Case Study 1: Clustering the epileptic.qol Dataset

Group-based trajectory modeling using the gbmt 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))
```

group-based trajectory modeling (gbmt package)

```
# install.packages("gbmt")
library(gbmt)
data(epileptic.qol)
N <- length(unique(epileptic.qol$id))
epileptic.qol$time_month <- epileptic.qol$time/30.25
epileptic.qol$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) )

# gbmt does not allow the time variable to be identical within an individual
# therefore, for individuals with an identical time value, add 0.0001
epileptic.qol[epileptic.qol$id==98,]$time_month[2]  <-
              epileptic.qol[epileptic.qol$id==98,]$time_month[2] + 0.0001
epileptic.qol[epileptic.qol$id==242,]$time_month[2] <-
              epileptic.qol[epileptic.qol$id==242,]$time_month[2] + 0.0001
epileptic.qol[epileptic.qol$id==302,]$time_month[2] <-
              epileptic.qol[epileptic.qol$id==302,]$time_month[2] + 0.0001
epileptic.qol[epileptic.qol$id==387,]$time_month[2] <-
              epileptic.qol[epileptic.qol$id==387,]$time_month[2] + 0.0001
epileptic.qol[epileptic.qol$id==389,]$time_month[2] <-
              epileptic.qol[epileptic.qol$id==389,]$time_month[2] + 0.0001
epileptic.qol[epileptic.qol$id==390,]$time_month[2] <-
              epileptic.qol[epileptic.qol$id==390,]$time_month[2] + 0.0001
epileptic.qol[epileptic.qol$id==486,]$time_month[2] <-
              epileptic.qol[epileptic.qol$id==486,]$time_month[2] + 0.0001
epileptic.qol[epileptic.qol$id==509,]$time_month[2] <-
              epileptic.qol[epileptic.qol$id==509,]$time_month[2] + 0.0001

# fitting GBTM using the gbmt function
```

```r
varNames <- c("anxiety_scale", "depress_scale","aep_scale")

# # not run to reduce compiling time
#bic <- NULL
#for (kk in 1:8){
#   fit.gbmt <- gbmt(x.names=varNames, unit="id", time="time_month",
#                d=1, ng=kk, data=epileptic.qol, scaling=0)
#   bic <- c(bic,fit.gbmt$ic[2])   # extract bic from the results
#   }
# print the best number of clusters with the smallest BIC
# num.clust.gbmt <- which.min(bic);num.clust.gbmt
```

```r
num.clust.gbmt <- 8  # optimal number of clusters based on bic
# fitting the final model with the optimal number of clusters
fit_gbmt <- gbmt(x.names=varNames, unit="id", time="time_month",
                     d=1, ng=num.clust.gbmt, data=epileptic.qol, scaling=0)
```

