

```
## Load libraries
library(splines)
library(MASS)

##source functions
source("../functions.R")
```

1 Probability of being a false positive as a linear function of time

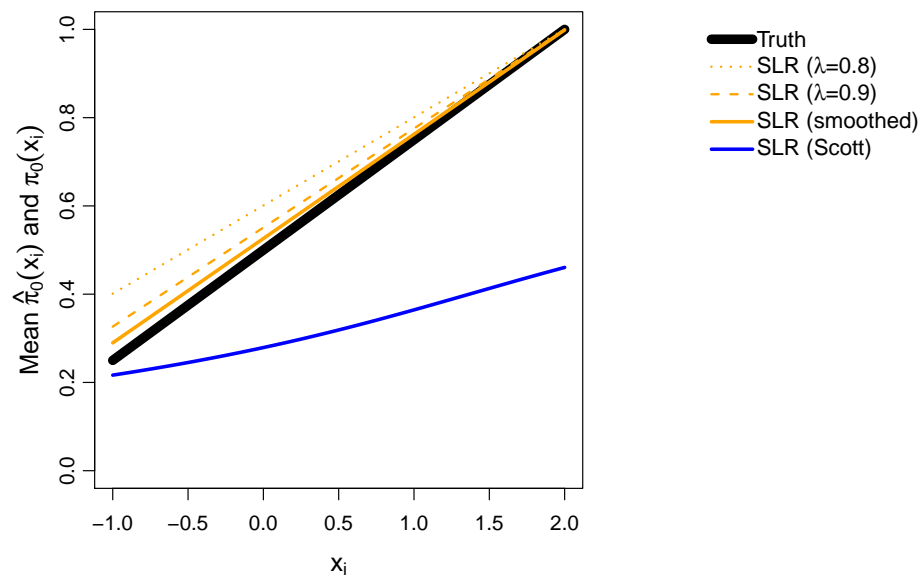
Load results:

```
load("simResults_pi0x_noThresh_1.RData")
load("simResults_pi0x_Scott_1.RData")
```

1.1 Plot for means

```
plotMeanPi0(pi0, pi0MeansVars, pi0hatScottMean, tme=tme)

legend("topright", inset=c(-0.71,0),
      legend=c("Truth", expression(paste("SLR (", lambda, "=0.8)")),
               expression(paste("SLR (", lambda, "=0.9)")),
               "SLR (smoothed)", "SLR (Scott)"),
      col=c("black", "orange", "orange", "orange", "blue"), bty="n",
      lwd=c(8,2,2,3,3), lty=c(1,3,2,1,1),
      cex=1.2, x.intersp=0.2, y.intersp=1.0)
```



2 Probability of being a false positive as a smooth function of time

Load results:

```
load("simResults_pi0x_noThresh_2.RData")
load("simResults_pi0x_Scott_2.RData")
```

