Using knitr with Multilevel Models and Decision Support Trees

Som Bohora

Febeuary 03, 2015

```
rm(list = ls(all = TRUE))
library(foreign)
library(reshape2)
library(sqldf)
library(reshape)
library(Hmisc)
```

Reading data and manipulation

```
  depressData <- read.csv("./SCUG Presentation/Final Presentation/Final/Demonstration/Data/data1.csv", header=T)
    header = TRUE)

depressData$CM <- with(depressData, ifelse(hurt1w1 %in% c(1, 2, 3, 4) | badnamesw1 %in%
    1:4 | needsw1 %in% 1:4 | sexabusew1 == 1, CM <- 1, ifelse(hurt1w1 == 0 &
    badnamesw1 == 0 & needsw1 == 0 & sexabusew1 == 0, CM <- 0, NA)))

depressData$hurt1w1 <- depressData$badnamesw1 <- depressData$needsw1 <- depressData$sexabusew1 <- NULL
names(depressData)
```

```
   [1] "ID"          "Group"       "bdiw1"       "bdiw2"       "bdiw3"
   [6] "PHASTcw1"    "PHASTcw2"    "PHASTcw3"    "singpsyrw1" "singpsyrw2"
  [11] "singpsyrw3" "spstotw1"    "spstotw2"    "spstotw3"    "TOTFRSw1"
  [16] "TOTFRSw2"    "TOTFRSw3"    "pregw1"      "pregw2"      "pregw3"
  [21] "helpfw1"     "helpfw2"     "helpfw3"     "CM"
```

```
catVars <- names(depressData) %in% c("Group", "CM", "singpsyrw1", "singpsyrw2",
    "singpsyrw3", "pregw1", "pregw2", "pregw3", "helpfw1", "helpfw2", "helpfw3",
    "depresw1", "depresw2", "depresw3")

allFactorVars <- data.frame(depressData[!catVars], sapply(depressData[catVars],
    function(x) as.factor(x)))

numVars <- !names(allFactorVars) %in% c("Group", "CM", "singpsyrw1", "singpsyrw2",
    "singpsyrw3", "pregw1", "pregw2", "pregw3", "helpfw1", "helpfw2", "helpfw3",
    "depresw1", "depresw2", "depresw3")

allNumericFactorVars <- data.frame(allFactorVars[!numVars], sapply(allFactorVars[numVars],
    function(x) as.numeric(x)))
```

Imputation and restructure

```
library(mice)
imputed1 <- mice(allNumericFactorVars, me =c("", "logreg","logreg","logreg","logreg","logreg","logreg",
                                             "logreg","logreg","logreg","logreg","","","","",
                                             "norm","norm","norm","norm","norm","norm","norm","norm","norm"))

# Removing these variables as predictors
pred <- imputed1$pred
pred[,c("ID","Group", "bdiw1", "bdiw2", "bdiw3")] <- 0

# Removing `pred` variables as predictors and re-running `MICE`
imputed2 <- mice(allNumericFactorVars, seed= 98765431, me =c("", "logreg","logreg","logreg","logreg","logreg",
```