```
##  EM iteration 0. Log likelihood: -6245.9635  EM iteration 1. Log likelihood: -5689.7679  EM iteratio
```

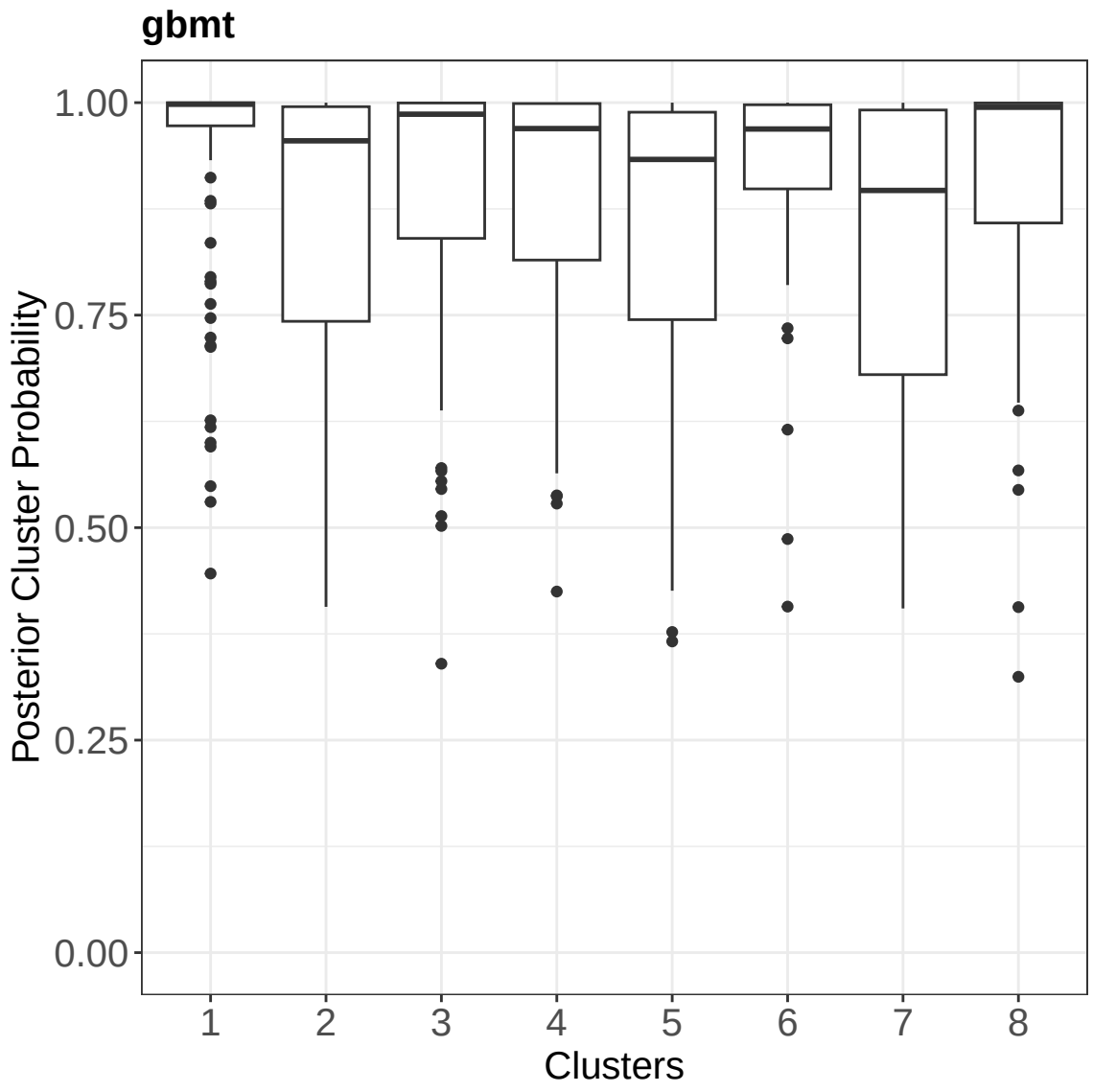
```r
# relabel the clusters
# this step is for making the results comparable to other methods
cluster.re <-   (fit_gbmt$assign==6)*1 +
                (fit_gbmt$assign==3)*2 +
                (fit_gbmt$assign==8)*3 +
                      (fit_gbmt$assign==4)*4 +
                (fit_gbmt$assign==7)*5 +
                (fit_gbmt$assign==5)*6 +
                      (fit_gbmt$assign==1)*7 +
                (fit_gbmt$assign==2)*8

# compute and plot the posterior cluster probability
postprob <- apply(posterior(fit_gbmt),1,max)
dat.postprob <- data.frame(postprob,cluster=fit_gbmt$assign,cluster.re=cluster.re)

library(ggplot2)
```

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

```r
# Posterior cluster probability
bp.gbmt <- ggplot(dat.postprob, aes(x=factor(cluster.re), y=postprob)) +
            geom_boxplot() + ggtitle("gbmt") +
            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.gbmt
```




