resultados

Cristiano

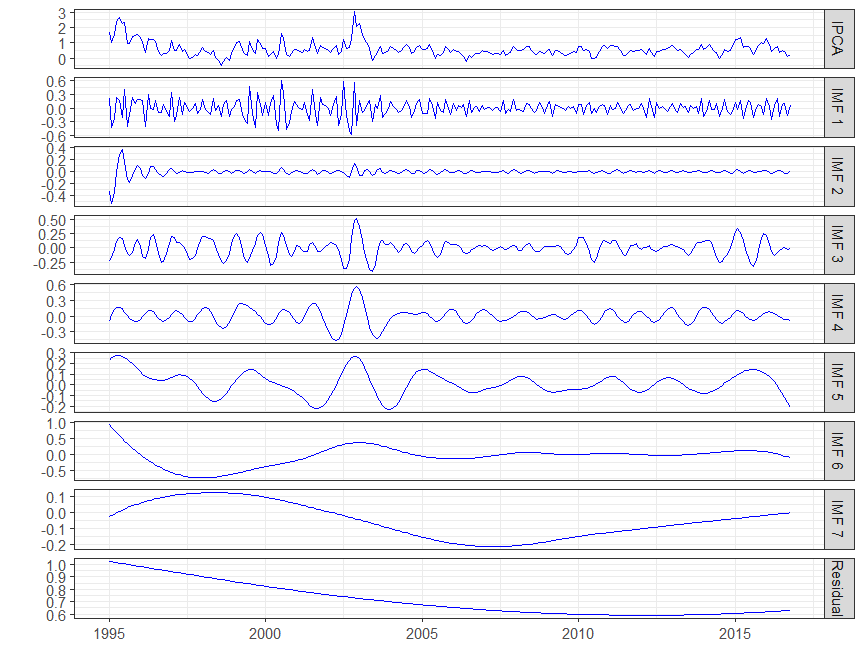
12 de outubro de 2016

O resumo deste para realizar os seguintes procedimentos:

* Decomposição com EMD do IPCA
* Estatísticas descritivas das IMFs

## Decomposição do IPCA com CEEMDAN

devtools::load\_all()  
library(nimcno)  
ipca <- ipca95[,"ipca"]  
demd <- Rlibeemd::ceemdan(ipca, noise\_strength = 0.4)  
x <- cbind(ipca, demd)  
colnames(x) <- c("IPCA", colnames(demd))  
tsplot(x)



## Estatísticas descritivas das IMFs

Período médio é o número total de observações dividido pelo número total de máximos locais de cada IMF.

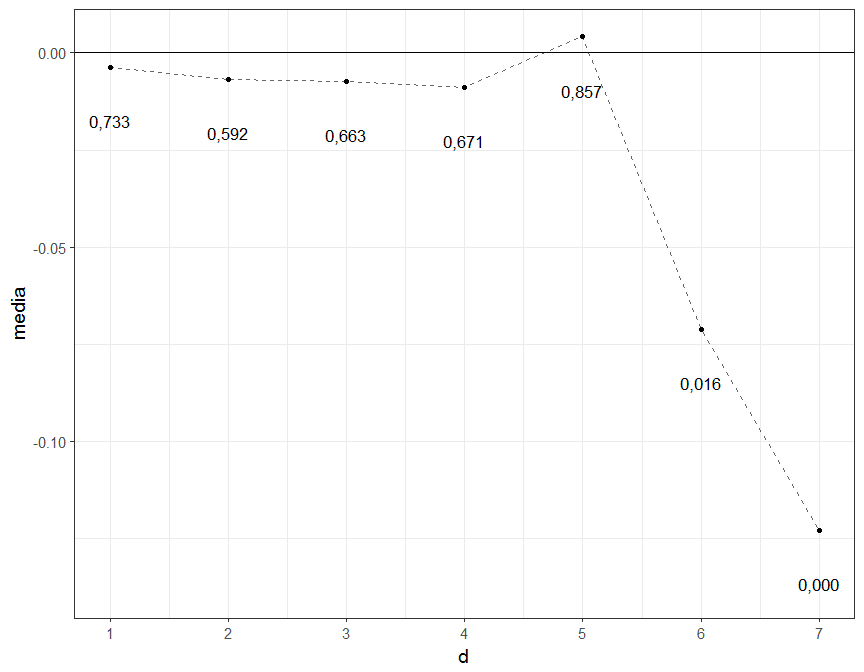
correlação entre a IMF e a serie IPCA

n <- length(demd[,1])  
ntml <- apply(demd, 2, function(x) length(Rlibeemd::extrema(x)$maxima[,1]))  
pm <- n/ntml  
vari <- apply(demd, 2, var)  
varip <- (vari/var(ipca))\*100  
# tabela com os resultados  
tab <- n2tab(cbind(pm, vari, varip), 2)  
tab <- rbind(obs=c(rep("",1), n2tab(var(ipca), 2), ""),  
 tab,  
 soma=c(rep("",2), n2tab(sum(varip), 2)))  
colnames(tab) <- c("Período Médio (mês)",  
 "Variância",  
 "variância como % da variância observada")  
knitr::kable(tab)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Período Médio (mês) | Variância | variância como % da variância observada |
| obs |  | 0,23 |  |
| IMF 1 | 3,05 | 0,03 | 13,72 |
| IMF 2 | 4,85 | 0,00 | 1,75 |
| IMF 3 | 6,72 | 0,02 | 8,82 |
| IMF 4 | 11,39 | 0,02 | 9,09 |
| IMF 5 | 23,82 | 0,01 | 5,17 |
| IMF 6 | 43,67 | 0,09 | 38,35 |
| IMF 7 | 87,33 | 0,01 | 5,22 |
| Residual | 131,00 | 0,02 | 7,83 |
| soma |  |  | 89,96 |

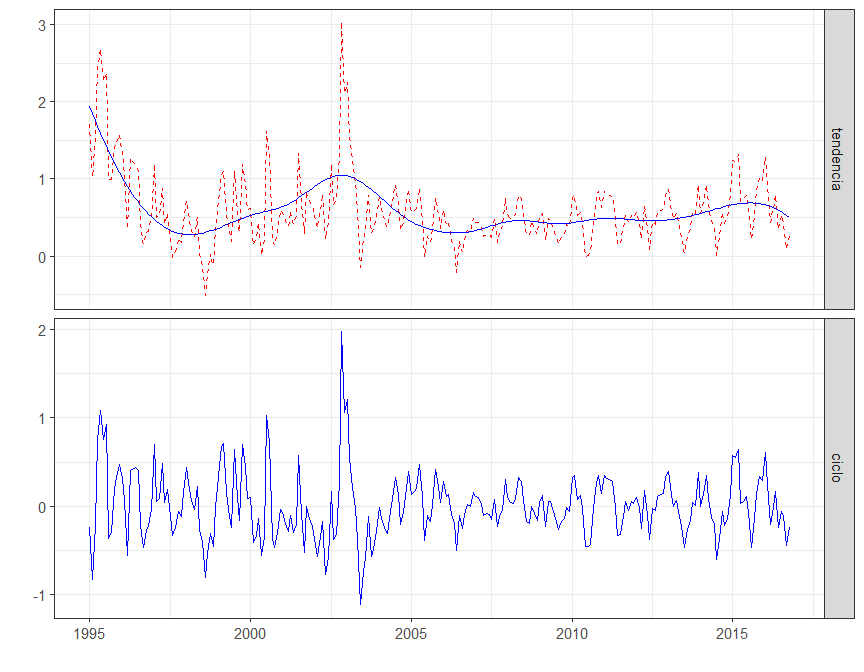
## Escolha da tendencia

# media das reconstrucoes parciais   
fc <- vector()  
rp <- demd  
for(i in 1:(length(demd[1,])-1)){  
 rp[,i]<- apply(as.matrix(demd[,1:i]), 1, sum)  
 fc[i] <- mean(rp[,i])  
}  
  
