Case Study 1: Clustering the epileptic.qol Dataset

latent class mixed effect model using the lcmm package

```
# install.packages("joineRML")
library(joineRML)
data(epileptic.qol)
# convert days to months
epileptic.qol$time_month <- epileptic.qol$time/30.25
# sort by id and time
epileptic.qol <- epileptic.qol[order(epileptic.qol$id,epileptic.qol$time_month),]

# scaling the clustering variables prior to analysis
epileptic.qol$anxiety_scale <- as.numeric(scale(epileptic.qol$anxiety))
epileptic.qol$depress_scale <- as.numeric(scale(epileptic.qol$depress))
epileptic.qol$aep_scale <- as.numeric(scale(epileptic.qol$aep))
```

latent class mixed effect model (lcmm package)

```
# install.packages("lcmm")
library(lcmm)
```

```
## Warning: package 'randtoolbox' was built under R version 4.2.2
```

```
## Warning: package 'rngWELL' was built under R version 4.2.2
```

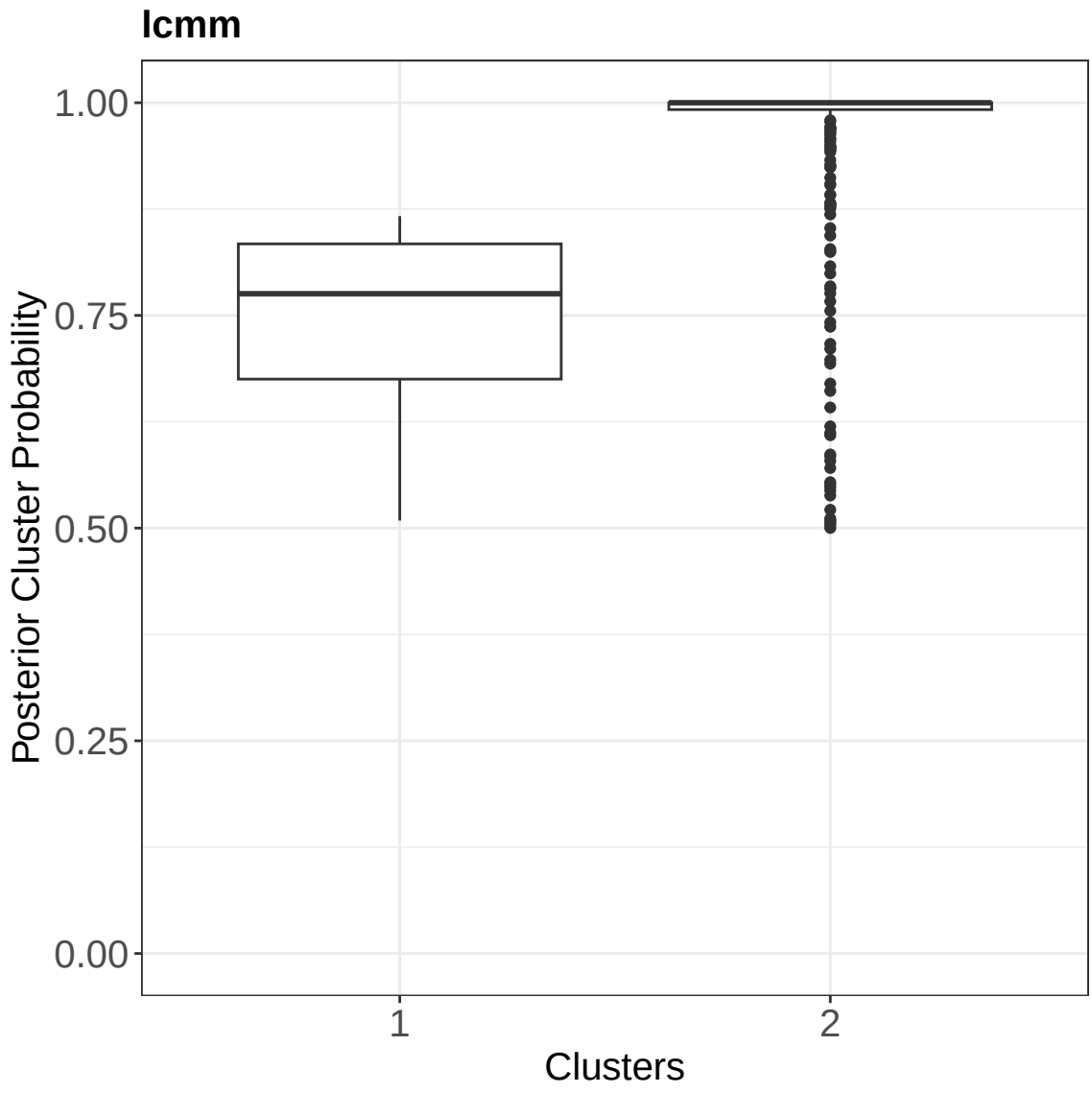
```
# fitting lcmm with K=1 to obtain initial values for models with K > 1
# each model assumes random intercept and random slope
mult1a <- multlcmm(anxiety_scale + depress_scale + aep_scale~ time_month,
            random =~ time_month ,
            subject='id',
            data = epileptic.qol,
            randomY = TRUE,
            verbose = FALSE,
            ng = 1)

# not run to reduce compiling time
#BIC <- NULL
#for (kk in 2:8){
#fit.multlcmm <- multlcmm(anxiety_scale + depress_scale + aep_scale ~ time_month,
#            mixture = ~ time_month,
#            random =~ time_month ,
#            subject='id',
#            data = epileptic.qol,
#            randomY = TRUE,
#            ng = kk,
#            B =mult1a  )
#BIC <- c(BIC,fit.multlcmm$BIC)
#}
```

```r
# print the number of clusters with the smallest BIC
# num.clust.multlcmm <- which.min(BIC) + 1; num.clust.multlcmm
```

```r
num.clust.multlcmm  <- 2 # optimal number of clusters based on bic
ptm <- proc.time()
# fitting the final model
fit.multlcmm <- multlcmm(anxiety_scale + depress_scale + aep_scale ~ time_month,
            mixture = ~ time_month,
            random =~ time_month ,
            subject='id',
            data = epileptic.qol,
            randomY = TRUE,
            verbose = FALSE,
            nwg = TRUE,
            ng = num.clust.multlcmm, B =mult1a  )
run.time.lcmm <- as.numeric((proc.time() - ptm)[3])
postprob <- apply(fit.multlcmm$pprob[,-c(1,2)],1,max)

# relabel cluster if needed
# here no relabeling is performed as the labels are appropriate
cluster.re <- fit.multlcmm$pprob$class
dnew_uq <- epileptic.qol[!duplicated(epileptic.qol$id, fromLast=TRUE),] # Keep last observation per id
dnew_uq$postprob <- postprob
dnew_uq$cluster.lcmm <-  cluster.re

# Posterior cluster probability
library(ggplot2)
```

```
## Warning: package 'ggplot2' was built under R version 4.2.2
```

```r
bp.lcmm <- ggplot(dnew_uq, aes(x=factor(cluster.lcmm), y=postprob)) +
            geom_boxplot() + ggtitle("lcmm") +
            xlab("Clusters") + ylab("Posterior Cluster Probability") +
