A Bayesian Beta-BinomialPiecewise Growth Mixture Model for Longitudinal Overdispersed Binomial Data

2024-08-14

Package Installed

```
library(VGAM)     # for rbetabinom
library(mvtnorm)  # for rmvt proposal
library(truncnorm)# for rtruncnorm()
library(tmvtnorm) # for trmvt()
library(msm)
library(aod)
library(MCMCpack)   # for Iwish
library(Hmisc)
library(BayesLogit) # for rpg function
library(ggplot2)
library(dplyr)
library(ggpubr)
```

Data Generating

```
rm(list=ls())

set.seed(25)
n<-250                               # number of subjects
ntrial<-7                      # TLFB upper bound
nis<-sample(1:12,n,replace=T)  # number obs per subject
id<-rep(1:n,nis)
N<-length(id)

# Covariate
t<-rep(0,N)
for (i in 1:n) t[id==i]<-sort(sample(1:12,nis[i]))  # time variable
t<-t-5                                              # centered time (T=-4,-3,..,7,8)
trt<-rbinom(n,1,.5)
tx<-rep(trt,nis)                                    # trt indicator

Xs<-cbind(t,tx,t*tx)                                # design matrix (with cp spline)

kappa<-true.kappa<-c(2,2,0,1,1,3)                   # class-specfic cps in 3 classes
xg11<-(t-kappa[1])*(t>kappa[1])*(tx==0)             # spline function for place gp in class 1
xg12<-(t-kappa[2])*(t>kappa[2])*(tx==1)             # tx gp in class 1
xg21<-(t-kappa[3])*(t>kappa[3])*(tx==0)             # pl gp in class 2
xg22<-(t-kappa[4])*(t>kappa[4])*(tx==1)             # tx gp in class 2
xg31<-(t-kappa[5])*(t>kappa[5])*(tx==0)             # pl gp in class 3
xg32<-(t-kappa[6])*(t>kappa[6])*(tx==1)             # tx gp in class 3
```

```
xg1<-xg11+xg12                  # slope after cp in class 1
xg2<-xg21+xg22                  # slope after cp in class 2
xg3<-xg31+xg32                  # slope after cp in class 3

X1<-cbind(Xs,xg11,xg12)         # design matix (class 1)
X2<-cbind(Xs,xg21,xg22)         # design matix (class 2)
X3<-cbind(Xs,xg31,xg32)         # design matix (class 3)

p1<-ncol(X1)                    # number of predictor in class 1
p2<-ncol(X2)                    # number of predictor in class 2
p3<-ncol(X3)                    # number of predictor in class 3



# Fixed effect param
# Class 1
beta10<-true.beta10<--2.15
beta1<-true.beta1<-c(0.35,-0.50,0.30,-0.95,-0.75)
# Class 2
beta20<-true.beta20<-0.40
beta2<-true.beta2<-c(0.25,0.65,0.15,-0.75,-0.60)
# Class 3
beta30<-true.beta30<-1.85
beta3<-true.beta3<-c(-0.65,  -0.30,  -0.25,  1.25, 1.55)

# Multinomial logit model
w1<-rbinom(n,1,.5) # Class Covs and Allocations
w2<-rnorm(n)
W<-cbind(1,w1,w2)
g<-ncol(W)
true.gamma1<-gamma1<-c(-.5,0.5,-.5)        # gamma3 is ref
true.gamma2<-gamma2<-c(0.25,-1,0.5)
etagam1<-W%*%gamma1;expetagam1<-exp(etagam1)
etagam2<-W%*%gamma2;expetagam2<-exp(etagam2)

pi3<-1/(1+expetagam1+expetagam2)
pi1<-expetagam1*pi3
pi2<-expetagam2*pi3
pi<-cbind(pi1,pi2,pi3)
c<-rep(0,n)
for(i in 1:n) c[i]<-(1:3)[rmultinom(1,1,pi[i,])==1]
true.c<-c             # true class assignment
# table(true.c)         # number of subjects in each class
n1<-length(c[c==1])   # number of subjects in class 1
n2<-length(c[c==2])   # number of subjects in class 2
n3<-n-n1-n2           # number of subjects in class 3
N1<-sum(nis[c==1])    # number of obs in each class
N2<-sum(nis[c==2])
N3<-N-N1-N2
C<-rep(c,nis)         # Class ID

nis1<-nis[c==1]       # number of obs per subject in class 1
nis2<-nis[c==2]       # class 2
```

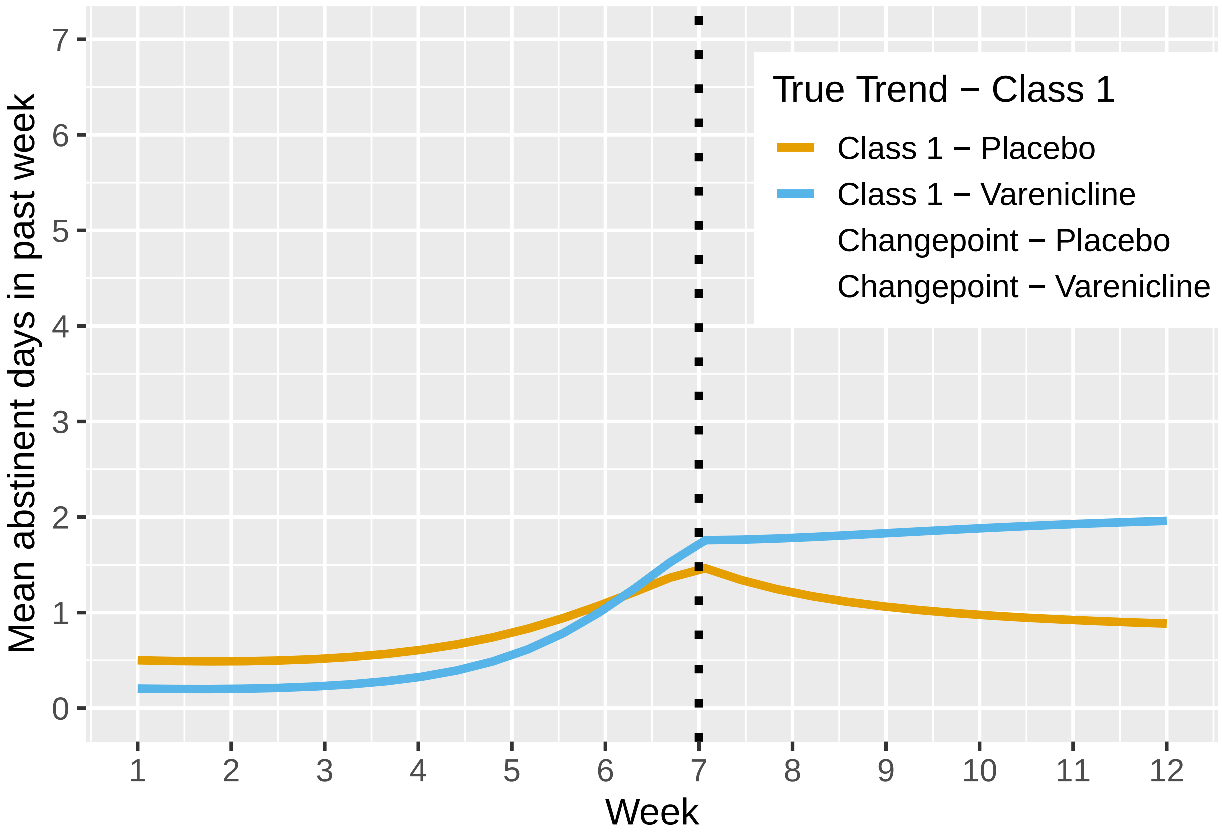
```
nis3<-nis[c==3]       # class 3


# Random Intercept/Slope/Second Slope
# class 1
truesigmab1<-sigmab1<-matrix(c(0.50,0.15,0.10,
                               0.15,0.25,0.05,
                               0.10,0.05,0.10),3,3)
# class 2
truesigmab2<-sigmab2<-matrix(c(0.20,0.05,0.05,
                               0.05,0.15,0.05,
                               0.05,0.05,0.10),3,3)
# class 3
truesigmab3<-sigmab3<-matrix(c(0.15,0.05,0.06,
                               0.05,0.15,-0.04,
                               0.06,-0.04,0.10),3,3)

# Cov matrix -> Corr matrix
# cov2cor(truesigmab1)
# cov2cor(truesigmab2)
# cov2cor(truesigmab3)


q<-3   # num of random effects in each class
trueb1<-b1<-rmvnorm(n1,sigma=sigmab1) # true random effect var in class 1
trueb2<-b2<-rmvnorm(n2,sigma=sigmab2) # true random effect var in class 2
trueb3<-b3<-rmvnorm(n3,sigma=sigmab3) # true random effect var in class 3

B11<-rep(b1[,1],nis1)
B12<-rep(b1[,2],nis1)
B13<-rep(b1[,3],nis1)

B21<-rep(b2[,1],nis2)
B22<-rep(b2[,2],nis2)
B23<-rep(b2[,3],nis2)

B31<-rep(b3[,1],nis3)
B32<-rep(b3[,2],nis3)
B33<-rep(b3[,3],nis3)


# Response
eta1<-beta10+X1[C==1,]%*%beta1+B11+B12*t[C==1]+B13*xg1[C==1]
eta2<-beta20+X2[C==2,]%*%beta2+B21+B22*t[C==2]+B23*xg2[C==2]
eta3<-beta30+X3[C==3,]%*%beta3+B31+B32*t[C==3]+B33*xg3[C==3]

mu1<-exp(eta1)/(1+exp(eta1))
mu2<-exp(eta2)/(1+exp(eta2))
mu3<-exp(eta3)/(1+exp(eta3))

truerho1<-rho1<-0.20               # corr. parameter in BB (class 1)
truerho2<-rho2<-0.15               # class 2
truerho3<-rho3<-0.35               # class 3
```

```r
# Outcome
y<-rep(NA,n)
y[C==1]<-rbetabinom(N1,size=ntrial,prob=mu1,rho=rho1) # class 1
y[C==2]<-rbetabinom(N2,size=ntrial,prob=mu2,rho=rho2) # class 2
y[C==3]<-rbetabinom(N3,size=ntrial,prob=mu3,rho=rho3) # class 3
```

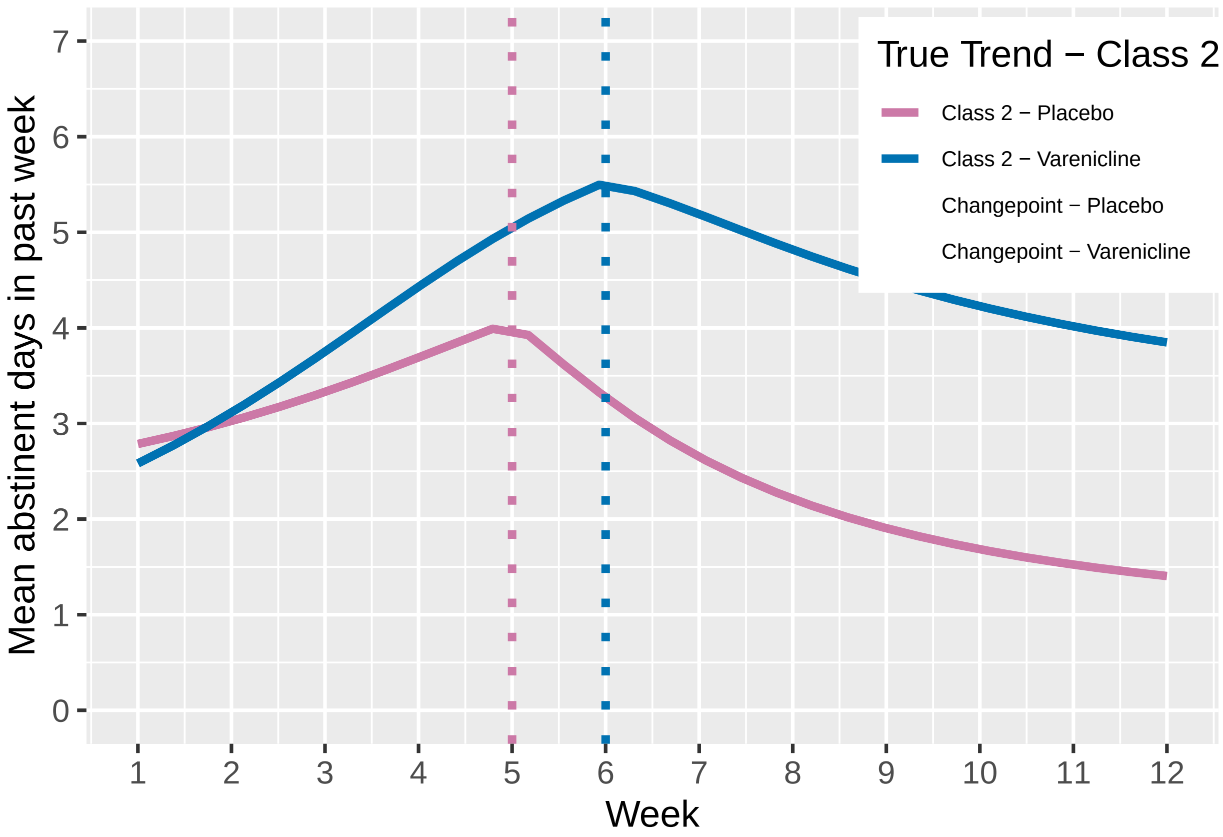
True abstinents day trends b/t groups in each class

```r
#------------#
# True Trend #
#------------#

num=30
gridind<-seq(-4,7,length.out=num)

#---------#
# Class 1 #
#---------#

X1p<-cbind(1,gridind,0,0,(gridind-true.kappa[1])*(gridind>true.kappa[1]),0)
X1t<-cbind(1,gridind,1,gridind,0,(gridind-true.kappa[2])*(gridind>true.kappa[2]))

n1p<-sum(trt[true.c==1]==0) # number of subject in class 1 (placebo)
n1t<-sum(trt[true.c==1]==1) # number of subject in class 1 (tx)


Y1p<-matrix(NA,n1p,num)     # Observed subject-level trend in class 1 (placebo)
Y1t<-matrix(NA,n1t,num)     # Observed subject-level trend in class 1 (tx)

# Class 1 - Placebo #
for(l in 1:n1p){
  eta1p<-X1p%*%c(true.beta10,true.beta1)+(trueb1[trt[true.c==1]==0,1])[l]+
    (trueb1[trt[true.c==1]==0,2])[l]*gridind+
    (trueb1[trt[true.c==1]==0,3])[l]*(gridind-true.kappa[1])*(gridind>true.kappa[1])
  y1p<-1/(1+exp(-eta1p))*7

  Y1p[l,]<-y1p
}

# Class 1 - Tx #
for(l in 1:n1t){
  eta1t<-X1t%*%c(true.beta10,true.beta1)+(trueb1[trt[true.c==1]==1,1])[l]+
    (trueb1[trt[true.c==1]==1,2])[l]*gridind+
    (trueb1[trt[true.c==1]==1,3])[l]*(gridind-true.kappa[2])*(gridind>true.kappa[2])
  y1t<-1/(1+exp(-eta1t))*7

  Y1t[l,]<-y1t
}


my1p<-colMeans(Y1p)
my1t<-colMeans(Y1t)
```