# teste t  
rpt <- apply(rp, 2, function(x) t.test(x)$p.value)  
  
library(ggplot2)  
n <- length(demd[1,])  
df <- data.frame(media=fc, rpt=n2tab(rpt[-n]), d=1:(n-1))  
g <- ggplot(df, aes(y = media, x = d)) +  
 geom\_line(linetype="dashed", alpha=.6) +  
 geom\_point() +   
 geom\_hline(yintercept=0) +  
 scale\_x\_continuous(breaks=1:max(df$d)) +  
 geom\_text(aes(y=media+.5\*mean(media), label=rpt, vjust=0), size=4.5, family="Times", position = "dodge") +  
 theme\_bw(base\_size = 14)  
g



## Núcleo como tedência da inflação

core <- apply(as.matrix(demd[,-(1:5)]), 1, sum)  
attributes(core) <- attributes(ipca)  
u <- ipca - core  
x <- cbind(tendencia=core, ciclo=u)  
y <- cbind(ipca, NA)  
tsplot(x, y)



## Teste de estacionaridade

library(nimcno)  
x <- cbind(ipca, core, ipca95[,-1])  
tab <- tab.stationary(acum(x))  
knitr::kable(tab)

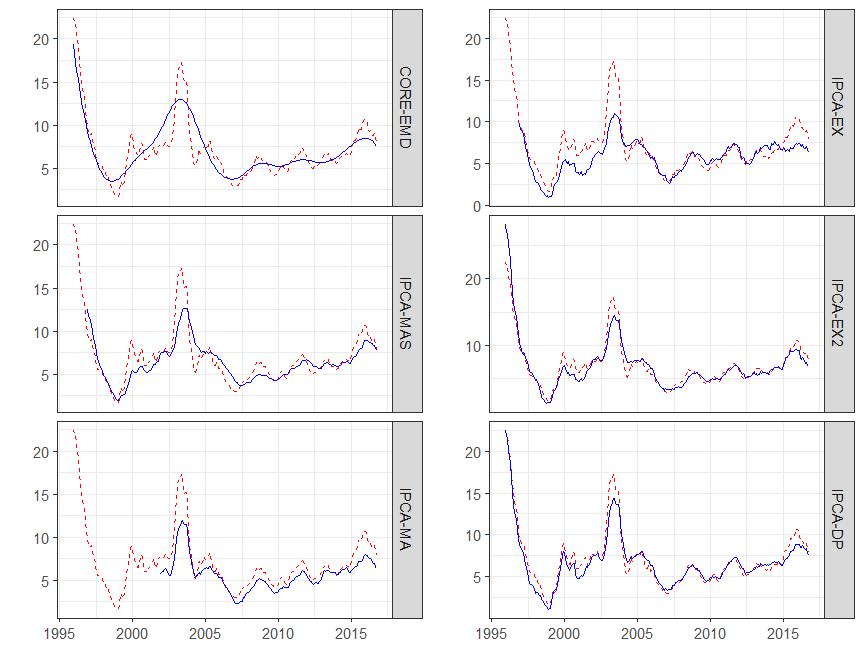
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Variavel | tendencia | ADF.lag | ADF | KPSS |
| ipca | sim | 14 | -3,235\* | 0,194\*\* |
| core | sim | 8 | -2,876 | 0,206\*\* |
| ipca95[, -1].ipca.mas | sim | 15 | -3,001 | 0,271\*\*\* |
| ipca95[, -1].ipca.ma | sim | 14 | -2,474 | 0,538\*\*\* |
| ipca95[, -1].ipca.ex | sim | 14 | -2,933 | 0,199\*\* |
| ipca95[, -1].ipca.ex2 | sim | 14 | -3,138\* | 0,195\*\* |
| ipca95[, -1].ipca.dp | sim | 14 | -3,478\*\* | 0,157\*\* |
| diff.ipca | nao | 13 | -4,448\*\*\* | 0,314 |
| diff.core | nao | 15 | -2,865\* | 0,660\*\* |
| diff.ipca95[, -1].ipca.mas | nao | 13 | -3,455\*\* | 0,326 |
| diff.ipca95[, -1].ipca.ma | nao | 13 | -4,309\*\*\* | 0,066 |
| diff.ipca95[, -1].ipca.ex | nao | 13 | -3,986\*\*\* | 0,161 |
| diff.ipca95[, -1].ipca.ex2 | nao | 13 | -4,231\*\*\* | 0,592\*\* |
| diff.ipca95[, -1].ipca.dp | nao | 13 | -4,040\*\*\* | 0,441\* |

x <- cbind(core, ipca95[,-1])  
tab <- tab.marques(y=acum(ipca), x=acum(x))  
knitr::kable(tab)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| nucleos | ADF | t.alpha | t.gamma | t.lambda | F.thetas |
| core | -4,084\*\*\* | 0,566 | 0,000 | 0,169 | 0,132 |
| ipca95[, -1].ipca.mas | -2,748\* | 0,226 | 0,096 | 0,397 | 0,000 |
| ipca95[, -1].ipca.ma | -4,031\*\*\* | 0,006 | 0,001 | 0,039 | 0,044 |
| ipca95[, -1].ipca.ex | -1,688 | 0,306 | 0,346 | 0,249 | 0,001 |
| ipca95[, -1].ipca.ex2 | -3,185\*\* | 0,136 | 0,054 | 0,795 | 0,013 |
| ipca95[, -1].ipca.dp | -2,237 | 0,354 | 0,370 | 0,875 | 0,091 |

## Grafico das series

library(grid)  
library(gridExtra)  
x <- cbind(core, ipca95[,-1])  
colnames(x) <- c("CORE-EMD", "IPCA-MAS", "IPCA-MA", "IPCA-EX", "IPCA-EX2", "IPCA-DP")  
y <- ipca  
p1 <- tsplot(acum(x[,1:3]), acum(y))  
p2 <- tsplot(acum(x[,4:6]), acum(y))  
grid.arrange(p1, p2, ncol = 2)



## Teste de Previsão

# Previsao fora da amostra  
library(pimfc)

##   
## Attaching package: 'pimfc'

## The following object is masked from 'package:nimcno':  
##   
## acum

library(zoo)