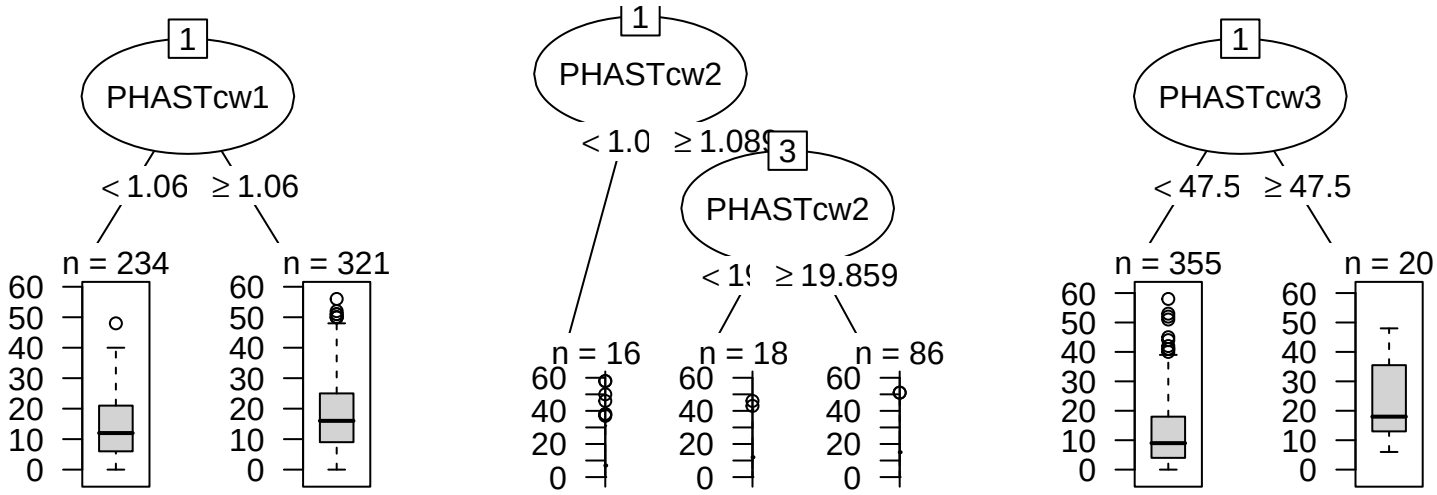
```
                                    "logreg","logreg","logreg","logreg","logreg","","","", "","norm","norm","norm",
                                    "norm","norm","norm","norm","norm","norm"), pred = pred)

completeImputed <- complete(imputed2, inc=FALSE)

long <-reshape(completeImputed, varying =list(c("bdiw1","bdiw2", "bdiw3"),
                                              c("PHASTcw1","PHASTcw2","PHASTcw3"),
                                              c("singpsyrw1", "singpsyrw2", "singpsyrw3"),
                                              c("TOTFRSw1", "TOTFRSw2", "TOTFRSw3"),
                                              c("pregw1", "pregw2", "pregw3"),
                                              c("helpfw1", "helpfw2", "helpfw3"),
                                              c("spstotw1","spstotw2","spstotw3")),
#                                               c("datew1","datew2","datew3")),
                  v.names=c("bdiScore", "IPV", "relationship", "famRes", "preg", "depressMed", "socialSupport"),
                        times = 1:3, new.row.names = NULL, direction = "long")

keepLong <- subset(long,
            select = c(ID, Group,CM, time,bdiScore,IPV,relationship,famRes,preg,depressMed,socialSupport))
nrow(keepLong)

keepLong[order(keepLong$time),]

nrow(depressData[complete.cases(keepLong),])

isComplete <- complete.cases(keepLong)
sum(isComplete)
finalLong <- keepLong[isComplete,]
nrow(finalLong)

tapply(finalLong$IPV, finalLong$time, summary)
summary(finalLong$IPV)
```

The mean of `IPV` is 11.1574516 and correlation between IPV and depression is 0.1753468 which is less than 0.494.

Regression tree on IPV

Regression tree on IPV;

```
library(rpart)
library(party)
library(partykit)
library(rpart.plot)

name1 <- c("bdiw1","bdiw2","bdiw3")
name2 <- c("PHASTcw1", "PHASTcw2", "PHASTcw3")

forReg <- depressData[, c("bdiw1", "PHASTcw1")]

reglist1 <- list()

for (i in 1:3){
  reg1 <- paste(name1[i],name2[i], sep='~')
#   print(regress)
  reglist1[[i]] <- rpart(as.formula(reg1), maxdepth=2, method ='anova', data=completeImputed)
  plots1 <- plot(as.party(reglist1[[i]]),tp_args = list(id = FALSE))
}
```




```
with(finalLong,tapply(IPV,time,summary))
```

```
$`1`
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -31.55    0.00    4.00   12.58   20.28  170.00

$`2`
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -31.28    0.00    4.00   10.12   15.72  108.00

$`3`
   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
 -19.53    0.00    4.00   10.27   12.00  125.00
```