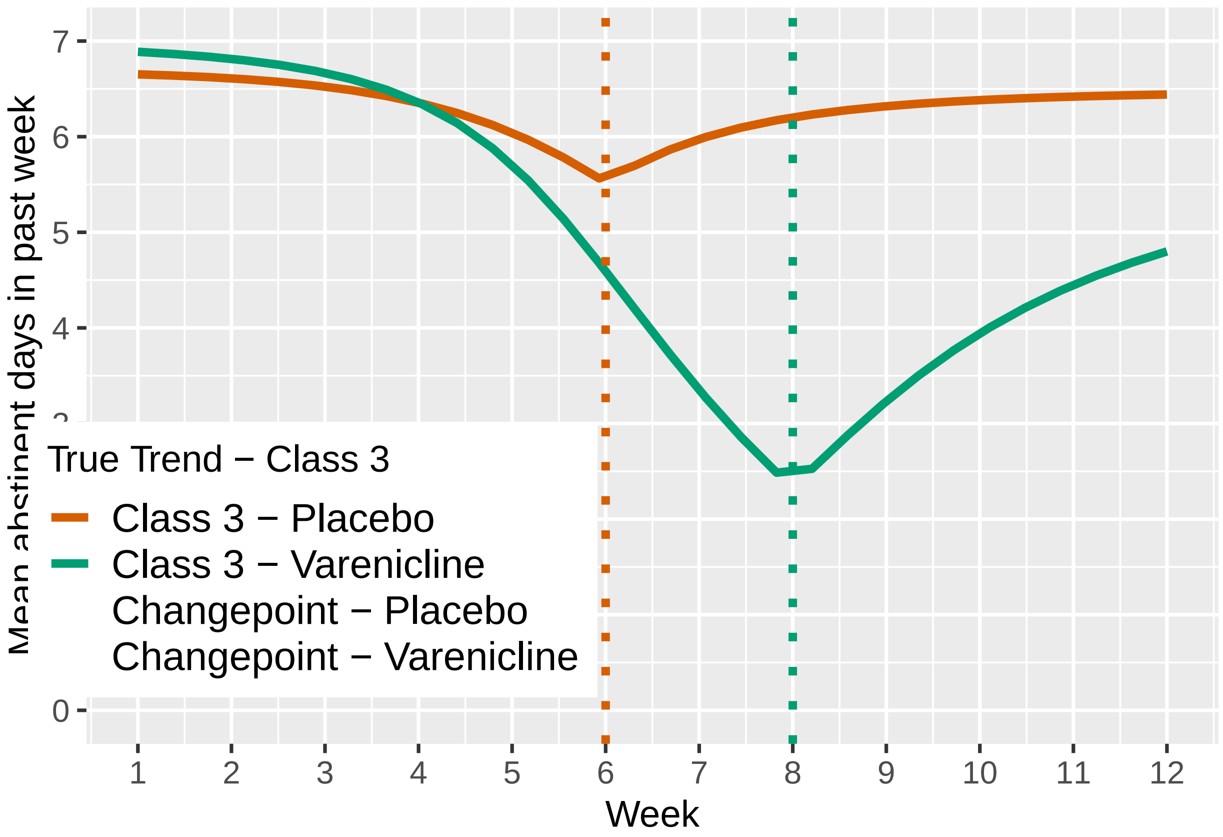
```r
dplotc0_c1<-data.frame(y=c(my1p,my1t),
                       t=rep(gridind+5,2),
                       gp=rep(1:2,each=num))

dplotc0_c1$gp<-factor(dplotc0_c1$gp,levels = c(1,2),
                      labels=c("Class 1 - Placebo","Class 1 - Varenicline"))


ggplot(dplotc0_c1,aes(x=t,y=y,shape=gp,color=gp))+
  geom_line(linewidth=1.5)+
  geom_vline(aes(xintercept=7,col="Changepoint - Placebo"),
             linetype=3,linewidth=1.5,show.legend = F)+
  geom_vline(aes(xintercept=7,col="Changepoint - Varenicline"),
             linetype=3,linewidth=1.5,show.legend = F)+
  xlab("Week")+ylab("Mean abstinent days")+
  scale_x_continuous(breaks = 1:12)+scale_y_continuous(limits = c(0,7),breaks = seq(0,7,by=1))+
  scale_color_manual(breaks = c("Class 1 - Placebo",
                                "Class 1 - Varenicline","Changepoint - Placebo",
                                "Changepoint - Varenicline"),
                     values = c("#E69F00","#56B4E9","black","black"))+
  scale_shape_manual(breaks = c("Class 1 - Placebo","Class 1 - Varenicline"),
                     values = c(2,7))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="True Trend - Class 1",
                              override.aes = list(
                                linetype = c(1,1,3,3),
                                shape=c(2,7,NA,NA)),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.position = c(0.80,0.75),legend.text=element_text(size=12),
        axis.text=element_text(size=12),
        axis.title=element_text(size=14))
```



```
#---------#
# Class 2 #
#---------#

X2p<-cbind(1,gridind,0,0,(gridind-true.kappa[3])*(gridind>true.kappa[3]),0)
X2t<-cbind(1,gridind,1,gridind,0,(gridind-true.kappa[4])*(gridind>true.kappa[4]))

n2p<-sum(trt[true.c==2]==0) # number of subject in class 2 (placebo)
n2t<-sum(trt[true.c==2]==1) # number of subject in class 2 (tx)


Y2p<-matrix(NA,n2p,num)      # Observed subject-level trend in class 2 (placebo)
Y2t<-matrix(NA,n2t,num)      # Observed subject-level trend in class 2 (tx)

# Class 2 - Placebo #
for(l in 1:n2p){
  eta2p<-X2p%*%c(true.beta20,true.beta2)+(trueb2[trt[true.c==2]==0,1])[l]+
    (trueb2[trt[true.c==2]==0,2])[l]*gridind+
    (trueb2[trt[true.c==2]==0,3])[l]*(gridind-true.kappa[3])*(gridind>true.kappa[3])
  y2p<-1/(1+exp(-eta2p))*7

  Y2p[l,]<-y2p
}

# Class 2 - Tx #
for(l in 1:n2t){
  eta2t<-X2t%*%c(true.beta20,true.beta2)+(trueb2[trt[true.c==2]==1,1])[l]+
    (trueb2[trt[true.c==2]==1,2])[l]*gridind+
    (trueb2[trt[true.c==2]==1,3])[l]*(gridind-true.kappa[4])*(gridind>true.kappa[4])
  y2t<-1/(1+exp(-eta2t))*7

  Y2t[l,]<-y2t
}


my2p<-colMeans(Y2p)
my2t<-colMeans(Y2t)


dplotc0_c2<-data.frame(y=c(my2p,my2t),
                       t=rep(gridind+5,2),
                       gp=rep(1:2,each=num))

dplotc0_c2$gp<-factor(dplotc0_c2$gp,levels = c(1,2),
                      labels=c("Class 2 - Placebo","Class 2 - Varenicline"))

ggplot(dplotc0_c2,aes(x=t,y=y,shape=gp,color=gp))+
  geom_line(linewidth=1.5)+
  geom_vline(aes(xintercept=5,col="Changepoint - Placebo"),
             linetype=3,linewidth=1.5,show.legend = F)+
  geom_vline(aes(xintercept=6,col="Changepoint - Varenicline"),
             linetype=3,linewidth=1.5,show.legend = F)+
  xlab("Week")+ylab("Mean abstinent days")+
  scale_x_continuous(breaks = 1:12)+scale_y_continuous(limits = c(0,7),breaks = seq(0,7,by=1))+
  scale_color_manual(breaks = c("Class 2 - Placebo","Class 2 - Varenicline","Changepoint - Placebo","Cha
                     values = c("#CC79A7","#0072B2","#CC79A7","#0072B2"))+
  scale_shape_manual(breaks = c("Class 2 - Placebo","Class 2 - Varenicline"),
                     values = c(2,7))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="True Trend - Class 2",
                              override.aes = list(
                                linetype = c(1,1,3,3),
                                shape=c(2,7,NA,NA)),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.position = c(0.85,0.80),legend.text=element_text(size=8),
        axis.text=element_text(size=12),
        axis.title=element_text(size=14))
```



```
#---------#
# Class 3 #
#---------#

X3p<-cbind(1,gridind,0,0,(gridind-true.kappa[5])*(gridind>true.kappa[5]),0)
X3t<-cbind(1,gridind,1,gridind,0,(gridind-true.kappa[6])*(gridind>true.kappa[6]))

n3p<-sum(trt[true.c==3]==0) # number of subject in class 3 (placebo)
n3t<-sum(trt[true.c==3]==1) # number of subject in class 3 (tx)


Y3p<-matrix(NA,n3p,num)     # Observed subject-level trend in class 3 (placebo)
Y3t<-matrix(NA,n3t,num)     # Observed subject-level trend in class 3 (tx)

# Class 3 - Placebo #
for(l in 1:n3p){
  eta3p<-X3p%*%c(true.beta30,true.beta3)+(trueb3[trt[true.c==3]==0,1])[l]+(trueb3[trt[true.c==3]==0,2])
  y3p<-1/(1+exp(-eta3p))*7

  Y3p[l,]<-y3p
}

# Class 3 - Tx #
for(l in 1:n3t){
  eta3t<-X3t%*%c(true.beta30,true.beta3)+(trueb3[trt[true.c==3]==1,1])[l]+(trueb3[trt[true.c==3]==1,2])
  y3t<-1/(1+exp(-eta3t))*7
  Y3t[l,]<-y3t
}


my3p<-colMeans(Y3p)
my3t<-colMeans(Y3t)


dplotc0_c3<-data.frame(y=c(my3p,my3t),
                       t=rep(gridind+5,2),
                       gp=rep(1:2,each=num))

dplotc0_c3$gp<-factor(dplotc0_c3$gp,levels = c(1,2),
                      labels=c("Class 3 - Placebo","Class 3 - Varenicline"))


ggplot(dplotc0_c3,aes(x=t,y=y,shape=gp,color=gp))+
  geom_line(linewidth=1.5)+
  geom_vline(aes(xintercept=6,col="Changepoint - Placebo"),
             linetype=3,linewidth=1.5,show.legend = F)+
  geom_vline(aes(xintercept=8,col="Changepoint - Varenicline"),
             linetype=3,linewidth=1.5,show.legend = F)+
  xlab("Week")+ylab("Mean abstinent days")+
  scale_x_continuous(breaks = 1:12)+scale_y_continuous(limits = c(0,7),breaks = seq(0,7,by=1))+
  scale_color_manual(breaks = c("Class 3 - Placebo","Class 3 - Varenicline","Changepoint - Placebo","Cha
                     values = c("#D55E00","#009E73","#D55E00","#009E73"))+
  scale_shape_manual(breaks = c("Class 3 - Placebo","Class 3 - Varenicline"),
                     values = c(2,7))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="True Trend - Class 3",
                              override.aes = list(
                                linetype = c(1,1,3,3),
                                shape=c(2,7,NA,NA)),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.position = c(0.20,0.25),legend.text=element_text(size=15),
        axis.text=element_text(size=12),
        axis.title=element_text(size=14))
```



Mean abstinent days in past week
7
6
5
4
3
0
1
2
3
4
5
6
7
8
9
10
11
12
Week
True Trend − Class 3
Class 3 − Placebo
Class 3 − Varenicline
Changepoint − Placebo
Changepoint − Varenicline

MCMC Set-Up

```
#------------------------------------------------------------------------------#
# MCMC prep #
#-----------#

# Priors
beta01<-rep(0,p1)
V0b1<-diag(100,p1)        # prior precision for beta1
beta02<-rep(0,p2)
V0b2<-diag(100,p2)        # prior precision for beta2
beta03<-rep(0,p3)
V0b3<-diag(100,p3)        # prior precision for beta3
a0<-b0<-c0<-1        # Hyeprparms  for beta class probs
d0<-q+1
C0<-diag(q)
T0g<-diag(0.01,g)         # prior precision for gamma
gamma0<-rep(0,g)


# proposal
covb1<-diag(.01,p1)
covb2<-diag(.01,p2)       # Proposal covariance
covb3<-diag(.01,p3)       # Proposal covariance
sigmar0<-0.005            # proposal var for rho
sigkap11<-0.55            # propodal var for kap11
sigkap12<-0.55            # propodal var for kap12
sigkap21<-0.30            # propodal var for kap21
sigkap22<-0.30            # propodal var for kap22
sigkap31<-0.5             # propodal var for kap31
sigkap32<-0.5             # propodal var for kap32
sigmab110<-0.05           # proposal var for b11
sigmab120<-0.01           # proposal var for b12
sigmab130<-0.01           # proposal var for b13
sigmab210<-0.10           # proposal var for b21
sigmab220<-0.05           # proposal var for b22
sigmab230<-0.05           # proposal var for b23
sigmab310<-0.05           # proposal var for b31
sigmab320<-0.05           # proposal var for b32
sigmab330<-0.05           # proposal var for b33


L0<--3
U0<-6

# Init
beta10<--2
beta20<-0
beta30<-1
beta1<-rep(0,p1)
beta2<-rep(0,p2)
beta3<-rep(0,p3)
rho1<-0.5
rho2<-0.5
```

```
rho3<-0.5
A10<-A20<-A30<-A1<-A2<-A3<-A4<-A5<-A6<-Ak11<-Ak12<-Ak21<-Ak22<-Ak31<-Ak32<-0   # Acceptance counter
c<-sample(1:3,n,replace=T)      # random class
n1<-length(c[c==1])
n2<-length(c[c==2])
n3<-n-n1-n2
gamma1<-rep(0,g)
gamma2<-rep(0,g)

# Init. design matrix
kappa<-c(0,0,0,0,0,0) # Init. 6 Cps
xg11<-(t-kappa[1])*(t>kappa[1])*(tx==0)
xg12<-(t-kappa[2])*(t>kappa[2])*(tx==1)
xg21<-(t-kappa[3])*(t>kappa[3])*(tx==0)
xg22<-(t-kappa[4])*(t>kappa[4])*(tx==1)
xg31<-(t-kappa[5])*(t>kappa[5])*(tx==0)
xg32<-(t-kappa[6])*(t>kappa[6])*(tx==1)

xg1<-xg11+xg12
xg2<-xg21+xg22
xg3<-xg31+xg32

X1<-cbind(Xs,xg11,xg12) # class 1
X2<-cbind(Xs,xg21,xg22) # class 2
X3<-cbind(Xs,xg31,xg32) # class 3


# Random Intercept/slope
b1<-rnorm(n,sd=0.1)   # random intercept for all subjects
b2<-rnorm(n,sd=0.1)   # random slope for all subjects
b3<-rnorm(n,sd=0.1)   # second random slope for all subjects

b11<-b1[c==1]
b12<-b2[c==1]
b13<-b3[c==1]
b21<-b1[c==2]
b22<-b2[c==2]
b23<-b3[c==2]
b31<-b1[c==3]
b32<-b2[c==3]
b33<-b3[c==3]

Bmat1<-cbind(b11,b12,13)
sigmab1<-cov(Bmat1)

Bmat2<-cbind(b21,b22,b23)
sigmab2<-cov(Bmat2)

Bmat3<-cbind(b31,b32,b33)
sigmab3<-cov(Bmat3)


#################
```

```
# Store Samples #
#################
nsim<-80000                   # Number of MCMC Iterations
thin<-1                           # Thinnisng interval
burn<-50000                   # Burnisn
lastit<-(nsim-burn)/thin        # Last stored value
Beta1tmp<-matrix(NA,nsim,p1)
Beta1<-matrix(NA,lastit,p1+1)
Beta2tmp<-matrix(NA,nsim,p2)
Beta2<-matrix(NA,lastit,p2+1)
Beta3tmp<-matrix(NA,nsim,p3)
Beta3<-matrix(NA,lastit,p3+1)
Rho1<-rep(NA,lastit)
Rho2<-rep(NA,lastit)
Rho3<-rep(NA,lastit)
Cs<-matrix(NA,lastit,n)     # Indivdual Probs + LC Indicators
Kappa<-matrix(NA,lastit,6)
PI<-matrix(NA,lastit,3)
P1s<-matrix(NA,lastit,n)
P2s<-matrix(NA,lastit,n)
P3s<-matrix(NA,lastit,n)
Gamma1<-matrix(NA,lastit,g)
Gamma2<-matrix(NA,lastit,g)
Sigmab1<-matrix(NA,lastit,q^2)
Sigmab2<-matrix(NA,lastit,q^2)
Sigmab3<-matrix(NA,lastit,q^2)
B1s<-B2s<-B3s<-matrix(NA,lastit,n)
```