## Warning: package 'zoo' was built under R version 3.3.2

##   
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':  
##   
## as.Date, as.Date.numeric

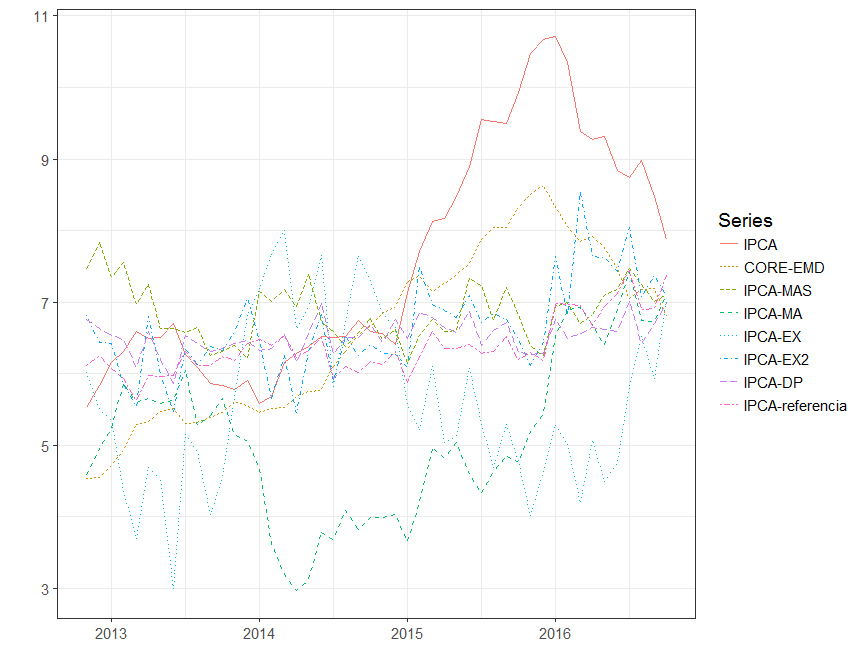
cores <- x  
cores <- acum(cores)  
ipca12 <- acum(ipca)  
  
out12.mdd <- vector("list")  
for(i in 1:ncol(cores)){  
 out12.mdd[[i]] <- outsample.mdd(yh=ipca12, yt=ipca12, x=cores[ ,i], m=6, p=6, n=48, h=12)  
}  
out12.mdd[[i+1]] <- outsample.mdd(yh=ipca12, yt=ipca12, x=NULL, m=6, p=6, n=48, h=12)  
  
# matriz de dados com as previsoes  
x <-sapply(out12.mdd, function(x) x$fcast)  
atr <- tsp(out12.mdd[[1]]$fcast)  
x <- ts(x, start = atr[1], end = atr[2], frequency = atr[3])  
  
# ipca em 12 meses no periodo fora da amostra  
ipca12 <- window(acum(ipca), start=start(x), end=end(x))  
  
colnames(x) <- c("core", colnames(ipca95[,-1]), "benchmark")  
dados <- cbind(ipca12, x)  
  
tab1 <- tab.reqm(dados, obs = "ipca12", ref = "x.benchmark")  
tab2 <- tab.enctest(dados, obs = "ipca12", ref = "x.benchmark")

## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)  
## Called from: model.matrix.dyn(x)  
## debug: cat("WE ARE IN model.matrix.dyn", "\n")  
## WE ARE IN model.matrix.dyn   
## debug: model.matrix(terms(object), model.frame(object), ...)

tab <- cbind(tab1[,-3], c("",tab2[,3]))  
knitr::kable(tab)

|  |  |  |  |
| --- | --- | --- | --- |
|  | reqm | eqmr |  |
| x.benchmark | 1,84 | 1,00 |  |
| x.core | 1,19 | 0,41 | 1,21 (0,00) |
| x.ipca.mas | 1,77 | 0,92 | -0,92 (0,21) |
| x.ipca.ma | 2,85 | 2,38 | 0,42 (0,30) |
| x.ipca.ex | 2,98 | 2,62 | -0,65 (0,06) |
| x.ipca.ex2 | 1,63 | 0,78 | 1,01 (0,00) |
| x.ipca.dp | 1,87 | 1,02 | -1,38 (0,19) |

colnames(dados) <-c("IPCA", "CORE-EMD", "IPCA-MAS", "IPCA-MA", "IPCA-EX", "IPCA-EX2", "IPCA-DP", "IPCA-referencia")  
tsplot(dados, facet = F)



## Plot emd shifting

library(EMD)

## Warning: package 'EMD' was built under R version 3.3.2

## Loading required package: fields

## Warning: package 'fields' was built under R version 3.3.2

## Loading required package: spam

## Warning: package 'spam' was built under R version 3.3.2

## Spam version 1.4-0 (2016-08-29) is loaded.  
## Type 'help( Spam)' or 'demo( spam)' for a short introduction   
## and overview of this package.  
## Help for individual functions is also obtained by adding the  
## suffix '.spam' to the function name, e.g. 'help( chol.spam)'.

