The effect of Alcohol consumption on Student Grades

Olufemi George

28 May 2017

The Project is to determine the correlation (if any) between alcohol consumption by students and their grades in 2 Portuguese Secondary Schools. We will also be looking at the importance/relevance of other variables in determining passing grades in this dataset based on the best performing model.

This data shows Secondary School student achievement for two Portuguese schools.The data attributes include student grades, demographic, social and school related features) and was collected using school reports, surveys and questionnaires.

The goal was to determine the correlation (if any) between alcohol consumption by students and their grades in school.We will also be looking at the importance/relevance of other variables in determining passing grades in this dataset based on the best performing model.

Merge both datasets into one

```
d1=read.table("student-mat.csv",sep=",",header=TRUE)
d2=read.table("student-por.csv",sep=",",header=TRUE)
d4=merge(d1,d2,by=c("school","sex","age","address","famsize","Pstatus",
                    "Medu","Fedu","Mjob","Fjob","reason",
                    "guardian","traveltime","studytime","failures",
                    "schoolsup","famsup","activities","nursery","higher","internet","romantic",
                    "famrel","freetime","goout","Dalc","Walc","health","absences"))
d4$meanMath <- rowMeans(subset(d4, select = c(G1.x, G2.x,G3.x)), na.rm = TRUE)
d4$meanPort <- rowMeans(subset(d4, select = c(G1.y, G2.y,G3.y)), na.rm = TRUE)
d3<-rbind(d1,d2)
df.merged<-d3 %>% distinct(school,sex,age,address,famsize,Pstatus,
                           Medu,Fedu,Mjob,Fjob,reason,
                           guardian,traveltime,studytime,failures,
                           schoolsup, famsup,activities,nursery,higher,internet,
                           romantic,famrel,freetime,goout,Dalc,Walc,health,absences, .keep_all = TRUE)
```

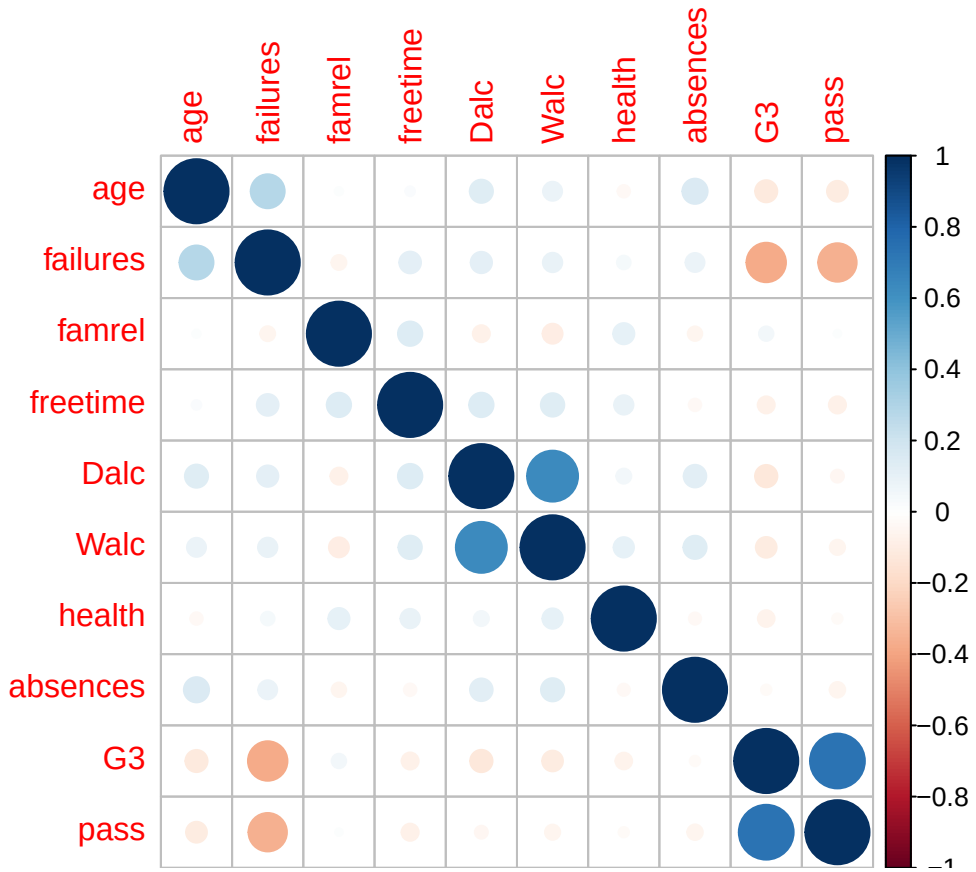
```
df.merged<-df.merged[,-31:-32]
#names(df.merged)
df.merged$pass<- ifelse(df.merged$G3>=9,1,0)
```

```
df.merged$activities<-as.character(df.merged$activities)
df.merged$romantic<-as.character(df.merged$romantic)
df.merged$internet<-as.character(df.merged$internet)
df.merged$higher<-as.character(df.merged$higher)
df.merged$nursery<-as.character(df.merged$nursery)
df.merged$famsup<-as.character(df.merged$famsup)
df.merged$schoolsup<-as.character(df.merged$schoolsup)
df.merged$activities<-ifelse(df.merged$activities=="no","N","Y")
df.merged$romantic<-ifelse(df.merged$romantic=="no","N","Y")
df.merged$internet<-ifelse(df.merged$internet=="no","N","Y")
df.merged$higher<-ifelse(df.merged$higher=="no","N","Y")
df.merged$nursery<-ifelse(df.merged$nursery=="no","N","Y")
df.merged$paid<-ifelse(df.merged$paid=="no","N","Y")
df.merged$famsup<-ifelse(df.merged$famsup=="no","N","Y")
df.merged$schoolsup<-ifelse(df.merged$schoolsup=="no","N","Y")
```

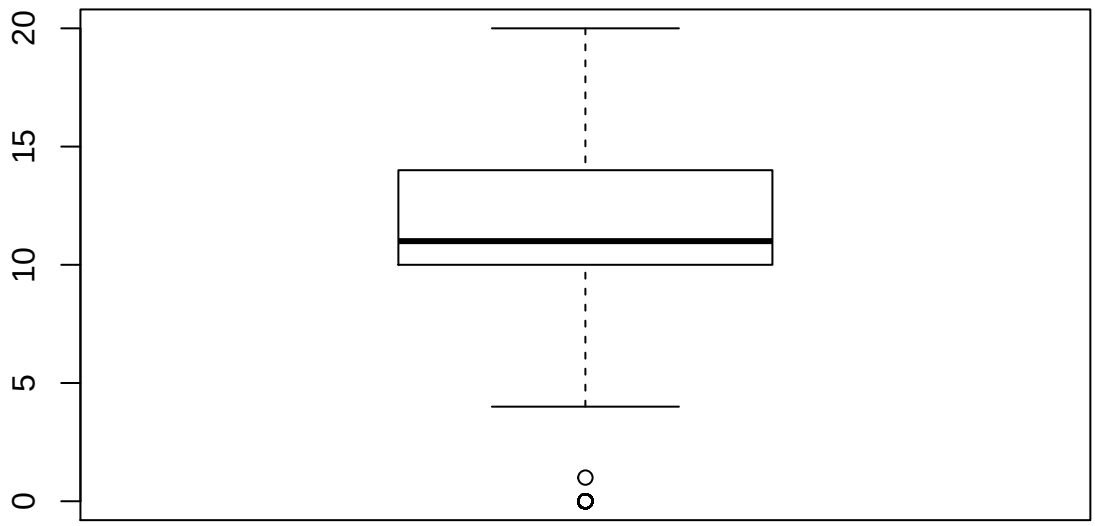
```
df.merged$activities<-as.factor(df.merged$activities)
df.merged$romantic<-as.factor(df.merged$romantic)
df.merged$internet<-as.factor(df.merged$internet)
df.merged$higher<-as.factor(df.merged$higher)
df.merged$nursery<-as.factor(df.merged$nursery)
df.merged$famsup<-as.factor(df.merged$famsup)
df.merged$schoolsup<-as.factor(df.merged$schoolsup)
df.merged$paid<-as.factor(df.merged$paid)
##
df.merged$reason<-as.character(df.merged$reason)
df.merged$reason[df.merged$reason == "home"] <- "athome"
df.merged$reason<-as.factor(df.merged$reason)
df.merged$reason<-as.character(df.merged$reason)
df.merged$reason[df.merged$reason == "home"] <- "athome"
df.merged$Mjob<-as.factor(df.merged$Mjob)
df.merged$Mjob<-as.character(df.merged$Mjob)
df.merged$Mjob[df.merged$Mjob == "at_home"] <- "stayhome"
df.merged$Mjob<-as.factor(df.merged$Mjob)
df.merged$Fjob<-as.character(df.merged$Fjob)
df.merged$Fjob[df.merged$Fjob == "at_home"] <- "stayhome"
df.merged$Fjob<-as.factor(df.merged$Fjob)
## Medu
df.merged$Medu[df.merged$Medu == "0"] <- "No-Grade"
df.merged$Medu[df.merged$Medu == "1"] <- "forththPass"
df.merged$Medu[df.merged$Medu == "2"] <- "fifth-9th-Grade"
df.merged$Medu[df.merged$Medu == "3"] <- "Secondary-Education"
df.merged$Medu[df.merged$Medu == "4"] <- "Higher-Education"
df.merged$Medu<-as.factor(df.merged$Medu)
#goout
df.merged$goout[df.merged$goout == "1"] <- "xx1"
df.merged$goout[df.merged$goout == "2"] <- "xx2"
df.merged$goout[df.merged$goout == "3"] <- "xx3"
df.merged$goout[df.merged$goout == "4"] <- "xx4"
df.merged$goout[df.merged$goout == "5"] <- "xx5"
df.merged$goout<-as.factor(df.merged$goout)
# Fedu
df.merged$Fedu[df.merged$Fedu == "0"] <- "No-Grade"
df.merged$Fedu[df.merged$Fedu == "1"] <- "forththPass"
df.merged$Fedu[df.merged$Fedu == "2"] <- "fifth-9th-Grade"
df.merged$Fedu[df.merged$Fedu == "3"] <- "Secondary-Education"
df.merged$Fedu[df.merged$Fedu == "4"] <- "Higher-Education"
df.merged$Fedu<-as.factor(df.merged$Fedu)
#recode traveltime
df.merged$traveltime[df.merged$traveltime == "1"] <- "under15mins"
df.merged$traveltime[df.merged$traveltime == "2"] <- "fifteen-30mins"
df.merged$traveltime[df.merged$traveltime == "3"] <- "thirtymin-1hour"
df.merged$traveltime[df.merged$traveltime == "4"] <- "over1hour"
df.merged$traveltime<-as.factor(df.merged$traveltime)
#recode studytime
df.merged$studytime[df.merged$studytime == "1"] <- "under2hours"
df.merged$studytime[df.merged$studytime == "2"] <- "two-5hours"
df.merged$studytime[df.merged$studytime == "3"] <- "thirtymin-1hour"
df.merged$studytime[df.merged$studytime == "4"] <- "five-10hours"
```

```
df.merged$studytime<-as.factor(df.merged$studytime)
```

```
# check correlations
correlations <- cor(df.merged[,c(3,15,24,25,27,28,29,30,31,32)])
corrplot(correlations, method="circle")
```