        ylim(c(0,1)) +
        theme_bw() +
        theme(legend.position =  "none",
            plot.title = element_text(size = 15, face = "bold"),
            axis.text=element_text(size=15),
            axis.title=element_text(size=15),
            axis.text.x = element_text(angle = 0 ),
            strip.text.x = element_text(size = 15, angle = 0),
            strip.text.y = element_text(size = 15,face="bold"))
bp.lcmm
```




```
N <- length(unique(epileptic.qol$id))
per <- paste(round(100*table(cluster.re)/N,1),"%",sep="")
cluster.lcmm <- factor(cluster.re, label=paste("Cluster ",1:num.clust.multlcmm," (",per,")",sep=""))
dat.cluster <- data.frame(fit.multlcmm$pprob$id,cluster.lcmm)
colnames(dat.cluster) <- c("id","cluster.multlcmm")
dnew_uq <- merge(dnew_uq,dat.cluster,by="id")
dnew <- merge(epileptic.qol,dat.cluster,by="id")
```

```
library(ggplot2)
library(cowplot)
# plotting the first feature (anxiety) by clusters
p1.lcmm <- ggplot(data =dnew, aes(x =time_month, y = anxiety,
                color=cluster.multlcmm,
                linetype=cluster.multlcmm,
                fill=cluster.multlcmm))+
  ggtitle("lcmm") +
  geom_smooth(aes(x =time_month, y = anxiety,
```

```
            color=cluster.multlcmm,
                  linetype=cluster.multlcmm,
            fill=cluster.multlcmm), method = "loess",
              linewidth = 3,se = FALSE,span=2)+
  theme_bw() +
  theme(legend.position = "none",
        plot.title = element_text(size = 15, face = "bold"),
        axis.text=element_text(size=15),
        axis.title=element_text(size=15),
        axis.text.x = element_text(angle = 0 ),
        strip.text.x = element_text(size = 15, angle = 0),
        strip.text.y = element_text(size = 15,face="bold")) +
  guides(   fill=guide_legend(title=NULL,nrow = 1,byrow=TRUE),
        color=guide_legend(title=NULL,nrow = 1,byrow=TRUE),
            linetype=guide_legend(title=NULL,nrow = 1,byrow=TRUE)) +
  xlab("Time (months)") + ylab("anxiety") +
  ylim(c(min(dnew$anxiety,na.rm=TRUE),max(dnew$anxiety,na.rm=TRUE)))+
  scale_color_manual(values=c("green", "black"))+
  scale_fill_manual(values=c("green", "black"))

# plotting the second feature (depress) by clusters
p2.lcmm <- ggplot(data =dnew, aes(x =time_month, y = depress,
                                  color=cluster.multlcmm,
                                  linetype=cluster.multlcmm,
                                  fill=cluster.multlcmm))+
  ggtitle("lcmm") +
  geom_smooth(aes(x =time_month, y = depress,
            color=cluster.multlcmm,
                  linetype=cluster.multlcmm,
            fill=cluster.multlcmm),
              method = "loess", linewidth = 3,se = FALSE,span=2)+
  theme_bw() +
  theme(legend.position = "none",
        plot.title = element_text(size = 15, face = "bold"),
        axis.text=element_text(size=15),
        axis.title=element_text(size=15),
        axis.text.x = element_text(angle = 0 ),
        strip.text.x = element_text(size = 15, angle = 0),