2.1 Plot for means

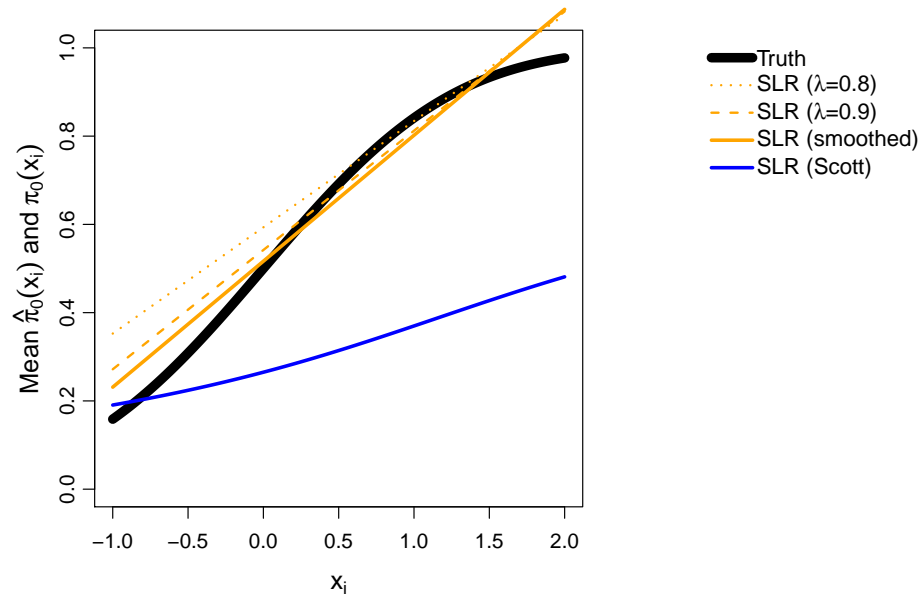
```
plotMeanPi0(pi0, pi0Lin.MeansVars, pi0hatLin.ScottMean, tme=tme)

legend("topright", inset=c(-0.71,0), ##x=-0.2, y=0.45, ##"bottomright", ##x=-100, y=0.3,
       legend=c("Truth",
                 expression(paste("SLR (", lambda, "=0.8)")),
                 expression(paste("SLR (", lambda, "=0.9)")),
                 "SLR (smoothed)",
                 "SLR (Scott)"),
       col=c("black",
             rep("orange", 3),
             "blue"))
```

```

"blue"),
bty="n",
lwd=c(8,2,2,3,3), lty=c(1,3,2,1,1),
cex=1.2, x.intersp=0.2, y.intersp=1.0)

```

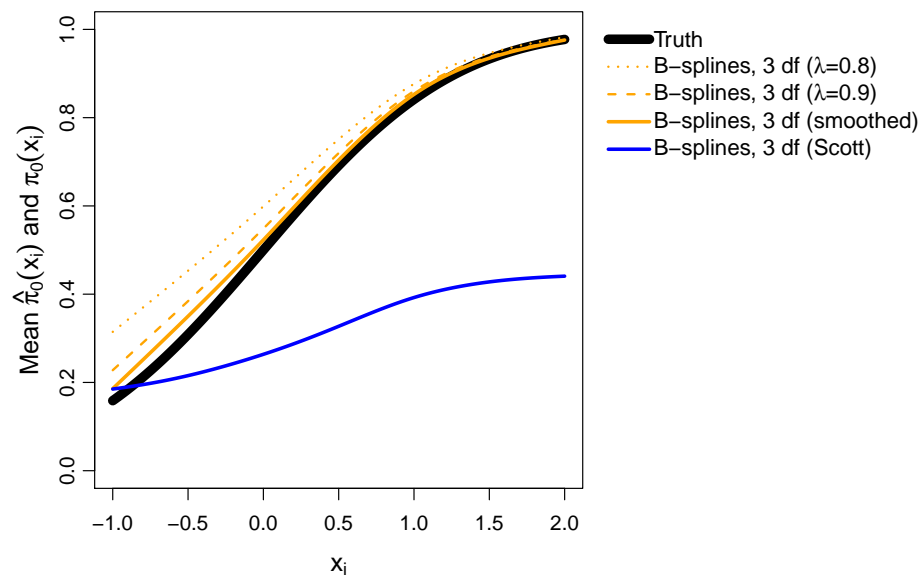


```

plotMeanPi0(pi0, pi0Spl.MeansVars, pi0hatSpl.ScottMean, tme=tme)

legend("topright", inset=c(-0.71,0), ##x=-0.2, y=0.45, ##"bottomright", ##x=-100, y=0.3,
       legend=c("Truth",
                 expression(paste("B-splines, 3 df (", lambda, "=0.8)")),
                 expression(paste("B-splines, 3 df (", lambda, "=0.9)")),
                 "B-splines, 3 df (smoothed)",
                 "B-splines, 3 df (Scott)"),
       col=c("black",
              rep("orange",3),
              "blue"),
       bty="n",
       lwd=c(8,2,2,3,3), lty=c(1,3,2,1,1),
       cex=1.2, x.intersp=0.2, y.intersp=1.0)

