In [2]:

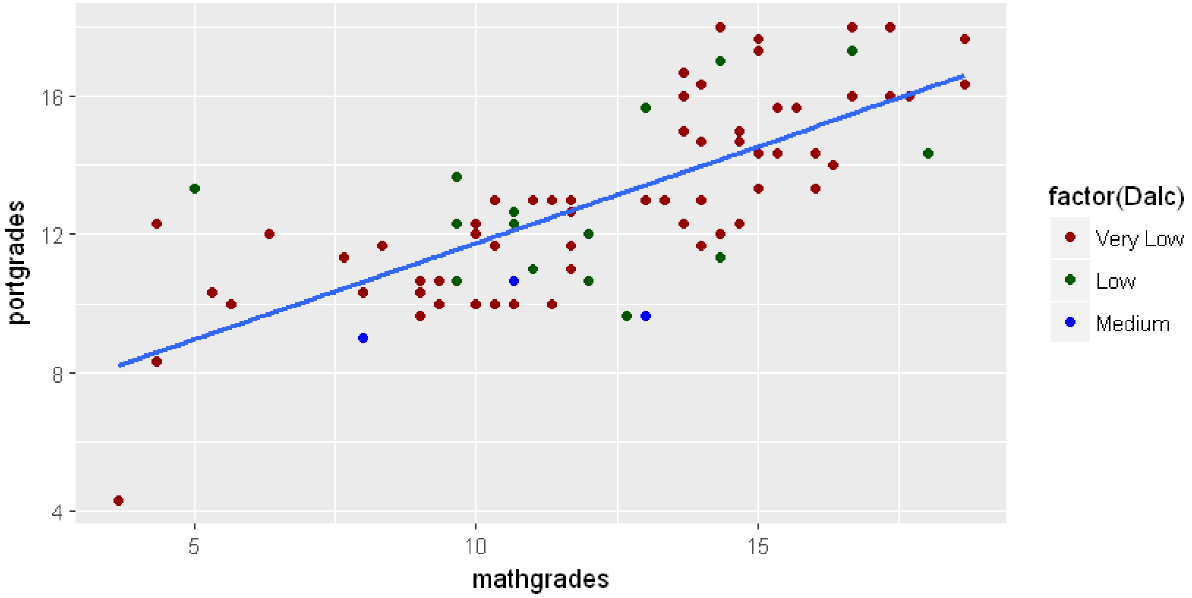
```
library(ggplot2)
library(plyr)
library(dplyr)
library(gridExtra)
library(alluvial)
library(extrafont)

d1=read.table("C:/Program Files (x86)/R/R-2.2.0/student-mat.csv",sep=",",header=T
d2=read.table("C:/Program Files (x86)/R/R-2.2.0/student-por.csv",sep=",",header=T

#Following the suggestion of Carlo Ventrella, one of the attributes, "paid," is c
#rather than student specific, so I am eliminating it from the list of attributes
# are matched matched
data.source=merge(d1,d2,by=c("school","sex","age","address","famsize","Pstatus",
                            "Medu","Fedu","Mjob","Fjob","reason","nursery","inter
                            "guardian","guardian","traveltime","studytime","failu
                            "schoolsup","famsup","activities","higher","romantic"
                            "famrel","freetime","goout","Dalc","Walc","health","a
print(nrow(data.source))
```

