0.144666 0.079771 1.814 0.070232 .

famsupyes 0.059651 0.057192 1.043 0.297351

paidyes 0.067767 0.059867 1.132 0.258087

activitiesyes -0.079941 0.054266 -1.473 0.141224

nurseryyes -0.116294 0.067679 -1.718 0.086235 .

higheryes 0.196401 0.124626 1.576 0.115551

internetyes 0.076681 0.074830 1.025 0.305885

romanticyes 0.055150 0.057877 0.953 0.341014

famrel -0.030495 0.029792 -1.024 0.306417

freetime 0.088547 0.028698 3.085 0.002122 \*\*

goout -0.019645 0.027212 -0.722 0.470603

Walc 0.410729 0.024550 16.730 < 2e-16 \*\*\*

health 0.019348 0.019413 0.997 0.319321

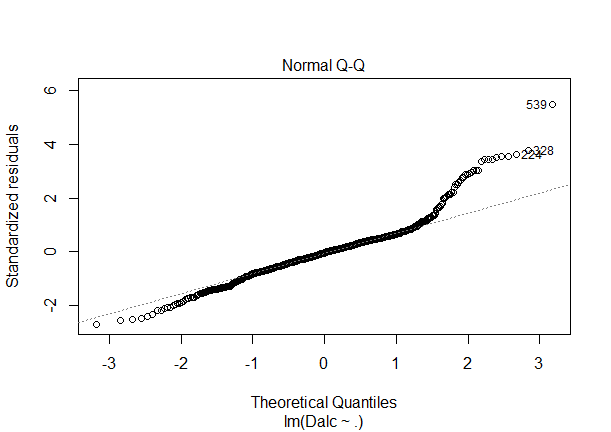
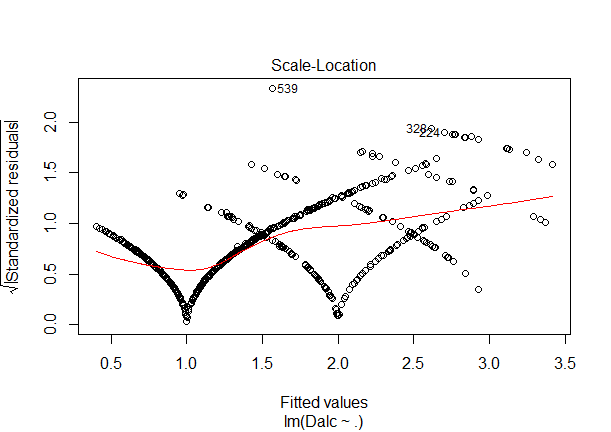
absences 0.002600 0.003961 0.656 0.511829

G1 -0.004112 0.017563 -0.234 0.814944

G2 0.012143 0.022047 0.551 0.581984

G3 -0.011537 0.015813 -0.730 0.465926

plot(linear)

#plotting independent variables

ggplot (aes(x = Dalc,fill=sex),data = d3) + geom\_histogram(binwidth = 1,na.rm = T) +

facet\_grid(sex~.,scale="free") +mytheme1+mytheme2

ggplot (aes(x = Dalc,fill=age),data = d3) + geom\_histogram(binwidth = 1,na.rm = T) +

facet\_grid(age~.,scale="free") +mytheme1+mytheme2

ggplot (aes(x = Dalc,fill=Medu),data = d3) + geom\_histogram(binwidth = 1,na.rm = T) +

facet\_grid(Medu~.,scale="free") +mytheme1+mytheme2

ggplot (aes(x = Dalc,fill=Mjob),data = d3) + geom\_histogram(binwidth = 1,na.rm = T) +

facet\_grid(Mjob~.,scale="free") +mytheme1+mytheme2

ggplot (aes(x = Dalc,fill=Fedu),data = d3) + geom\_histogram(binwidth = 1,na.rm = T) +

