DPMs applied to ROC curve estimation

2023-12-12

```
require(ROCnReg)
require(gridExtra)
require(grid)
require(ggplot2)
require(coda)
require(nor1mix)

# Summary of the dataset
data("endosyn")
summary(endosyn)
```

```
##     cvd_idf           age           gender          bmi
##   Min.   :0.0000   Min.   :18.25   Men  :1317   Min.   :12.60
##   1st Qu.:0.0000   1st Qu.:29.57   Women:1523   1st Qu.:23.19
##   Median :0.0000   Median :39.28                Median :26.24
##   Mean   :0.2433   Mean   :41.43                Mean   :26.69
##   3rd Qu.:0.0000   3rd Qu.:50.84                3rd Qu.:29.74
##   Max.   :1.0000   Max.   :84.66                Max.   :46.20
```

```
set.seed(123, "L'Ecuyer-CMRG")  # for reproducibility
pROC_dpm <- pooledROC.dpm(marker = "bmi", group = "cvd_idf",
    tag.h = 0, data = endosyn, standardise = TRUE, p = seq(0,
        1, l = 101), ci.level = 0.95, compute.lpml = TRUE, compute.WAIC = TRUE,
    compute.DIC = TRUE, pauc = pauccontrol(compute = TRUE, focus = "FPF",
        value = 0.1), density = densitycontrol(compute = TRUE),
    prior.h = priorcontrol.dpm(L = 10), prior.d = priorcontrol.dpm(L = 10),
    mcmc = mcmccontrol(nsave = 8000, nburn = 2000, nskip = 1),
    parallel = "multicore", ncpus = 10)
summary(pROC_dpm)
```

```
##
## Call:
## pooledROC.dpm(marker = "bmi", group = "cvd_idf", tag.h = 0, data = endosyn,
##     standardise = TRUE, p = seq(0, 1, l = 101), ci.level = 0.95,
##     compute.lpml = TRUE, compute.WAIC = TRUE, compute.DIC = TRUE,
##     pauc = pauccontrol(compute = TRUE, focus = "FPF", value = 0.1),
##     density = densitycontrol(compute = TRUE), prior.h = priorcontrol.dpm(L = 10),
##     prior.d = priorcontrol.dpm(L = 10), mcmc = mcmccontrol(nsave = 8000,
##         nburn = 2000, nskip = 1), parallel = "multicore", ncpus = 10)
##
## Approach: Pooled ROC curve - Bayesian DPM
## -----------------------------------------------
## Area under the pooled ROC curve: 0.759 (0.74, 0.777)*
```

```
## Partial area under the pooled ROC curve (FPF = 0.1): 0.168 (0.139, 0.199)*
##  * Credible level:  0.95
##
## Model selection criteria:
##                      Group H      Group D
## WAIC              12490.485     4017.063
## WAIC (Penalty)        8.431        5.468
## LPML              -6245.247    -2008.541
## DIC               12490.276     4016.920
## DIC (Penalty)         8.326        5.396
##
##
## Sample sizes:
##                      Group H      Group D
## Number of observations   2149          691
## Number of missing data      0            0
```

```r
plot(pROC_dpm, cex.main = 1.5, cex.lab = 1.5, cex.axis = 1.5,
    cex = 1.5)
```