Categorizing IPV based on regression tree cut-off

```
library(lme4)
library(MASS)
library(lmerTest)
library(nlme)
library(plyr)

finalLong$IPVCat <- ifelse(finalLong$IPV < 1.115, 0, 1)
table(finalLong$IPVCat)
```

```
##
##   0   1
## 547 822
```

```
# finalLong <- ddply(finalLong, "time", function(x) {within(x, {
#               if (time[1] == '1') {
#             IPVCat <- ifelse(IPV < 1.5, 0, 1)
#
#             } else if (time[1] == '2') {
#             IPVCat <- ifelse(IPV < 2.5, 0, 1)
#
#            } else {
#             IPVCat <- ifelse(IPV < 47.5, 0, 1)
#           }})
#       })
#
# #finalLong[,c("time","IPV", "IPVCat")]
finalLong$IPVCat <- as.factor(finalLong$IPVCat)
```

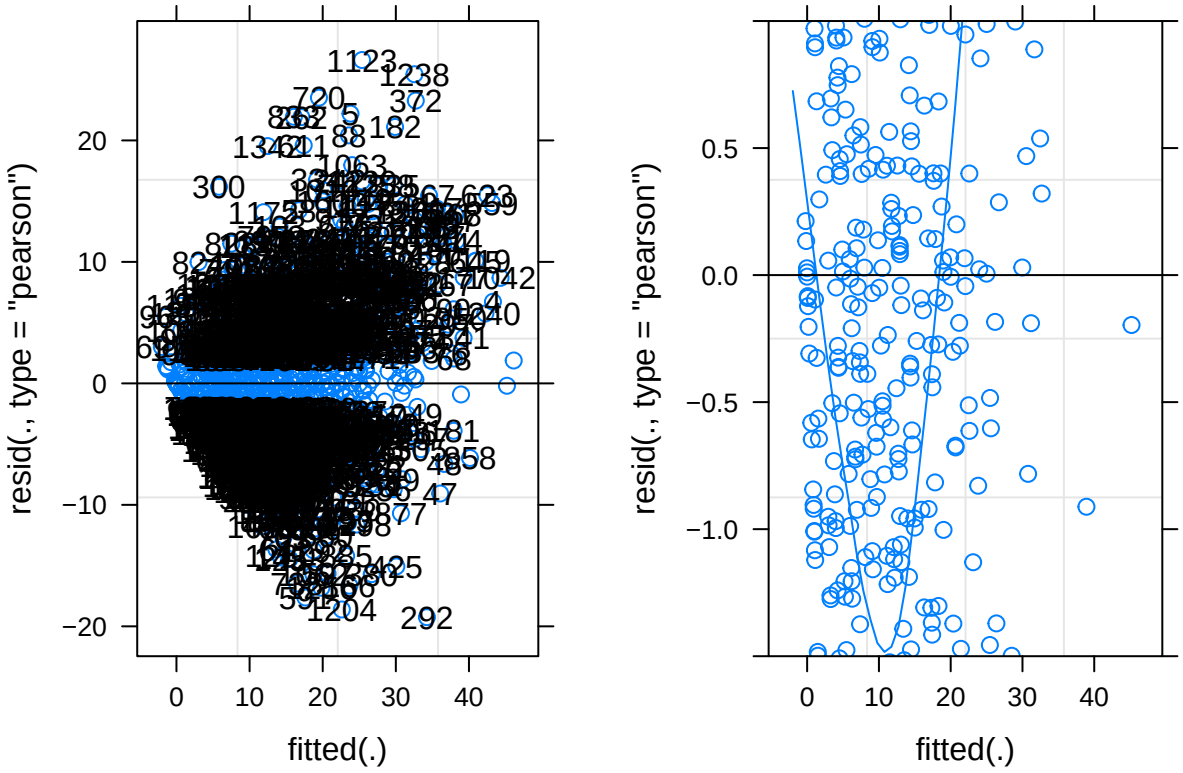
Different mixed-effects models (linear and quadratic) with IPV as categorical predictor

Results from linear mixed-effects model;