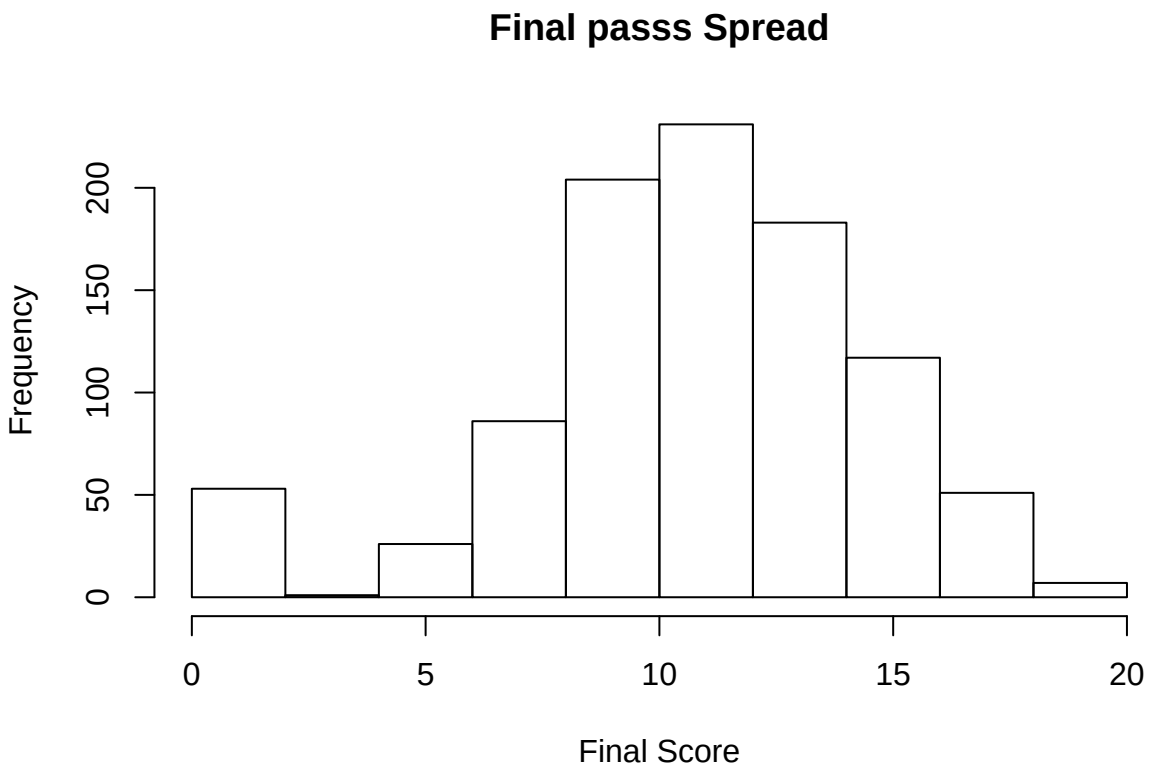

```
boxplot(df.merged$G3, main='Final Score Central Tendency')
```



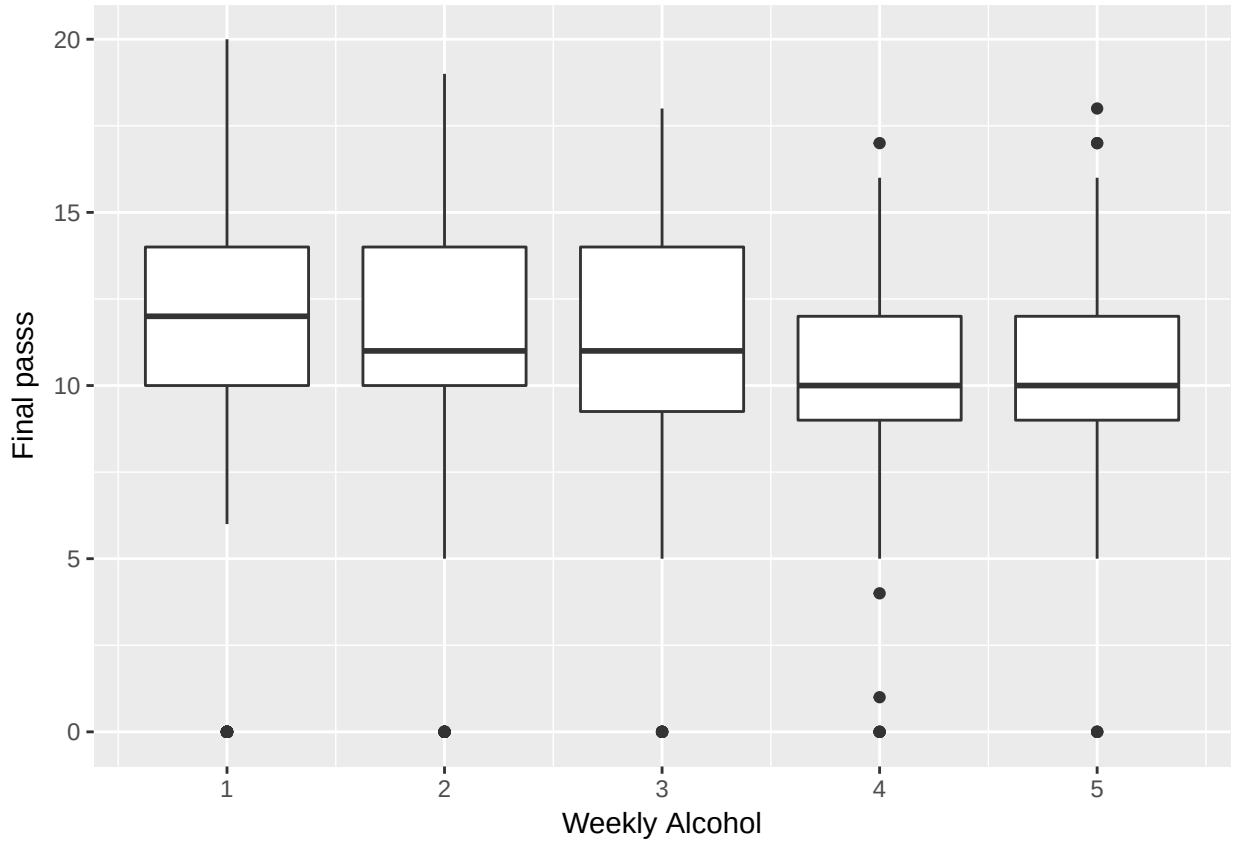

```
prop.table(table(df.merged$pass))
```

```
##
##        0        1
## 0.173097 0.826903
```

```
hist(df.merged$G3, main="Final passs Spread", xlab="Final Score")
```



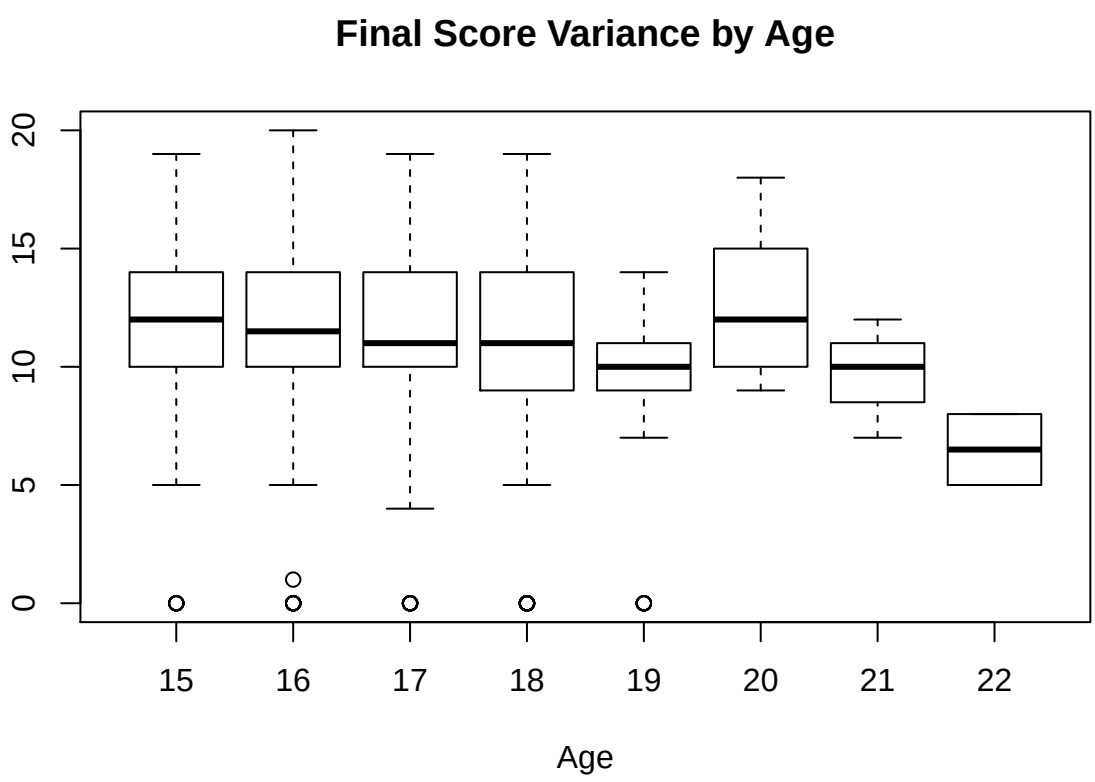

```
ggplot(df.merged, aes(x=Walc,y=G3, group=Walc)) +
  geom_boxplot() +
  xlab("Weekly Alcohol") +
  ylab("Final passs")
```