Pooled ROC curve – Bayesian DPM
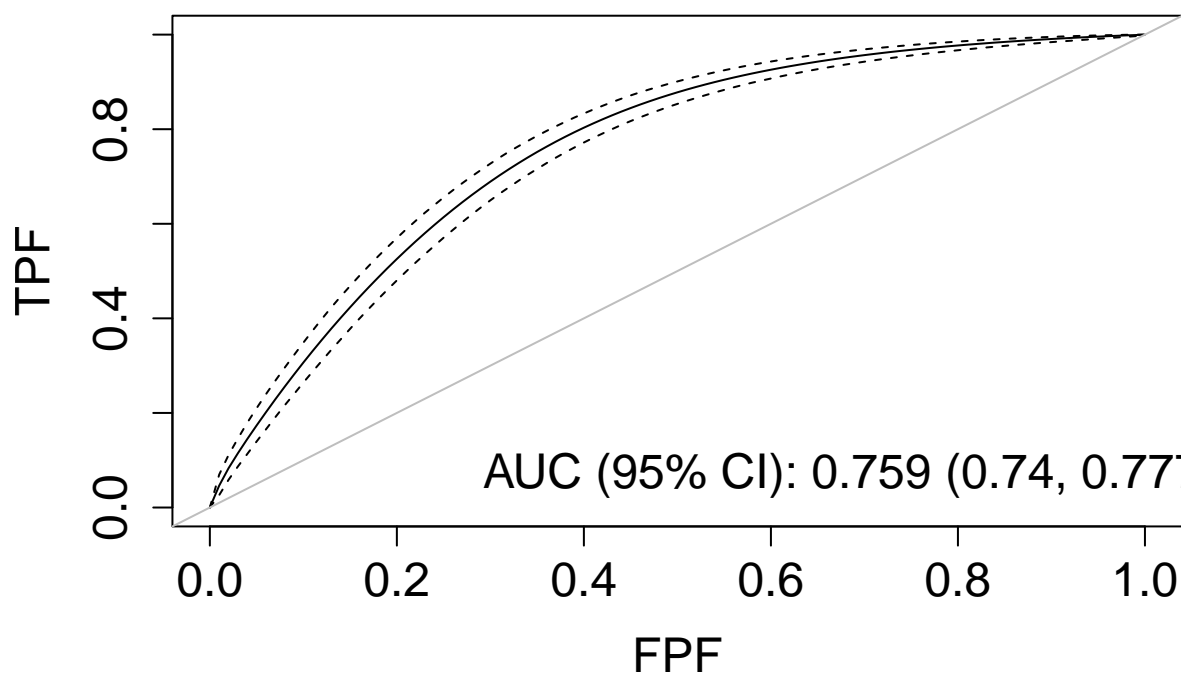
```r
# Densities of test ouctomes in each group Nondiseased
df_den_h <- data.frame(x = rep(pROC_dpm$dens$h$grid, each = pROC_dpm$mcmc$nsave),
    y = c(pROC_dpm$dens$h$dens))
p1 <- ggplot() + geom_histogram(aes(x = pROC_dpm$marker$h, y = ..density..),
    fill = "grey", color = "black") + stat_summary(data = df_den_h,
    aes(x = x, y = y), geom = "line", fun = "mean", col = "red",
    size = 1) + stat_summary(data = df_den_h, aes(x = x, y = y),
    geom = "line", fun = "quantile", fun.args = list(0.025),
    col = "red", linetype = "dashed") + stat_summary(data = df_den_h,