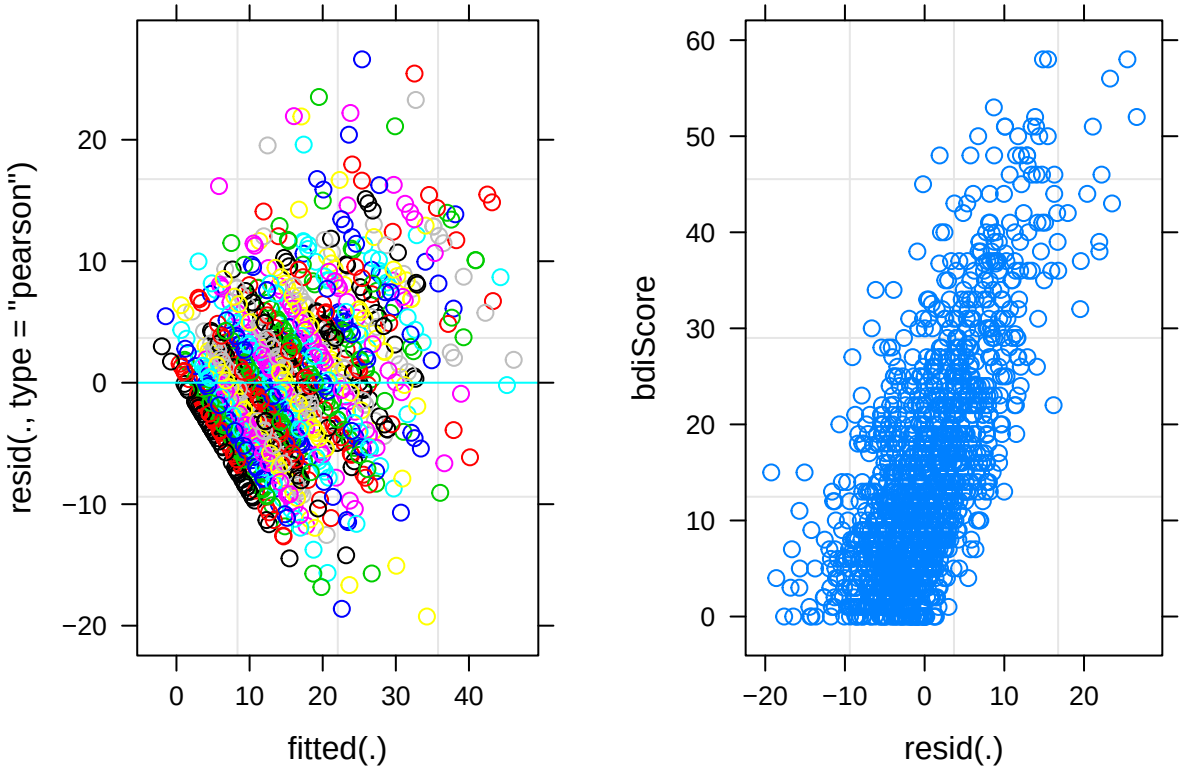
```
library(lme4)
library(lmerTest)
library(gridExtra)
library(coefplot)

mixedModel1 <- lmer(bdiScore ~ Group*factor(time) + CM*factor(time) + IPVCat + socialSupport +
                IPVCat:socialSupport + relationship + famRes + IPVCat:famRes + preg + depressMed +
                (1 | ID), data = finalLong)

anova(mixedModel1, ddf ="Kenward-Roger")
```

```
  Analysis of Variance Table of type 3  with  Kenward-Roger
  approximation for degrees of freedom
                       Sum Sq Mean Sq NumDF   DenDF F.value    Pr(>F)
  Group                   4.2     4.2     1  548.32   0.083  0.772931
  factor(time)          657.0   328.5     2  913.91   6.526  0.001534 **
  CM                    935.7   935.7     1  566.98  18.588 1.913e-05 ***
  IPVCat                218.4   218.4     1 1298.86   4.339  0.037447 *
  socialSupport        5010.9  5010.9     1 1327.09  99.544 < 2.2e-16 ***
  relationship          127.1   127.1     1 1322.09   2.524  0.112357
  famRes               6386.0  6386.0     1 1340.70 126.862 < 2.2e-16 ***
  preg                   72.8    72.8     1 1100.35   1.446  0.229452
  depressMed           2117.2  2117.2     1 1346.54  42.059 1.241e-10 ***