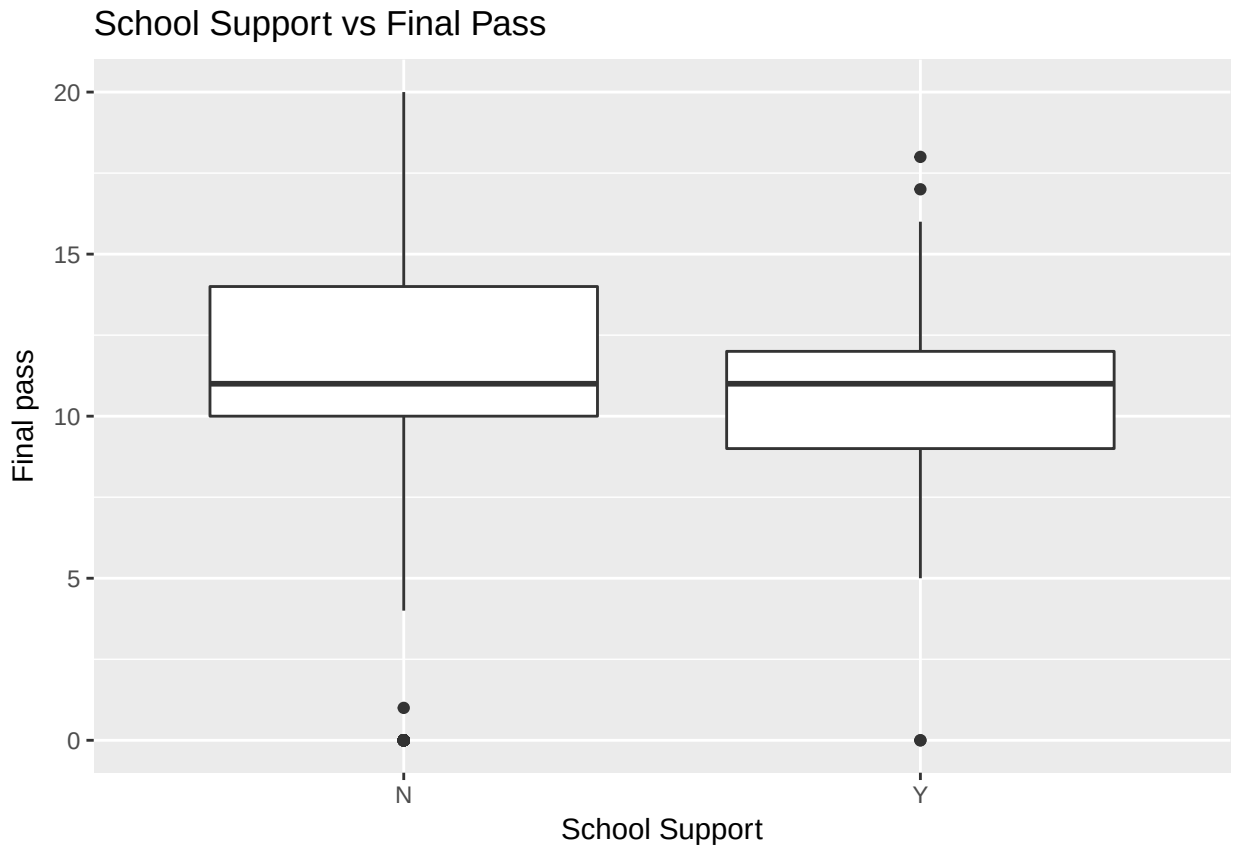

```
ggtitle("Weekly Alcohol Consumption vs Final Pass")
```

```
## $title
## [1] "Weekly Alcohol Consumption vs Final Pass"
##
## $subtitle
## NULL
##
## attr(,"class")
## [1] "labels"
```

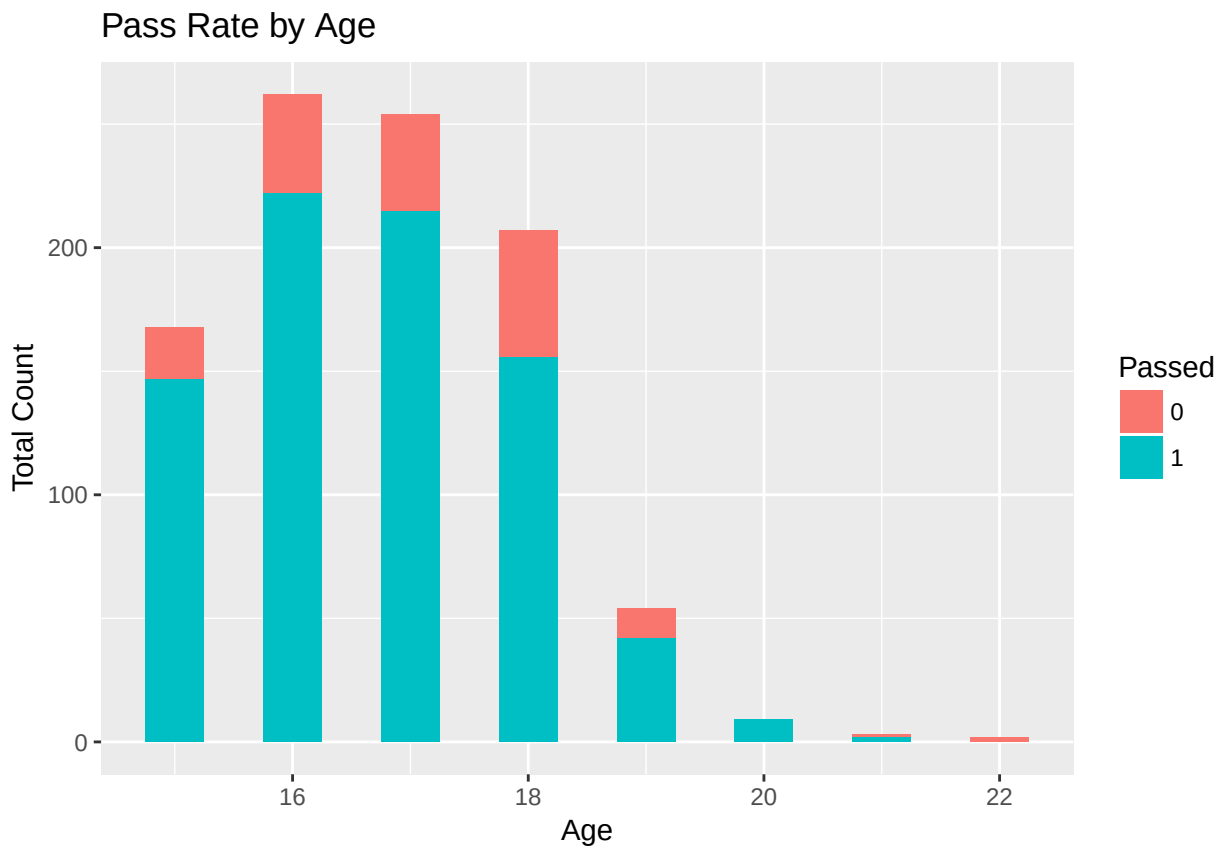
```
boxplot(df.merged$G3~df.merged$age, main='Final Score Variance by Age', xlab="Age")
```



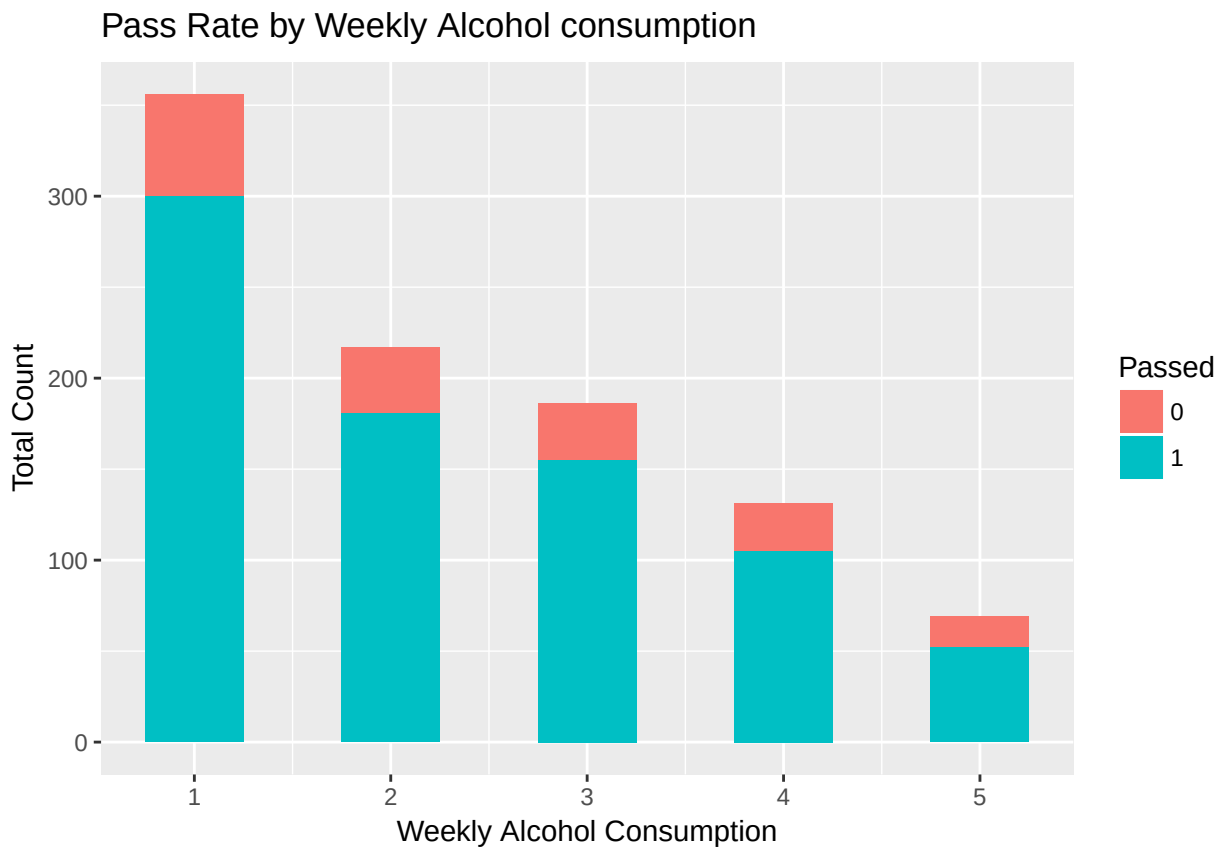

```
ggplot(df.merged, aes(x=schoolsup, y=G3, group=schoolsup)) +
  geom_boxplot() +
  xlab("School Support") +
  ylab("Final pass") +
  ggtitle("School Support vs Final Pass")
```