```

```
    aes(x = x, y = y), geom = "line", fun = "quantile", fun.args = list(0.975),
    col = "red", linetype = "dashed") + labs(x = "BMI", y = "Density",
    title = "Nondiseased population") + theme_bw() + theme(plot.title = element_text(hjust = 0.5,
    size = 20), axis.text = element_text(size = 20), axis.title = element_text(size = 20),
    )
```

```
## Warning: Using `size` aesthetic for lines was deprecated in ggplot2 3.4.0.
## i Please use `linewidth` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```
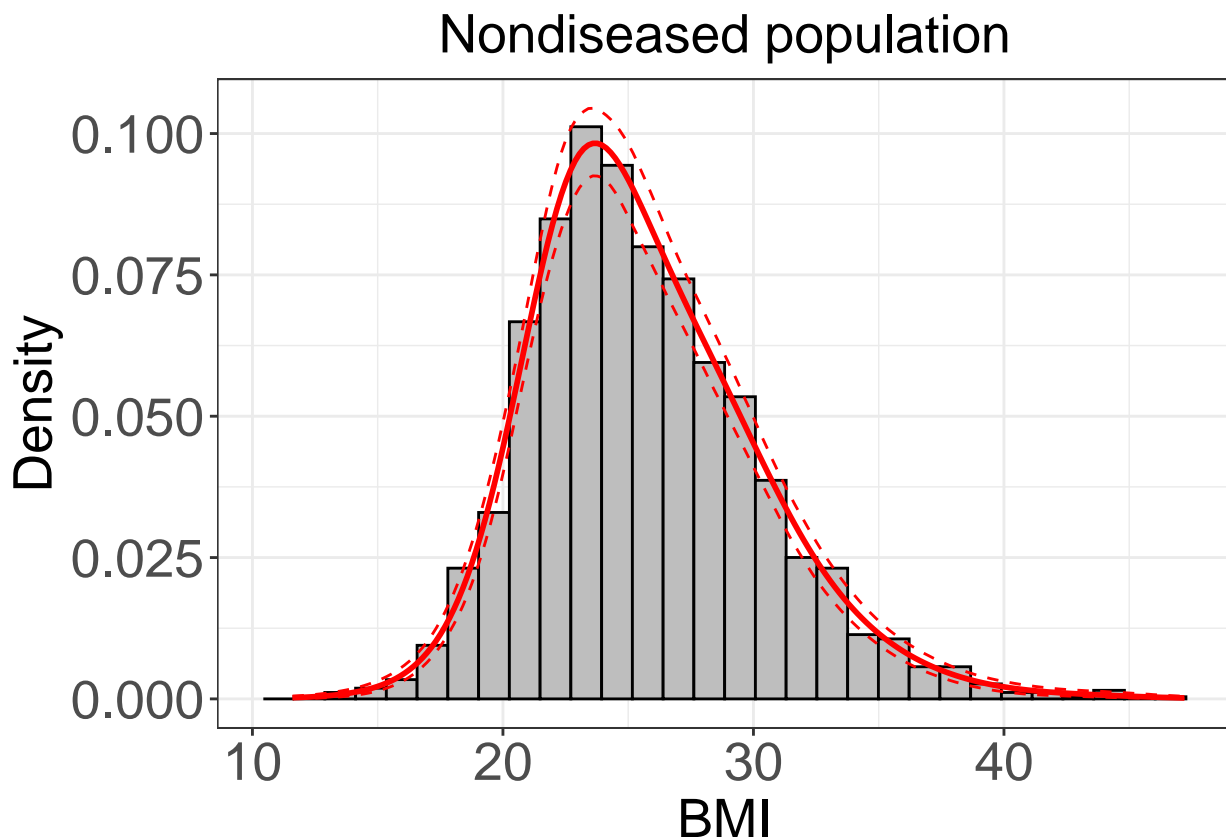
```
print(p1)
```