##   
## Attaching package: 'spam'

## The following objects are masked from 'package:base':  
##   
## backsolve, forwardsolve

## Loading required package: maps

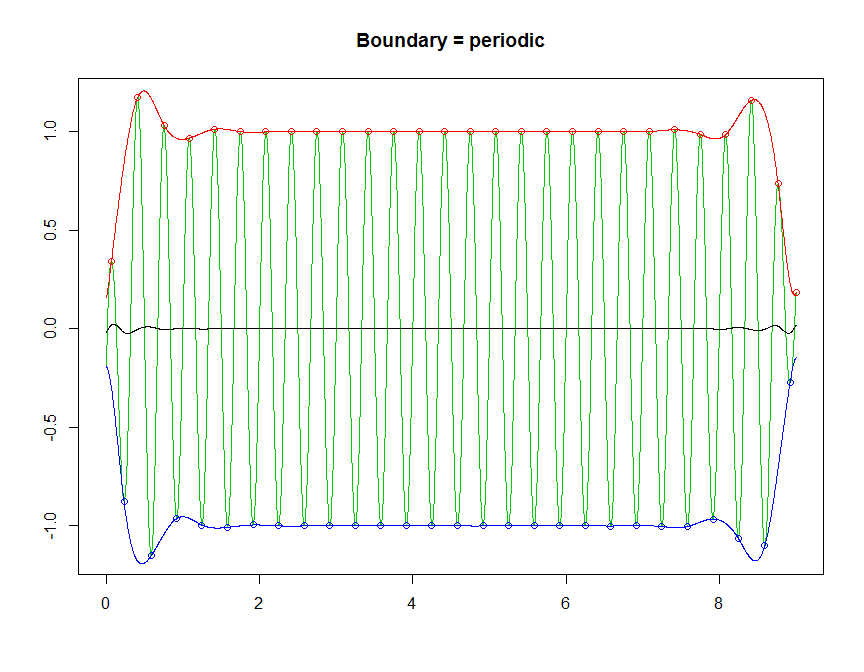
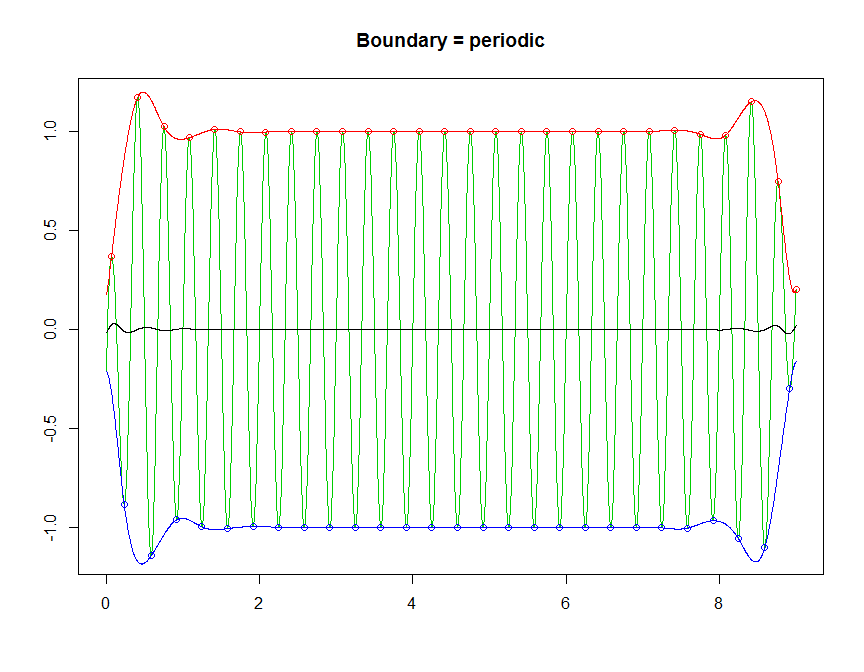