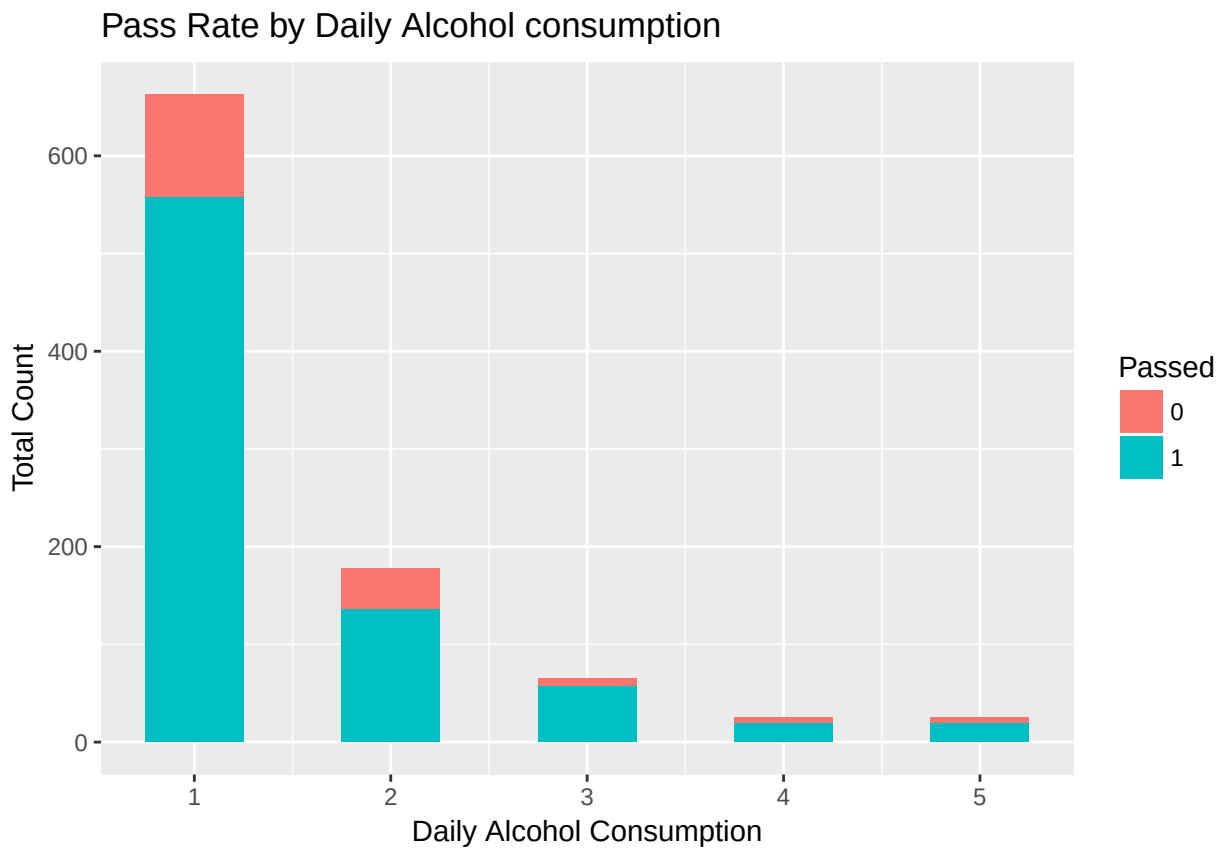

```
ggplot(df.merged, aes(x=age, fill=factor(pass))) +
    geom_bar(width=0.5)+
    xlab("Age") +
    ylab("Total Count") +
    labs(fill='Passed') +
    ggtitle("Pass Rate by Age")
```



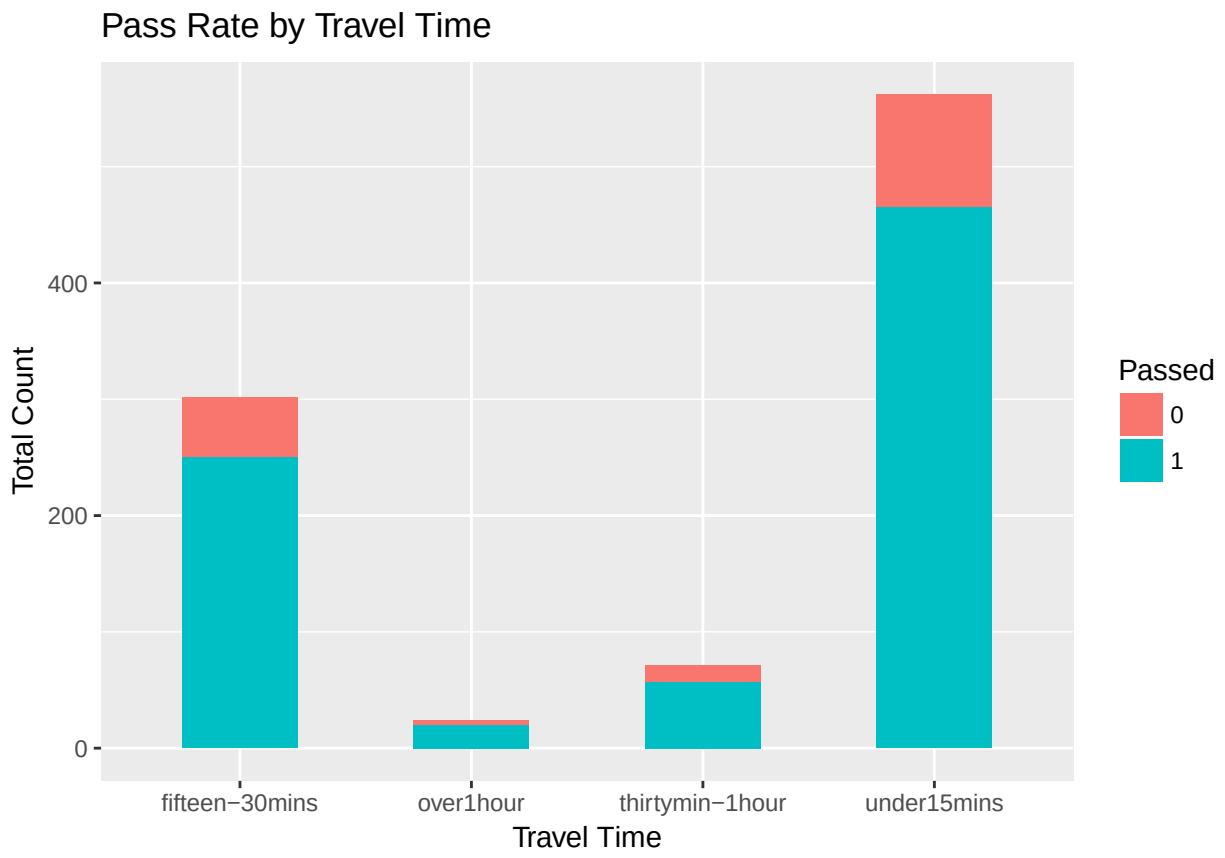

```
ggplot(df.merged, aes(x=Walc, fill=factor(pass))) +
    geom_bar(width=0.5)+
    xlab("Weekly Alcohol Consumption") +
    ylab("Total Count") +
    labs(fill='Passed') +
    ggtitle("Pass Rate by Weekly Alcohol consumption")
```



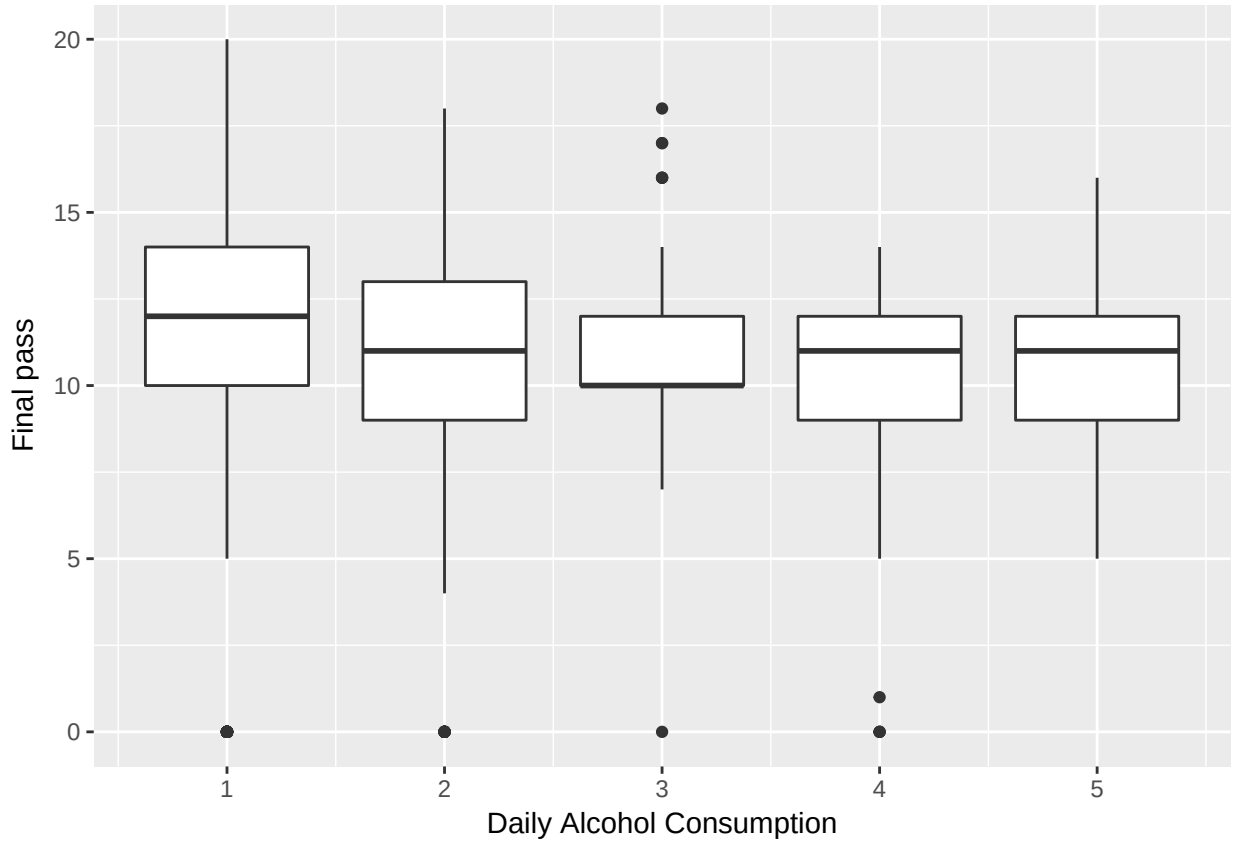

```
ggplot(df.merged, aes(x=Dalc, fill=factor(pass))) +
    geom_bar(width=0.5)+
    xlab("Daily Alcohol Consumption") +
    ylab("Total Count") +
    labs(fill='Passed') +
    ggtitle("Pass Rate by Daily Alcohol consumption")
```




```
ggplot(df.merged, aes(x=traveltime, fill=factor(pass))) +
   geom_bar(width=0.5)+
   xlab("Travel Time") +
   ylab("Total Count") +
   labs(fill='Passed') +
   ggtitle("Pass Rate by Travel Time")
```