MCMC Algorithm (not run)

```
set.seed(1234)
time.start<-proc.time()
for (i in 1:nsim){
  #--------------------#
  # Mult. logit Model #
  #--------------------#
  # Update gamma for category 1
  etagam2<-W%*%gamma2
  c1<-log(1+exp(etagam2))
  eta1<-W%*%gamma1-c1
  w1<-rpg(n,1,eta1)
  u1<-1*(c==1)
  z1<-(u1-1/2)/w1+c1
  v<-solve(crossprod(W*sqrt(w1))+T0g)
  m<-v%*%(T0g%*%gamma0+t(w1*W)%*%z1)
  gamma1<-c(rmvnorm(1,m,v))

  # Update gamma for category 2
  etagam1<-W%*%gamma1
  c2<-log(1+exp(etagam1))
  eta2<-W%*%gamma2-c2
  w2<-rpg(n,1,eta2)
  u2<-1*(c==2)
```

```
z2<-(u2-1/2)/w2+c2
v<-solve(crossprod(W*sqrt(w2))+T0g)
m<-v%*%(T0g%*%gamma0+t(w2*W)%*%z2)
gamma2<-c(rmvnorm(1,m,v))

gamma<-rbind(gamma1,gamma2)
eta<-cbind(W%*%t(gamma),rep(0,n))   # Reference group k=3
pi<-exp(eta)/(1+apply(as.matrix(exp(eta[,-3])),1,sum))  # n x K matrix of cluster pr

#--------------------------------#
# Update latent class hyperparms #
#--------------------------------#
pi1<-pi[,1]
pi2<-pi[,2]
pi3<-pi[,3]
eta1<-beta10+X1%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
eta2<-beta20+X2%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
eta3<-beta30+X3%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
mu1<-1/(1+exp(-eta1))
mu2<-1/(1+exp(-eta2))
mu3<-1/(1+exp(-eta3))

# Update C (latent class indicator)
p1<-pi1*tapply(dbetabinom(y,ntrial,mu1,rho1),id,prod)/
  (pi1*tapply(dbetabinom(y,ntrial,mu1,rho1),id,prod)+
     pi2*tapply(dbetabinom(y,ntrial,mu2,rho2),id,prod)+
     pi3*tapply(dbetabinom(y,ntrial,mu3,rho3),id,prod))

p2<-pi2*tapply(dbetabinom(y,ntrial,mu2,rho2),id,prod)/
  (pi1*tapply(dbetabinom(y,ntrial,mu1,rho1),id,prod)+
     pi2*tapply(dbetabinom(y,ntrial,mu2,rho2),id,prod)+
     pi3*tapply(dbetabinom(y,ntrial,mu3,rho3),id,prod))

p3<-1-p1-p2

P<-matrix(c(p1,p2,p3),n,3)

P[is.na(P)==TRUE]<-rep(1/3,length(P[is.na(P)==TRUE]))
c<-c(rMultinom(P,1))              # 1 draw for each row w.p row(p)

n1<-length(c[c==1])
n2<-length(c[c==2])
n3<-n-n1-n2
C<-rep(c,nis)

nis1<-nis[c==1]
nis2<-nis[c==2]
nis3<-nis[c==3]

#---------------#
# Fixed effects #
#---------------#
```

```
# update beta10
eta<-beta10+X1%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==1],size=ntrial,prob=mu[C==1],rho=rho1,log=T))

beta10new<-beta10+rnorm(1,sd=0.5)
eta<-beta10new+X1%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==1],size=ntrial,prob=mu[C==1],rho=rho1,log=T))

# Acceptance prob on log scale =log(lnew x prior) - log (lold x prior)
r10<-lnew+dnorm(beta10new,0,10,log=T)-(lold+dnorm(beta10,0,10,log=T))
if(log(runif(1))<r10) {
  beta10<-beta10new
  A10<-A10+1
}

# update beta1
eta<-beta10+X1%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==1],size=ntrial,prob=mu[C==1],rho=rho1,log=T))

beta1new<-beta1+rmvnorm(1,sigma=.05*covb1)  # Draw from "symmetric" MV dist
eta<-beta10+X1%*%c(beta1new)+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==1],size=ntrial,prob=mu[C==1],rho=rho1,log=T))


# Acceptance prob on log scale =log(lnew x prior) - log (lold x prior)
r1<-lnew+dmvnorm(beta1new,beta01,V0b1,log=T)-(lold+dmvnorm(beta1,beta01,V0b1,log=T))
if(log(runif(1))<r1) {
  beta1<-c(beta1new)
  A1<-A1+1
}

Beta1tmp[i,]<-beta1
if (i==nsim/2) covb1<-cov(Beta1tmp[(nsim/4+1):nsim/2,])  # Update proposal cov


# update beta20
eta<-beta20+X2%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==2],size=ntrial,prob=mu[C==2],rho=rho2,log=T))

beta20new<-beta20+rnorm(1,sd=0.5)
eta<-beta20new+X2%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==2],size=ntrial,prob=mu[C==2],rho=rho2,log=T))

# Acceptance prob on log scale =log(lnew x prior) - log (lold x prior)
r20<-lnew+dnorm(beta20new,0,10,log=T)-(lold+dnorm(beta20,0,10,log=T))
if(log(runif(1))<r20) {
  beta20<-beta20new
  A20<-A20+1
}


# update beta2
eta<-beta20+X2%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==2],size=ntrial,prob=mu[C==2],rho=rho2,log=T))

beta2new<-beta2+rmvnorm(1,sigma=.05*covb2)  # Draw from "symmetric" MV dist
eta<-beta20+X2%*%c(beta2new)+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==2],size=ntrial,prob=mu[C==2],rho=rho2,log=T))


# Acceptance prob on log scale =log(lnew x prior) - log (lold x prior)
r2<-lnew+dmvnorm(beta2new,beta02,V0b2,log=T)-(lold+dmvnorm(beta2,beta02,V0b2,log=T))
if(log(runif(1))<r2) {
  beta2<-c(beta2new)
  A2<-A2+1
}

Beta2tmp[i,]<-beta2
if (i==nsim/2) covb2<-cov(Beta2tmp[(nsim/4+1):nsim/2,])  # Update proposal cov

# update beta30
eta<-beta30+X3%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==3],size=ntrial,prob=mu[C==3],rho=rho3,log=T))

beta30new<-beta30+rnorm(1,sd=0.5)
eta<-beta30new+X3%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==3],size=ntrial,prob=mu[C==3],rho=rho3,log=T))

# Acceptance prob on log scale =log(lnew x prior) - log (lold x prior)
r30<-lnew+dnorm(beta30new,0,10,log=T)-(lold+dnorm(beta30,0,10,log=T))
if(log(runif(1))<r30) {
  beta30<-beta30new
  A30<-A30+1
}


# update beta3
eta<-beta30+X3%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==3],size=ntrial,prob=mu[C==3],rho=rho3,log=T))

beta3new<-beta3+rmvnorm(1,sigma=.1*covb3)  # Draw from "symmetric" MV dist
eta<-beta30+X3%*%c(beta3new)+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==3],size=ntrial,prob=mu[C==3],rho=rho3,log=T))
```

```
# Acceptance prob on log scale =log(lnew x prior) - log (lold x prior)
r3<-lnew+dmvnorm(beta3new,beta03,V0b3,log=T)-(lold+dmvnorm(beta3,beta03,V0b3,log=T))
if(log(runif(1))<r3) {
  beta3<-c(beta3new)
  A3<-A3+1
}

Beta3tmp[i,]<-beta3
if (i==nsim/2) covb3<-cov(Beta3tmp[(nsim/4+1):nsim/2,])  # Update proposal cov

#--------------------#
# Correlation parm. #
#--------------------#

# update rho1
# Current likelihood
eta<-beta10+X1%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==1],size=ntrial,prob=mu[C==1],rho=rho1,log=T))

# Draw candidate rho and compute likelihood from truncated noraml
rho1new<-rtnorm(1,rho1,sqrt(sigmar0),0,1)           # Draw from truncated normal
lnew<-sum(dbetabinom(y[C==1],size=ntrial,prob=mu[C==1],rho=rho1new,log=T))

# Acceptance prob on log scale
rrho1<-lnew-lold+dtnorm(rho1,rho1new,sqrt(sigmar0),0,1,log=T)-
  dtnorm(rho1new,rho1,sqrt(sigmar0),0,1,log=T)
if(log(runif(1))<rrho1) {
  rho1<-rho1new
  A4<-A4+1
}

# update rho2
# Current likelihood
eta<-beta20+X2%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==2],size=ntrial,prob=mu[C==2],rho=rho2,log=T))

# Draw candidate rho and compute likelihood from truncated noraml
rho2new<-rtnorm(1,rho2,sqrt(sigmar0),0,1)           # Draw from truncated normal
lnew<-sum(dbetabinom(y[C==2],size=ntrial,prob=mu[C==2],rho=rho2new,log=T))

# Acceptance prob on log scale
rrho2<-lnew-lold+dtnorm(rho2,rho2new,sqrt(sigmar0),0,1,log=T)-
  dtnorm(rho2new,rho2,sqrt(sigmar0),0,1,log=T)
if(log(runif(1))<rrho2) {
  rho2<-rho2new
  A5<-A5+1
}

# update rho3
# Current likelihood
eta<-beta30+X3%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
```

```
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==3],size=ntrial,prob=mu[C==3],rho=rho3,log=T))

# Draw candidate rho and compute likelihood from truncated noraml
rho3new<-rtnorm(1,rho3,sqrt(sigmar0),0,1)          # Draw from truncated normal
lnew<-sum(dbetabinom(y[C==3],size=ntrial,prob=mu[C==3],rho=rho3new,log=T))

# Acceptance prob on log scale
rrho3<-lnew-lold+dtnorm(rho3,rho3new,sqrt(sigmar0),0,1,log=T)-
  dtnorm(rho3new,rho3,sqrt(sigmar0),0,1,log=T)
if(log(runif(1))<rrho3) {
  rho3<-rho3new
  A6<-A6+1
}

#-------------------#
# Random-effect var #
#-------------------#

# Update random effect variance
Bmat1<-cbind(b1[c==1],b2[c==1],b3[c==1])
sigmab1<-riwish(d0+n1,C0+crossprod(Bmat1))

Bmat2<-cbind(b1[c==2],b2[c==2],b3[c==2])
sigmab2<-riwish(d0+n2,C0+crossprod(Bmat2))

Bmat3<-cbind(b1[c==3],b2[c==3],b3[c==3])
sigmab3<-riwish(d0+n3,C0+crossprod(Bmat3))

#--------------------------------------------#
# Random effects: intercept/slope/sec. slope #
#--------------------------------------------#

# update b11 (random intercept in class 1)
priorvar<-sigmab1[1,1]-sigmab1[1,-1]%*%solve(sigmab1[-1,-1])%*%sigmab1[-1,1]
priormean<-Bmat1[,-1]%*%t(sigmab1[1,-1]%*%solve(sigmab1[-1,-1]))

b11new<-rnorm(n1,b1[c==1],sqrt(sigmab110))

eta<-beta10+X1[C==1,]%*%beta1+rep(b11new,nis1)+
  rep(b2[c==1],nis1)*t[C==1]+rep(b3[c==1],nis1)*xg1[C==1]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==1],size=ntrial,prob=mu,rho=rho1,log=T),id[C==1],sum)

eta<-beta10+X1[C==1,]%*%beta1+rep(b1[c==1],nis1)+
  rep(b2[c==1],nis1)*t[C==1]+rep(b3[c==1],nis1)*xg1[C==1]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==1],size=ntrial,prob=mu,rho=rho1,log=T),id[C==1],sum)

ratio<-lnew+dnorm(b11new,priormean,sqrt(priorvar),log=T)-(lold+dnorm(b1[c==1],priormean,sqrt(priorvar)
utmp<-1*(log(runif(n1))<ratio)
b1[c==1][utmp==1]<-b11new[utmp==1]


# update b12 (random slope in class 1)
priorvar<-sigmab1[2,2]-sigmab1[2,-2]%*%solve(sigmab1[-2,-2])%*%sigmab1[-2,2]
priormean<-Bmat1[,-2]%*%t(sigmab1[2,-2]%*%solve(sigmab1[-2,-2]))

b12new<-rnorm(n1,b2[c==1],sqrt(sigmab120))

eta<-beta10+X1[C==1,]%*%beta1+rep(b1[c==1],nis1)+
  rep(b12new,nis1)*t[C==1]+rep(b3[c==1],nis1)*xg1[C==1]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==1],size=ntrial,prob=mu,rho=rho1,log=T),id[C==1],sum)

eta<-beta10+X1[C==1,]%*%beta1+rep(b1[c==1],nis1)+
  rep(b2[c==1],nis1)*t[C==1]+rep(b3[c==1],nis1)*xg1[C==1]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==1],size=ntrial,prob=mu,rho=rho1,log=T),id[C==1],sum)

ratio<-lnew+dnorm(b12new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b2[c==1],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n1))<ratio)
b2[c==1][utmp==1]<-b12new[utmp==1]

# update b13 (random slope after CP in class 1)
priorvar<-sigmab1[3,3]-sigmab1[3,-3]%*%solve(sigmab1[-3,-3])%*%sigmab1[-3,3]
priormean<-Bmat1[,-3]%*%t(sigmab1[3,-3]%*%solve(sigmab1[-3,-3]))

b13new<-rnorm(n1,b3[c==1],sqrt(sigmab130))

eta<-beta10+X1[C==1,]%*%beta1+rep(b1[c==1],nis1)+
  rep(b2[c==1],nis1)*t[C==1]+rep(b13new,nis1)*xg1[C==1]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==1],size=ntrial,prob=mu,rho=rho1,log=T),id[C==1],sum)

eta<-beta10+X1[C==1,]%*%beta1+rep(b1[c==1],nis1)+
  rep(b2[c==1],nis1)*t[C==1]+rep(b3[c==1],nis1)*xg1[C==1]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==1],size=ntrial,prob=mu,rho=rho1,log=T),id[C==1],sum)

ratio<-lnew+dnorm(b13new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b3[c==1],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n1))<ratio)
b3[c==1][utmp==1]<-b13new[utmp==1]
```

```
# update b21 (random intercept in class 2)
priorvar<-sigmab2[1,1]-sigmab2[1,-1]%*%solve(sigmab2[-1,-1])%*%sigmab2[-1,1]
priormean<-Bmat2[,-1]%*%t(sigmab2[1,-1]%*%solve(sigmab2[-1,-1]))

b21new<-rnorm(n2,b1[c==2],sqrt(sigmab210))

eta<-beta20+X2[C==2,]%*%beta2+rep(b21new,nis2)+
  rep(b2[c==2],nis2)*t[C==2]+rep(b3[c==2],nis2)*xg2[C==2]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==2],size=ntrial,prob=mu,rho=rho2,log=T),id[C==2],sum)

eta<-beta20+X2[C==2,]%*%beta2+rep(b1[c==2],nis2)+
  rep(b2[c==2],nis2)*t[C==2]+rep(b3[c==2],nis2)*xg2[C==2]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==2],size=ntrial,prob=mu,rho=rho2,log=T),id[C==2],sum)

ratio<-lnew+dnorm(b21new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b1[c==2],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n2))<ratio)
b1[c==2][utmp==1]<-b21new[utmp==1]



# update b22 (random slope in class 2)
priorvar<-sigmab2[2,2]-sigmab2[2,-2]%*%solve(sigmab2[-2,-2])%*%sigmab2[-2,2]
priormean<-Bmat2[,-2]%*%t(sigmab2[2,-2]%*%solve(sigmab2[-2,-2]))

b22new<-rnorm(n2,b2[c==2],sqrt(sigmab220))

eta<-beta20+X2[C==2,]%*%beta2+rep(b1[c==2],nis2)+
  rep(b22new,nis2)*t[C==2]+rep(b3[c==2],nis2)*xg2[C==2]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==2],size=ntrial,prob=mu,rho=rho2,log=T),id[C==2],sum)

eta<-beta20+X2[C==2,]%*%beta2+rep(b1[c==2],nis2)+
  rep(b2[c==2],nis2)*t[C==2]+rep(b3[c==2],nis2)*xg2[C==2]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==2],size=ntrial,prob=mu,rho=rho2,log=T),id[C==2],sum)

ratio<-lnew+dnorm(b22new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b2[c==2],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n2))<ratio)
b2[c==2][utmp==1]<-b22new[utmp==1]

# update b23 (random slope in class 2)
priorvar<-sigmab2[3,3]-sigmab2[3,-3]%*%solve(sigmab2[-3,-3])%*%sigmab2[-3,3]
priormean<-Bmat2[,-3]%*%t(sigmab2[3,-3]%*%solve(sigmab2[-3,-3]))
```

```
b23new<-rnorm(n2,b3[c==2],sqrt(sigmab230))

eta<-beta20+X2[C==2,]%*%beta2+rep(b1[c==2],nis2)+
  rep(b2[c==2],nis2)*t[C==2]+rep(b23new,nis2)*xg2[C==2]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==2],size=ntrial,prob=mu,rho=rho2,log=T),id[C==2],sum)

eta<-beta20+X2[C==2,]%*%beta2+rep(b1[c==2],nis2)+
  rep(b2[c==2],nis2)*t[C==2]+rep(b3[c==2],nis2)*xg2[C==2]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==2],size=ntrial,prob=mu,rho=rho2,log=T),id[C==2],sum)

ratio<-lnew+dnorm(b23new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b3[c==2],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n2))<ratio)
b3[c==2][utmp==1]<-b23new[utmp==1]


# update b31 (random intercept in class 3)
priorvar<-sigmab3[1,1]-sigmab3[1,-1]%*%solve(sigmab3[-1,-1])%*%sigmab3[-1,1]
priormean<-Bmat3[,-1]%*%t(sigmab3[1,-1]%*%solve(sigmab3[-1,-1]))

b31new<-rnorm(n3,b1[c==3],sqrt(sigmab310))

eta<-beta30+X3[C==3,]%*%beta3+rep(b31new,nis3)+
  rep(b2[c==3],nis3)*t[C==3]+rep(b3[c==3],nis3)*xg3[C==3]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==3],size=ntrial,prob=mu,rho=rho3,log=T),id[C==3],sum)

eta<-beta30+X3[C==3,]%*%beta3+rep(b1[c==3],nis3)+
  rep(b2[c==3],nis3)*t[C==3]+rep(b3[c==3],nis3)*xg3[C==3]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==3],size=ntrial,prob=mu,rho=rho3,log=T),id[C==3],sum)

ratio<-lnew+dnorm(b31new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b1[c==3],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n3))<ratio)
b1[c==3][utmp==1]<-b31new[utmp==1]


# update b32 (random slope in class 3)
priorvar<-sigmab3[2,2]-sigmab3[2,-2]%*%solve(sigmab3[-2,-2])%*%sigmab3[-2,2]
priormean<-Bmat3[,-2]%*%t(sigmab3[2,-2]%*%solve(sigmab3[-2,-2]))

b32new<-rnorm(n3,b2[c==3],sqrt(sigmab320))

eta<-beta30+X3[C==3,]%*%beta3+rep(b1[c==3],nis3)+
  rep(b32new,nis3)*t[C==3]+rep(b3[c==3],nis3)*xg3[C==3]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==3],size=ntrial,prob=mu,rho=rho3,log=T),id[C==3],sum)

eta<-beta30+X3[C==3,]%*%beta3+rep(b1[c==3],nis3)+
  rep(b2[c==3],nis3)*t[C==3]+rep(b3[c==3],nis3)*xg3[C==3]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==3],size=ntrial,prob=mu,rho=rho3,log=T),id[C==3],sum)

ratio<-lnew+dnorm(b32new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b2[c==3],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n3))<ratio)
b2[c==3][utmp==1]<-b32new[utmp==1]

# update b33 (random slope in class 3)
priorvar<-sigmab3[3,3]-sigmab3[3,-3]%*%solve(sigmab3[-3,-3])%*%sigmab3[-3,3]
priormean<-Bmat3[,-3]%*%t(sigmab3[3,-3]%*%solve(sigmab3[-3,-3]))

b33new<-rnorm(n3,b3[c==3],sqrt(sigmab330))

eta<-beta30+X3[C==3,]%*%beta3+rep(b1[c==3],nis3)+
  rep(b2[c==3],nis3)*t[C==3]+rep(b33new,nis3)*xg3[C==3]
mu<-1/(1+exp(-eta))

lnew<-tapply(dbetabinom(y[C==3],size=ntrial,prob=mu,rho=rho3,log=T),id[C==3],sum)

eta<-beta30+X3[C==3,]%*%beta3+rep(b1[c==3],nis3)+
  rep(b2[c==3],nis3)*t[C==3]+rep(b3[c==3],nis3)*xg3[C==3]
mu<-1/(1+exp(-eta))

lold<-tapply(dbetabinom(y[C==3],size=ntrial,prob=mu,rho=rho3,log=T),id[C==3],sum)

ratio<-lnew+dnorm(b33new,priormean,sqrt(priorvar),log=T)-
  (lold+dnorm(b3[c==3],priormean,sqrt(priorvar),log=T))

utmp<-1*(log(runif(n3))<ratio)
b3[c==3][utmp==1]<-b33new[utmp==1]

#--------------#
# Changepoints #
#--------------#

# update CP in the class 1 (Control gp)
kappa11new<-rtnorm(1,kappa[1],sqrt(sigkap11),L0,U0)
xg11new<-(t-kappa11new)*(t>kappa11new)*(tx==0)
xg1new<-xg11new+xg12
X1new<-cbind(Xs,xg11new,xg12)

eta<-beta10+X1new%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1new
mu<-1/(1+exp(-eta))
```

```r
lnew<-sum(dbetabinom(y[C==1 & tx==0],ntrial,mu[C==1 & tx==0],rho1,log=T))

eta<-beta10+X1%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==1 & tx==0],ntrial,mu[C==1 & tx==0],rho1,log=T))

rk11<-lnew-lold+dtnorm(kappa[1],kappa11new,sqrt(sigkap11),L0,U0,log=T)-
  dtnorm(kappa11new,kappa[1],sqrt(sigkap11),L0,U0,log=T)

if(log(runif(1))<rk11) {
  kappa[1]<-kappa11new
  xg11<-xg11new
  xg1<-xg1new
  X1<-X1new
  Ak11<-Ak11+1
}


# update CP in the class 1 (Tx group)
kappa12new<-rtnorm(1,kappa[2],sqrt(sigkap12),L0,U0)
xg12new<-(t-kappa12new)*(t>kappa12new)*(tx==1)
xg1new<-xg11+xg12new
X1new<-cbind(Xs,xg11,xg12new)

eta<-beta10+X1new%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1new
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==1 & tx==1],ntrial,mu[C==1 & tx==1],rho1,log=T))

eta<-beta10+X1%*%beta1+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg1
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==1 & tx==1],ntrial,mu[C==1 & tx==1],rho1,log=T))

rk12<-lnew-lold+dtnorm(kappa[2],kappa12new,sqrt(sigkap12),L0,U0,log=T)-
  dtnorm(kappa12new,kappa[2],sqrt(sigkap12),L0,U0,log=T)

if(log(runif(1))<rk12) {
  kappa[2]<-kappa12new
  xg12<-xg12new
  xg1<-xg1new
  X1<-X1new
  Ak12<-Ak12+1
}

# update CP in the class 2 (Control gp)
kappa21new<-rtnorm(1,kappa[3],sqrt(sigkap21),L0,U0)
xg21new<-(t-kappa21new)*(t>kappa21new)*(tx==0)
xg2new<-xg21new+xg22
X2new<-cbind(Xs,xg21new,xg22)

eta<-beta20+X2new%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2new
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==2 & tx==0],ntrial,mu[C==2 & tx==0],rho2,log=T))
```

```
eta<-beta20+X2%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==2 & tx==0],ntrial,mu[C==2 & tx==0],rho2,log=T))

rk21<-lnew-lold+dtnorm(kappa[3],kappa21new,sqrt(sigkap21),L0,U0,log=T)-
  dtnorm(kappa21new,kappa[3],sqrt(sigkap21),L0,U0,log=T)

if(log(runif(1))<rk21) {
  kappa[3]<-kappa21new
  xg21<-xg21new
  xg2<-xg2new
  X2<-X2new
  Ak21<-Ak21+1
}


# update CP in the class 2 (Tx group)
kappa22new<-rtnorm(1,kappa[4],sqrt(sigkap22),L0,U0)
xg22new<-(t-kappa22new)*(t>kappa22new)*(tx==1)
xg2new<-xg21+xg22new
X2new<-cbind(Xs,xg21,xg22new)

eta<-beta20+X2new%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2new
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==2 & tx==1],ntrial,mu[C==2 & tx==1],rho2,log=T))

eta<-beta20+X2%*%beta2+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg2
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==2 & tx==1],ntrial,mu[C==2 & tx==1],rho2,log=T))

rk22<-lnew-lold+dtnorm(kappa[4],kappa22new,sqrt(sigkap22),L0,U0,log=T)-
  dtnorm(kappa22new,kappa[4],sqrt(sigkap22),L0,U0,log=T)

if(log(runif(1))<rk22) {
  kappa[4]<-kappa22new
  xg22<-xg22new
  xg2<-xg2new
  X2<-X2new
  Ak22<-Ak22+1
}

# update CP in the class 3 (Control gp)
kappa31new<-rtnorm(1,kappa[5],sqrt(sigkap31),L0,U0)
xg31new<-(t-kappa31new)*(t>kappa31new)*(tx==0)
xg3new<-xg31new+xg32
X3new<-cbind(Xs,xg31new,xg32)

eta<-beta30+X3new%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3new
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==3 & tx==0],ntrial,mu[C==3 & tx==0],rho3,log=T))

eta<-beta30+X3%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
mu<-1/(1+exp(-eta))
```

```r
lold<-sum(dbetabinom(y[C==3 & tx==0],ntrial,mu[C==3 & tx==0],rho3,log=T))

rk31<-lnew-lold+dtnorm(kappa[5],kappa31new,sqrt(sigkap31),L0,U0,log=T)-
  dtnorm(kappa31new,kappa[5],sqrt(sigkap31),L0,U0,log=T)

if(log(runif(1))<rk31) {
  kappa[5]<-kappa31new
  xg31<-xg31new
  xg3<-xg3new
  X3<-X3new
  Ak31<-Ak31+1
}


# update CP in the class 3 (Tx group)
kappa32new<-rtnorm(1,kappa[6],sqrt(sigkap32),L0,U0)
xg32new<-(t-kappa32new)*(t>kappa32new)*(tx==1)
xg3new<-xg31+xg32new
X3new<-cbind(Xs,xg31,xg32new)

eta<-beta30+X3new%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3new
mu<-1/(1+exp(-eta))
lnew<-sum(dbetabinom(y[C==3 & tx==1],ntrial,mu[C==3 & tx==1],rho3,log=T))

eta<-beta30+X3%*%beta3+rep(b1,nis)+rep(b2,nis)*t+rep(b3,nis)*xg3
mu<-1/(1+exp(-eta))
lold<-sum(dbetabinom(y[C==3 & tx==1],ntrial,mu[C==3 & tx==1],rho3,log=T))

rk32<-lnew-lold+dtnorm(kappa[6],kappa32new,sqrt(sigkap32),L0,U0,log=T)-
  dtnorm(kappa32new,kappa[6],sqrt(sigkap32),L0,U0,log=T)

if(log(runif(1))<rk32) {
  kappa[6]<-kappa32new
  xg32<-xg32new
  xg3<-xg3new
  X3<-X3new
  Ak32<-Ak32+1
}


#################
# Store Results #
#################
if (i> burn & i%%thin==0) {
  j<-(i-burn)/thin
  Beta1[j,]<-c(beta10,beta1)
  Beta2[j,]<-c(beta20,beta2)
  Beta3[j,]<-c(beta30,beta3)
  Rho1[j]<-rho1
  Rho2[j]<-rho2
  Rho3[j]<-rho3
  Cs[j,]<-c
  Gamma1[j,]<-gamma1
```

```
    Gamma2[j,]<-gamma2
    Kappa[j,]<-kappa
    P1s[j,]<-p1
    P2s[j,]<-p2
    P3s[j,]<-p3
    Sigmab1[j,]<-c(sigmab1)
    Sigmab2[j,]<-c(sigmab2)
    Sigmab3[j,]<-c(sigmab3)
    B1s[j,]<-b1
    B2s[j,]<-b2
    B3s[j,]<-b3
  }

  if (i%%5==0) {
    print(i)
    print(c(true.beta10,true.beta1))
    print(c(beta10,beta1))
    print(c(true.beta20,true.beta2))
    print(c(beta20,beta2))
    print(c(true.beta30,true.beta3))
    print(c(beta30,beta3))
    print(true.kappa)
    print(kappa)
    print(c(true.gamma1,true.gamma2))
    print(c(gamma1,gamma2))
    print(sigmab1)
    print(sigmab2)
    print(sigmab3)
    print(table(c,true.c))
  }

}
(time.tol<-proc.time()-time.start)
```