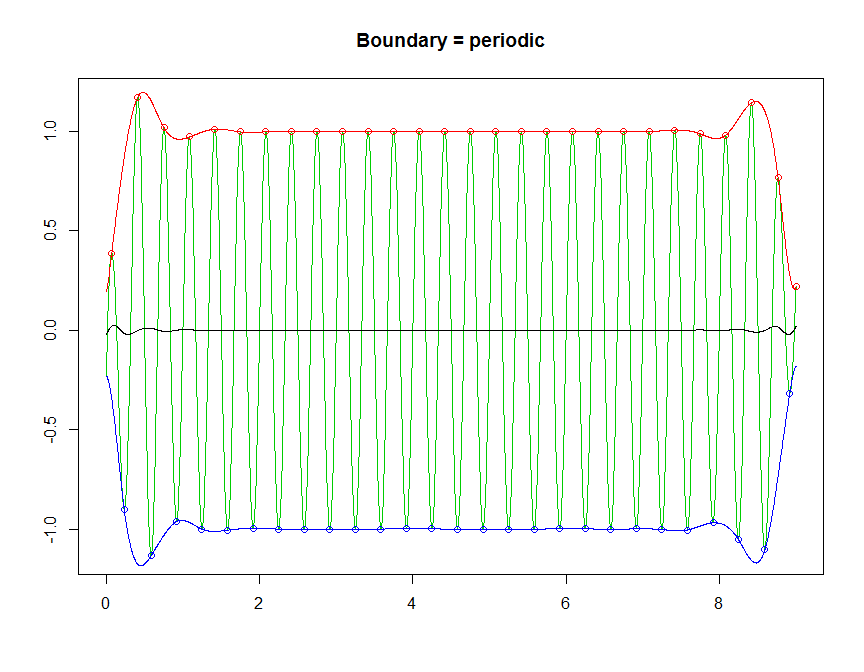
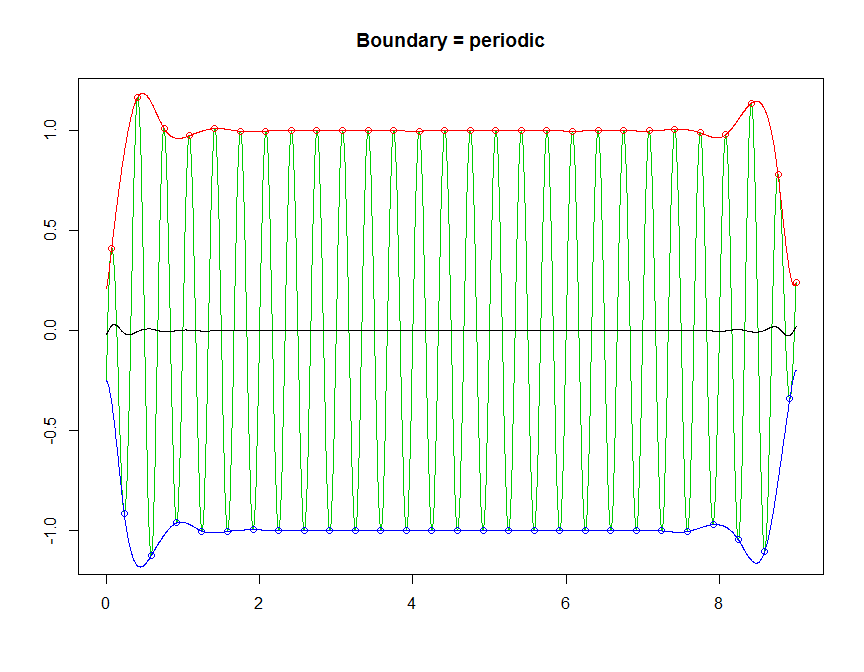
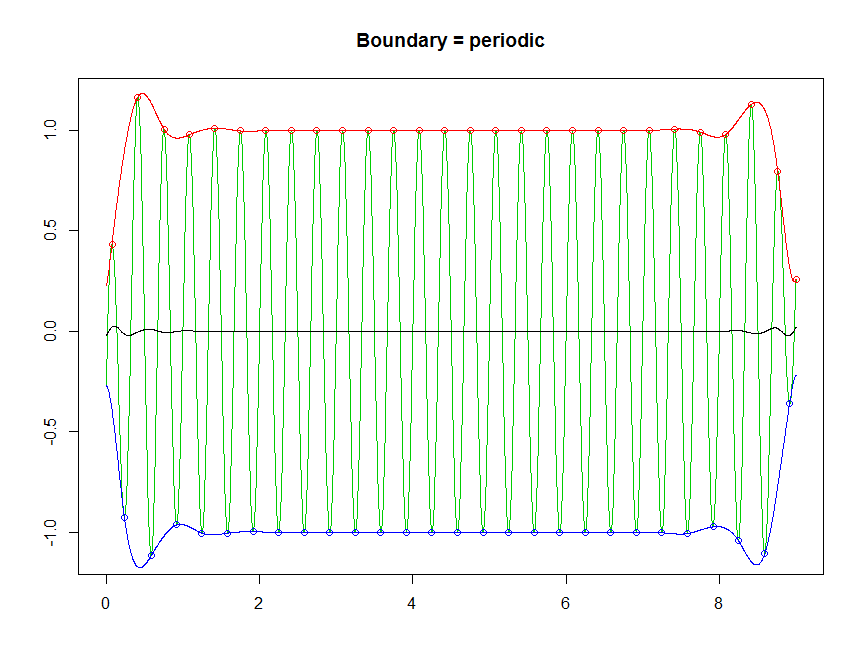
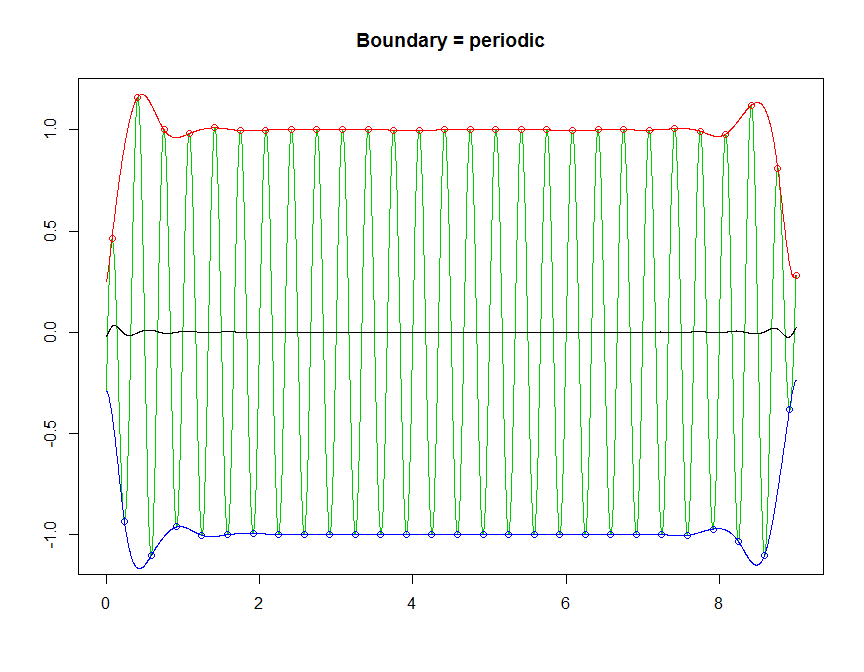
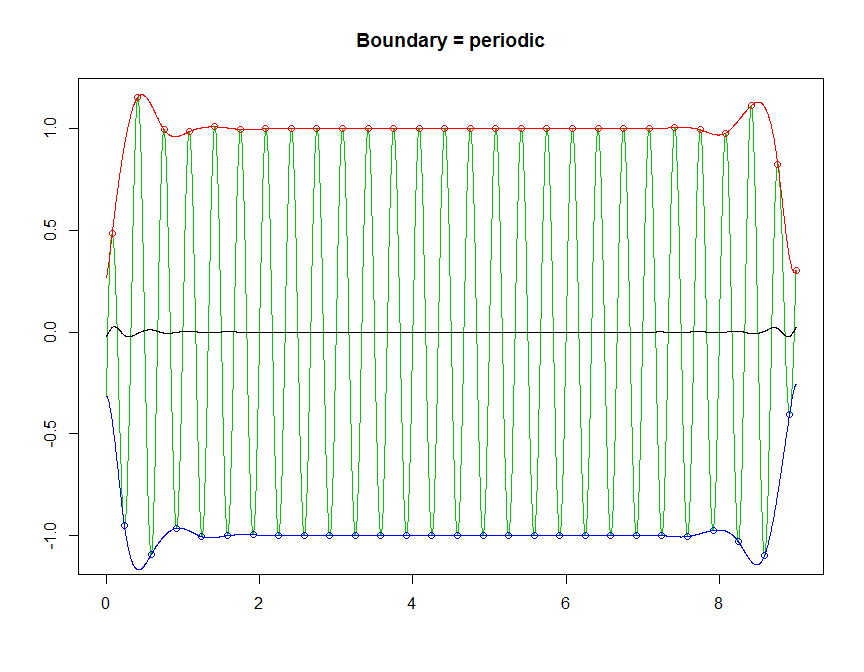
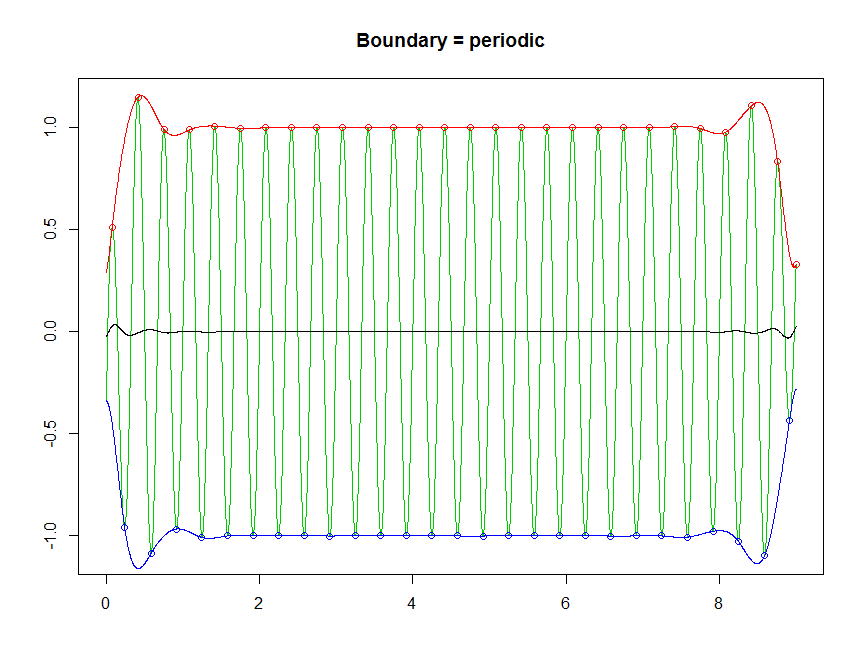
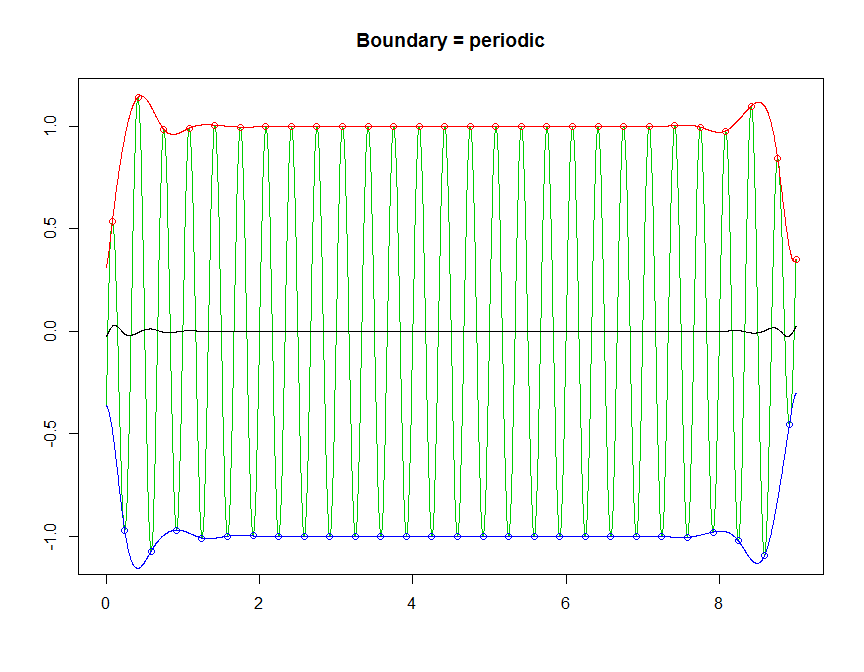