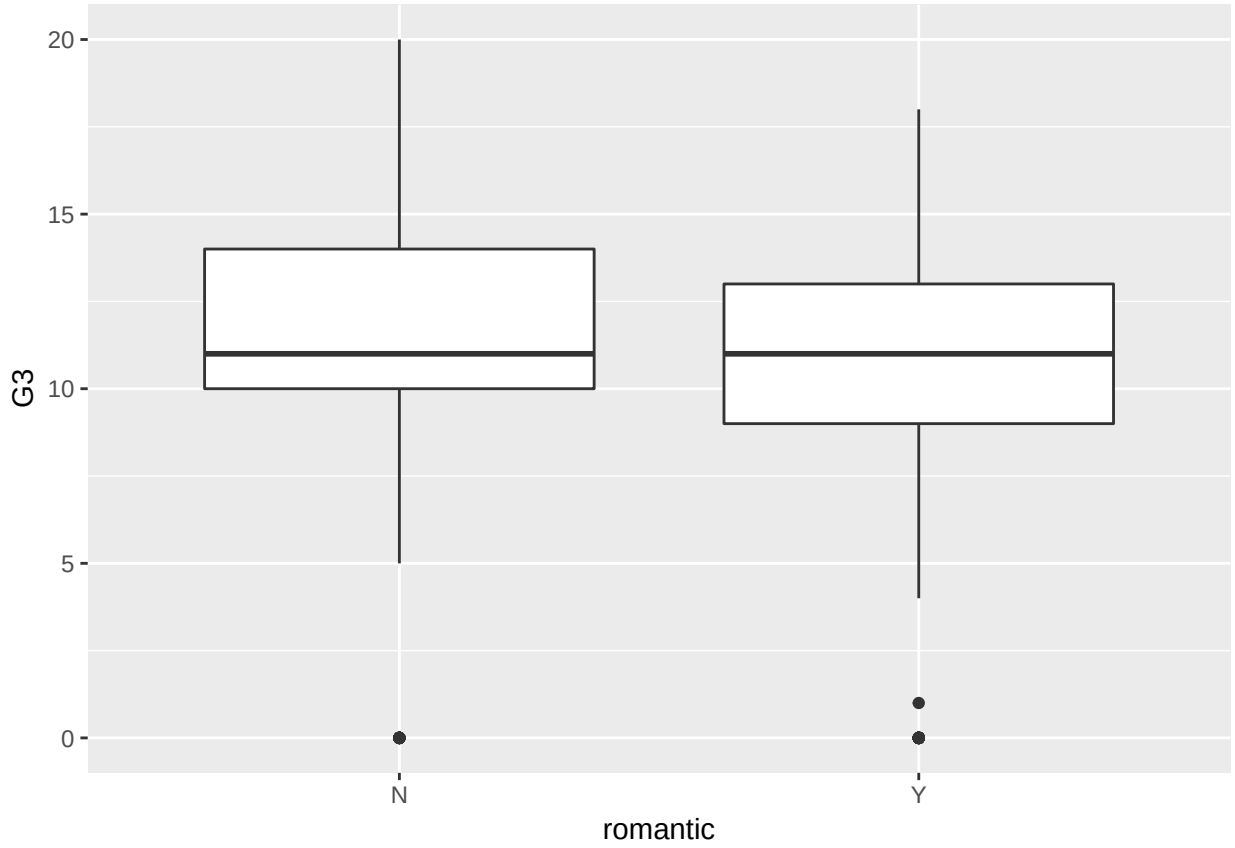

```
ggplot(df.merged, aes(x=Dalc, y=G3, group=Dalc)) +
  geom_boxplot()+
  xlab("Daily Alcohol Consumption") +
  ylab("Final pass")
```



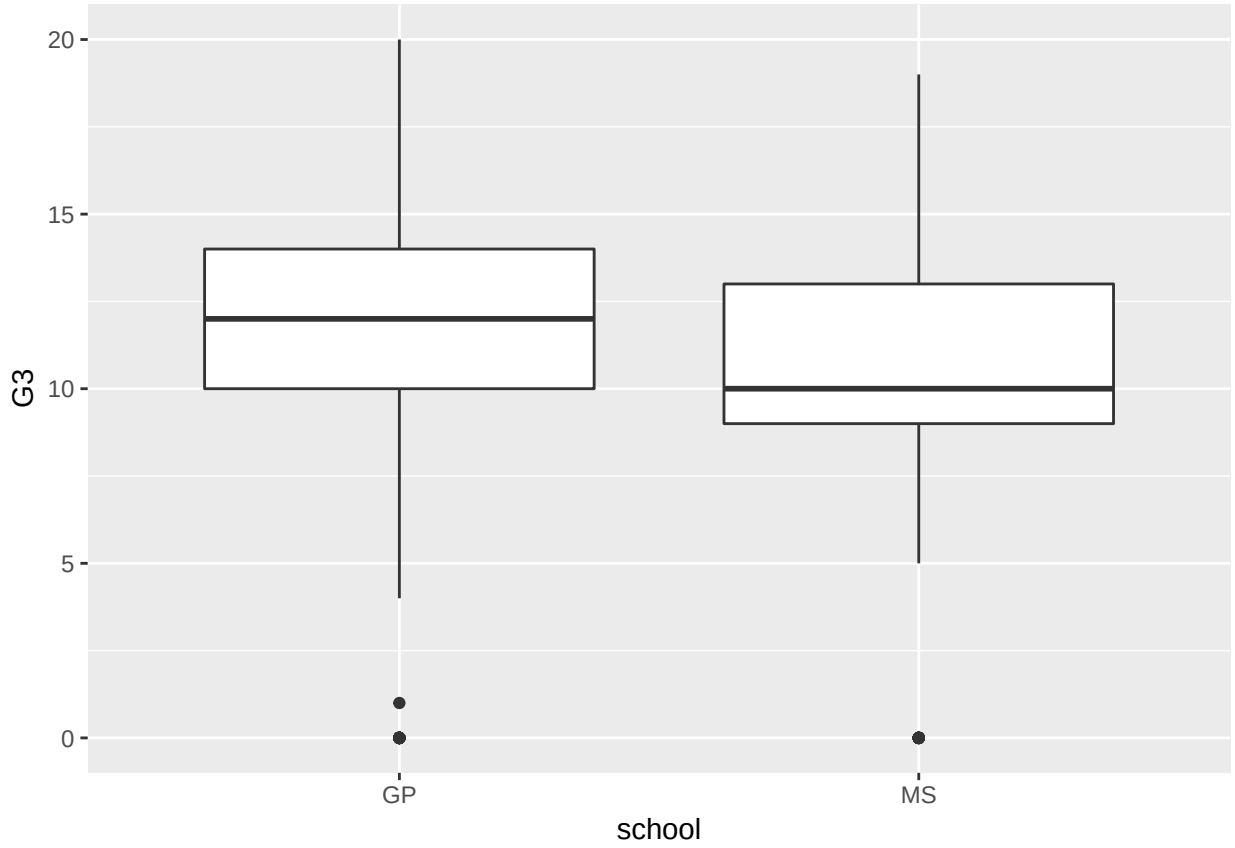

```
ggtitle("Daily Alcohol Consumption vs Final pass")
```

```
## $title
## [1] "Daily Alcohol Consumption vs Final pass"
##
## $subtitle
## NULL
##
## attr(,"class")
## [1] "labels"
```