```
## Warning: The dot-dot notation (`..density..`) was deprecated in ggplot2 3.4.0.
## i Please use `after_stat(density)` instead.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.
```



```
# Diseased
df_den_d <- data.frame(x = rep(pROC_dpm$dens$d$grid, each = pROC_dpm$mcmc$nsave),
    y = c(pROC_dpm$dens$d$dens))
p2 <- ggplot() + geom_histogram(aes(x = pROC_dpm$marker$d, y = ..density..),
    fill = "grey", color = "black") + stat_summary(data = df_den_d,
```

```
    aes(x = x, y = y), geom = "line", fun = "mean", col = "red",
    size = 1) + stat_summary(data = df_den_d, aes(x = x, y = y),
    geom = "line", fun = "quantile", fun.args = list(0.025),
    col = "red", linetype = "dashed") + stat_summary(data = df_den_d,
    aes(x = x, y = y), geom = "line", fun = "quantile", fun.args = list(0.975),
    col = "red", linetype = "dashed") + labs(x = "BMI", y = "Density",
    title = "Diseased population") + theme_bw() + theme(plot.title = element_text(hjust = 0.5,
    size = 20), axis.text = element_text(size = 20), axis.title = element_text(size = 20))

print(p2)
```
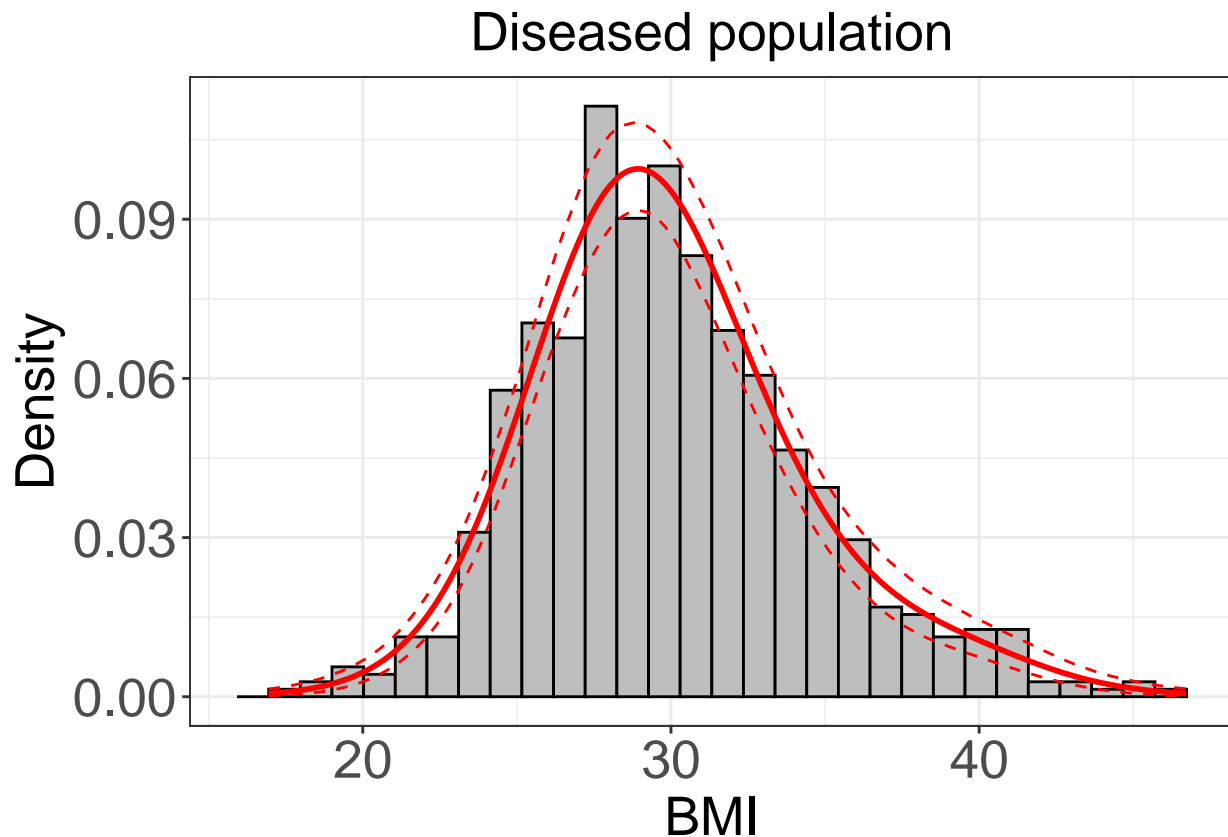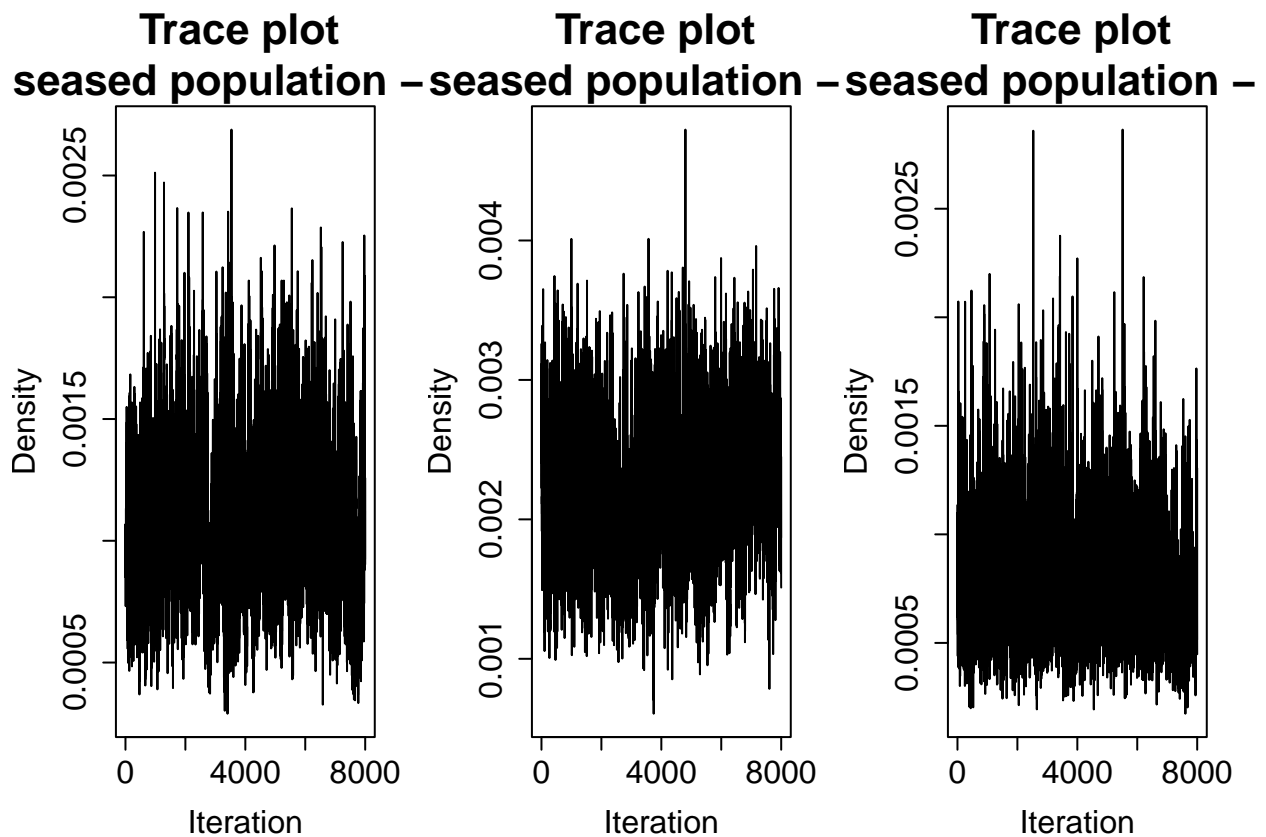