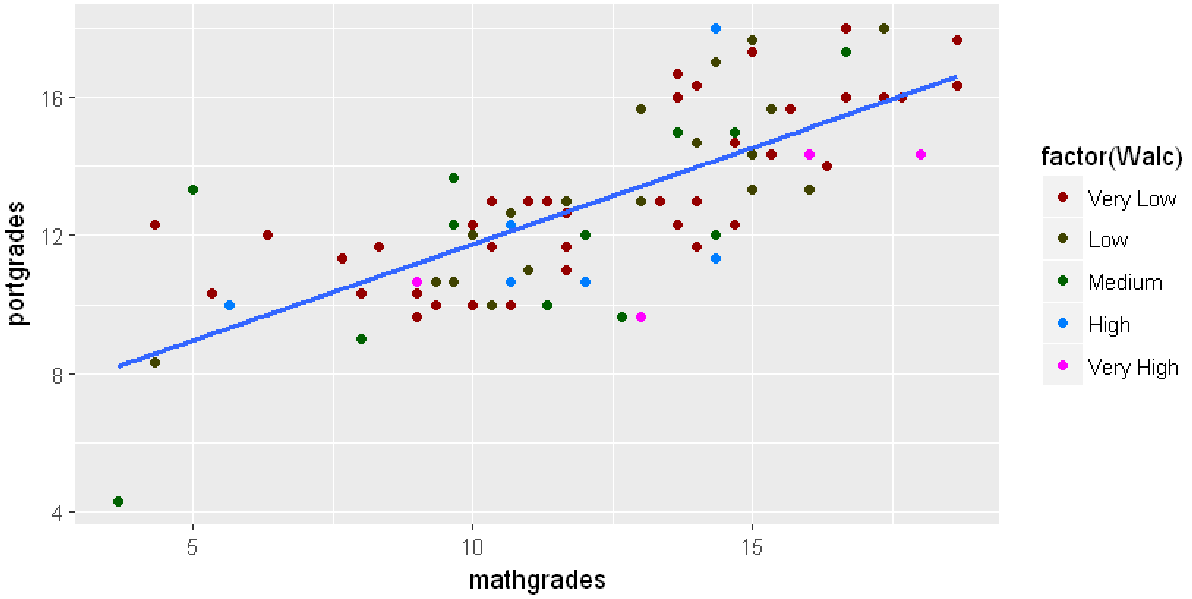
```
[1] 85
```

In [3]:

```
data.source$mathgrades=rowMeans(cbind(data.source$G1.x,data.source$G2.x,data.sour
data.source$portgrades=rowMeans(cbind(data.source$G1.y,data.source$G2.y,data.sour

data.source$Dalc <- as.factor(data.source$Dalc)
data.source$Dalc <- mapvalues(data.source$Dalc,
                              from = 1:5,
                              to = c("Very Low", "Low", "Medium", "High", "Very H

str1=ggplot(data.source, aes(x=mathgrades, y=portgrades)) +
 geom_point(aes(colour=factor(Dalc)))+ scale_colour_hue(l=25,c=150)+
geom_smooth(method = "lm", se = FALSE)

data.source$Walc <- as.factor(data.source$Walc)
data.source$Walc <- mapvalues(data.source$Walc,
                              from = 1:5,
                              to = c("Very Low", "Low", "Medium", "High", "Very H

str2=ggplot(data.source, aes(x=mathgrades, y=portgrades))+
geom_point(aes(colour=factor(Walc)))+ scale_colour_hue(l=25,c=150)+
geom_smooth(method = "lm", se = FALSE)

grid.arrange(str1,str2,nrow=2)
```

```
The following `from` values were not present in `x`: 4, 5
```