```
per <- paste(round(100*table(cluster.re)/N,1),"%",sep="")
cluster.gbmt <- factor(cluster.re, labels=paste("Cluster ",1:num.clust.gbmt," (",per,")",sep=""))
# Keep last observation per id
dnew_uq <- epileptic.qol[!duplicated(epileptic.qol$id, fromLast=TRUE),]
dnew_uq$cluster.gbmt <- cluster.gbmt
dnew_uq$postprob <- postprob
dat.cluster <- data.frame(dnew_uq$id,dnew_uq$cluster.gbmt)
colnames(dat.cluster) <- c("id","cluster.gbmt")
dnew <- merge(epileptic.qol,dat.cluster,by="id")
dnew$time_month <- dnew$time/30.25
```

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

```
            ggtitle("gbmt")+
        geom_smooth(aes(x =time_month, y = anxiety,
                color=cluster.gbmt,
                linetype=cluster.gbmt,
                fill=cluster.gbmt),
                    method = "loess", linewidth = 3,se = FALSE,span=2)+
        theme_bw() +
    ylim(c(min(dnew$anxiety,na.rm=TRUE),max(dnew$anxiety,na.rm=TRUE)))+
        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 = 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("anxiety") +
        scale_color_manual(values=c("green", "black","blue","red",
                            "purple","goldenrod3","dimgray","darkorange3"))+
        scale_fill_manual(values=c("green", "black","blue","red",
                            "purple","goldenrod3","dimgray","darkorange3"))

# plotting the second feature (depress) by clusters
p2.gbmt <- ggplot(data =dnew, aes(x =time_month, y = depress,
                                  color=cluster.gbmt,
                                  linetype=cluster.gbmt,
                                  fill=cluster.gbmt))+
     ggtitle("gbmt")+
         geom_smooth(aes(x =time_month, y = depress,
                    color=cluster.gbmt,
                    linetype=cluster.gbmt,
                    fill=cluster.gbmt),
                     method = "loess", linewidth = 3,se = FALSE,span=2)+
        theme_bw() +
    ylim(c(min(dnew$depress,na.rm=TRUE),max(dnew$depress,na.rm=TRUE)))+
        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 = 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")  +
        scale_color_manual(values=c("green", "black","blue","red",
                            "purple","goldenrod3","dimgray","darkorange3"))+
        scale_fill_manual(values=c("green", "black","blue","red",
                            "purple","goldenrod3","dimgray","darkorange3"))
```

```
# plotting the third feature (aep) by clusters
p3.gbmt <- ggplot(data =dnew, aes(x =time_month, y = aep,
                                  color=cluster.gbmt,
                                  linetype=cluster.gbmt,
                                  fill=cluster.gbmt)) +
        ggtitle("gbmt")+
        geom_smooth(aes(x =time_month, y = aep,
                    color=cluster.gbmt,
                    linetype=cluster.gbmt,
                    fill=cluster.gbmt),
                    method = "loess", linewidth = 3,se = FALSE,span=2)+
        theme_bw() +
    ylim(c(min(dnew$aep,na.rm=TRUE),max(dnew$aep,na.rm=TRUE)))+
        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 = 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("aep")  +
        scale_color_manual(values=c("green", "black","blue","red",
                                  "purple","goldenrod3","dimgray","darkorange3"))+
        scale_fill_manual(values=c("green", "black","blue","red",
                                  "purple","goldenrod3","dimgray","darkorange3"))