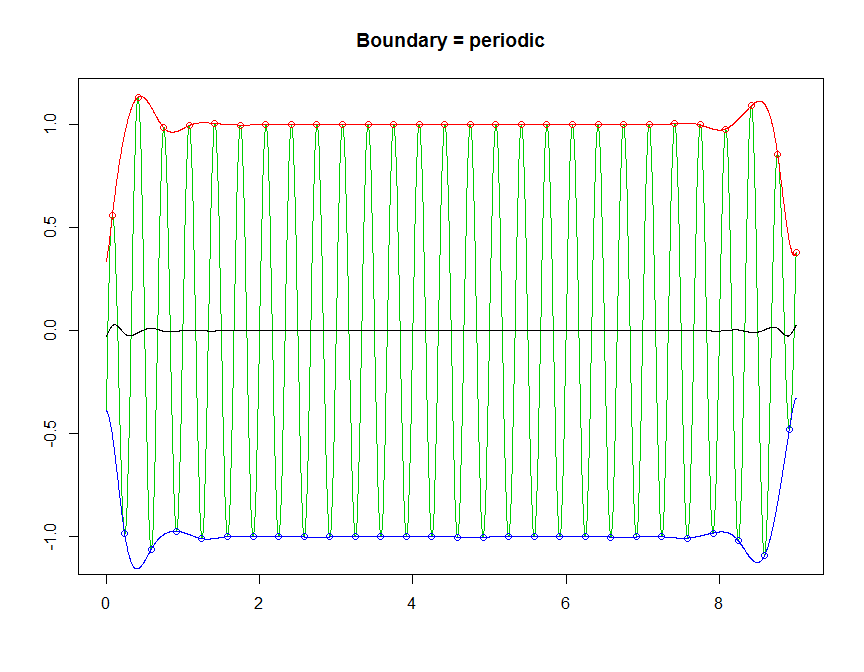
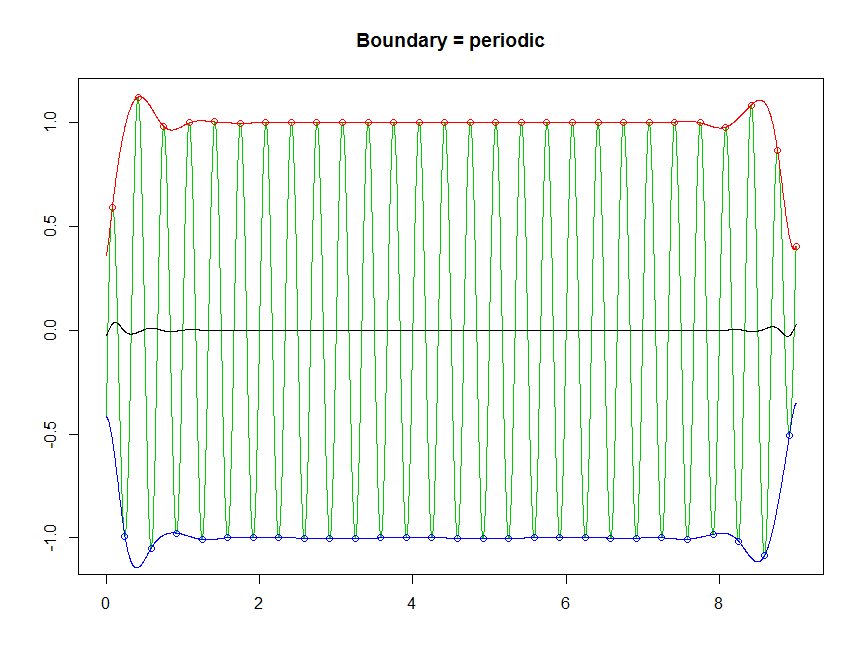
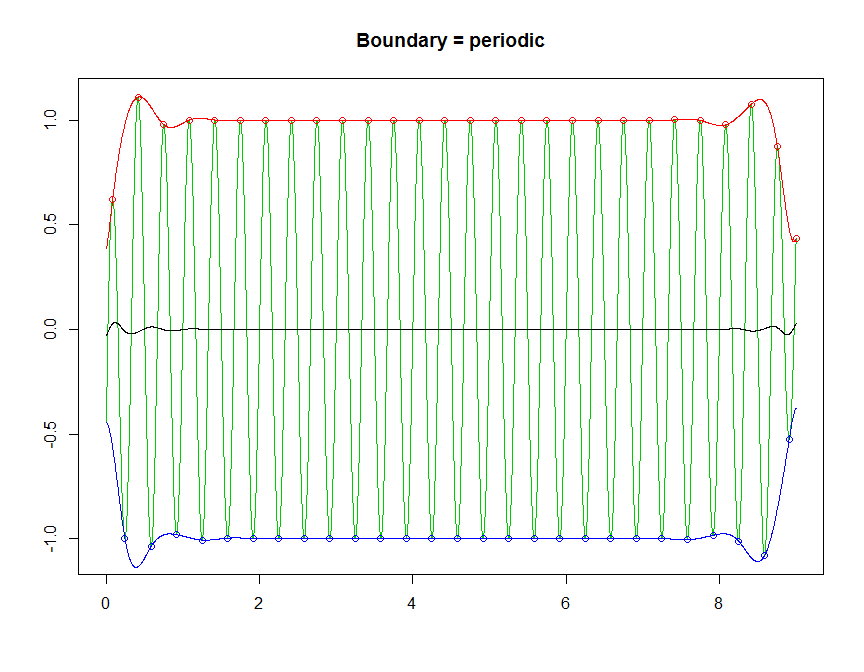
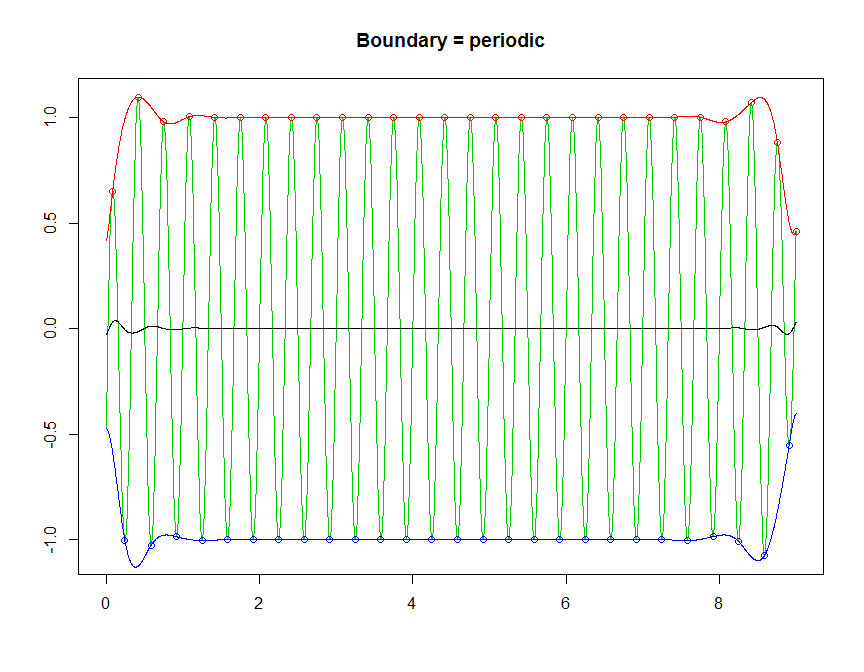
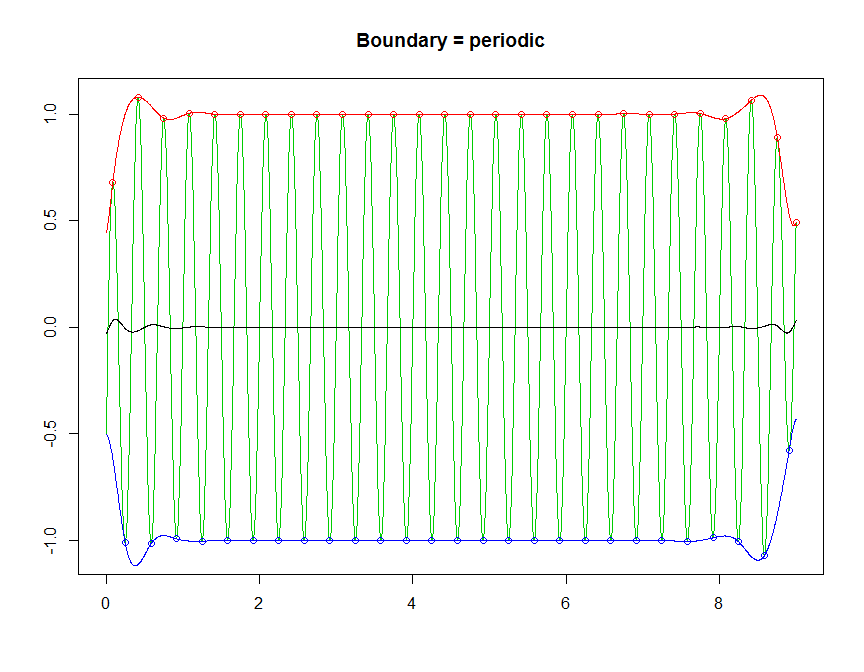
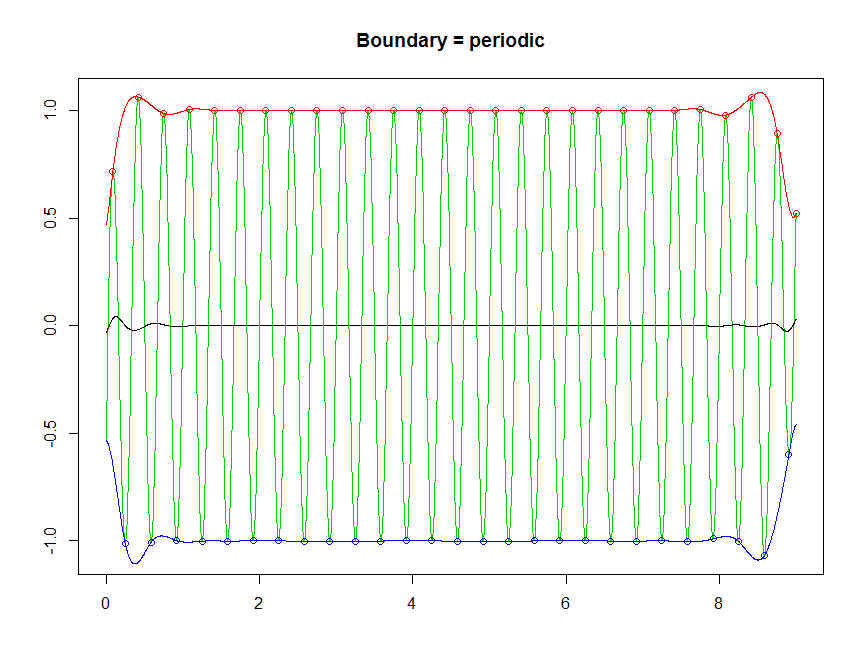