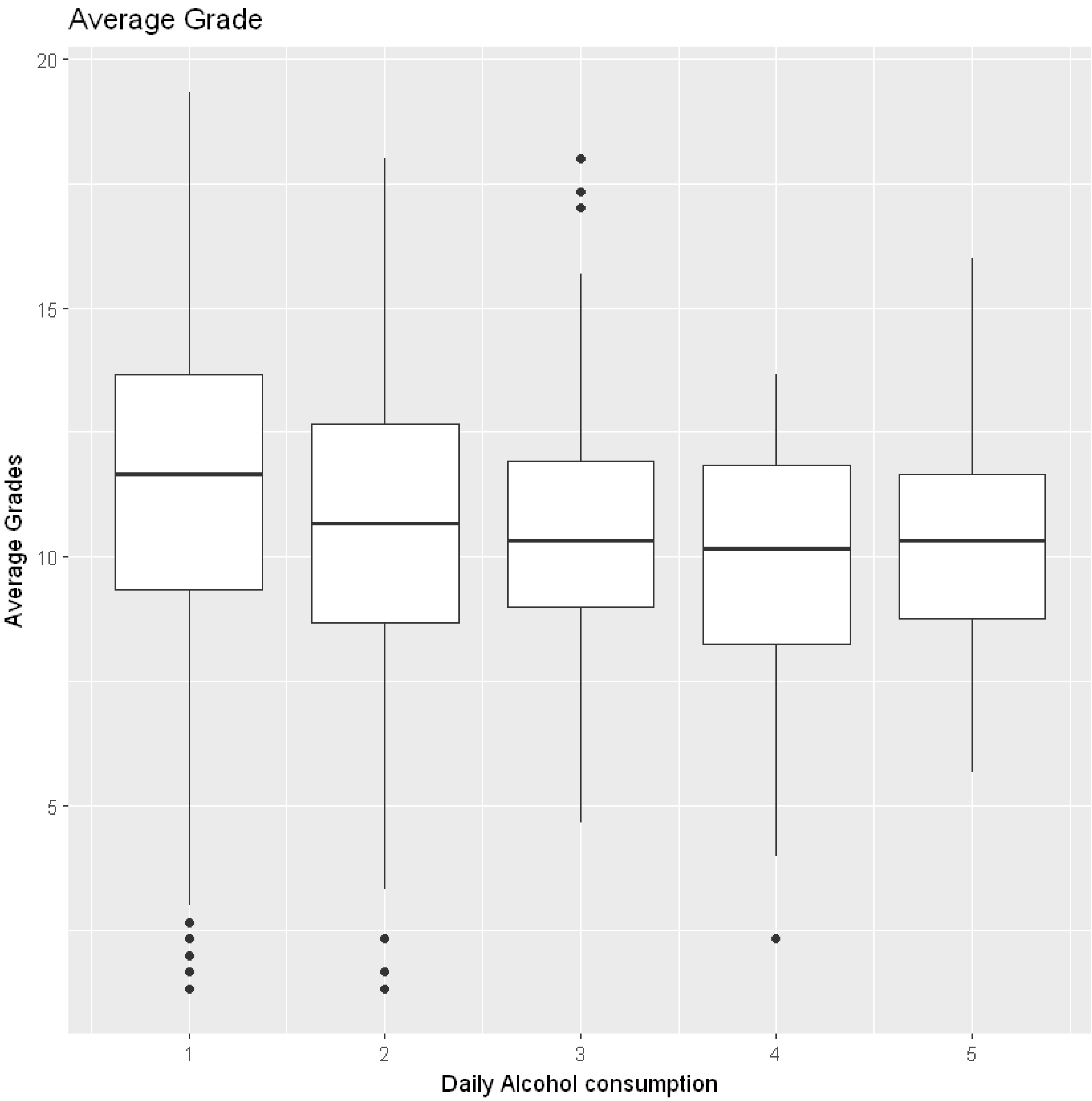




In [4]:
```
d3<-rbind(d1,d2) #combine the two datasets
# and eliminate the repeats:
d3norepeats<-d3 %>% distinct(school,sex,age,address,famsize,Pstatus,
                Medu,Fedu,Mjob,Fjob,reason,
                guardian,traveltime,studytime,failures,
                schoolsup, famsup,activities,nursery,higher,internet,
                romantic,famrel,freetime,goout,Dalc,Walc,health,absences, .keep_a
#add a column with average grades (math or Portuguese, whichever is available)
d3norepeats$avggrades=rowMeans(cbind(d3norepeats$G1,d3norepeats$G2,d3norepeats$G3
# and drop grades in 3 marking periods.
d3norepeats<-d3norepeats[,-(31:33)]
```

In [7]:

```
ggplot(d3norepeats, aes(x=Dalc, y=avggrades, group=Dalc))+
  geom_boxplot()+
  theme(legend.position="none")+
  scale_fill_manual(values=waffle.col)+
  xlab("Daily Alcohol consumption")+
  ylab("Average Grades")+
  ggtitle("Average Grade")
```



In [8]:

```
failureind<-which(names(d3norepeats)=="failures")
d3norepeats<-d3norepeats[,-failureind]
```

In [10]:

```
# 1) multiple regression
lm2<-lm(avggrades~., data=d3norepeats[,1:30])
summary(lm2)
```