```



3 Probability of being a false positive as a sine + step function

Load results:

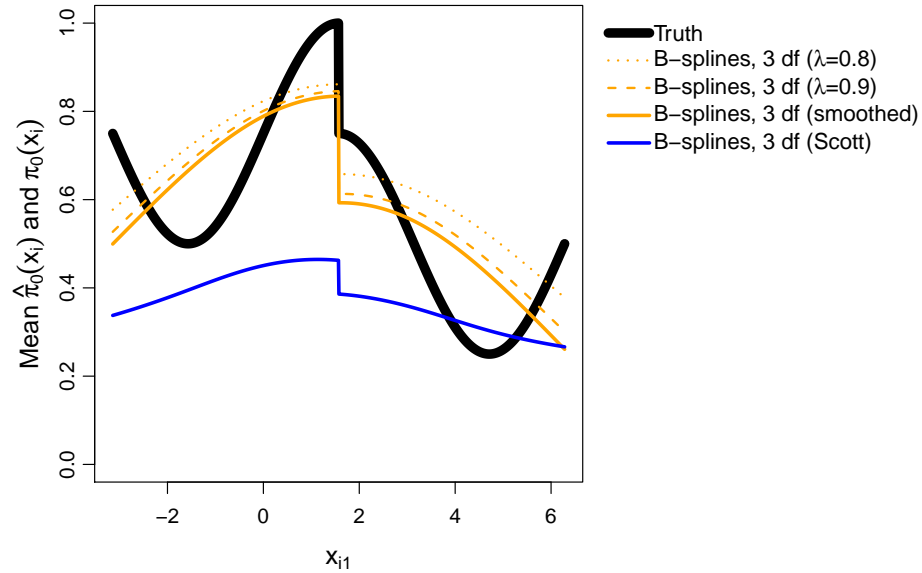
```
load("simResults_pi0x_noThresh_3.RData")
load("simResults_pi0x_Scott_3.RData")
```

3.1 Plot for means

```
plotMeanPi0(pi0, pi0_3.MeansVars, pi0hat3.ScottMean, tme=tme1, TRUE)

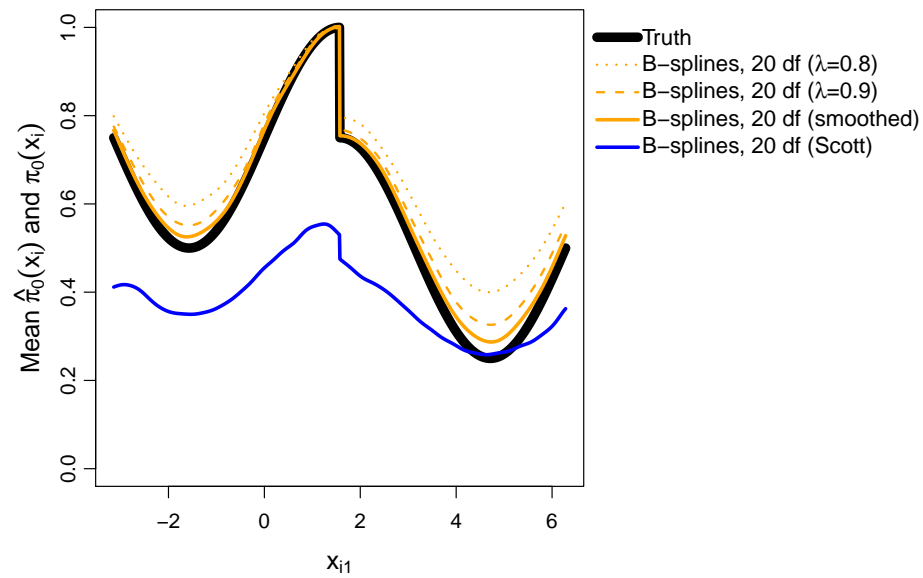
legend("topright", inset=c(-0.71,0),
      legend=c("Truth",
               expression(paste("B-splines, 3 df (", lambda, "=0.8)")),
               expression(paste("B-splines, 3 df (", lambda, "=0.9)")),
               "B-splines, 3 df (smoothed)",
               "B-splines, 3 df (Scott)"),
      col=c("black", rep("orange",3), "blue"),
      bty="n",
```

```
lwd=c(8,2,2,3,3), lty=c(1,3,2,1,1),
cex=1.2, x.intersp=0.2, y.intersp=1.0)
```



```
plotMeanPi0(pi0, pi0_20.MeanVars, pi0hat20.ScottMean, tme=tme1, TRUE)

legend("topright", inset=c(-0.71,0),
      legend=c("Truth",
                expression(paste("B-splines, 20 df (", lambda, "=0.8)")),
                expression(paste("B-splines, 20 df (", lambda, "=0.9)")),
                "B-splines, 20 df (smoothed)",
                "B-splines, 20 df (Scott)"),
      col=c("black", rep("orange",3), "blue"),
      bty="n",
      lwd=c(8,2,2,3,3), lty=c(1,3,2,1,1),
      cex=1.2, x.intersp=0.2, y.intersp=1.0)
```