```
# Trace plots for several BMI values (randomly selected),
# for healthy and diseased population Nondiseased
set.seed(123, "Mersenne-Twister")  # for reproducibility
# BMI values (randomly selected)
pos_h_den <- sort(sample(1:length(pROC_dpm$dens$h$grid), 3))
bmi_h_den <- round(pROC_dpm$dens$h$grid[pos_h_den])

op <- par(mfrow = c(1, 3))
for (i in 1:3) {
    plot(1:pROC_dpm$mcmc$nsave, pROC_dpm$dens$h$dens[, pos_h_den[i]],
        type = "l", xlab = "Iteration", ylab = "Density", main = paste0("Trace plot \n Nondiseased popul
            round(bmi_h_den[i])), cex.main = 2, cex.lab = 1.5,
        cex.axis = 1.5)
}
```

Trace plot
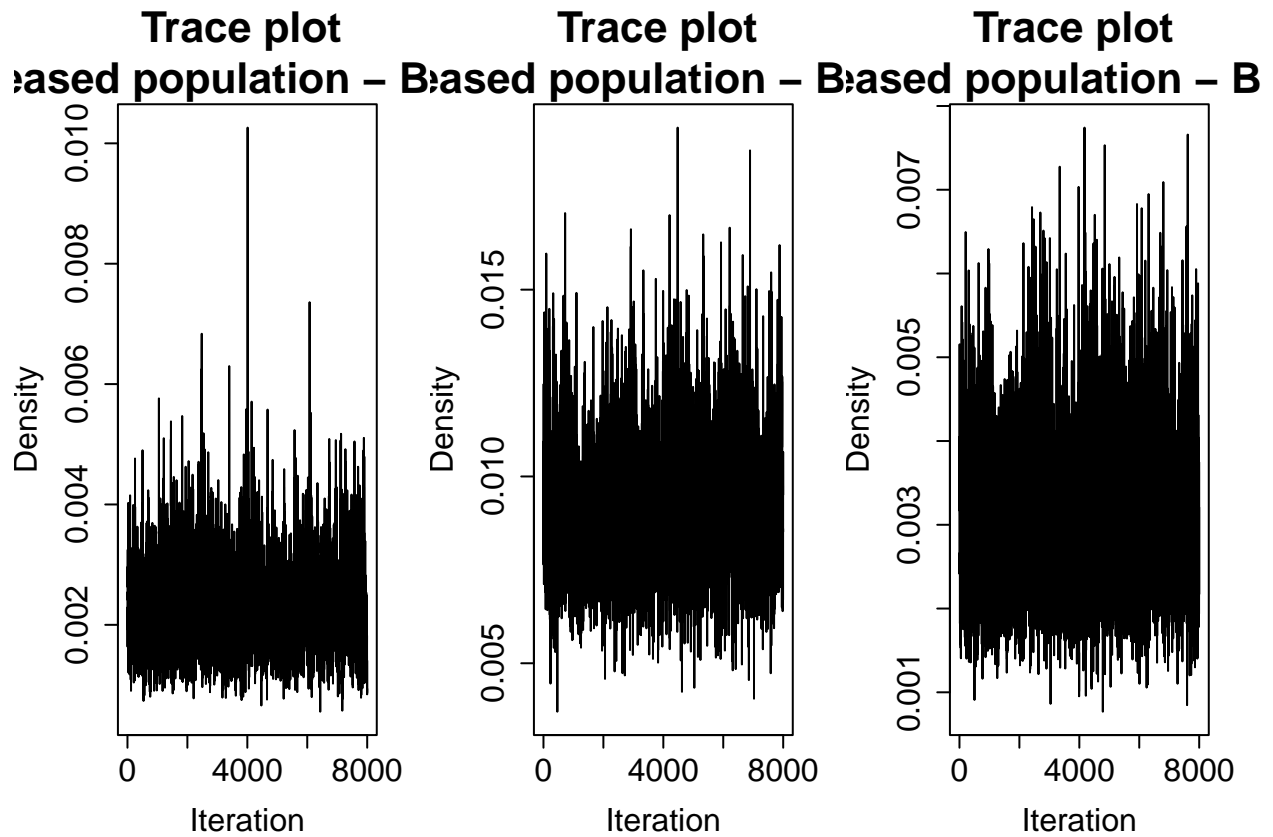### seased population – seased population – seased population –