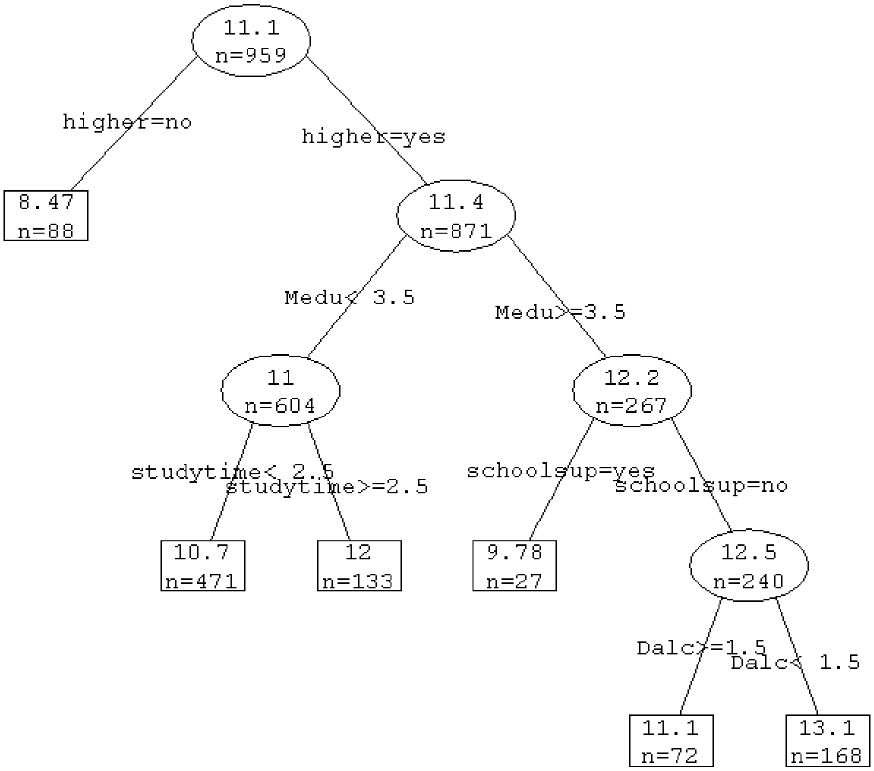
```
Call:
lm(formula = avggrades ~ ., data = d3norepeats[, 1:30])

Residuals:
     Min       1Q   Median       3Q      Max 
-10.8048  -1.5890   0.1455   1.8676   8.4868 

Coefficients:
                   Estimate Std. Error t value Pr(>|t|)    
(Intercept)      10.6017499  1.8139110   5.845 7.04e-09 ***
schoolMS         -0.3723012  0.2574810  -1.446  0.14854    
sexM             -0.1215773  0.2255528  -0.539  0.59000    
age              -0.1198027  0.0906465  -1.322  0.18661    
addressU          0.2930648  0.2427090   1.207  0.22756    
famsizeLE3        0.4975437  0.2219491   2.242  0.02522 *  
PstatusT          0.0640413  0.3168516   0.202  0.83987    
Medu              0.1848194  0.1396447   1.323  0.18600    
Fedu              0.1642152  0.1234646   1.330  0.18383    
Mjobhealth        0.7195927  0.4873500   1.477  0.14014    
Mjobother         0.0878501  0.2874717   0.306  0.75998    
Mjobservices      0.3595006  0.3392675   1.060  0.28959    
Mjobteacher       0.1089511  0.4567083   0.239  0.81150    
Fjobhealth       -0.2298997  0.6841685  -0.336  0.73693    
Fjobother         0.0004672  0.4281211   0.001  0.99913    
Fjobservices     -0.2993809  0.4493361  -0.666  0.50540    
Fjobteacher       1.1650673  0.6051943   1.925  0.05452 .  
reasonhome        0.2418873  0.2555691   0.946  0.34416    
reasonother       0.3335054  0.3412013   0.977  0.32861    
reasonreputation  0.4954622  0.2654763   1.866  0.06232 .  
guardianmother   -0.1771667  0.2439794  -0.726  0.46793    
guardianother    -0.2497580  0.4596099  -0.543  0.58698    
traveltime       -0.1100901  0.1449817  -0.759  0.44785    
studytime         0.5080568  0.1272501   3.993 7.06e-05 ***
schoolsupyes     -1.5979812  0.3190187  -5.009 6.56e-07 ***
famsupyes        -0.3691163  0.2091140  -1.765  0.07787 .  
paidyes          -0.6938560  0.2413583  -2.875  0.00414 ** 
activitiesyes     0.0980887  0.2022346   0.485  0.62777    
nurseryyes        0.0309093  0.2473158   0.125  0.90057    
higheryes         1.9236766  0.3618564   5.316 1.33e-07 ***
internetyes       0.4269133  0.2569866   1.661  0.09701 .  
romanticyes      -0.4996783  0.2097075  -2.383  0.01739 *  
famrel            0.1569738  0.1056496   1.486  0.13768    
freetime         -0.0399315  0.1029791  -0.388  0.69828    
goout            -0.2127131  0.0995379  -2.137  0.03286 *  
Dalc             -0.1303141  0.1389503  -0.938  0.34857    
Walc             -0.0210543  0.1092166  -0.193  0.84718    
health           -0.1562929  0.0709717  -2.202  0.02790 *  
absences         -0.0181346  0.0162657  -1.115  0.26519    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 2.943 on 920 degrees of freedom
```

```
Multiple R-squared:  0.2015,    Adjusted R-squared:  0.1685 
F-statistic:  6.11 on 38 and 920 DF,  p-value: < 2.2e-16
```