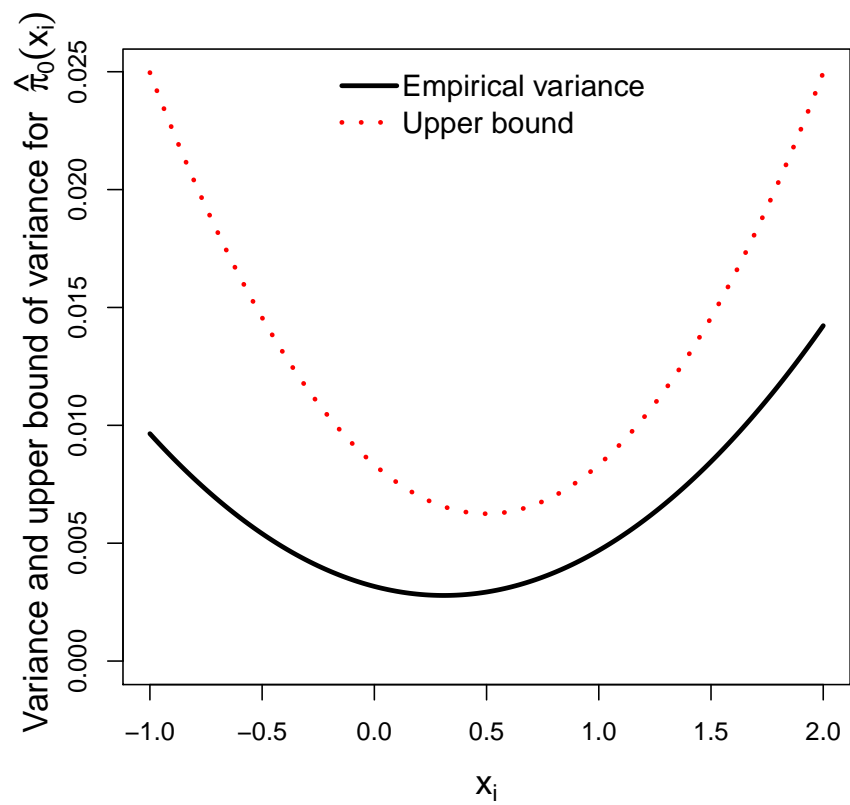


3.2 Plots for variances

```
ntest <- length(pi0MeansVars$pi0hatVar0.8)

tme <- seq(-1,2,length=ntest)

##Get the variance bound:
pi0hatVarBound <- getVarBound(tme, 0.8)
##Make plot:
plotVarBound(pi0hatVarBound, pi0MeansVars$pi0hatVar0.8, tme)
```