Save/Load MCMC Samples

```
#------------------------------------------------------------------------------#
# Save/Load MCMC Samples #
#------------------------#
# samples<-list(Beta1=Beta1,
#               Beta2=Beta2,
#               Beta3=Beta3,
#               Rho1=Rho1,
#               Rho2=Rho2,
#               Rho3=Rho3,
#               Gamma1=Gamma1,
#               Gamma2=Gamma2,
#               Sigmab1=Sigmab1,
#               Sigmab2=Sigmab2,
#               Sigmab3=Sigmab3,
#               Kappa=Kappa,
#               Cs=Cs,
#               P1s=P1s,
#               P2s=P2s,
#               P3s=P3s,
#               B1s=B1s,
#               B2s=B2s,
#               B3s=B3s)

dir.sav<-"C:\\Users\\chech\\OneDrive - Medical University of South Carolina\\Research\\BB Mixture Model
# save(samples,file=paste(dir.sav,"Simulation_bbmixcp.Rda",sep=""))
load(file=paste(dir.sav,"Simulation_bbmixcp.Rda",sep=""))

Beta1=samples$Beta1
Beta2=samples$Beta2
Beta3=samples$Beta3
Rho1=samples$Rho1
Rho2=samples$Rho2
Rho3=samples$Rho3
Gamma1=samples$Gamma1
Gamma2=samples$Gamma2
Sigmab1=samples$Sigmab1
Sigmab2=samples$Sigmab2
Sigmab3=samples$Sigmab3
Kappa=samples$Kappa
Cs=samples$Cs
P1s=samples$P1s
P2s=samples$P2s
P3s=samples$P3s
B1s=samples$B1s
B2s=samples$B2s
B3s=samples$B3s
```