facet\_grid(Fedu~.,scale="free") +mytheme1+mytheme2

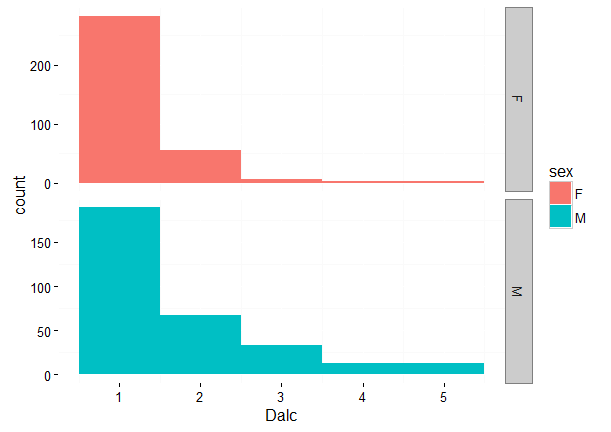
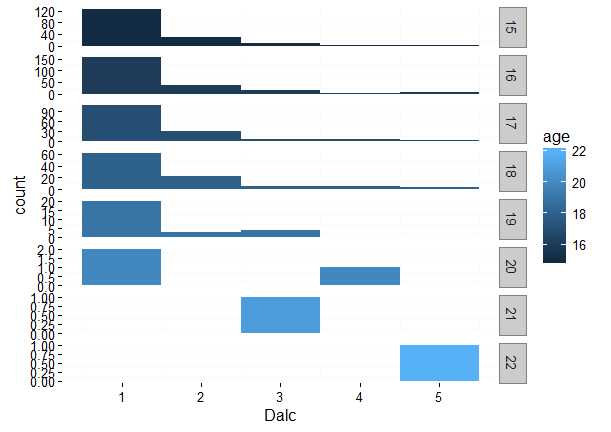
ggplot (aes(x = Dalc,fill=Fjob),data = d3) + geom\_histogram(binwidth = 1,na.rm = T) +

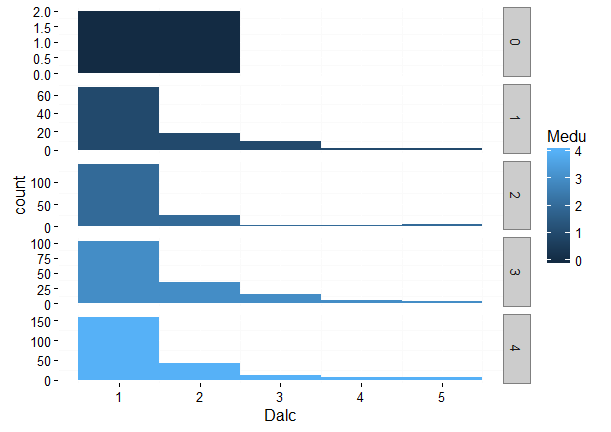
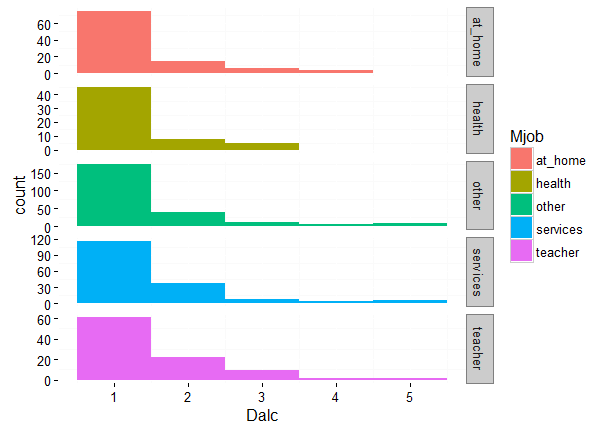
facet\_grid(Fjob~.,scale="free") +mytheme1+mytheme2

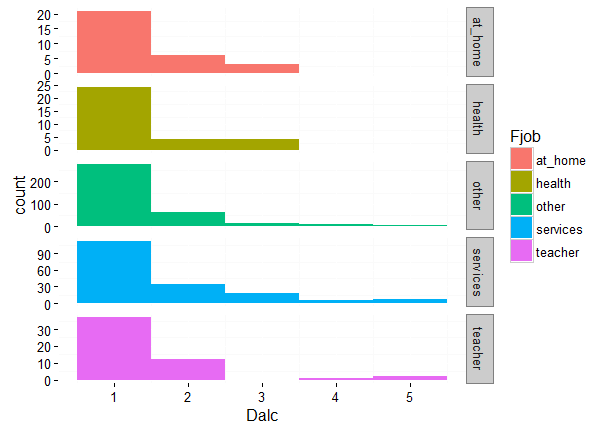
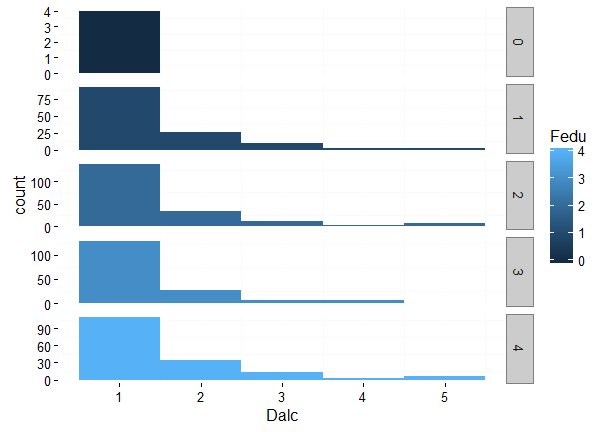
ggplot (aes(x = Dalc,fill=freetime),data = d3) + geom\_histogram(binwidth = 1,na.rm = T) +