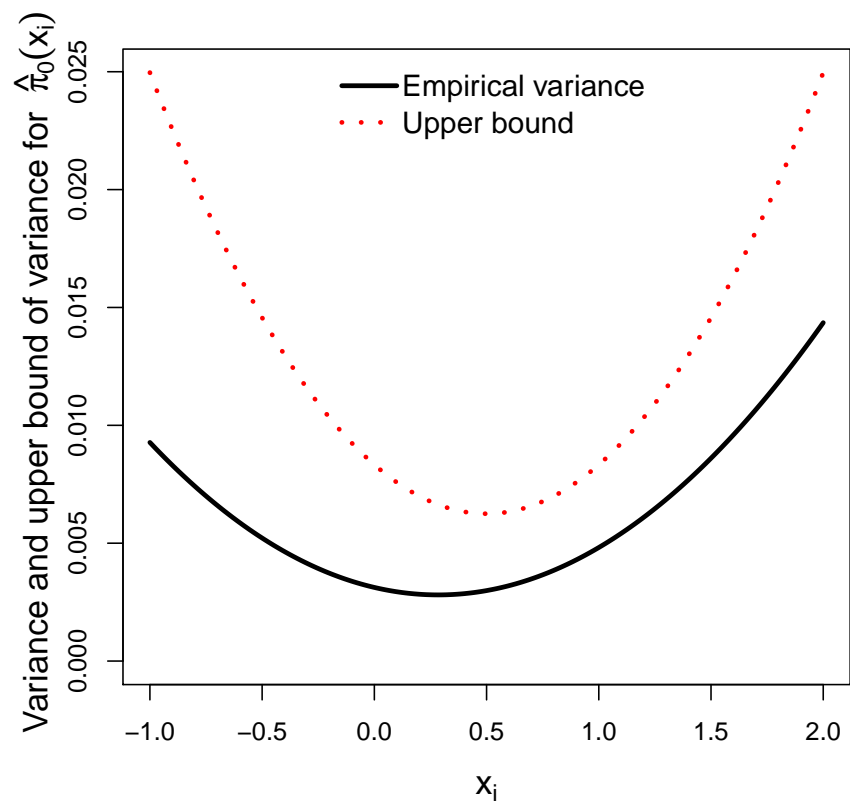
```

ntest <- length(pi0Lin.MeanVars$pi0hatVar0.8)

tme <- seq(-1,2,length=ntest)

##Get the variance bound:
pi0hatVarBound <- getVarBound(tme, 0.8)
##Make plot:
plotVarBound(pi0hatVarBound, pi0Lin.MeanVars$pi0hatVar0.8, tme)

```



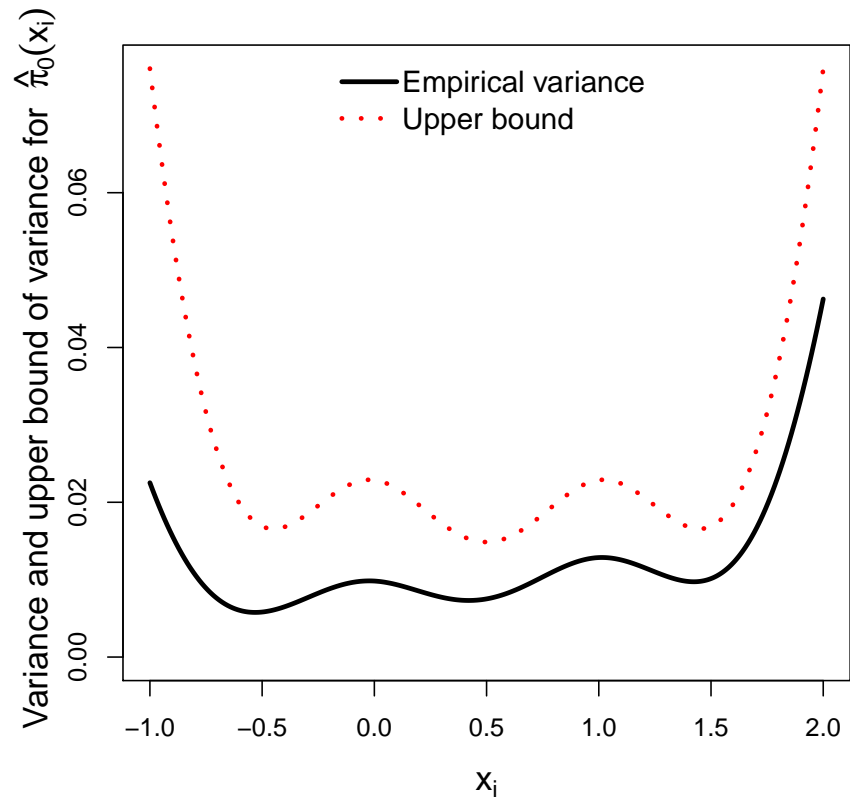
```

ntest <- length(pi0Spl.MeanVars$pi0hatVar0.8)

tme <- seq(-1,2,length=ntest)
splineMat <- ns(tme,df=3)

##Get the variance bound:
pi0hatVarBound <- getVarBound(splineMat, 0.8)
##Make plot:
plotVarBound(pi0hatVarBound, pi0Spl.MeanVars$pi0hatVar0.8, tme)