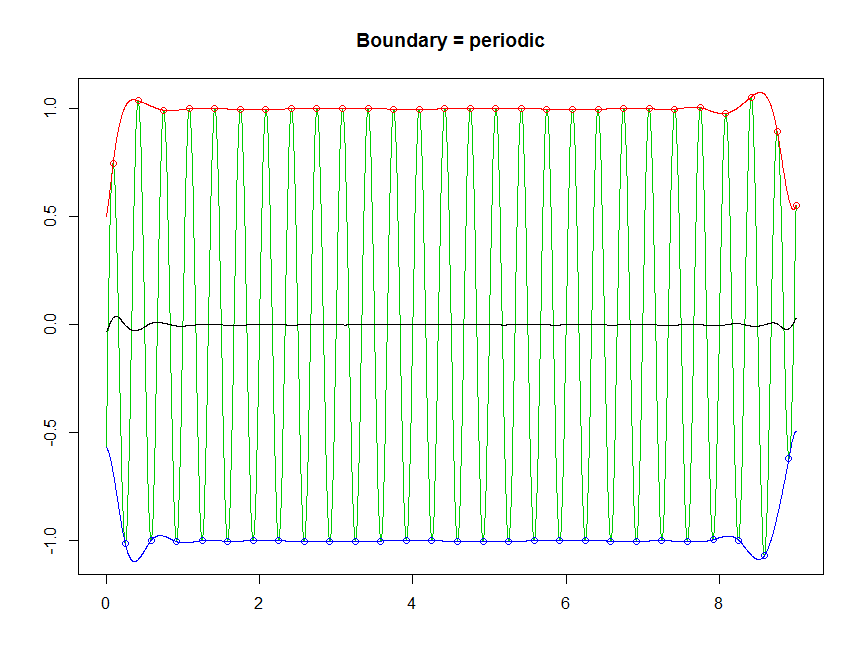
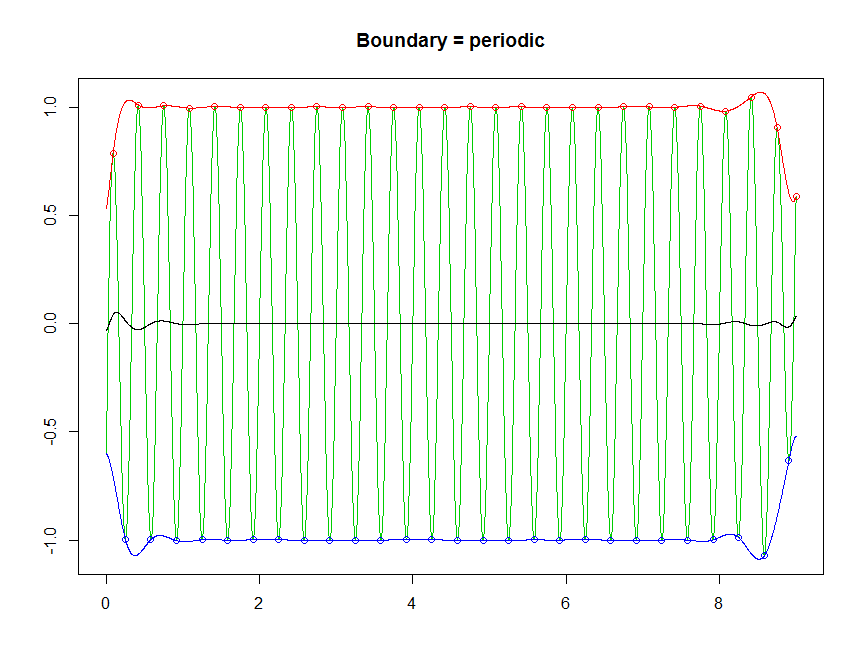
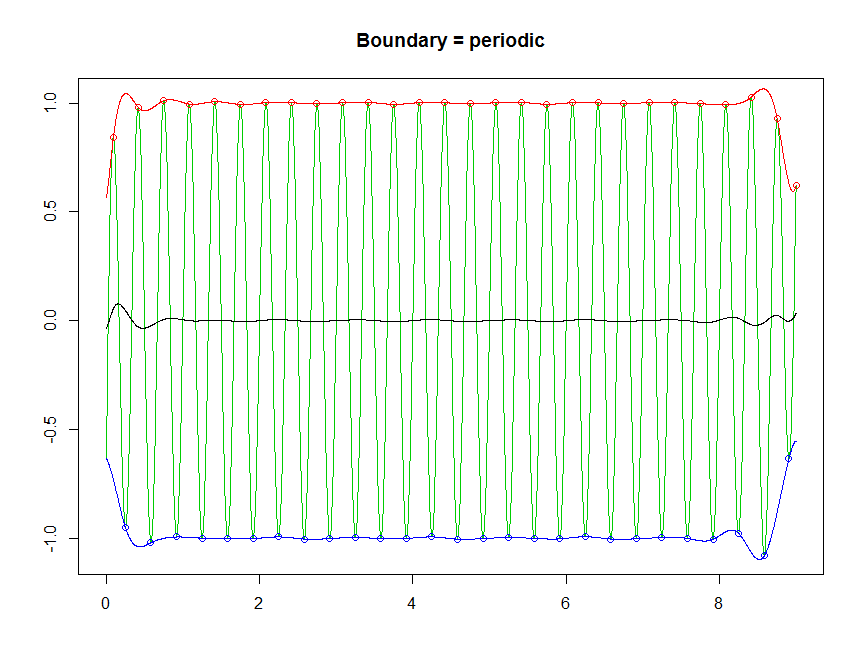
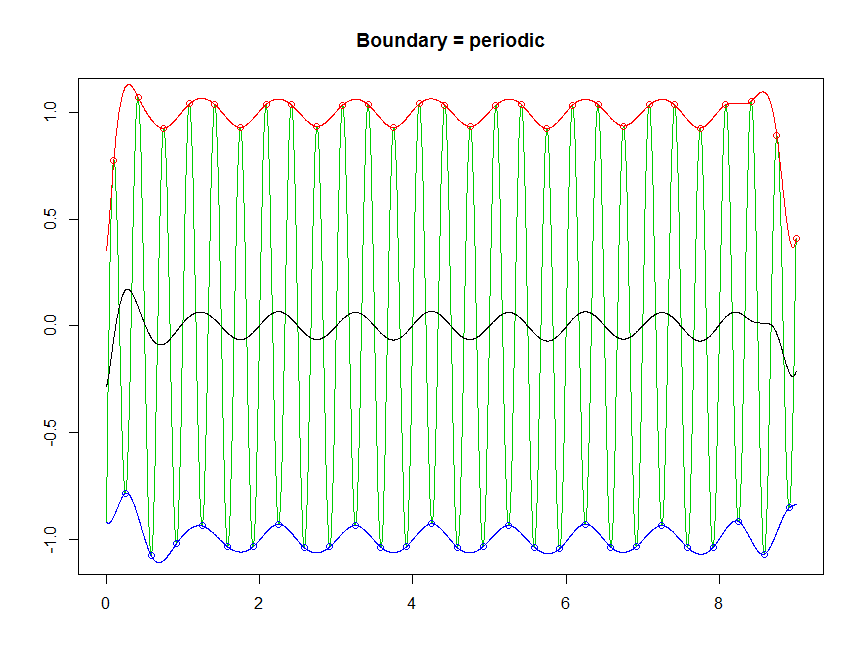
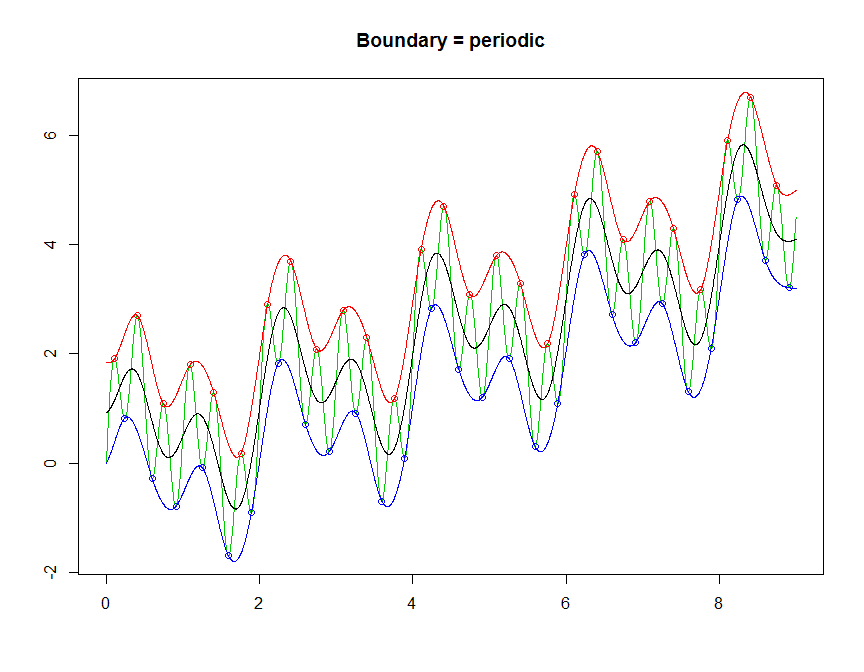
## Warning: package 'maps' was built under R version 3.3.2