Label switching (not run)

```
#------------------------------------------------------------------------------#
# Label Switching #
#-----------------#
library(label.switching)

ls<-label.switching(method=c("ECR"),
                    zpivot=Cs[c(216,70),],z = Cs,K = 3)

#------#
# Beta #
#------#

mcmc.Beta<-array(c(rbind(Beta1,Beta2,Beta3)),dim=c(lastit,3,5))


matplot(permute.mcmc(mcmc.Beta,ls$permutations$"ECR-1")$output[,,1],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(beta[0]))
matplot(permute.mcmc(mcmc.Beta,ls$permutations$"ECR-1")$output[,,2],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(beta[1]))
matplot(permute.mcmc(mcmc.Beta,ls$permutations$"ECR-1")$output[,,3],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(beta[2]))
```

```
matplot(permute.mcmc(mcmc.Beta,ls$permutations$"ECR-1")$output[,,4],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(beta[3]))
matplot(permute.mcmc(mcmc.Beta,ls$permutations$"ECR-1")$output[,,5],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(beta[4]))


#-------------#
# Rho + Kappa #
#-------------#

Rho<-cbind(Rho1,Rho2,Rho3)


mcmc.par<-array(dim=c(lastit,3,3))
mcmc.par[,,1]<-Rho
mcmc.par[,,2]<-Kappa[,c(1,3,5)]
mcmc.par[,,3]<-Kappa[,c(2,4,6)]


matplot(permute.mcmc(mcmc.par,ls$permutations$"ECR-1")$output[,,1],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(rho))
matplot(permute.mcmc(mcmc.par,ls$permutations$"ECR-1")$output[,,2],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(kappa[1]))
matplot(permute.mcmc(mcmc.par,ls$permutations$"ECR-1")$output[,,2],type="l",
        xlab="iteration",main="ECR (1st pivot)",ylab =expression(kappa[2]))
```

Results

```
# Results
mbeta1<-colMeans(Beta1)
qbeta1<-apply(Beta1,2,quantile,c(0.025,0.975))
mbeta2<-colMeans(Beta2)
qbeta2<-apply(Beta2,2,quantile,c(0.025,0.975))
mbeta3<-colMeans(Beta3)
qbeta3<-apply(Beta3,2,quantile,c(0.025,0.975))

mrho1<-mean(Rho1)
qrho1<-quantile(Rho1,c(0.025,0.975))
mrho2<-mean(Rho2)
qrho2<-quantile(Rho2,c(0.025,0.975))
mrho3<-mean(Rho3)
qrho3<-quantile(Rho3,c(0.025,0.975))

mkap<-colMeans(Kappa)
qkap<-apply(Kappa,2,quantile,c(0.025,0.975))

msigmab1<-colMeans(Sigmab1)
qsigmab1<-apply(Sigmab1,2,quantile,c(0.025,0.975))

msigmab2<-colMeans(Sigmab2)
qsigmab2<-apply(Sigmab2,2,quantile,c(0.025,0.975))

msigmab3<-colMeans(Sigmab3)
```

```
qsigmab3<-apply(Sigmab3,2,quantile,c(0.025,0.975))

mgamma1<-colMeans(Gamma1)
qgamma1<-apply(Gamma1,2,quantile,c(0.025,0.975))
mgamma2<-colMeans(Gamma2)
qgamma2<-apply(Gamma2,2,quantile,c(0.025,0.975))


c(true.beta10,true.beta1)
```

```
## [1] -2.15  0.35 -0.50  0.30 -0.95 -0.75
```

```
mbeta1
```

```
## [1] -2.1596034  0.4063089 -0.6805026  0.3101057 -1.0998630 -1.0756436
```

```
qbeta1
```

```
##            [,1]      [,2]        [,3]       [,4]       [,5]       [,6]
## 2.5%  -2.685318 0.1229546 -1.36155839 -0.1008205 -1.5546479 -1.4728374
## 97.5% -1.651959 0.7306093  0.01883402  0.7389754 -0.6638178 -0.7029726
```

```
c(true.beta20,true.beta2)
```

```
## [1]  0.40  0.25  0.65  0.15 -0.75 -0.60
```

```
mbeta2
```

```
## [1]  0.57364776  0.33336300  0.41202968  0.04340568 -0.82911692 -0.49281241
```

```
qbeta2
```

```
##            [,1]      [,2]       [,3]       [,4]       [,5]      [,6]
## 2.5%  0.1355096 0.1122630 -0.5686688 -0.2805500 -1.1500904 -0.739473
## 97.5% 1.4639433 0.6167213  0.9575524  0.3223428 -0.5372002 -0.261294
```

```
c(true.beta30,true.beta3)
```

```
## [1]  1.85 -0.65 -0.30 -0.25  1.25  1.55
```

```
mbeta3
```

```
## [1]  1.7325029 -0.5786790  0.1081196 -0.4776618  1.1598116  1.7672218
```

```
qbeta3
```

```
##            [,1]       [,2]       [,3]        [,4]      [,5]     [,6]
## 2.5%  0.9554192 -1.0266317 -0.6267593 -0.96988788 0.6341991 1.318369
## 97.5% 2.4034874 -0.1872149  0.9629639  0.05367751 1.7029667 2.230436
```

```
truerho1
```

```
## [1] 0.2
```

```
mrho1
```

```
## [1] 0.2584947
```

```
qrho1
```

```
##      2.5%     97.5%
## 0.1986003 0.3251143
```

```r
truerho2
```

```
## [1] 0.15
```

```r
mrho2
```

```
## [1] 0.111947
```

```r
qrho2
```

```
##       2.5%      97.5% 
## 0.07635195 0.14959976
```

```r
truerho3
```

```
## [1] 0.35
```

```r
mrho3
```

```
## [1] 0.3467842
```

```r
qrho3
```

```
##      2.5%     97.5% 
## 0.2813869 0.4151928
```

```r
true.kappa+5
```

```
## [1] 7 7 5 6 6 8
```

```r
mkap+5
```

```
## [1] 7.032032 7.302580 4.821537 5.931274 6.050066 7.812290
```

```r
qkap+5
```

```
##           [,1]     [,2]     [,3]     [,4]     [,5]     [,6]
## 2.5%  5.704464 6.375126 4.084279 5.173086 4.905656 7.318242
## 97.5% 8.114194 8.335112 5.443013 6.654941 7.611106 8.266948
```

```r
truesigmab1
```

```
##      [,1] [,2] [,3]
## [1,] 0.50 0.15 0.10
## [2,] 0.15 0.25 0.05
## [3,] 0.10 0.05 0.10
```

```r
msigmab1
```

```
## [1]  0.273383799  0.049493749  0.009890524  0.049493749  0.277005513
## [6] -0.002309905  0.009890524 -0.002309905  0.223837889
```

```r
qsigmab1
```

```
##            [,1]        [,2]       [,3]        [,4]      [,5]       [,6]
## 2.5%  0.1092469 -0.08803465 -0.1316687 -0.08803465 0.1564469 -0.1366369
## 97.5% 0.5762047  0.19410260  0.1463547  0.19410260 0.4512921  0.1181061
##             [,7]       [,8]       [,9]
## 2.5%  -0.1316687 -0.1366369 0.09612847
## 97.5%  0.1463547  0.1181061 0.43471049
```

```r
truesigmab2
```

```
##      [,1] [,2] [,3]
## [1,] 0.20 0.05 0.05
## [2,] 0.05 0.15 0.05
## [3,] 0.05 0.05 0.10
```

```
msigmab2
```

```
## [1]  0.269203921 -0.016284626 -0.033516543 -0.016284626  0.181117844
## [6]  0.002117587 -0.033516543  0.002117587  0.186934363
```

```
qsigmab2
```

```
##            [,1]        [,2]        [,3]        [,4]      [,5]        [,6]
## 2.5%  0.1195922 -0.11134630 -0.15508666 -0.11134630 0.1059829 -0.08129388
## 97.5% 0.5285508  0.07644354  0.06383251  0.07644354 0.2914647  0.07105591
##              [,7]        [,8]       [,9]
## 2.5%  -0.15508666 -0.08129388 0.09358608
## 97.5%  0.06383251  0.07105591 0.33031657
```

```
truesigmab3
```

```
##      [,1]  [,2]  [,3]
## [1,] 0.15  0.05  0.06
## [2,] 0.05  0.15 -0.04
## [3,] 0.06 -0.04  0.10
```

```
msigmab3
```

```
## [1]  0.330672106  0.003284934 -0.015727173  0.003284934  0.161289096
## [6] -0.019848222 -0.015727173 -0.019848222  0.241466025
```

```
qsigmab3
```

```
##            [,1]       [,2]       [,3]       [,4]       [,5]        [,6]
## 2.5%  0.1275295 -0.1102541 -0.1771025 -0.1102541 0.08570479 -0.13564313
## 97.5% 0.6759391  0.1087623  0.1403035  0.1087623 0.27759266  0.06617912
##             [,7]        [,8]      [,9]
## 2.5%  -0.1771025 -0.13564313 0.1000480
## 97.5%  0.1403035  0.06617912 0.5171723
```

```
true.gamma1
```

```
## [1] -0.5  0.5 -0.5
```

```
mgamma1
```

```
## [1] -0.5615652  0.6152190 -0.7143338
```

```
qgamma1
```

```
##              [,1]       [,2]       [,3]
## 2.5%  -1.24118043 -0.1217305 -1.1887382
## 97.5%  0.07060882  1.3647706 -0.2873936
```

```
true.gamma2
```

```
## [1]  0.25 -1.00  0.50
```

```
mgamma2
```

```
## [1]  0.5323925 -1.3549421  0.3576034
```

```r
qgamma2
```

```
##              [,1]      [,2]        [,3]
## 2.5%  0.02984561 -2.159617 -0.05566537
## 97.5% 1.06100860 -0.600257  0.80137453
```

```r
# Class
chat<-round(colMeans(Cs))
table(chat,true.c)
```

```
##     true.c
## chat  1  2  3
##    1 73  2  0
##    2  7 78 10
##    3  0  2 78
```

```r
# Class proportion
table(true.c)/n
```

```
## true.c
##     1     2     3 
## 0.320 0.328 0.352
```

```r
Cstmp<-t(apply(Cs,1,table)/n)
colMeans(Cstmp)
```

```
##         1         2         3 
## 0.3156067 0.3531007 0.3312927
```

```r
apply(Cstmp,2,quantile,c(.025,.975))
```

```
##           1     2     3
## 2.5%  0.284 0.312 0.300
## 97.5% 0.344 0.396 0.364
```

Figures

Anbsinent trends b/t group in each class

```
#------------------------------------------------------------------------------#
# Population-Average Figures + Difference plot + Selected subject trajectories #
#------------------------------------------------------------------------------#
num=20
gridind<-seq(-4,7,length.out=num)

trt<-tx[!duplicated(id)]

tp<-rep(gridind,n)
nisp<-rep(num,n)
txp<-rep(trt,nisp)

idp<-rep(1:n,eac=num)

#-------------------#
# True Trajectories #
#-------------------#w

#---------#
# Class 1 #
#---------#

X1p<-cbind(1,gridind,0,0,(gridind-true.kappa[1])*(gridind>true.kappa[1]),0)
X1t<-cbind(1,gridind,1,gridind,0,(gridind-true.kappa[2])*(gridind>true.kappa[2]))

n1p<-sum(trt[true.c==1]==0) # number of subject in class 1 (placebo)
n1t<-sum(trt[true.c==1]==1) # number of subject in class 1 (tx)


Y1p<-matrix(NA,n1p,num)     # Observed subject-level trend in class 1 (placebo)
Y1t<-matrix(NA,n1t,num)     # Observed subject-level trend in class 1 (tx)

# Class 1 - Placebo #
for(l in 1:n1p){
  eta1p<-X1p%*%c(true.beta10,true.beta1)+(trueb1[trt[true.c==1]==0,1])[l]+
    (trueb1[trt[true.c==1]==0,2])[l]*gridind+
    (trueb1[trt[true.c==1]==0,3])[l]*(gridind-true.kappa[1])*(gridind>true.kappa[1])
  y1p<-1/(1+exp(-eta1p))*7

  Y1p[l,]<-y1p
}

# Class 1 - Tx #
for(l in 1:n1t){
  eta1t<-X1t%*%c(true.beta10,true.beta1)+(trueb1[trt[true.c==1]==1,1])[l]+
    (trueb1[trt[true.c==1]==1,2])[l]*gridind+
    (trueb1[trt[true.c==1]==1,3])[l]*(gridind-true.kappa[2])*(gridind>true.kappa[2])
  y1t<-1/(1+exp(-eta1t))*7

  Y1t[l,]<-y1t
```