```

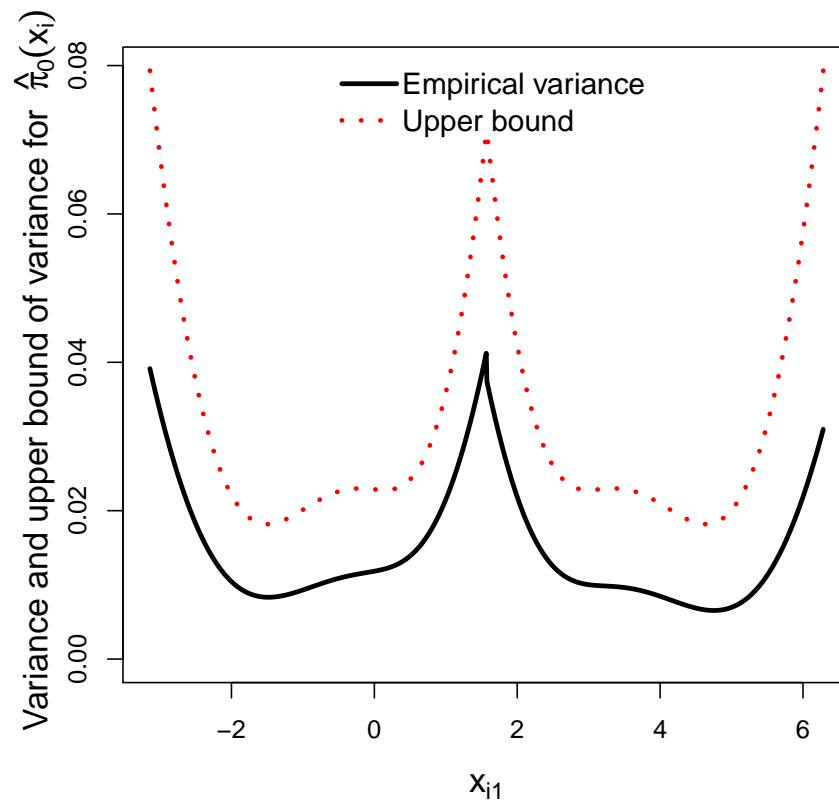
```

ntest <- length(pi0_3.MeanVars$pi0hatVar0.8)

tme1 <- seq(-1*pi, 2*pi, length=ntest)
tme2 <- rep(1:0, each=ntest/2)
splineMat3 <- cbind(ns(tme1, df=3), tme2)

##Get the variance bound:
pi0hatVarBound <- getVarBound(splineMat3, 0.8)
##Make plot:
plotVarBound(pi0hatVarBound, pi0_3.MeanVars$pi0hatVar0.8, tme1, TRUE)

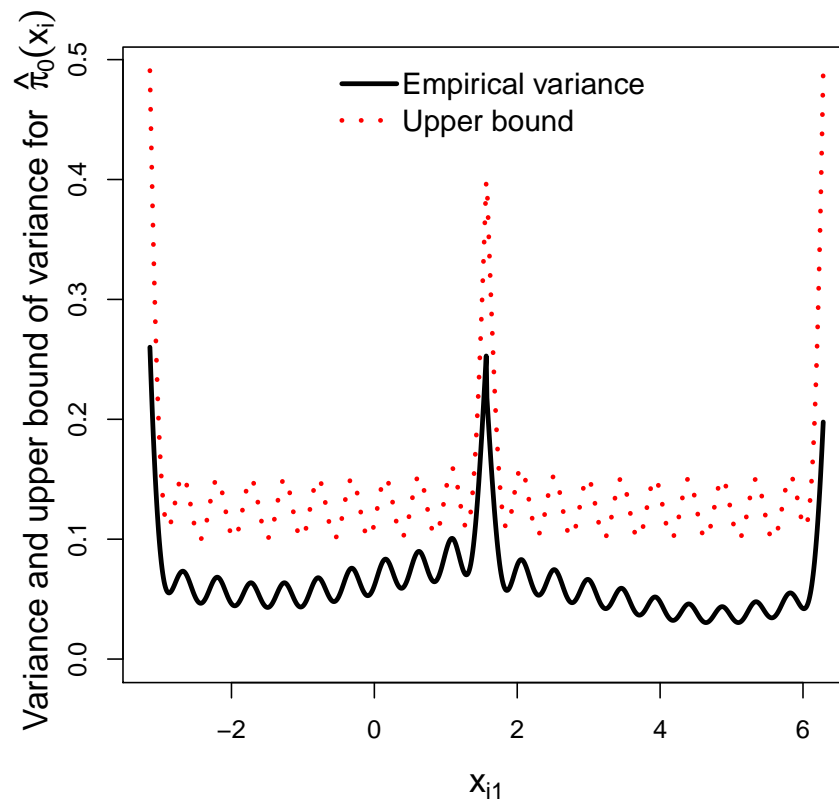
```



```
ntest <- length(pi0_3.MeanVars$pi0hatVar0.8)

tme1 <- seq(-1*pi, 2*pi, length=ntest)
tme2 <- rep(1:0, each=ntest/2)
splineMat20 <- cbind(ns(tme1, df=20), tme2)

##Get the variance bound:
pi0hatVarBound <- getVarBound(splineMat20, 0.8)
##Make plot:
plotVarBound(pi0hatVarBound, pi0_20.MeanVars$pi0hatVar0.8, tme1, TRUE)
```



Session info:

```
devtools::session_info()

## Session info -----
## setting value
## version R version 3.3.1 (2016-06-21)
## system x86_64, mingw32
## ui RTerm
## language (EN)
## collate English_United States.1252
## tz America/New_York
## date 2017-01-04

## Packages -----
## package * version date source
```

```
## devtools      1.12.0 2016-06-24 CRAN (R 3.3.1)
## digest        0.6.9 2016-01-08 CRAN (R 3.3.1)
## evaluate      0.10   2016-10-11 CRAN (R 3.3.2)
## highr         0.6    2016-05-09 CRAN (R 3.3.1)
## knitr         * 1.15.1 2016-11-22 CRAN (R 3.3.2)
## magrittr      1.5    2014-11-22 CRAN (R 3.3.1)
## MASS         * 7.3-45 2016-04-21 CRAN (R 3.3.1)
## memoise       1.0.0 2016-01-29 CRAN (R 3.3.1)
## stringi       1.1.1 2016-05-27 CRAN (R 3.3.0)
## stringr       1.0.0 2015-04-30 CRAN (R 3.3.1)
## withr         1.0.2 2016-06-20 CRAN (R 3.3.1)
```