        strip.text.y = element_text(size = 15,face="bold")) +
  guides(fill=guide_legend(title=NULL,nrow = 1,byrow=TRUE),
        color=guide_legend(title=NULL,nrow = 1,byrow=TRUE),
         linetype=guide_legend(title=NULL,nrow = 1,byrow=TRUE)) +
  xlab("Time (months)") + ylab("depress") +
  ylim(c(min(dnew$depress,na.rm=TRUE),max(dnew$depress,na.rm=TRUE)))+
  scale_color_manual(values=c("green", "black"))+
  scale_fill_manual(values=c("green", "black"))

# plotting the third feature (aep) by clusters
p3.lcmm <- ggplot(data =dnew, aes(x =time_month, y = aep,
            color=cluster.multlcmm,
            linetype=cluster.multlcmm,
            fill=cluster.multlcmm))+
  ggtitle("lcmm") +
```

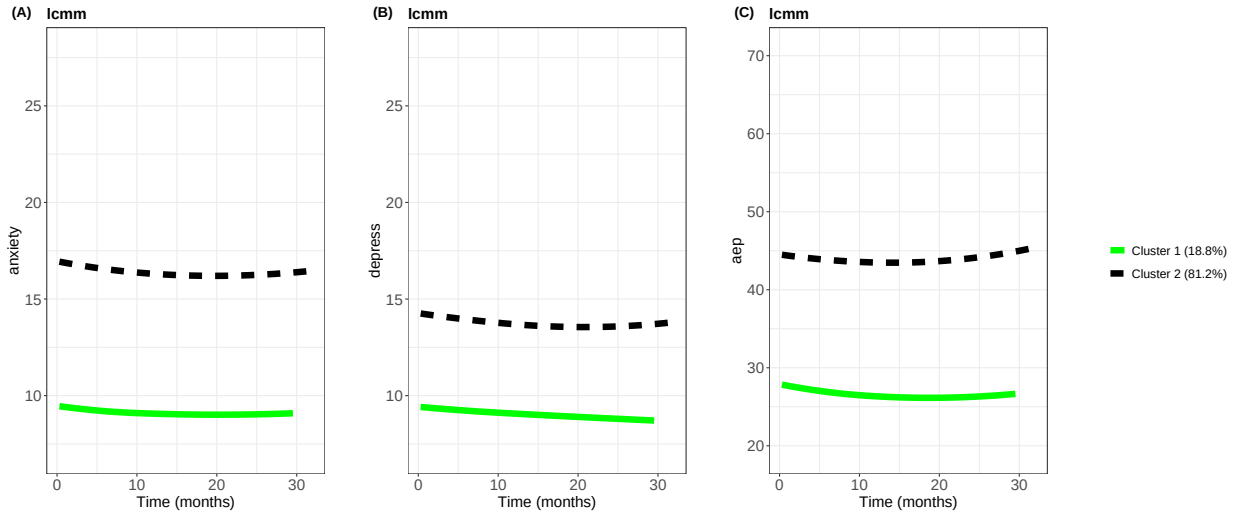
```
  geom_smooth(aes(x =time_month, y = aep,
            color=cluster.multlcmm,
                  linetype=cluster.multlcmm,
            fill=cluster.multlcmm), method = "loess",
              linewidth= 3,se = FALSE,span=2)+
  theme_bw() +
  theme(legend.position = "none",
        plot.title = element_text(size = 15, face = "bold"),
        axis.text=element_text(size=15),
        axis.title=element_text(size=15),
        axis.text.x = element_text(angle = 0 ),
        strip.text.x = element_text(size = 15, angle = 0),
        strip.text.y = element_text(size = 15,face="bold")) +
  guides(fill=guide_legend(title=NULL,nrow = 1,byrow=TRUE),
        color=guide_legend(title=NULL,nrow = 1,byrow=TRUE),
         linetype=guide_legend(title=NULL,nrow = 1,byrow=TRUE)) +
  xlab("Time (months)") + ylab("aep")  +
  ylim(c(min(dnew$aep,na.rm=TRUE),max(dnew$aep,na.rm=TRUE)))+
  scale_color_manual(values=c("green", "black")) +
  scale_fill_manual(values=c("green", "black"))

#-----------------------------------------------------------------#
# extract a legend
legend.lcmm <- get_legend(ggplot(data =dnew, aes(x =time_month, y = depress,
                color=cluster.multlcmm,
                linetype=cluster.multlcmm,
                fill=cluster.multlcmm))+
                         ggtitle("lcmm") +
                         geom_smooth(aes(x =time_month, y = depress,
                            color=cluster.multlcmm,
                                        linetype=cluster.multlcmm,
                            fill=cluster.multlcmm),
                                    method = "loess", linewidth = 3,se = FALSE,span=2)+
                         theme_bw() +
                         theme(legend.position = c(0.5,0.5),
                               legend.text = element_text(size = 12),
                               plot.title = element_text(size = 15, face = "bold"),
                               axis.text=element_text(size=15),
                               axis.title=element_text(size=15),
                               axis.text.x = element_text(angle = 0 ),
                               strip.text.x = element_text(size = 15, angle = 0),
                               strip.text.y = element_text(size = 15,face="bold")) +