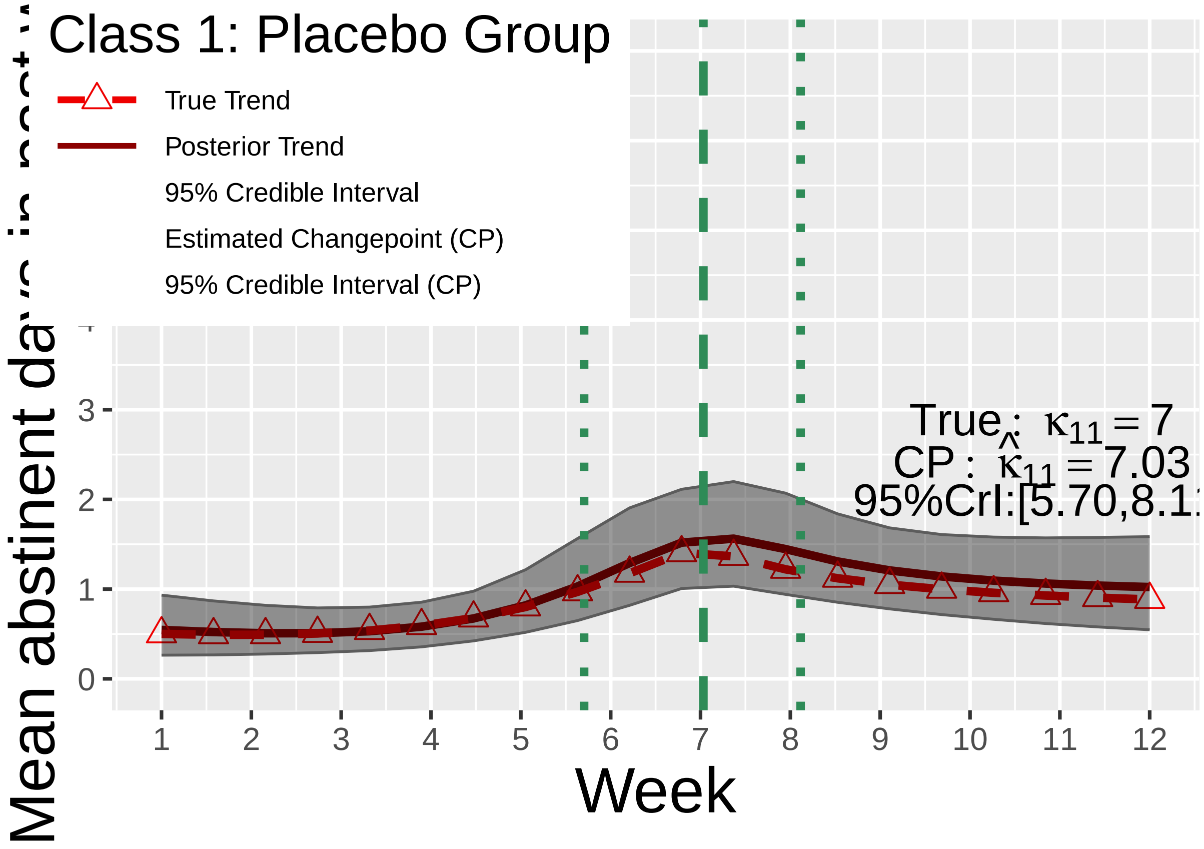
```r
}


my1p<-colMeans(Y1p)
my1t<-colMeans(Y1t)


#---------#
# Class 2 #
#---------#

X2p<-cbind(1,gridind,0,0,(gridind-true.kappa[3])*(gridind>true.kappa[3]),0)
X2t<-cbind(1,gridind,1,gridind,0,(gridind-true.kappa[4])*(gridind>true.kappa[4]))

n2p<-sum(trt[true.c==2]==0) # number of subject in class 2 (placebo)
n2t<-sum(trt[true.c==2]==1) # number of subject in class 2 (tx)


Y2p<-matrix(NA,n2p,num)     # Observed subject-level trend in class 2 (placebo)
Y2t<-matrix(NA,n2t,num)     # Observed subject-level trend in class 2 (tx)

# Class 2 - Placebo #
for(l in 1:n2p){
  eta2p<-X2p%*%c(true.beta20,true.beta2)+(trueb2[trt[true.c==2]==0,1])[l]+
    (trueb2[trt[true.c==2]==0,2])[l]*gridind+
    (trueb2[trt[true.c==2]==0,3])[l]*(gridind-true.kappa[3])*(gridind>true.kappa[3])
  y2p<-1/(1+exp(-eta2p))*7

  Y2p[l,]<-y2p
}

# Class 2 - Tx #
for(l in 1:n2t){
  eta2t<-X2t%*%c(true.beta20,true.beta2)+(trueb2[trt[true.c==2]==1,1])[l]+
    (trueb2[trt[true.c==2]==1,2])[l]*gridind+
    (trueb2[trt[true.c==2]==1,3])[l]*(gridind-true.kappa[4])*(gridind>true.kappa[4])
  y2t<-1/(1+exp(-eta2t))*7

  Y2t[l,]<-y2t
}


my2p<-colMeans(Y2p)
my2t<-colMeans(Y2t)

#---------#
# Class 3 #
#---------#

X3p<-cbind(1,gridind,0,0,(gridind-true.kappa[5])*(gridind>true.kappa[5]),0)
X3t<-cbind(1,gridind,1,gridind,0,(gridind-true.kappa[6])*(gridind>true.kappa[6]))

n3p<-sum(trt[true.c==3]==0) # number of subject in class 3 (placebo)
```

```
n3t<-sum(trt[true.c==3]==1) # number of subject in class 3 (tx)


Y3p<-matrix(NA,n3p,num)     # Observed subject-level trend in class 3 (placebo)
Y3t<-matrix(NA,n3t,num)     # Observed subject-level trend in class 3 (tx)

# Class 3 - Placebo #
for(l in 1:n3p){
  eta3p<-X3p%*%c(true.beta30,true.beta3)+(trueb3[trt[true.c==3]==0,1])[l]+
    (trueb3[trt[true.c==3]==0,2])[l]*gridind+
    (trueb3[trt[true.c==3]==0,3])[l]*(gridind-true.kappa[5])*(gridind>true.kappa[5])
  y3p<-1/(1+exp(-eta3p))*7

  Y3p[l,]<-y3p
}

# Class 3 - Tx #
for(l in 1:n3t){
  eta3t<-X3t%*%c(true.beta30,true.beta3)+(trueb3[trt[true.c==3]==1,1])[l]+
    (trueb3[trt[true.c==3]==1,2])[l]*gridind+
    (trueb3[trt[true.c==3]==1,3])[l]*(gridind-true.kappa[6])*(gridind>true.kappa[6])
  y3t<-1/(1+exp(-eta3t))*7

  Y3t[l,]<-y3t
}


my3p<-colMeans(Y3p)
my3t<-colMeans(Y3t)

#--------#
# Fitted #
#--------#
YPOSc1pl<-array(0,dim=c(lastit,num))  # Class 1: Placebo
YPOSc1tx<-array(0,dim=c(lastit,num))  # Class 1: Tx

YPOSc2pl<-array(0,dim=c(lastit,num))  # Class 2: Placebo
YPOSc2tx<-array(0,dim=c(lastit,num))  # Class 2: Tx

YPOSc3pl<-array(0,dim=c(lastit,num))  # Class 3: Placebo
YPOSc3tx<-array(0,dim=c(lastit,num))  # Class 3: Tx

YPOSc1diff<-array(0,dim=c(lastit,num)) # Difference in abs.: class 1
YPOSc2diff<-array(0,dim=c(lastit,num)) # Difference in abs.: class 2
YPOSc3diff<-array(0,dim=c(lastit,num)) # Difference in abs.: class 2



for (j in 1:lastit){
  beta1<-Beta1[j,]
  beta2<-Beta2[j,]
  beta3<-Beta3[j,]
  kappa11<-Kappa[j,1]
```

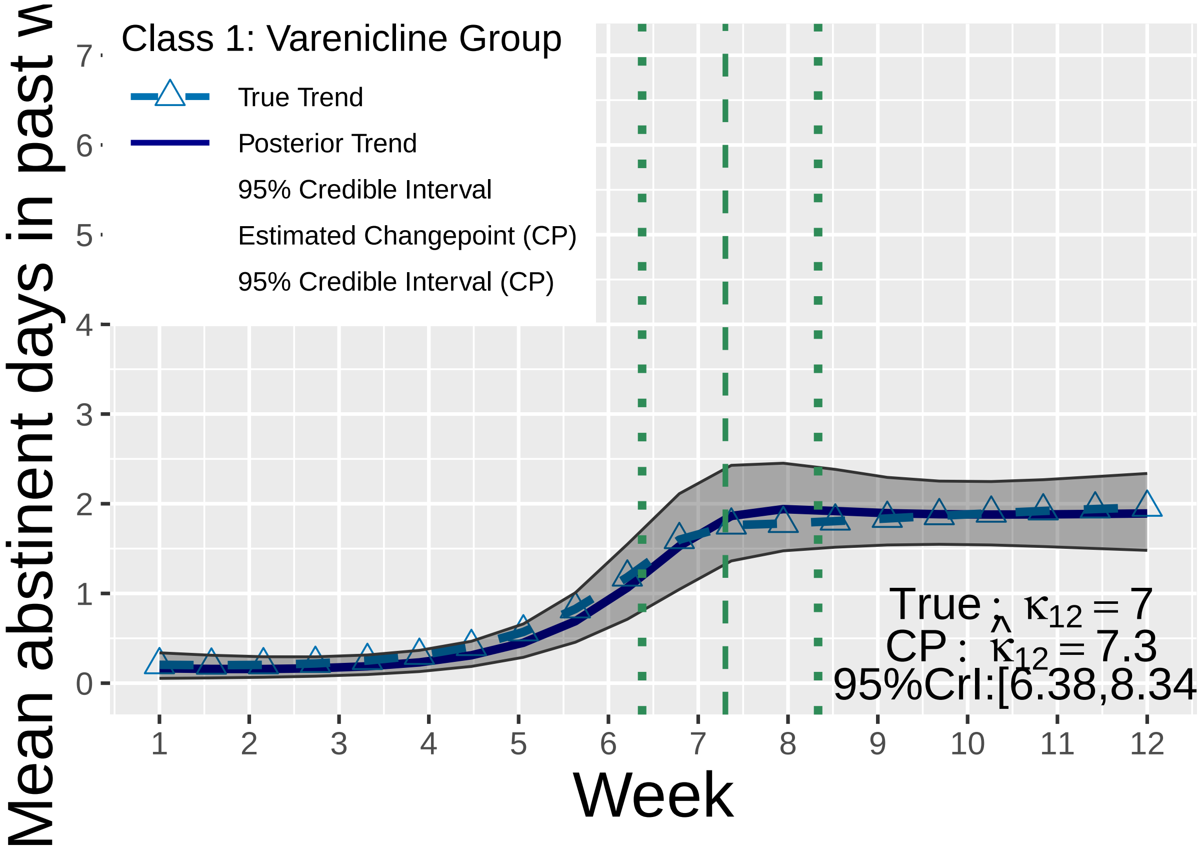
```r
kappa12<-Kappa[j,2]
kappa21<-Kappa[j,3]
kappa22<-Kappa[j,4]
kappa31<-Kappa[j,5]
kappa32<-Kappa[j,6]
c<-Cs[j,]
C<-rep(c,each=num)
nc1pl<-sum(c==1&trt==0)   # Number of subj at class 1 & Placebo
nc1tx<-sum(c==1&trt==1)   # Number of subj at class 1 & Tx
nc2pl<-sum(c==2&trt==0)   # Number of subj at class 2 & Placebo
nc2tx<-sum(c==2&trt==1)   # Number of subj at class 2 & Tx
nc3pl<-sum(c==3&trt==0)   # Number of subj at class 3 & Placebo
nc3tx<-sum(c==3&trt==1)   # Number of subj at class 3 & Tx

b1<-B1s[j,]
b2<-B2s[j,]
b3<-B3s[j,]

spg11<-(tp-kappa11)*(tp>kappa11)*(txp==0)
spg12<-(tp-kappa12)*(tp>kappa12)*(txp==1)
spg21<-(tp-kappa21)*(tp>kappa21)*(txp==0)
spg22<-(tp-kappa22)*(tp>kappa22)*(txp==1)
spg31<-(tp-kappa31)*(tp>kappa31)*(txp==0)
spg32<-(tp-kappa32)*(tp>kappa32)*(txp==1)

spg1<-spg11+spg12
spg2<-spg21+spg22
spg3<-spg31+spg32

X1<-cbind(1,tp,txp,tp*txp,spg11,spg12)
X2<-cbind(1,tp,txp,tp*txp,spg21,spg22)
X3<-cbind(1,tp,txp,tp*txp,spg31,spg32)

etac1pl<-X1[C==1 & txp==0,]%*%beta1+rep(b1[c==1&trt==0],each=num)+
  rep(b2[c==1&trt==0],each=num)*rep(gridind,nc1pl)+rep(b3[c==1&trt==0],each=num)*spg1[C==1&txp==0]
muc1pl<-1/(1+exp(-etac1pl))   # Predicted mean: class 1 placebo

etac1tx<-X1[C==1 & txp==1,]%*%beta1+rep(b1[c==1&trt==1],each=num)+
  rep(b2[c==1&trt==1],each=num)*rep(gridind,nc1tx)+rep(b3[c==1&trt==1],each=num)*spg1[C==1&txp==1]
muc1tx<-1/(1+exp(-etac1tx))   # Predicted mean: class 1 tx

etac2pl<-X2[C==2 & txp==0,]%*%beta2+rep(b1[c==2&trt==0],each=num)+
  rep(b2[c==2&trt==0],each=num)*rep(gridind,nc2pl)+rep(b3[c==2&trt==0],each=num)*spg2[C==2&txp==0]
muc2pl<-1/(1+exp(-etac2pl))   # Predicted mean: class 2 placebo

etac2tx<-X2[C==2 & txp==1,]%*%beta2+rep(b1[c==2&trt==1],each=num)+
  rep(b2[c==2&trt==1],each=num)*rep(gridind,nc2tx)+rep(b3[c==2&trt==1],each=num)*spg2[C==2&txp==1]
muc2tx<-1/(1+exp(-etac2tx))   # Predicted mean: class 2 t

etac3pl<-X3[C==3 & txp==0,]%*%beta3+rep(b1[c==3&trt==0],each=num)+
  rep(b2[c==3&trt==0],each=num)*rep(gridind,nc3pl)+rep(b3[c==3&trt==0],each=num)*spg3[C==3&txp==0]
muc3pl<-1/(1+exp(-etac3pl))   # Predicted mean: class 3 placebo
```

```
  etac3tx<-X3[C==3 & txp==1,]%*%beta3+rep(b1[c==3&trt==1],each=num)+
    rep(b2[c==3&trt==1],each=num)*rep(gridind,nc3tx)+rep(b3[c==3&trt==1],each=num)*spg3[C==3&txp==1]
  muc3tx<-1/(1+exp(-etac3tx))   # Predicted mean: class 3 tx


  yc1pl<-matrix(muc1pl,nc1pl,num,byrow=T)
  yc1tx<-matrix(muc1tx,nc1tx,num,byrow=T)
  yc2pl<-matrix(muc2pl,nc2pl,num,byrow=T)
  yc2tx<-matrix(muc2tx,nc2tx,num,byrow=T)
  yc3pl<-matrix(muc3pl,nc3pl,num,byrow=T)
  yc3tx<-matrix(muc3tx,nc3tx,num,byrow=T)

  YPOSc1pl[j,]<-colMeans(yc1pl)*7
  YPOSc1tx[j,]<-colMeans(yc1tx)*7
  YPOSc2pl[j,]<-colMeans(yc2pl)*7
  YPOSc2tx[j,]<-colMeans(yc2tx)*7
  YPOSc3pl[j,]<-colMeans(yc3pl)*7
  YPOSc3tx[j,]<-colMeans(yc3tx)*7

  YPOSc1diff[j,]<-YPOSc1tx[j,]-YPOSc1pl[j,]
  YPOSc2diff[j,]<-YPOSc2tx[j,]-YPOSc2pl[j,]
  YPOSc3diff[j,]<-YPOSc3tx[j,]-YPOSc3pl[j,]

  # if (j %% 100==0) print(j)
}



yposc1pl<-colMeans(YPOSc1pl)
yposc1tx<-colMeans(YPOSc1tx)
yposc2pl<-colMeans(YPOSc2pl)
yposc2tx<-colMeans(YPOSc2tx)
yposc3pl<-colMeans(YPOSc3pl)
yposc3tx<-colMeans(YPOSc3tx)
yposc1diff<-colMeans(YPOSc1diff)
yposc2diff<-colMeans(YPOSc2diff)
yposc3diff<-colMeans(YPOSc3diff)


yposc1plcl<-apply(YPOSc1pl,2,quantile,c(.025,.975))
yposc1txcl<-apply(YPOSc1tx,2,quantile,c(.025,.975))
yposc2plcl<-apply(YPOSc2pl,2,quantile,c(.025,.975))
yposc2txcl<-apply(YPOSc2tx,2,quantile,c(.025,.975))
yposc3plcl<-apply(YPOSc3pl,2,quantile,c(.025,.975))
yposc3txcl<-apply(YPOSc3tx,2,quantile,c(.025,.975))
yposc1diffcl<-apply(YPOSc1diff,2,quantile,c(.025,.975))
yposc2diffcl<-apply(YPOSc2diff,2,quantile,c(.025,.975))
yposc3diffcl<-apply(YPOSc3diff,2,quantile,c(.025,.975))




#---------#
```