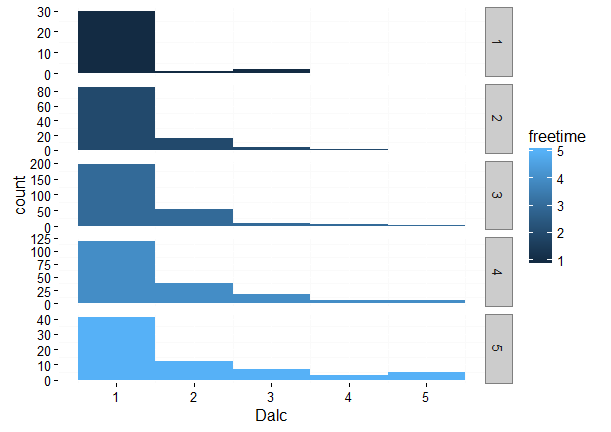
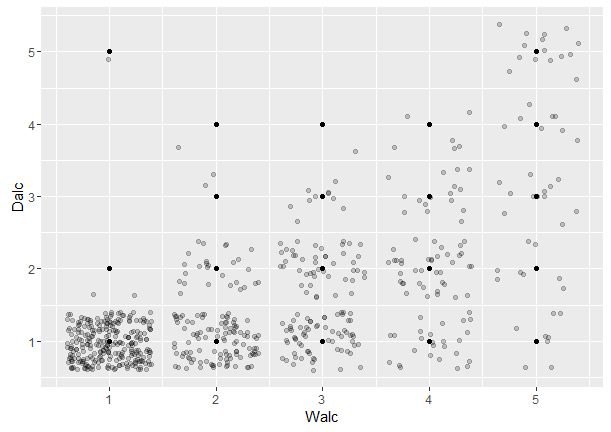
facet\_grid(freetime~.,scale="free") +mytheme1+mytheme2

ggplot (aes(x = Walc,y=Dalc),data = d3) + geom\_point()+geom\_jitter(alpha = 0.2)

Hi alcohol consumption

#preparing training and testing sets for the future work

library(caTools)

set.seed(76)

sample.d3 = sample.split(d3$Dalc, SplitRatio=0.7,group = NULL )

trainIdx = which(sample.d3 == TRUE)

trainData = d3[trainIdx,]

testIdx = which(sample.d3 == FALSE)

testData = d3[testIdx,]

#Display of distributed data

dim(trainData) [1] 465 33

dim(testData) [1] 201 33

#Logistic regression

set.seed(123)

#creating logistic regression model

train.glm<- glm(Dalc~ ., data=trainData,family= gaussian)

summary(train.glm)

plot(train.glm)

#predicting the

predicted.glm=predict(train.glm,type="response")

head(predicted.glm)

1 2 3 4 5 6

1.1628075 0.6538965 1.9595075 0.7408622 1.2479522 1.4182362

summary(predicted.glm)

tapply(predicted.glm,d3$Dalc,mean)

1 2 3 4 5

1.197307 1.906120 2.334112 2.682862 2.877123

table(d3$Dalc, predicted.glm >2.5) #with threshold 2.5

FALSE TRUE

1 467 5

2 111 11

3 22 18

4 6 10

5 1 15

We have a multilevel variable Dalc

levels(as.factor(d3$Dalc)) "1" "2" "3" "4" "5"

For the Daily alcohol consumption, we will use Multinomial Regression Model

#multinomial regression

require(foreign)

require(nnet)

require(ggplot2)

require(reshape2)

Executing a multinomial regression with independent variables on train data.

mult.regression <- multinom(as.factor(Dalc )~ . , data = trainData)

summary(mult.regression)