#-----------------------------------------------------------------------#
# extract a legend
legend.gbmt <- get_legend(ggplot(data =dnew, aes(x =time_month, y = depress,
                                                 color=cluster.gbmt,
                                                 linetype=cluster.gbmt,
                                                 fill=cluster.gbmt))+
                          ggtitle("gbmt")+
              geom_smooth(aes(x =time_month, y = depress,
                              color=cluster.gbmt,
                              linetype=cluster.gbmt,
                              fill=cluster.gbmt),
              method = "loess", linewidth = 3,se = FALSE,span=2)+
                          theme_bw() +
                             ylim(c(min(dnew$depress,na.rm=TRUE),max(dnew$depress,na.rm=TRUE)))+
                          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,ncol=1,byrow=TRUE),
                                 color=guide_legend(title=NULL,ncol=1,byrow=TRUE),
```

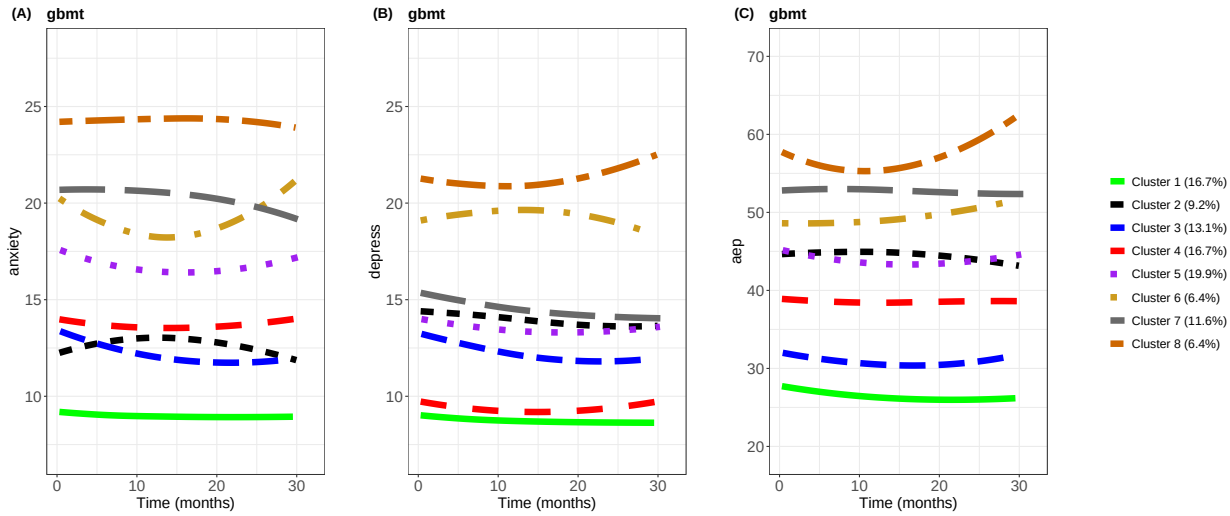
```
                              linetype=guide_legend(title=NULL,ncol=1,byrow=TRUE)) +
                        xlab("Time (months)") + ylab("depress")  +
                        scale_color_manual(values=c("green", "black","blue","red",
                           "purple","goldenrod3","dimgray","darkorange3"))+
                        scale_fill_manual(values=c("green", "black","blue","red",
                           "purple","goldenrod3","dimgray","darkorange3"))
)
```

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