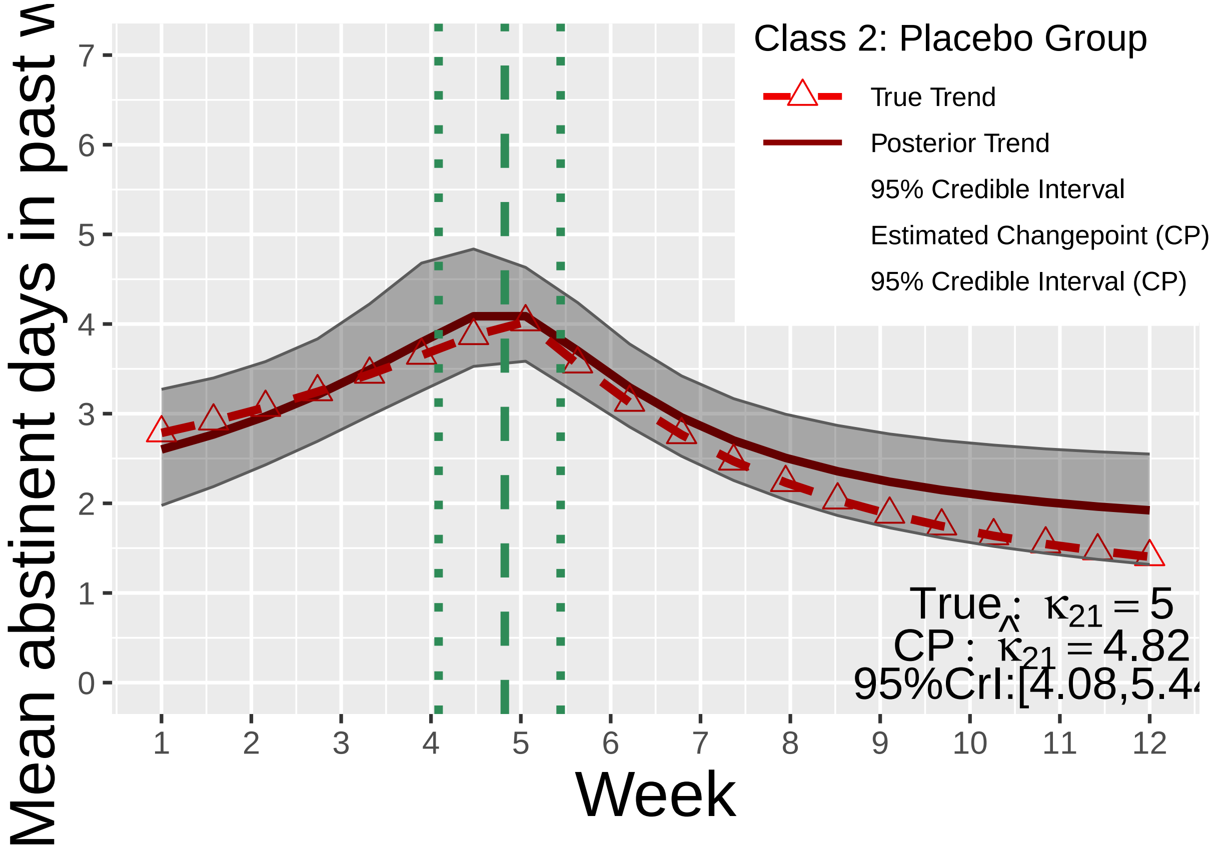
```
# Class 1 #
#---------#
dplotc1<-data.frame(grid=rep(gridind+5,2),
                    mmu1=c(my1p,yposc1pl),
                    lb1=c(rep(NA,num),yposc1plcl[1,]),
                    ub1=c(rep(NA,num),yposc1plcl[2,]),
                    gp=c(rep("True Trend",num),rep("Posterior Trend",num)),
                    mmu2=c(my1t,yposc1tx),
                    lb2=c(rep(NA,num),yposc1txcl[1,]),
                    ub2=c(rep(NA,num),yposc1txcl[2,]))



newlinetype=c("dashed","solid","solid","dashed","dotted")

ggplot(dplotc1,aes(x=grid,y=mmu1,col=gp,shape=gp))+
  geom_line(linetype=c(rep("solid",num),rep("dashed",num)),size=1.5)+
  geom_point(size=3.5)+
  geom_ribbon(aes(ymin = lb1, ymax = ub1,col="95% Credible Interval",fill="95% Credible Interval"),line
  geom_vline(aes(xintercept=mkap[1]+5,col="Estimated Changepoint (CP)"),
             linetype="dashed",size=1.5,show.legend = F)+
  geom_vline(aes(xintercept=qkap[1,1]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  geom_vline(aes(xintercept=qkap[2,1]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  scale_x_continuous(breaks = 1:12,limits=c(1,12))+
  scale_y_continuous(breaks = 0:7,limits=c(0,7))+
  scale_color_manual(breaks = c("True Trend","Posterior Trend",
                                "95% Credible Interval","Estimated Changepoint (CP)",
                                "95% Credible Interval (CP)"),
                     values = c("red2","red4","grey36","seagreen","seagreen"))+
  scale_shape_manual(breaks = c("True Trend","Posterior Trend"),
                     values = c(2,NA))+
  scale_fill_manual(breaks = c("95% Credible Interval"),
                    values = c("grey1"))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="Class 1: Placebo Group",
                              override.aes = list(
                                linetype = newlinetype,
                                shape=c(2,NA,NA,NA,NA),
                                linewidth=c(1.2,1,4,1.2,1.8),
                                fill=c("white","white","grey36","white","white")),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.key.width = unit(13,"mm"),
        legend.position = c(0.20,0.80),legend.text=element_text(size=10),
        axis.text=element_text(size=12),
        axis.title=element_text(size=24),title=element_text(size=20),
        plot.margin=unit(c(0.5, 1, 1, 0.5), "lines"))+
  annotate('text', x = 10.8, y = 2.85,
         label = "True:~kappa[11]==7.0",parse = TRUE,size=6)+
  annotate('text', x = 10.8, y = 2.45,
         label = "CP:~hat(kappa)[11]==7.03",parse = TRUE,size=6)+
  annotate(geom="text", x=10.8, y=2, label="95%CrI:[5.70,8.11]",
           color="black",size=6)
```



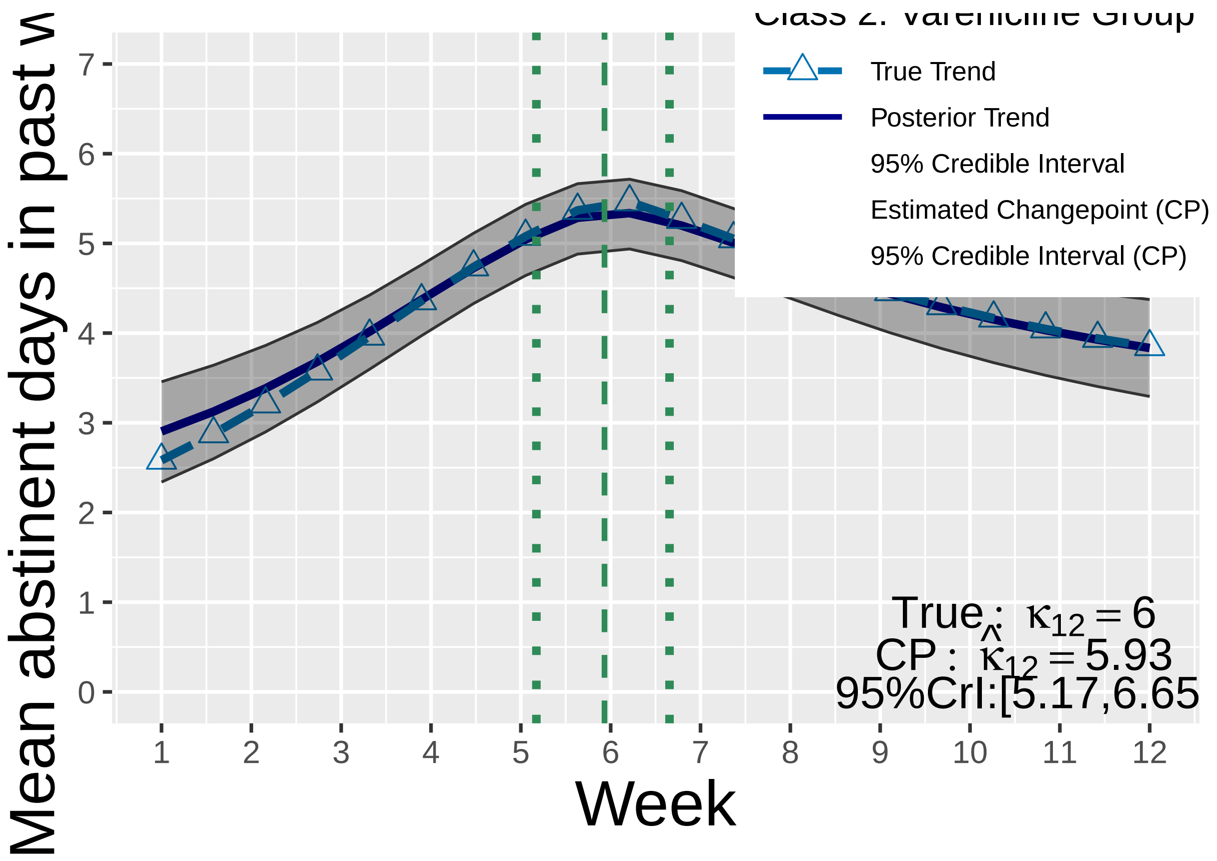
```
ggplot(dplotc1,aes(x=grid,y=mmu2,col=gp,shape=gp))+
  geom_line(linetype=c(rep("solid",num),rep("dashed",num)),size=1.5)+
  geom_point(size=3.5)+
  geom_ribbon(aes(ymin = lb2, ymax = ub2,col="95% Credible Interval",fill="95% Credible Interval"),line
  geom_vline(aes(xintercept=mkap[2]+5,col="Estimated Changepoint (CP)"),
             linetype="dashed",size=1,show.legend = F)+
  geom_vline(aes(xintercept=qkap[1,2]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  geom_vline(aes(xintercept=qkap[2,2]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  scale_x_continuous(breaks = 1:12,limits=c(1,12))+
  scale_y_continuous(breaks = 0:7,limits=c(0,7))+
  scale_color_manual(breaks = c("True Trend","Posterior Trend",
                                "95% Credible Interval","Estimated Changepoint (CP)",
                                "95% Credible Interval (CP)"),
                     values = c("#0072B2","darkblue","grey20","seagreen","seagreen"))+
  scale_shape_manual(breaks = c("True Trend","Posterior Trend"),
                     values = c(2,NA))+
  scale_fill_manual(breaks = c("95% Credible Interval"),
                    values = c("grey1"))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="Class 1: Varenicline Group",
                              override.aes = list(
                                linetype = newlinetype,
                                shape=c(2,NA,NA,NA,NA),
                                linewidth=c(1.2,1,4,1.2,1.8),
                                fill=c("white","white","grey20","white","white")),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.key.width = unit(13,"mm"),
        legend.position = c(0.22,0.80),legend.text=element_text(size=10),
        axis.text=element_text(size=12),
        axis.title=element_text(size=24),
        plot.margin=unit(c(0.5, 1, 1, 0.5), "lines"))+
  annotate('text', x = 10.6, y = 0.85,
           label = "True:~kappa[12]==7.0",parse = TRUE,size=6)+
  annotate('text', x = 10.6, y = 0.45,
           label = "CP:~hat(kappa)[12]==7.30",parse = TRUE,size=6)+
  annotate(geom="text", x=10.6, y=0, label="95%CrI:[6.38,8.34]",
           color="black",size=6)
```



```
#---------#
# Class 2 #
#---------#
dplotc2<-data.frame(grid=rep(gridind+5,2),
                    mmu1=c(my2p,yposc2pl),
                    lb1=c(rep(NA,num),yposc2plcl[1,]),
                    ub1=c(rep(NA,num),yposc2plcl[2,]),
                    gp=c(rep("True Trend",num),rep("Posterior Trend",num)),
                    mmu2=c(my2t,yposc2tx),
                    lb2=c(rep(NA,num),yposc2txcl[1,]),
                    ub2=c(rep(NA,num),yposc2txcl[2,]))

ggplot(dplotc2,aes(x=grid,y=mmu1,col=gp,shape=gp))+
  geom_line(linetype=c(rep("solid",num),rep("dashed",num)),size=1.5)+
  geom_point(size=3.5)+
  geom_ribbon(aes(ymin = lb1, ymax = ub1,col="95% Credible Interval",fill="95% Credible Interval"),linet
  geom_vline(aes(xintercept=mkap[3]+5,col="Estimated Changepoint (CP)"),
             linetype="dashed",size=1.5,show.legend = F)+
  geom_vline(aes(xintercept=qkap[1,3]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sho
  geom_vline(aes(xintercept=qkap[2,3]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sho
  scale_x_continuous(breaks = 1:12,limits=c(1,12))+
  scale_y_continuous(breaks = 0:7,limits=c(0,7))+
  scale_color_manual(breaks = c("True Trend","Posterior Trend",
                                "95% Credible Interval","Estimated Changepoint (CP)",
                                "95% Credible Interval (CP)"),
                     values = c("red2","red4","grey36","seagreen","seagreen"))+
  scale_shape_manual(breaks = c("True Trend","Posterior Trend"),
                     values = c(2,NA))+
  scale_fill_manual(breaks = c("95% Credible Interval"),
                    values = c("grey1"))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="Class 2: Placebo Group",
                              override.aes = list(
                                linetype = newlinetype,
                                shape=c(2,NA,NA,NA,NA),
                                linewidth=c(1.2,1,4,1.2,1.8),
                                fill=c("white","white","grey36","white","white")),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.key.width = unit(13,"mm"),
        legend.position = c(0.80,0.80),legend.text=element_text(size=10),
        axis.text=element_text(size=12),
        axis.title=element_text(size=24),
        plot.margin=unit(c(0.5, 1, 1, 0.5), "lines"))+
  annotate('text', x = 10.8, y = .85,
           label = "True:~kappa[21]==5.0",parse = TRUE,size=6)+
  annotate('text', x = 10.8, y = 0.45,
           label = "CP:~hat(kappa)[21]==4.82",parse = TRUE,size=6)+
  annotate(geom="text", x=10.8, y=0, label="95%CrI:[4.08,5.44]",
           color="black",size=6)
```

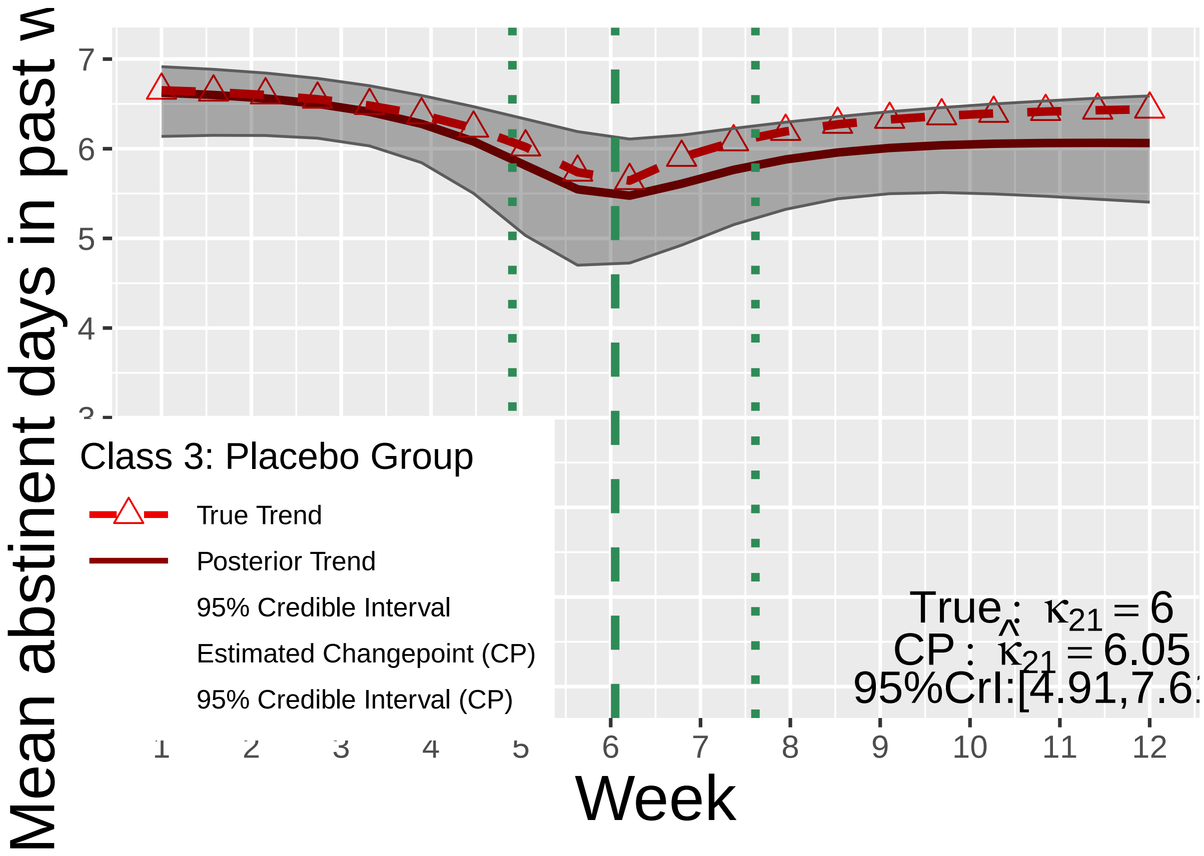


```
ggplot(dplotc2,aes(x=grid,y=mmu2,col=gp,shape=gp))+
  geom_line(linetype=c(rep("solid",num),rep("dashed",num)),size=1.5)+
  geom_point(size=3.5)+
  geom_ribbon(aes(ymin = lb2, ymax = ub2,col="95% Credible Interval",fill="95% Credible Interval"),line
  geom_vline(aes(xintercept=mkap[4]+5,col="Estimated Changepoint (CP)"),
             linetype="dashed",size=1,show.legend = F)+
  geom_vline(aes(xintercept=qkap[1,4]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  geom_vline(aes(xintercept=qkap[2,4]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  scale_x_continuous(breaks = 1:12,limits=c(1,12))+
  scale_y_continuous(breaks = 0:7,limits=c(0,7))+
  scale_color_manual(breaks = c("True Trend","Posterior Trend",
                                "95% Credible Interval","Estimated Changepoint (CP)",
                                "95% Credible Interval (CP)"),
                     values = c("#0072B2","darkblue","grey20","seagreen","seagreen"))+
  scale_shape_manual(breaks = c("True Trend","Posterior Trend"),
                     values = c(2,NA))+
  scale_fill_manual(breaks = c("95% Credible Interval"),
                    values = c("grey1"))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="Class 2: Varenicline Group",
                              override.aes = list(
                                linetype = newlinetype,
                                shape=c(2,NA,NA,NA,NA),
                                linewidth=c(1.2,1,4,1.2,1.8),
                                fill=c("white","white","grey36","white","white")),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.key.width = unit(13,"mm"),
        legend.position = c(0.80,0.85),legend.text=element_text(size=10),
        axis.text=element_text(size=12),
        axis.title=element_text(size=24),
        plot.margin=unit(c(0.5, 1, 1, 0.5), "lines"))+
  annotate('text', x = 10.6, y = 0.85,
           label = "True:~kappa[12]==6.0",parse = TRUE,size=6)+
  annotate('text', x = 10.6, y = 0.45,
           label = "CP:~hat(kappa)[12]==5.93",parse = TRUE,size=6)+
  annotate(geom="text", x=10.6, y=0, label="95%CrI:[5.17,6.65]",
           color="black",size=6)
```