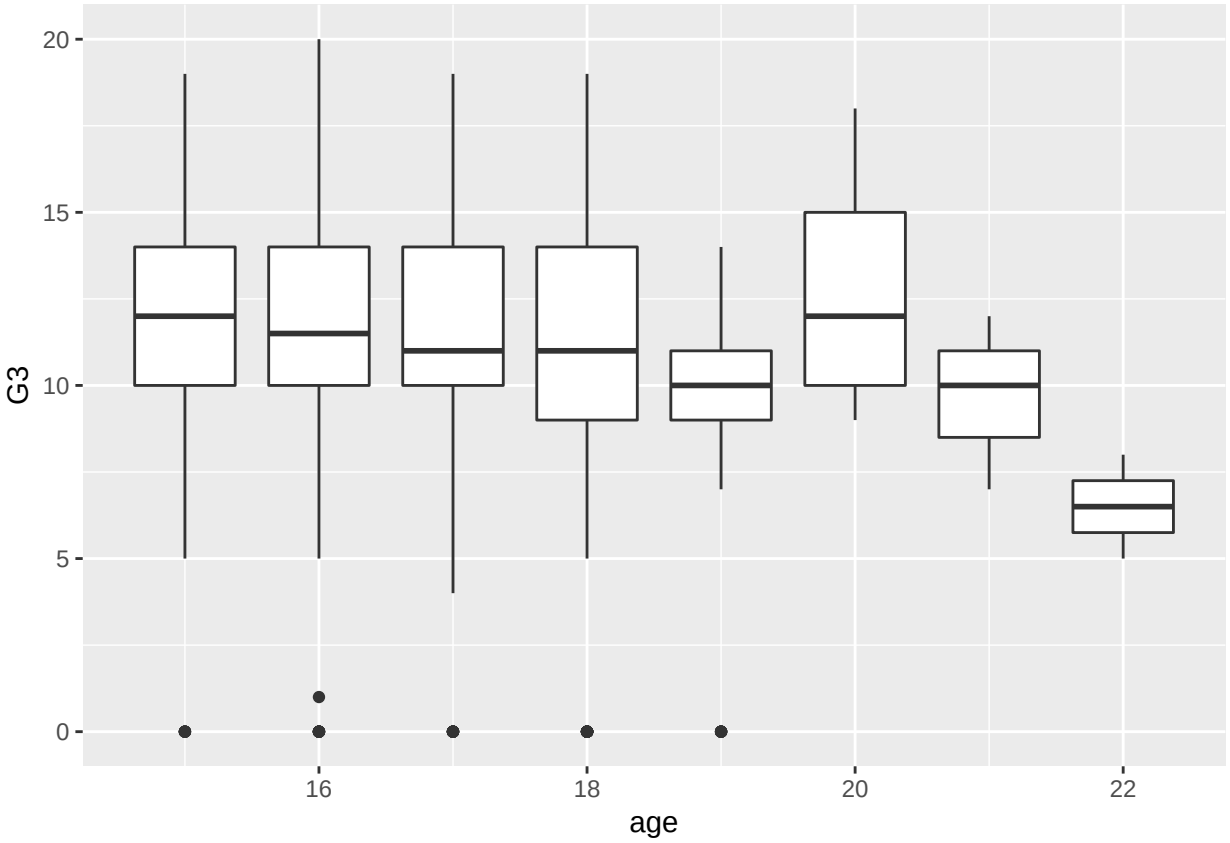
```
ggplot(df.merged, aes(x=romantic, y=G3, group=romantic)) +
  geom_boxplot()
```



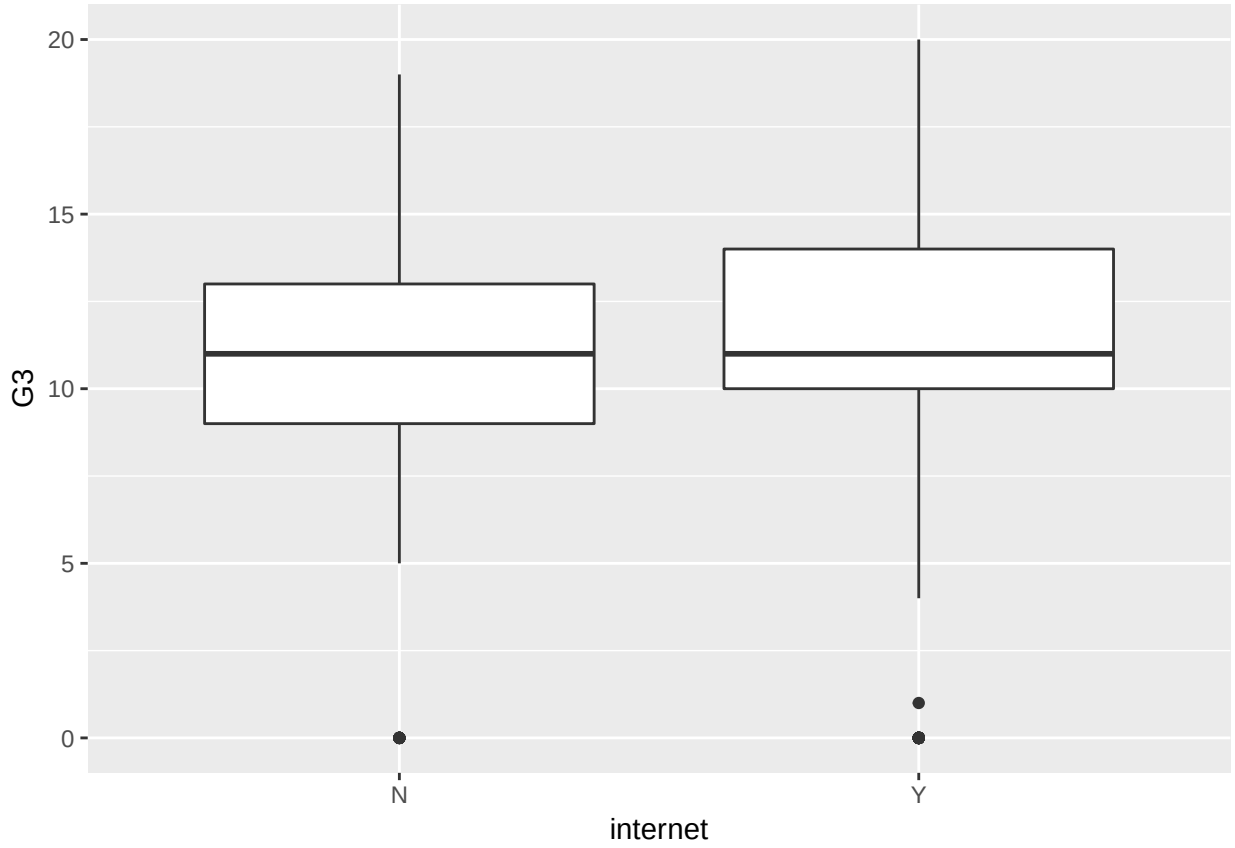

```
ggplot(df.merged, aes(x=school, y=G3, group=school)) +
  geom_boxplot()
```



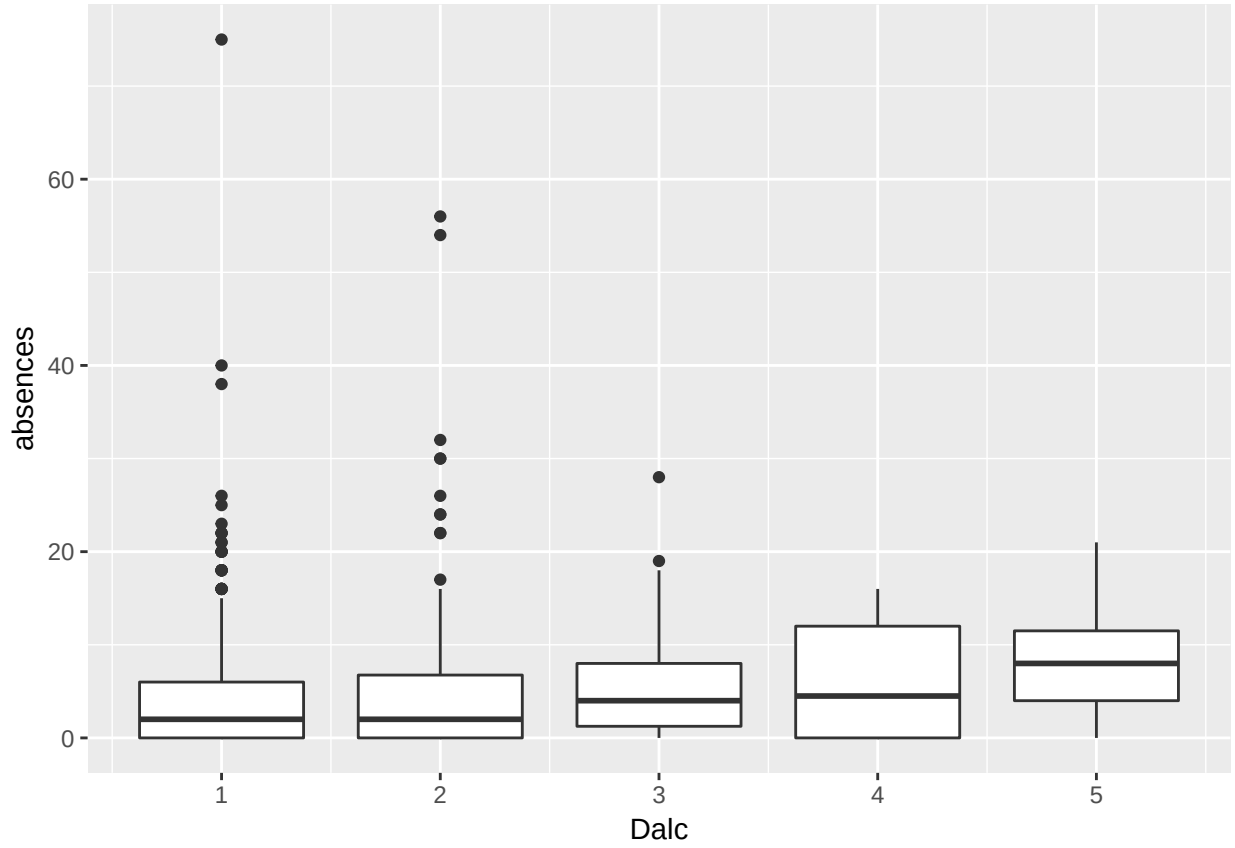

```
ggplot(df.merged, aes(x=age, y=G3, group=age)) +
  geom_boxplot()
```