## Loading required package: locfit

## Warning: package 'locfit' was built under R version 3.3.2

## locfit 1.5-9.1 2013-03-22

ndata <- 3000  
tt2 <- seq(0, 9, length=ndata)  
xt2 <- sin(pi \* tt2) + sin(2\* pi \* tt2) + sin(6 \* pi \* tt2) + 0.5 \* tt2  
  
tryimf <- extractimf(xt2, tt2, check = TRUE)



emin <- tryimf$emin[ ,1]  
emax <- tryimf$emax[ ,1]  
em <- tryimf$em[ ,1]  
  
  
aux <- extrema(xt2)  
idmin <- aux$minindex[,1]  
idmax <- aux$maxindex[,1]  
pmin <- xt2[idmin]  
pmax <- xt2[idmax]  
extr <- c(rep("min", length(pmin)),rep("max", length(pmax)))  
df1 <- data.frame(n1 = "(a)", n2 = "(b)", n3 = "(c)", n4 = "(d)",  
 tt2, xt2, emin, emax, em, resid=xt2-em)  
df2 <- data.frame(extr, p=c(pmin, pmax), id=c(tt2[idmin], tt2[idmax]))  
  
library(ggplot2)  
  
g1 <- ggplot(df1, aes(y = xt2, x=tt2)) +  
 geom\_line(colour = "green") +  
 geom\_point(data=df2, aes(y = p, x = id, colour = extr, shape=extr)) +  
 theme\_bw(base\_size = 14) +  
 theme(legend.position="none") +  
 labs(x="", y="") +  
 scale\_colour\_hue(l=45) +  
 facet\_grid(n1 ~. )+  
 theme(strip.text.y = element\_text(angle = 0), axis.text.x = element\_blank())  
  
g2 <- ggplot(df1, aes(x = tt2)) +  
 geom\_line(aes(y = xt2), colour = "green") +  
 geom\_line(aes(y = emax), colour = "red", linetype=2) +   
 geom\_line(aes(y = emin), colour = "blue", linetype=3) +  
 geom\_point(data=df2, aes(y = p, x = id, colour = extr, shape=extr)) +  
 theme\_bw(base\_size = 14) +  
 theme(legend.position="none") +  
 labs(x="", y="") +  
 scale\_colour\_hue(l=45) +  
 facet\_grid(n2 ~ .)+  
 theme(strip.text.y = element\_text(angle = 0), axis.text.x = element\_blank())  
  
g3 <- ggplot(df1, aes(x = tt2)) +  
 geom\_line(aes(y = xt2), colour = "green") +  
 geom\_line(aes(y = em), colour = "black", linetype=4) +  
 theme\_bw(base\_size = 14) +  
 theme(legend.position="none") +  
 labs(x="", y="") +  
 facet\_grid(n3 ~ .)+  
 theme(strip.text.y = element\_text(angle = 0), axis.text.x = element\_blank())  
  
g4 <- ggplot(df1, aes(x = tt2)) +  
 geom\_line(aes(y = resid), colour = "green") +  
 theme\_bw(base\_size = 14) +  
 theme(legend.position="none") +  
 labs(x="", y="") +  
 ylim(c(-2,4)) +  
 facet\_grid(n4 ~ .) +  
 theme(strip.text.y = element\_text(angle = 0))  
  
library(gridExtra)  
grid.arrange(g1, g2, g3, g4, ncol=1)