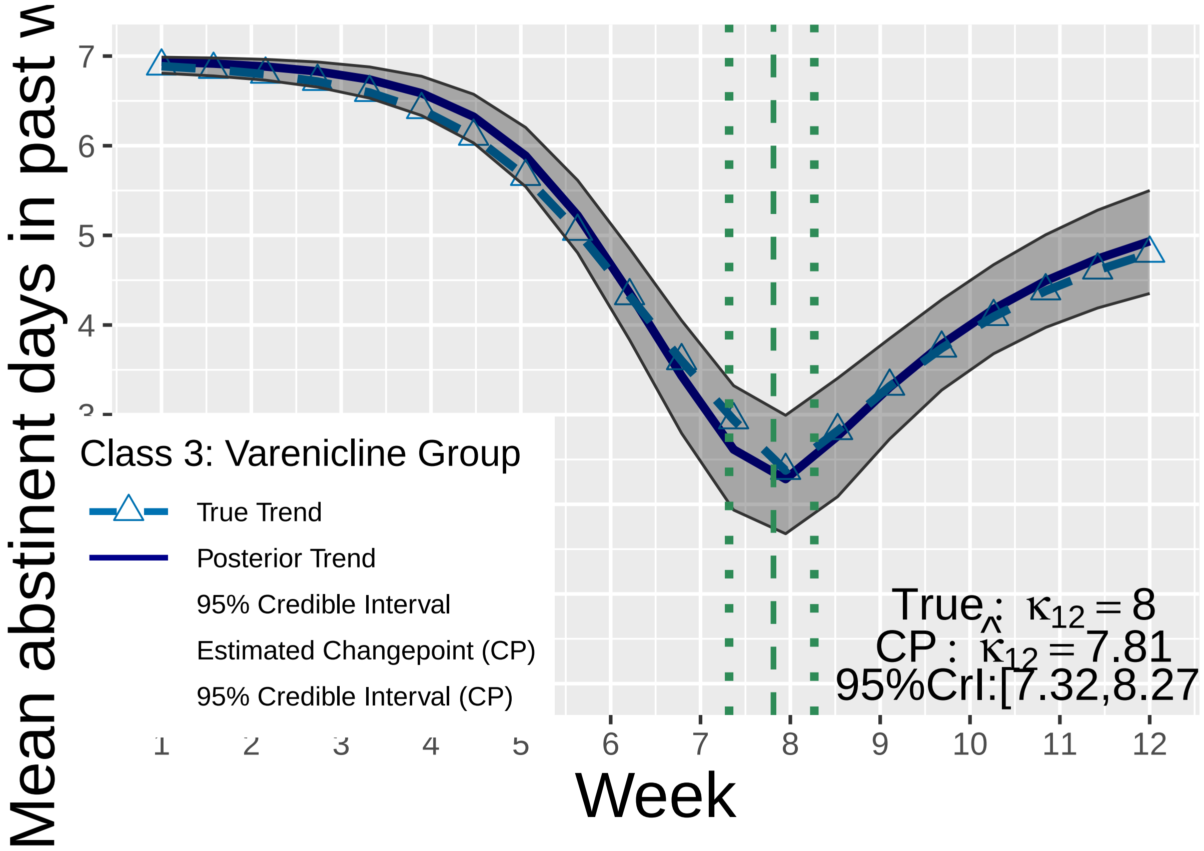


```
#---------#
# Class 3 #
#---------#

dplotc3<-data.frame(grid=rep(gridind+5,2),
                    mmu1=c(my3p,yposc3pl),
                    lb1=c(rep(NA,num),yposc3plcl[1,]),
                    ub1=c(rep(NA,num),yposc3plcl[2,]),
                    gp=c(rep("True Trend",num),rep("Posterior Trend",num)),
                    mmu2=c(my3t,yposc3tx),
                    lb2=c(rep(NA,num),yposc3txcl[1,]),
                    ub2=c(rep(NA,num),yposc3txcl[2,]))
```

```
ggplot(dplotc3,aes(x=grid,y=mmu1,col=gp,shape=gp))+
  geom_line(linetype=c(rep("solid",num),rep("dashed",num)),size=1.5)+
  geom_point(size=3.5)+
  geom_ribbon(aes(ymin = lb1, ymax = ub1,col="95% Credible Interval",fill="95% Credible Interval"),linet
  geom_vline(aes(xintercept=mkap[5]+5,col="Estimated Changepoint (CP)"),
             linetype="dashed",size=1.5,show.legend = F)+
  geom_vline(aes(xintercept=qkap[1,5]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sho
  geom_vline(aes(xintercept=qkap[2,5]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sho
  scale_x_continuous(breaks = 1:12,limits=c(1,12))+
  scale_y_continuous(breaks = 0:7,limits=c(0,7))+
  scale_color_manual(breaks = c("True Trend","Posterior Trend",
                                "95% Credible Interval","Estimated Changepoint (CP)",
                                "95% Credible Interval (CP)"),
                     values = c("red2","red4","grey36","seagreen","seagreen"))+
  scale_shape_manual(breaks = c("True Trend","Posterior Trend"),
                     values = c(2,NA))+
  scale_fill_manual(breaks = c("95% Credible Interval"),
                    values = c("grey1"))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="Class 3: Placebo Group",
                              override.aes = list(
                                linetype = newlinetype,
                                shape=c(2,NA,NA,NA,NA),
                                linewidth=c(1.2,1,4,1.2,1.8),
                                fill=c("white","white","grey36","white","white")),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.key.width = unit(13,"mm"),
        legend.position = c(0.18,0.20),legend.text=element_text(size=10),
        axis.text=element_text(size=12),
        axis.title=element_text(size=24),
        plot.margin=unit(c(0.5, 1, 1, 0.5), "lines"))+
  annotate('text', x = 10.8, y = .85,
           label = "True:~kappa[21]==6.0",parse = TRUE,size=6)+
  annotate('text', x = 10.8, y = 0.45,
           label = "CP:~hat(kappa)[21]==6.05",parse = TRUE,size=6)+
  annotate(geom="text", x=10.8, y=0, label="95%CrI:[4.91,7.61]",
           color="black",size=6)
```



```r
ggplot(dplotc3,aes(x=grid,y=mmu2,col=gp,shape=gp))+
  geom_line(linetype=c(rep("solid",num),rep("dashed",num)),size=1.5)+
  geom_point(size=3.5)+
  geom_ribbon(aes(ymin = lb2, ymax = ub2,col="95% Credible Interval",fill="95% Credible Interval"),line
  geom_vline(aes(xintercept=mkap[6]+5,col="Estimated Changepoint (CP)"),
             linetype="dashed",size=1,show.legend = F)+
  geom_vline(aes(xintercept=qkap[1,6]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  geom_vline(aes(xintercept=qkap[2,6]+5,col="95% Credible Interval (CP)"),linetype="dotted",size=1.5,sh
  scale_x_continuous(breaks = 1:12,limits=c(1,12))+
  scale_y_continuous(breaks = 0:7,limits=c(0,7))+
  scale_color_manual(breaks = c("True Trend","Posterior Trend","95% Credible Interval",
                                "Estimated Changepoint (CP)","95% Credible Interval (CP)"),
                     values = c("#0072B2","darkblue","grey20","seagreen","seagreen"))+
  scale_shape_manual(breaks = c("True Trend","Posterior Trend"),
                     values = c(2,NA))+
  scale_fill_manual(breaks = c("95% Credible Interval"),
                    values = c("grey1"))+
  xlab("Week")+ylab("Mean abstinent days in past week")+
  guides(color = guide_legend(title="Class 3: Varenicline Group",
                              override.aes = list(
                                linetype = newlinetype,
                                shape=c(2,NA,NA,NA,NA),
                                linewidth=c(1.2,1,4,1.2,1.8),
                                fill=c("white","white","grey36","white","white")),
                              reverse = F), fill="none",shape="none",linetype="none")+
  theme_gray(base_size = 14)+
  theme(legend.key = element_rect(fill = "white"),
        legend.key.width = unit(13,"mm"),
        legend.position = c(0.18,0.20),legend.text=element_text(size=10),
        axis.text=element_text(size=12),
        axis.title=element_text(size=24),
        plot.margin=unit(c(0.5, 1, 1, 0.5), "lines"))+
  annotate('text', x = 10.6, y = 0.85,
           label = "True:~kappa[12]==8.0",parse = TRUE,size=6)+
  annotate('text', x = 10.6, y = 0.45,
           label = "CP:~hat(kappa)[12]==7.81",parse = TRUE,size=6)+
  annotate(geom="text", x=10.6, y=0, label="95%CrI:[7.32,8.27]",
           color="black",size=6)
```

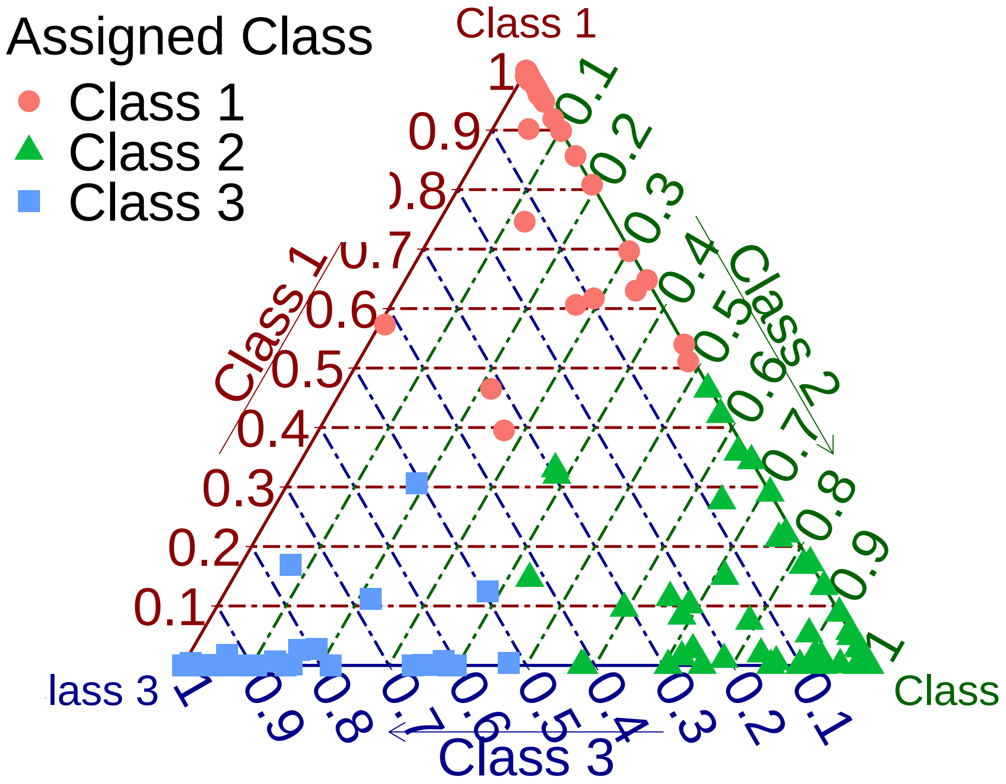


Triangle Plot

```
#------------------------------------------------------------------------------#
# Triangle plot #
#---------------#
library(ggtern)
```

```
## Registered S3 methods overwritten by 'ggtern':
##   method           from
##   grid.draw.ggplot ggplot2
##   plot.ggplot      ggplot2
##   print.ggplot     ggplot2
## --
## Remember to cite, run citation(package = 'ggtern') for further info.
## --

##
## Attaching package: 'ggtern'

## The following objects are masked from 'package:ggplot2':
##
##     aes, annotate, ggplot, ggplot_build, ggplot_gtable, ggplotGrob,
##     ggsave, layer_data, theme_bw, theme_classic, theme_dark,
##     theme_gray, theme_light, theme_linedraw, theme_minimal, theme_void
```

```
denom<-P1s+P2s+P3s
p1<-apply(P1s/denom,2,mean)   # Class 1 n=106 with posterior prob > .95
p2<-apply(P2s/denom,2,mean)   # Class 2 n=69 wiht posterior prob > .95
p3<-apply(P3s/denom,2,mean)   # Class 3 n=41 with posterior prob > .95




dtriax  <- data.frame(
  Class.1 =p1,
  Class.2 =p2,
  Class.3 =p3
)


dtriax2  <- data.frame(
  Class.1 =p1,
  Class.2 =p2,
  Class.3 =p3,
  colgp = as.factor(apply(dtriax,1,which.max))
)

dtriax2$colgp<-factor(dtriax2$colgp,labels=c("Class 1", "Class 2", "Class 3"))



ggtern(data=dtriax2,aes(Class.3,Class.1,Class.2,col=colgp,fill=colgp,shape=colgp)) +
  geom_mask() +
  geom_point(size=3.5) +
  theme_rgbw() +
  theme_showarrows() +
  theme_clockwise() +
  tern_limits(labels=c(0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9,1.0),
              breaks=seq(0.1,1,by=0.1))+
  guides(color = guide_legend(title="Assigned Class",
                              override.aes = list(shape=c(16,17,15)),
                              reverse = F), fill="none",shape="none")+
  Tlab("Class 1") + Llab("Class 3") + Rlab("Class 2") +
  Tarrowlab("Class 1") + Larrowlab("Class 3") + Rarrowlab("Class 2")+
  theme(legend.key = element_rect(fill = "white"),
        legend.title=element_text(size=20),
        legend.position = c(0.15,0.80),legend.text=element_text(size=20),
        axis.text=element_text(size=20),
        axis.title=element_text(size=16))
```


Assigned Class
● Class 1
▲ Class 2
■ Class 3
Class 1
).8
Class 1
Class 2
Class
lass 3
Class 3