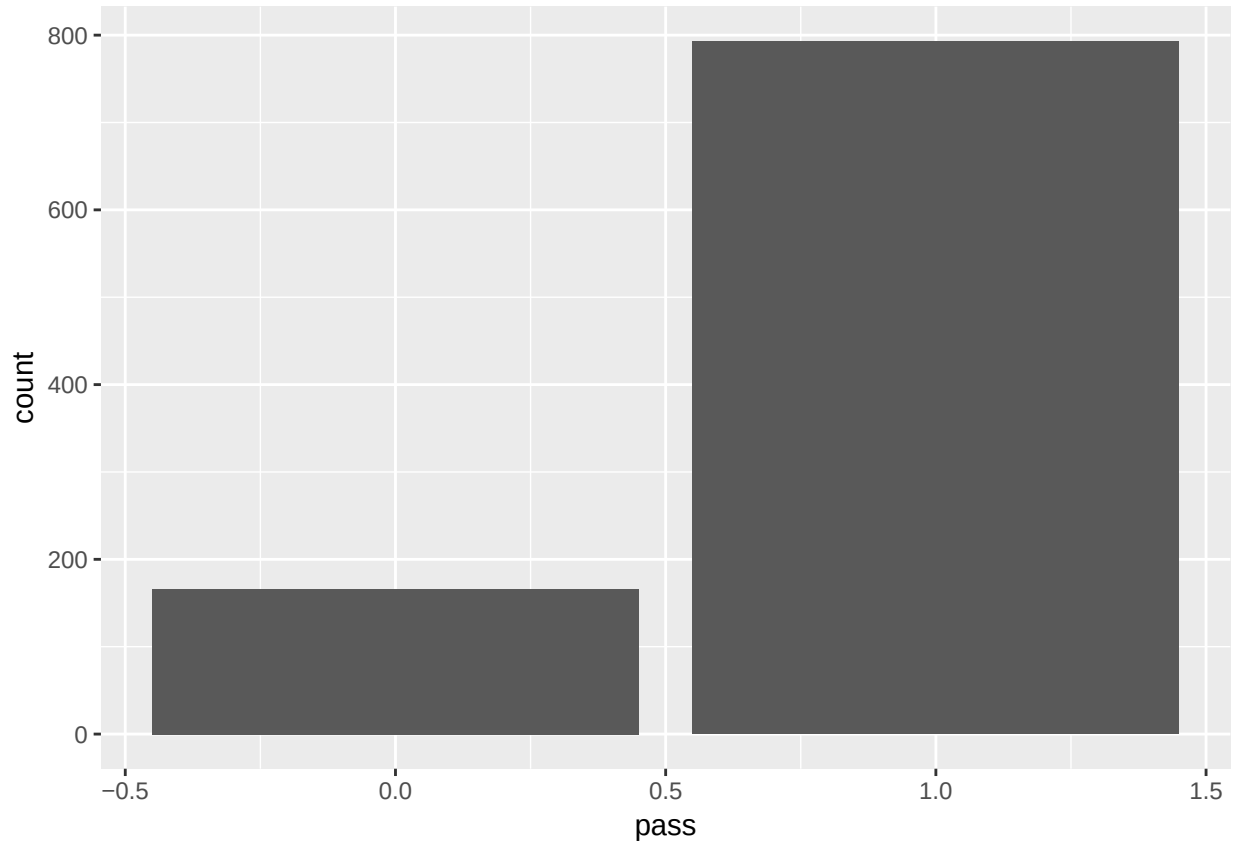

```
ggplot(df.merged, aes(x=internet, y=G3, group=internet)) +
  geom_boxplot()
```




```
ggplot(df.merged, aes(x=Dalc, y=absences, group=Dalc)) +
  geom_boxplot()
```




```
ggplot(df.merged, aes(x=pass)) +
  geom_bar()
```




```
df.merged$pass <- as.integer(df.merged$pass)
df.Dummy <- dummyVars("~.",data=df.merged,fullRank=T)
df.schools <- as.data.frame(predict(df.Dummy,df.merged))
prop.table(table(df.schools$pass))
```

```
##
##        0        1
## 0.173097 0.826903
```

```
cor.prob <- function (X, dfr = nrow(X) - 2) {
  R <- cor(X, use="pairwise.complete.obs")
  above <- row(R) < col(R)
  r2 <- R[above]^2
  Fstat <- r2 * dfr/(1 - r2)
  R[above] <- 1 - pf(Fstat, 1, dfr)
  R[row(R) == col(R)] <- NA
  R
}

flattenSquareMatrix <- function(m) {
  if( (class(m) != "matrix") | (nrow(m) != ncol(m))) stop("Must be a square matrix.")
  if(!identical(rownames(m), colnames(m))) stop("Row and column names must be equal.")
  ut <- upper.tri(m)
  data.frame(i = rownames(m)[row(m)[ut]],
             j = rownames(m)[col(m)[ut]],
             cor=t(m)[ut],
             p=m[ut])
```

```
}


corMasterList <- flattenSquareMatrix (cor.prob(df.schools))
print(head(corMasterList,20))
```

```
##              i                j         cor            p
## 1    school.MS            sex.M -0.08196236 1.111220e-02
## 2    school.MS              age  0.14061346 1.239447e-05
## 3        sex.M              age -0.03557143 2.711239e-01
## 4    school.MS        address.U -0.34156438 0.000000e+00
## 5        sex.M        address.U  0.01318405 6.834461e-01
## 6          age        address.U -0.05687328 7.834610e-02
## 7    school.MS      famsize.LE3  0.03369830 2.971824e-01
## 8        sex.M      famsize.LE3  0.09731404 2.554358e-03
## 9          age      famsize.LE3  0.01070619 7.405512e-01
## 10   address.U      famsize.LE3  0.04491855 1.645554e-01
## 11   school.MS        Pstatus.T  0.02107108 5.145659e-01
## 12       sex.M        Pstatus.T  0.05645908 8.054664e-02
## 13         age        Pstatus.T -0.01249649 6.991277e-01
## 14   address.U        Pstatus.T -0.07078291 2.838843e-02
## 15 famsize.LE3        Pstatus.T -0.22259750 3.124612e-12
## 16   school.MS Medu.forththPass  0.25593100 8.881784e-16
## 17       sex.M Medu.forththPass -0.04650884 1.501014e-01
## 18         age Medu.forththPass  0.08081408 1.229856e-02
## 19   address.U Medu.forththPass -0.15101590 2.633848e-06
## 20 famsize.LE3 Medu.forththPass  0.05816199 7.181020e-02
```

```
corList <- corMasterList[order(-abs(corMasterList$cor)),]
print(head(corList,10))
```

```
##                          i                    j        cor p
## 1431                    G3                 pass  0.7310482 0
## 190            Fjob.other         Fjob.services -0.7115110 0
## 528   studytime.two-5hours studytime.under2hours -0.6444438 0
## 1225                  Dalc                 Walc  0.6307247 0
## 144  Medu.Higher-Education          Mjob.teacher  0.5468440 0
## 63   Medu.Higher-Education Fedu.Higher-Education  0.5201226 0
## 299          reasoncourse      reasonreputation -0.4676946 0
## 222  Fedu.Higher-Education          Fjob.teacher  0.4477619 0
## 52        Medu.forththPass      Fedu.forththPass  0.4321979 0
## 120            Mjob.other         Mjob.services -0.4305323 0
```

```
selectedSub <- subset(corList, (abs(cor) > 0.10 & j == 'pass'))
#print(selectedSub)
```

```
#remove G3 variable
df.schools$G3<- NULL
#Sort out Outcome variable
outcomeName <- 'pass'
predictorsNames <- names(df.schools)[names(df.schools) != outcomeName]
df.schools$pass <- as.factor(ifelse(df.schools$pass==1,'P','F'))
#split data into test and training
```

```r
# Train the data
set.seed(1234)
splitIndex <- createDataPartition(df.schools[,outcomeName], p = .75, list = FALSE, times = 1)
trainDF <- df.schools[ splitIndex,]
testDF  <- df.schools[-splitIndex,]
```

```r
trainControl <- trainControl(method="repeatedcv", number=10, repeats=3, summaryFunction=twoClassSummary
metric <- "ROC"
```

```r
#RF
#set.seed(7)
fit.rf <- train(pass~., data=trainDF, method="rf", metric=metric, preProc=c("center", "scale"), trContro
```

```
## Loading required package: randomForest
```

```
## randomForest 4.6-12
```

```
## Type rfNews() to see new features/changes/bug fixes.
```

```
##
## Attaching package: 'randomForest'
```

```
## The following object is masked from 'package:dplyr':
##
##     combine
```

```
## The following object is masked from 'package:ggplot2':
##
##     margin
```

```r
#GLM
#set.seed(7)
fit.glm <- train(pass~., data=trainDF, method="glm", metric=metric, preProc=c("center", "scale"), trCon
# GLMNET
#set.seed(7)
fit.glmnet <- train(pass~., data=trainDF, method="glmnet", metric=metric, preProc=c("center", "scale"),
```

```
## Loading required package: glmnet
```

```
## Loading required package: Matrix
```

```
## Loading required package: foreach
```

```
## Loaded glmnet 2.0-5
```

```
##
## Attaching package: 'glmnet'
```

```
## The following object is masked from 'package:pROC':
##
##     auc
```

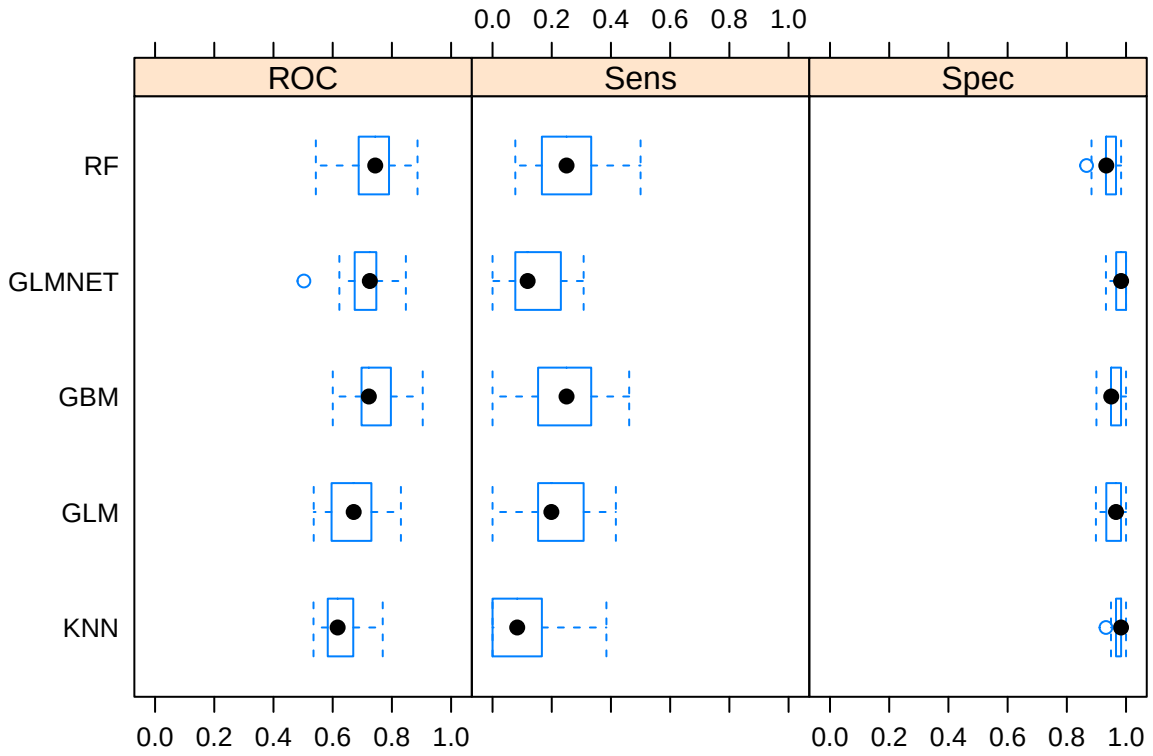
```r
# KNN
#set.seed(7)
fit.knn <- train(pass~., data=trainDF, method="knn", metric=metric, preProc=c("center", "scale"), trCon
# GBM
#set.seed(7)
fit.gbm <- train(pass~., data=trainDF, method="gbm", metric=metric, preProc=c("center", "scale"), trCon
```

```
## Loading required package: gbm
```

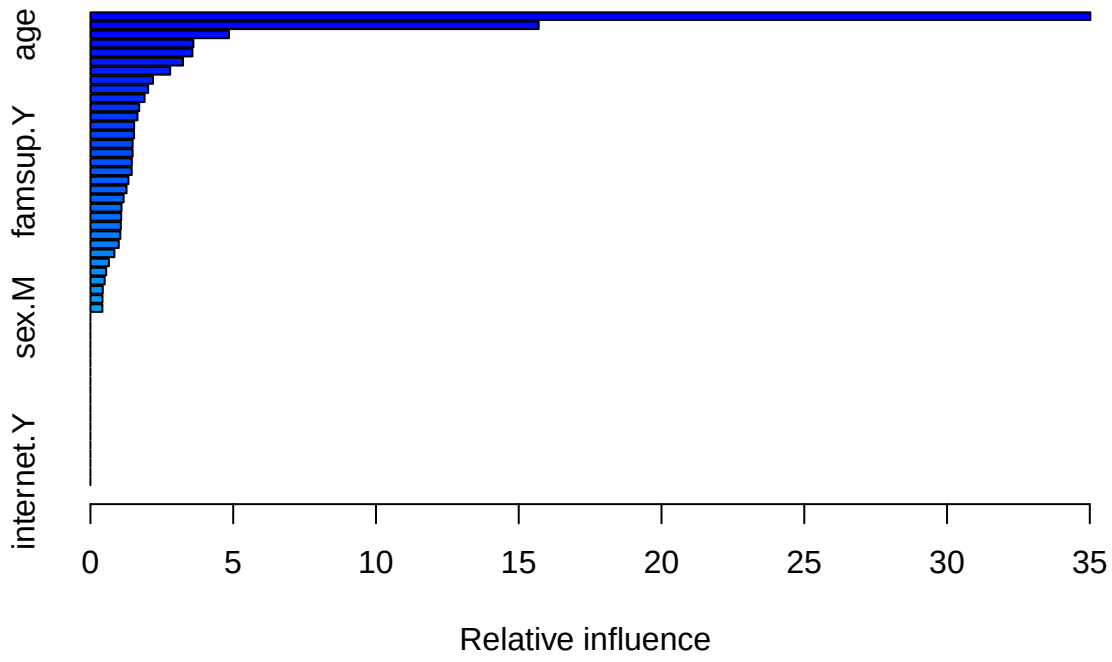
```
## Loading required package: survival
```

```
##
## Attaching package: 'survival'

## The following object is masked from 'package:caret':
##
##     cluster

## Loading required package: splines

## Loading required package: parallel

## Loaded gbm 2.1.3
```

```
#summarize results
set.seed(7)
results <- resamples(list(GLM=fit.glm, GBM=fit.gbm, RF=fit.rf, GLMNET=fit.glmnet, KNN=fit.knn))
#summary(results)
bwplot(results,layout = c(3,1))
```



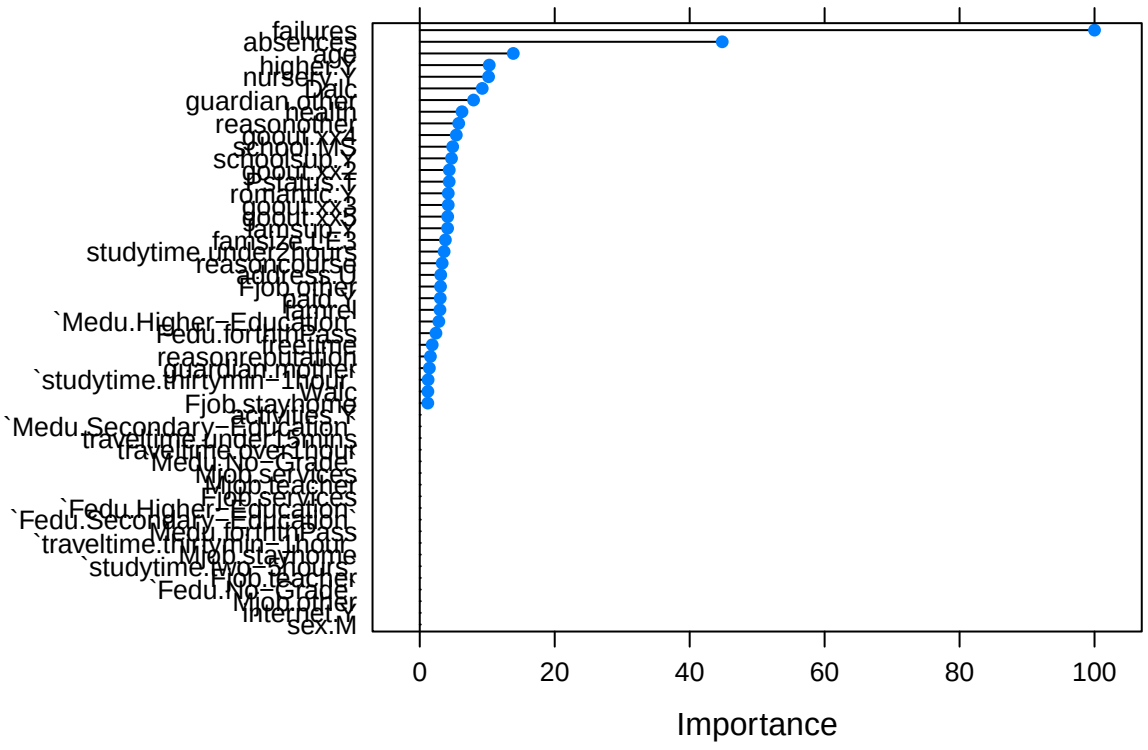

```
# view details of GBM Model
summary(fit.gbm)
```




```
##                                                  var    rel.inf
## failures                                    failures 35.0193076
## absences                                    absences 15.6986395
## age                                              age  4.8527627
## higher.Y                                    higher.Y  3.6059526
## nursery.Y                                  nursery.Y  3.5720584
## Dalc                                            Dalc  3.2429248
## guardian.other                        guardian.other  2.7933391
## health                                        health  2.1914183
## reasonother                              reasonother  2.0214988
## goout.xx4                                  goout.xx4  1.8971148
## school.MS                                  school.MS  1.7083856
## schoolsup.Y                              schoolsup.Y  1.6470635
## goout.xx2                                  goout.xx2  1.5307274
## Pstatus.T                                  Pstatus.T  1.5268054
## romantic.Y                                romantic.Y  1.4788798
## goout.xx3                                  goout.xx3  1.4772277
## goout.xx5                                  goout.xx5  1.4503881
## famsup.Y                                    famsup.Y  1.4443596
## famsize.LE3                              famsize.LE3  1.3304644
## studytime.under2hours          studytime.under2hours  1.2645152
## reasoncourse                            reasoncourse  1.1636369
## address.U                                  address.U  1.0886284
## Fjob.other                                Fjob.other  1.0779546
## paid.Y                                        paid.Y  1.0644430
## famrel                                        famrel  1.0483112
```

```
## `Medu.Higher-Education`           `Medu.Higher-Education`  0.9914274
## Fedu.forththPass                         Fedu.forththPass  0.8400525
## freetime                                         freetime  0.6458102
## reasonreputation                         reasonreputation  0.5532737
## guardian.mother                           guardian.mother  0.4996474
## `studytime.thirtymin-1hour`   `studytime.thirtymin-1hour`  0.4328122
## Walc                                                 Walc  0.4221567
## Fjob.stayhome                               Fjob.stayhome  0.4180123
## sex.M                                               sex.M  0.0000000
## Medu.forththPass                         Medu.forththPass  0.0000000
## `Medu.No-Grade`                           `Medu.No-Grade`  0.0000000
## `Medu.Secondary-Education`     `Medu.Secondary-Education`  0.0000000
## `Fedu.Higher-Education`           `Fedu.Higher-Education`  0.0000000
## `Fedu.No-Grade`                           `Fedu.No-Grade`  0.0000000
## `Fedu.Secondary-Education`     `Fedu.Secondary-Education`  0.0000000
## Mjob.other                                   Mjob.other  0.0000000
## Mjob.services                             Mjob.services  0.0000000
## Mjob.stayhome                             Mjob.stayhome  0.0000000
## Mjob.teacher                               Mjob.teacher  0.0000000
## Fjob.services                             Fjob.services  0.0000000
## Fjob.teacher                               Fjob.teacher  0.0000000
## traveltime.over1hour                 traveltime.over1hour  0.0000000
## `traveltime.thirtymin-1hour` `traveltime.thirtymin-1hour`  0.0000000
## traveltime.under15mins             traveltime.under15mins  0.0000000
## `studytime.two-5hours`             `studytime.two-5hours`  0.0000000
## activities.Y                                 activities.Y  0.0000000
## internet.Y                                     internet.Y  0.0000000
```

```
#Plot variable importance of GBM Model
plot(varImp(object=fit.gbm),main="GBM - Variable Importance")
```




```
predictions <- predict(object=fit.gbm, testDF[,predictorsNames], type='raw')
head(predictions)
```

```
## [1] P P P P P F
## Levels: F P
```

```
# Accuracy and Kappa
print(postResample(pred=predictions, obs=as.factor(testDF[,outcomeName])))
```

```
##  Accuracy     Kappa
## 0.8326360 0.1925676
```

```
## Probabilities
predictions <- predict(object=fit.gbm, testDF[,predictorsNames], type='prob')
head(predictions)
```

```
##            F         P
## 1 0.16858434 0.8314157
## 2 0.08680885 0.9131912
## 3 0.11194206 0.8880579
## 4 0.12577500 0.8742250
## 5 0.08038472 0.9196153
## 6 0.79209636 0.2079036
```

```
# AUC Score
auc <- roc(ifelse(testDF[,outcomeName]=="P",1,0), predictions[[2]])
print(auc$auc)
```

```
## Area under the curve: 0.6957
```