  Group:factor(time)    351.1   175.5     2  883.58   3.487  0.031010 *
  factor(time):CM       457.6   228.8     2  881.28   4.545  0.010869 *
  IPVCat:socialSupport  280.7   280.7     1 1279.02   5.577  0.018347 *
  IPVCat:famRes          14.4    14.4     1 1269.61   0.286  0.592983
  ---
  Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

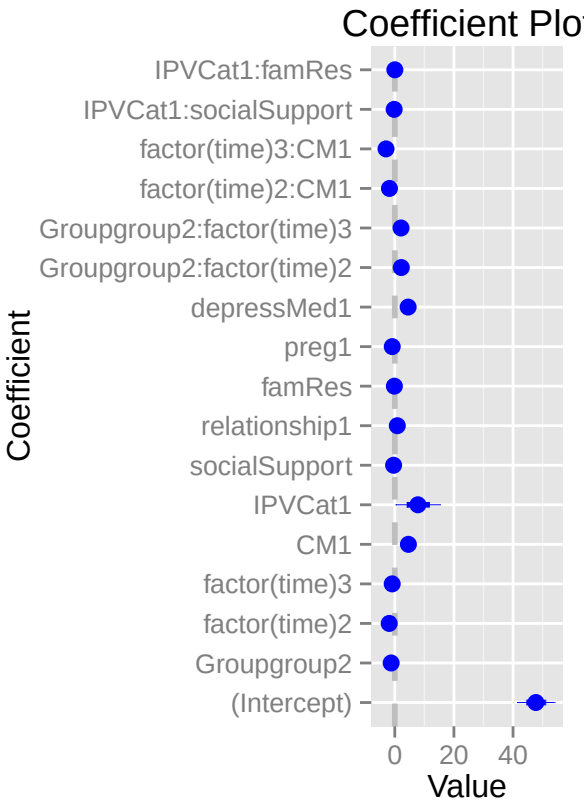
```
p1 <- plot(mixedModel1,id=0.05,idLabels=~.obs)
p2 <- plot(mixedModel1,ylim=c(-1.5,1),type=c("p","smooth"))
p3 <- plot(mixedModel1,col=finalLong$bdiScore)
p4 <- plot(mixedModel1,bdiScore~resid(.))
p5 <- coefplot(mixedModel1)
re<-ranef(mixedModel1,condVar = TRUE)
p6 <- qqmath(~re[[1]])
# p1; p2; p3; p4;p4;p5;p6
# Some regression diagnostics
grid.arrange(p1,p2,nrow=1)
```

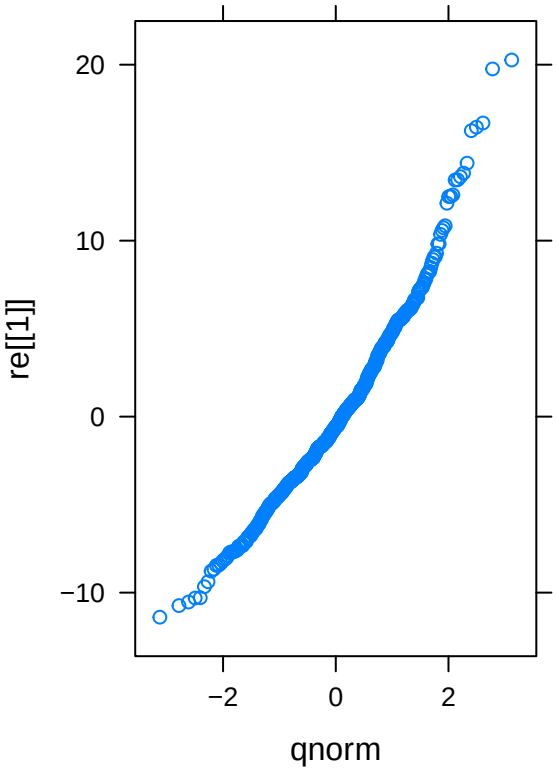



```
grid.arrange(p3,p4,nrow=1)
```




```
grid.arrange(p5,p6,nrow=1)
```


Coefficient Plot
IPVCat1:famRes
IPVCat1:socialSupport
factor(time)3:CM1
factor(time)2:CM1
Groupgroup2:factor(time)3
Groupgroup2:factor(time)2
depressMed1
preg1
famRes
relationship1
socialSupport
IPVCat1
CM1
factor(time)3
factor(time)2
Groupgroup2
(Intercept)
Coefficient
0
20
40
Value


20
10
0
−10
re[[1]]
−2
0
2
qnorm