```
par(op)

# Diseased
set.seed(123, "Mersenne-Twister")  # for reproducibility
# BMI values (randomly selected)
pos_d_den <- sort(sample(1:length(pROC_dpm$dens$d$grid), 3))
bmi_d_den <- round(pROC_dpm$dens$d$grid[pos_d_den])

op <- par(mfrow = c(1, 3))
for (i in 1:3) {
    plot(1:pROC_dpm$mcmc$nsave, pROC_dpm$dens$d$dens[, pos_d_den[i]],
        type = "l", xlab = "Iteration", ylab = "Density", main = paste0("Trace plot \n Diseased populati
            bmi_d_den[i]), cex.main = 2, cex.lab = 1.5, cex.axis = 1.5)
}
```
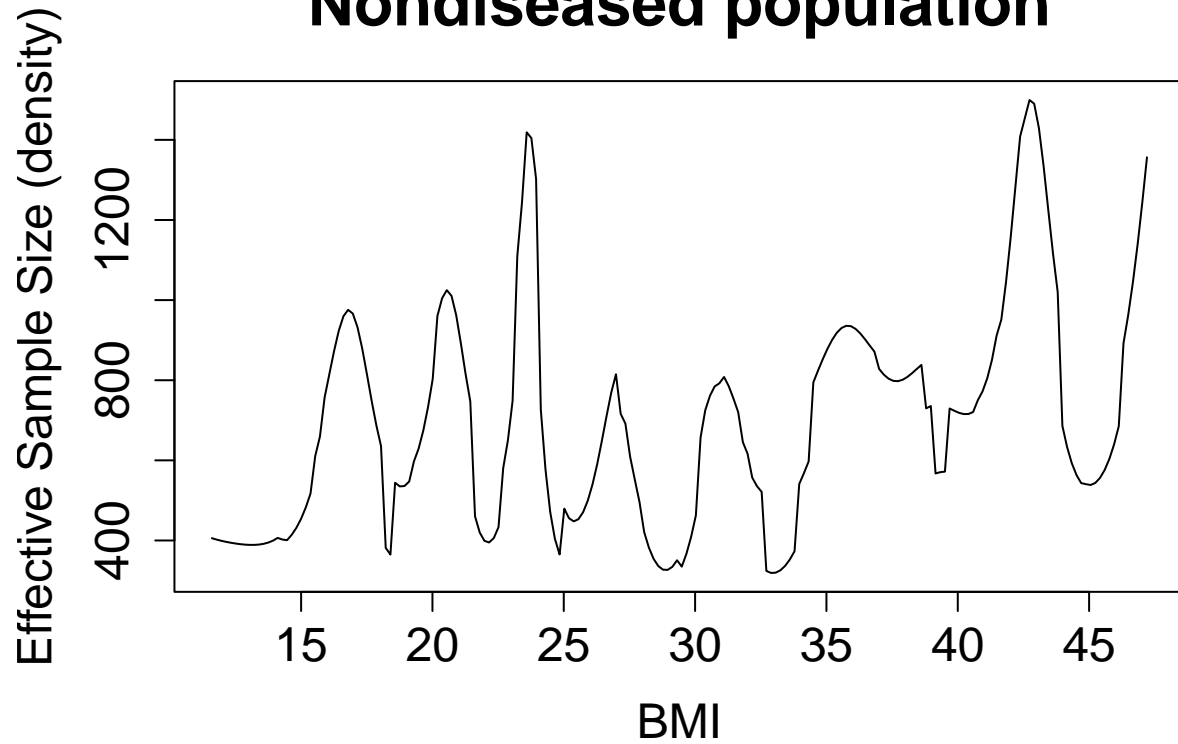
Trace plot
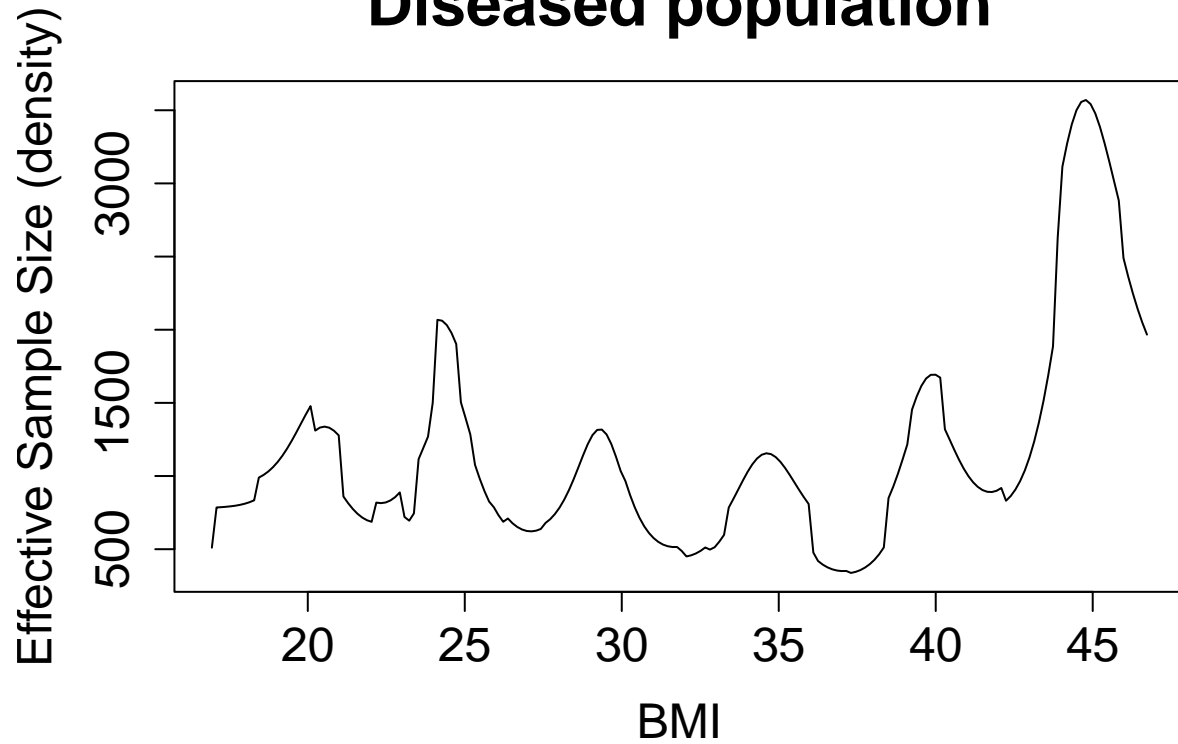## ~~~ased population – B~~~ased population – B~~~ased population – B