In [12]:

```
#2) Regression tree:
library(rpart)
library(DMwR)# I will be relying heavily on the DMwR library that comes with Torg
rt2<-rpart(avggrades~., data=d3norepeats[,1:30])
prettyTree(rt2)
```

```
Loading required package: lattice
Loading required package: grid

Attaching package: 'DMwR'

The following object is masked from 'package:plyr':

    join
```



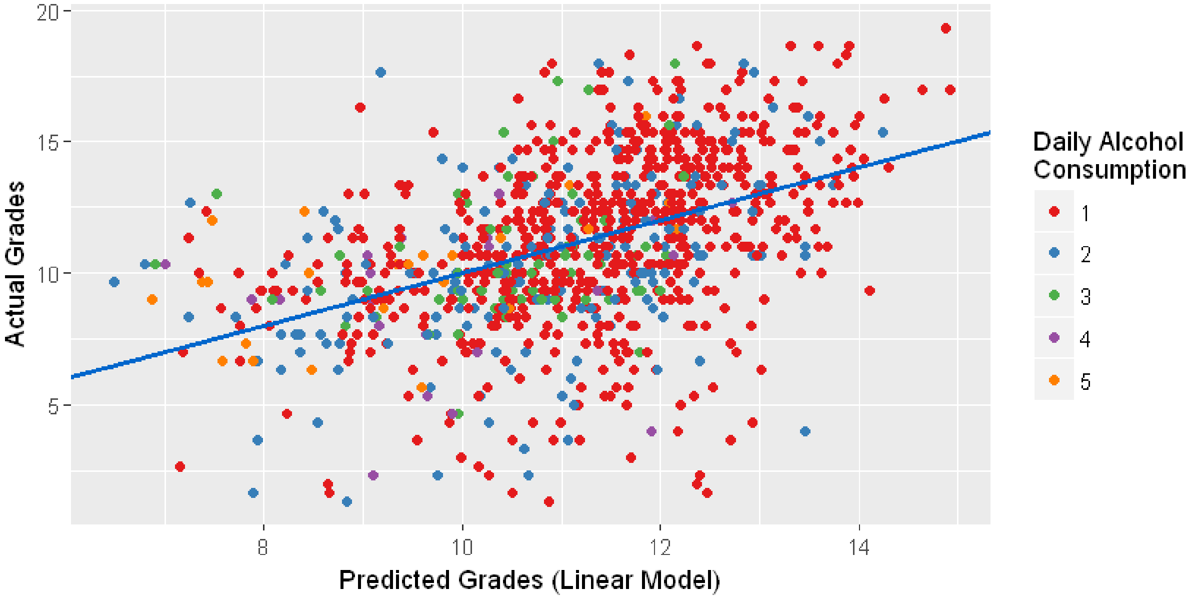
In [13]:

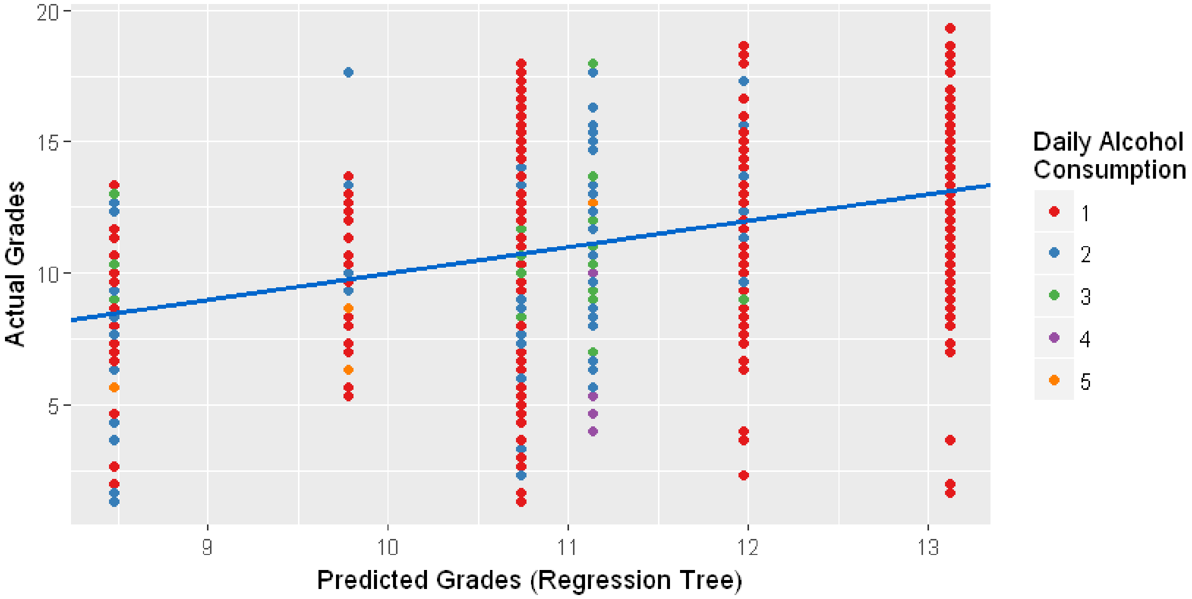
```
#predictions
lm.predictions<-predict(lm2,d3norepeats)
rt.predictions<-predict(rt2,d3norepeats)
nmse.lm<-mean((lm.predictions-d3norepeats[,"avggrades"])^2)/mean((mean(d3norepeat
nmse.rt<-mean((rt.predictions-d3norepeats[,"avggrades"])^2)/mean((mean(d3norepeat
print(nmse.lm) #0.79
print(nmse.rt) #0.85
```

```
[1] 0.7984877
[1] 0.849412
```

In [14]:

```
lmpltdata1=data.frame(cbind(lm.predictions,d3norepeats[,"avggrades"]))
colnames(lmpltdata1)<-c("lm.predictions","avggrades")
rtpltdata1=data.frame(cbind(rt.predictions,d3norepeats[,"avggrades"]))
colnames(rtpltdata1)<-c("rt.predictions","avggrades")

d3norepeats$Dalc<-as.factor(d3norepeats$Dalc)

errplt.lt1=ggplot(lmpltdata1,aes(lm.predictions,avggrades))+
                  geom_point(aes(color=d3norepeats[,"Dalc"]))+
                  xlab("Predicted Grades (Linear Model)")+
                  ylab("Actual Grades")+
                  geom_abline(intercept=0,slope=1,color="#0066CC",size=1)+
                  #geom_smooth(method = "lm", se = FALSE)+
                  scale_colour_brewer(palette = "Set1",name = "Daily Alcohol \nCo

errplt.rt1=ggplot(rtpltdata1,aes(rt.predictions,avggrades))+
  geom_point(aes(color=d3norepeats[,"Dalc"]))+
  xlab("Predicted Grades (Regression Tree)")+
  ylab("Actual Grades")+
  geom_abline(intercept=0,slope=1,color="#0066CC",size=1)+
  #geom_smooth(method = "lm", se = FALSE)+
  scale_colour_brewer(palette = "Set1",name = "Daily Alcohol \nConsumption")

grid.arrange(errplt.lt1,errplt.rt1,nrow=2)
```