```
plot_grid(p1.gbmt,NULL,p2.gbmt,NULL,p3.gbmt,NULL,legend.gbmt,
          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()`).
```



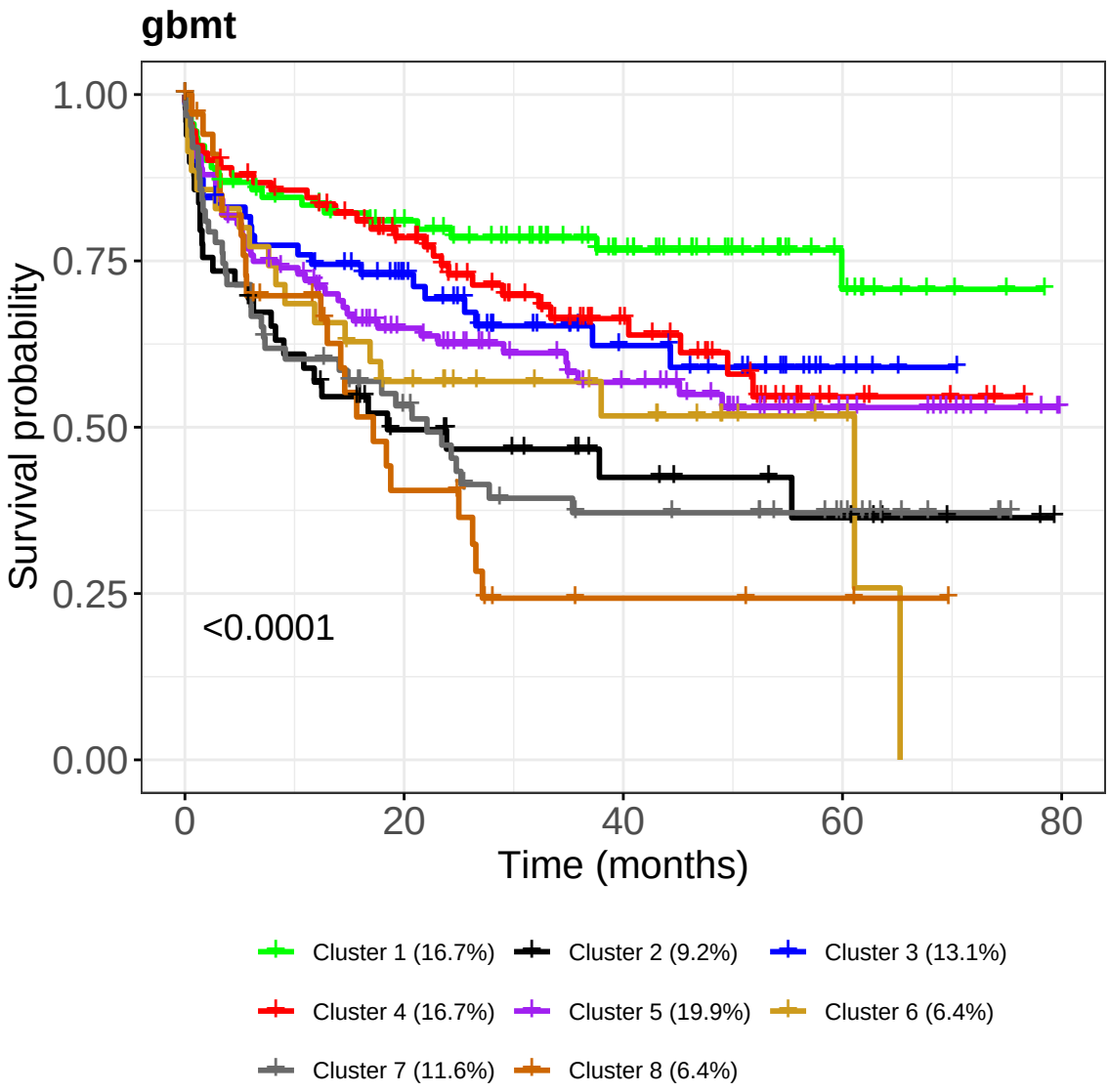

```
# weighted cox model by the posterior cluster probability
dnew_uq$with.time.month <- dnew_uq$with.time/30.25
fit <- survfit(Surv(with.time.month, with.status2) ~ cluster.gbmt, data = dnew_uq)
res.cox <- coxph(Surv(with.time.month, with.status2) ~ cluster.gbmt,
                 weights=postprob, data =  dnew_uq)
# extract the p-value for plotting
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.gbmt," (",per,")",sep="")
gp_survival.gbmt <- ggsurvplot(fit, data = dnew_uq, title="gbmt",
                           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.gbmt$plot <- gp_survival.gbmt$plot +
        guides(fill=guide_legend(title=NULL,nrow = 3,byrow=TRUE),
                  color=guide_legend(title=NULL,nrow = 3,byrow=TRUE),
                  linetype=guide_legend(title=NULL,nrow = 3,byrow=TRUE))+
        scale_color_manual(values=c("green", "black","blue","red",
                           "purple","goldenrod3","dimgray","darkorange3"))+
        scale_fill_manual(values=c("green", "black","blue","red",
                           "purple","goldenrod3","dimgray","darkorange3"))
gp_survival.gbmt$plot
```


gbmt
1.00
0.75
0.50
0.25
0.00
Survival probability
<0.0001
0
20
40
60
80
Time (months)
Cluster 1 (16.7%)
Cluster 2 (9.2%)
Cluster 3 (13.1%)
Cluster 4 (16.7%)
Cluster 5 (19.9%)
Cluster 6 (6.4%)
Cluster 7 (11.6%)
Cluster 8 (6.4%)