We’ll calculate Z score and p-Value for the variables in the model.

z <summary(mult.regression)$coefficients/

summary(mult.regression)$standard.errors

p <- (1 - pnorm(abs(z), 0, 1))\*2

predict.test.multinom<-predict(mult.regression,newdata = testData)

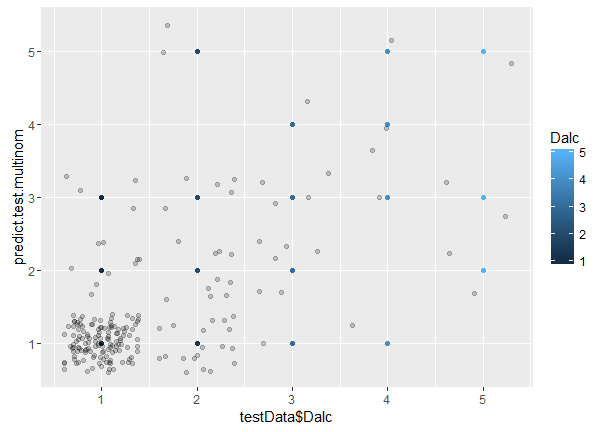
predict.test.multinom.prob<- predict(mult.regression, newdata = testData, "probs")

summary(predict.train.multinom.prob)

table(testData$Dalc,predict.test.multinom)

mean(as.character(predict.test.multinom) != as.character(testData$Dalc)) #misclassification error 27.3% low

ggplot(testData, aes(x=testData$Dalc, y=predict.test.multinom)) + geom\_point(aes(colour=Dalc))+geom\_jitter(alpha = 0.2)

The confusion matrix looks like this table predict.test.multinom

1 2 3 4 5

1 129 9 4 0 0

2 20 10 5 0 2

3 1 6 4 1 0

4 1 0 1 2 1

5 0 2 2 0 1

misclassification error 27.3% ,low.

#CVM regression

library(caret)

library(e1071)

trainModels=list()

**#forming set of 60 different values of cost and gamma**

**#and applying to SVM to finding the best model**

grid=expand.grid(cost=seq(1,901,100),gamma=seq(1,200,30))

for(i in 1:nrow(grid)){

trainModels[[i]]=svm(as.factor(Dalc) ~ sex+ age+famsize+Pstatus+ Medu+Fedu +

studytime +failures+ schoolsup+ activities+ higher +romantic

+famrel+freetime+goout, data = trainData,type= "C", kernel="radial",

cost=grid$cost[i], gamma = grid$gamma[i] ,probability=TRUE) }

summary(trainModels[5])#Will take 40 sec ,the best cost: 901 ,gamma: 181

train\_svmBest<-svm(as.factor(Dalc) ~ sex+ age+famsize+Pstatus+ Medu+Fedu +

studytime +failures+ schoolsup+ activities+ higher +romantic

+famrel+freetime+goout, data = trainData,type= "C", kernel="radial", cost=901,

gamma = 181,probability=TRUE)

**#predicting the test data**

svmmodel.predict<-predict(train\_svmBest,subset(testData,decision.values=TRUE))

svmmodel.class<-predict(train\_svmBest,testData,type="class")

svmmodel.labels<-testData$Dalc

**#analyzing result**

library(SDMTools)

svmmodel.confusion<-confusionMatrix(svmmodel.labels,svmmodel.class)

svmmodel.confusion **#Accuracy : 0.8408**

Confusion Matrix and Statistics

Reference

Prediction 1 2 3 4 5

1 142 0 0 0 0

2 22 15 0 0 0

3 5 0 7 0 0

4 1 0 0 4 0

5 4 0 0 0 1

Overall Statistics

Accuracy : 0.8408

95% CI : (0.7827, 0.8885)

No Information Rate : 0.8657

P-Value [Acc > NIR] : 0.8712

Kappa : 0.572

Mcnemar's Test P-Value : NA

Statistics by Class:

Class: 1 Class: 2 Class: 3 Class: 4 Class: 5

Sensitivity 0.8161 1.00000 1.00000 1.00000 1.000000

Specificity 1.0000 0.88172 0.97423 0.99492 0.980000

Pos Pred Value 1.0000 0.40541 0.58333 0.80000 0.200000

Neg Pred Value 0.4576 1.00000 1.00000 1.00000 1.000000

Prevalence 0.8657 0.07463 0.03483 0.01990 0.004975

Detection Rate 0.7065 0.07463 0.03483 0.01990 0.004975

Detection Prevalence 0.7065 0.18408 0.05970 0.02488 0.024876

Balanced Accuracy 0.9080 0.94086 0.98711 0.99746 0.990000

**#SVM with cross validation in R using caret**

ctrl <- trainControl(method = "repeatedcv", repeats = 10)**#setting up control**

set.seed(1500)

mod <- train(Dalc ~ sex+ age+famsize+Pstatus+ Medu+Fedu +

+ studytime +failures+ schoolsup+ activities+ higher +romantic

+ famrel+freetime+goout, data=trainData, method = "svmLinear", trControl = ctrl)

RMSE Rsquared

0.9186448 0.09766363

Tuning parameter 'C' was held constant at a value of 1

The last approach is to work with the CART model:

**#CART**

library(rpart)

library(rpart.plot)

library(e1071)

library(caret)

treeDalc1<-rpart(Dalc~.,data=trainData,method="poisson")

treeDalc2<-rpart(Dalc~.,data=trainData,method="class")

treeDalc3<-rpart(Dalc~.,data=trainData,method="anova")

prp(treeDalc1)

prp(treeDalc2)

prp(treeDalc3)

**#tunung CART model**

train.contr=trainControl(method="cv",number=20)

grid=expand.grid(.cp=(0:10)\*0.001)

training=train(Dalc~sex+Medu+Mjob+reason+traveltime+paid+higher+freetime,

data=trainData,method="rpart",

trControl=train.contr,tuneGrid=grid)

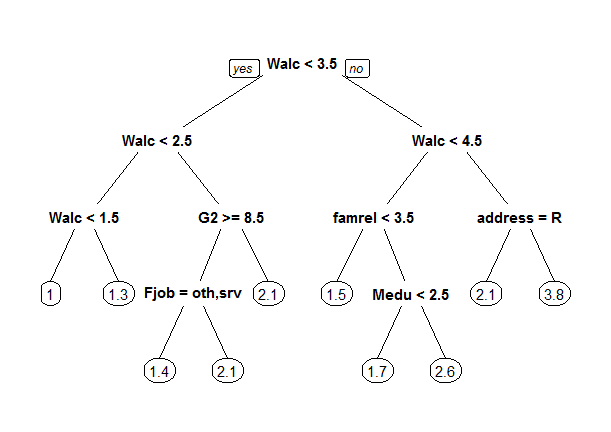
best=training$finalModel

prp(best)

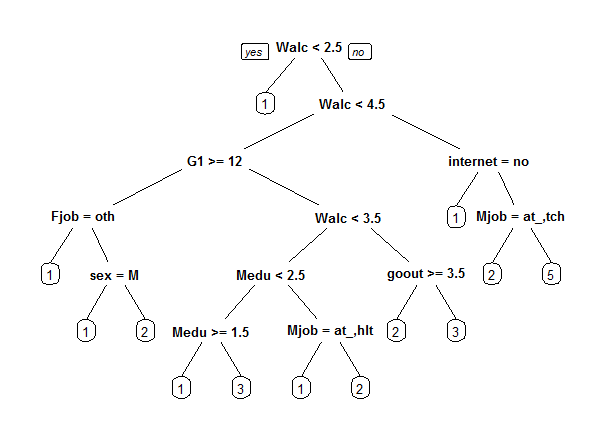
best.prediction= predict(best, data =testData )

sum(best.prediction - trainData$Dalc)^2

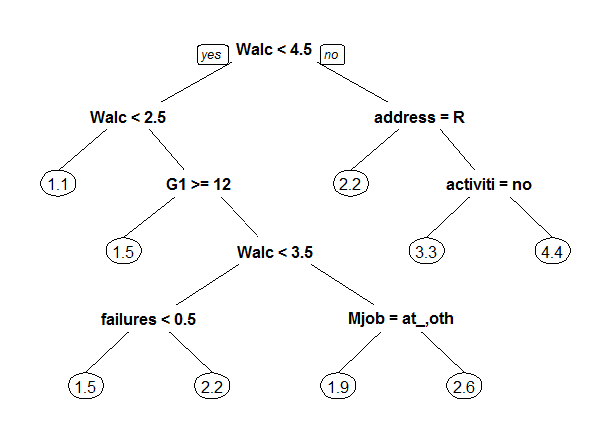
method=" poisson "



method="class"



method="anova"



Tuned CART model