                         guides(fill=guide_legend(title=NULL,nrow = 2,byrow=TRUE),
                                color=guide_legend(title=NULL,nrow = 2,byrow=TRUE),
                                linetype=guide_legend(title=NULL,nrow = 2,byrow=TRUE)) +
                         xlab("Time (months)") + ylab("depress") +
                         ylim(c(min(dnew$depress,na.rm=TRUE),max(dnew$depress,na.rm=TRUE)))+
                         scale_color_manual(values=c("green", "black"))+
                         scale_fill_manual(values=c("green", "black"))
)
```

```
## Warning: Removed 53 rows containing non-finite values (`stat_smooth()`).
```

```
plot_grid(p1.lcmm,NULL,p2.lcmm,NULL,p3.lcmm,NULL,legend.lcmm,
        labels=c("(A)","", "(B)","","(C)","",""),
            ncol = 7,
            rel_widths = c(1,0.1,1,0.1,1,0.1,0.5))
```

```
## Warning: Removed 57 rows containing non-finite values (`stat_smooth()`).
## Removed 53 rows containing non-finite values (`stat_smooth()`).
```

```
## Warning: Removed 93 rows containing non-finite values (`stat_smooth()`).
```




```
# Use Cox model to evaluate the association between
# clusters and time to treatment failure
dnew_uq$with.time.month <- dnew_uq$with.time/30.25
fit <- survfit(Surv(with.time.month, with.status2) ~ cluster.multlcmm,
               data = dnew_uq)
res.cox <- coxph(Surv(with.time.month, with.status2) ~ cluster.multlcmm,
                 weights=postprob,data =  dnew_uq)
pvalue <-  ifelse(summary(res.cox)$sctest[3] >= 0.0001,
                  summary(res.cox)$sctest[3],'<0.0001')

# Visualize with survminer package
library(survminer)
```

```
## Warning: package 'ggpubr' was built under R version 4.2.2
```

```
library(survival)
names(fit$strata) <-   paste("Cluster ",1:num.clust.multlcmm," (",per,")",sep="")
gp_survival.lcmm <- ggsurvplot(fit, data = dnew_uq, title="lcmm",
                     risk.table = TRUE,
                            risk.table.y.text.col = TRUE,
                            pval = pvalue,
                     legend = "bottom", # conf.int = TRUE,
                     xlab = "Time (months)",
                            legend.title="Clusters",
                          ggtheme = theme_bw() +
                               theme(legend.position ="none",legend.title=element_blank(),
                            plot.title = element_text(size = 15, face = "bold"),
                            axis.text=element_text(size=15),
```

```
                            axis.title=element_text(size=15),
                            strip.text.x = element_text(size=15),
                            strip.text.y = element_text(size=15)))

gp_survival.lcmm$plot <- gp_survival.lcmm$plot +
        guides(fill=guide_legend(title=NULL,nrow = 1, byrow=TRUE),
             color=guide_legend(title=NULL,nrow = 1, byrow=TRUE),
             linetype=guide_legend(title=NULL,nrow = 1, byrow=TRUE))+
        scale_color_manual(values=c("green", "black"))+
        scale_fill_manual(values=c("green", "black"))
gp_survival.lcmm$plot
```