```
par(op)

# Effective sample size in healthy and diseased populations
# Nondiseased
plot(pROC_dpm$dens$h$grid, effectiveSize(pROC_dpm$dens$h$dens),
    type = "l", xlab = "BMI", ylab = "Effective Sample Size (density)",
    main = "Nondiseased population", cex.main = 2, cex.lab = 1.5,
    cex.axis = 1.5)
```
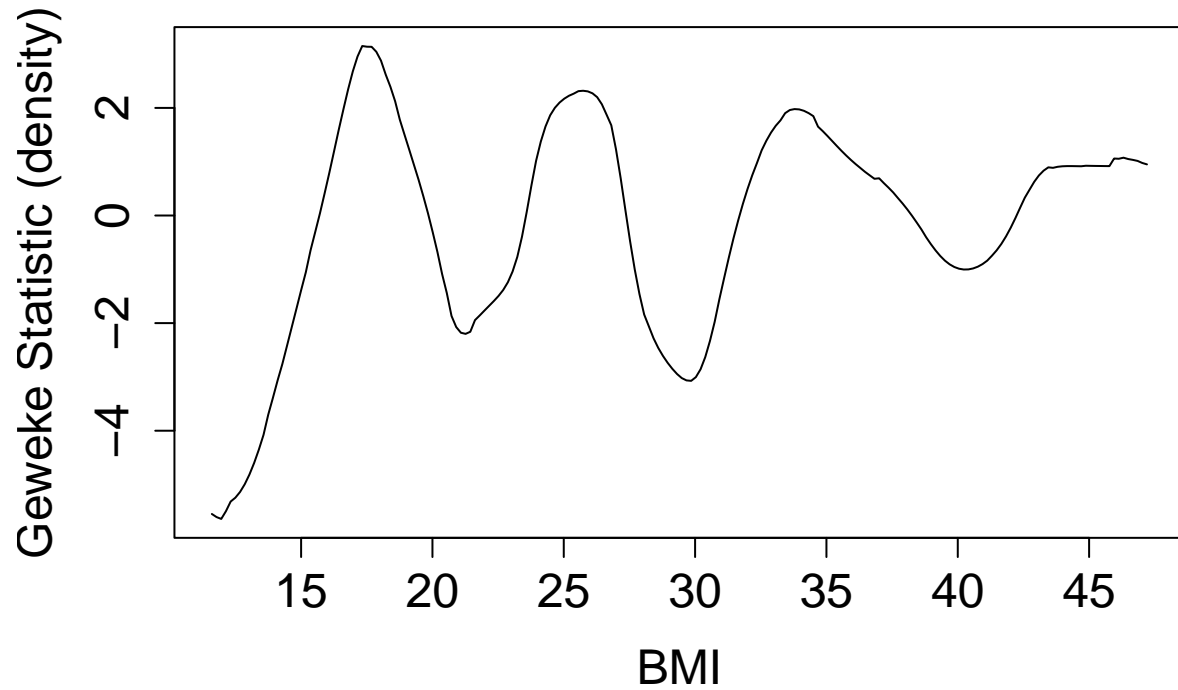
Nondiseased population


```
# Diseased
plot(pROC_dpm$dens$d$grid, effectiveSize(pROC_dpm$dens$d$dens),
    type = "l", xlab = "BMI", ylab = "Effective Sample Size (density)",
    main = "Diseased population", cex.main = 2, cex.lab = 1.5,
    cex.axis = 1.5)
```

Diseased population


```
# Geweke statistic in healthy and diseased populations
# Nondiseased
plot(pROC_dpm$dens$h$grid, geweke.diag(pROC_dpm$dens$h$dens)$z,
    type = "l", xlab = "BMI", ylab = "Geweke Statistic (density)",
    main = "Nondiseased population", cex.main = 2, cex.lab = 1.5,
    cex.axis = 1.5)
```

Nondiseased population


```
# Diseased
plot(pROC_dpm$dens$d$grid, geweke.diag(pROC_dpm$dens$d$dens)$z,
    type = "l", xlab = "BMI", ylab = "Geweke Statistic (density)",
    main = "Diseased population", cex.main = 2, cex.lab = 1.5,
    cex.axis = 1.5)
```

Diseased population