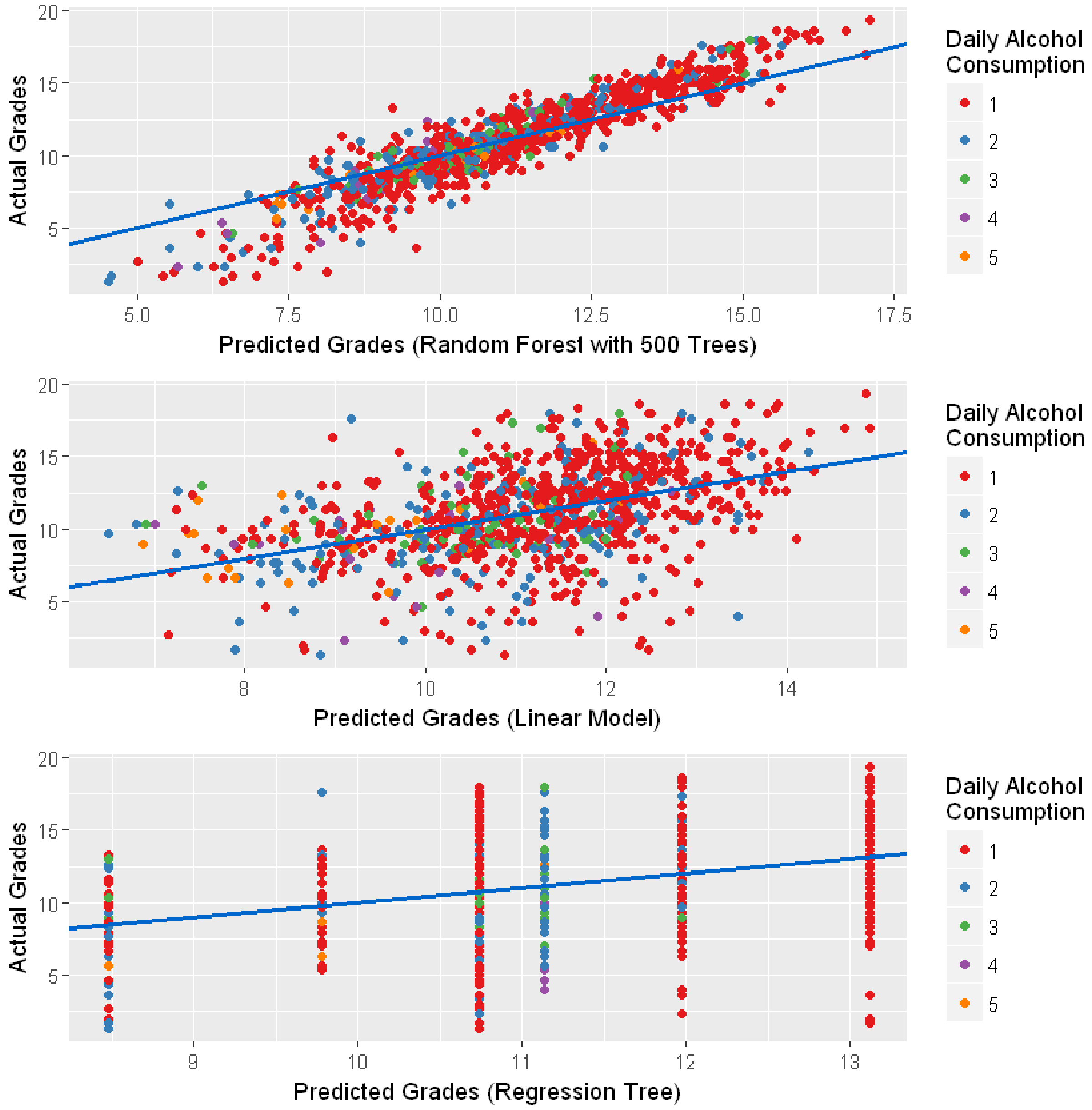
In [16]:

```
library(randomForest)
set.seed(4543)
rf2<-randomForest(avggrades~., data=d3norepeats[,1:30], ntree=500, importance=T)
rf.predictions<-predict(rf2,d3norepeats)
nmse.rf<-mean((rf.predictions-d3norepeats[,"avggrades"])^2)/mean((mean(d3norepeat
print(nmse.rf)
```

```
[1] 0.2038965
```

In [17]:

```
#first combine the rf predictions and actual scores in a single data frame
rfpltdata1=data.frame(cbind(rf.predictions,d3norepeats[,"avggrades"]))
colnames(rfpltdata1)<-c("rf.predictions","avggrades")

# then create the error plot.
errplt.rf1<-ggplot(rfpltdata1,aes(rf.predictions,avggrades))+
  geom_point(aes(color=d3norepeats[,"Dalc"]))+
  xlab("Predicted Grades (Random Forest with 500 Trees)")+
  ylab("Actual Grades")+
  geom_abline(intercept=0,slope=1,color="#0066CC",size=1)+
  #geom_smooth(method = "lm", se = FALSE)+
  scale_colour_brewer(palette = "Set1",name = "Daily Alcohol \nConsumption")
#finally, plot the error plot from the random forest with the error plots of the
grid.arrange(errplt.rf1, errplt.lt1,errplt.rt1,nrow=3)
```



In [ ]: