Algorithmic Analysis for L0TF

2023-04-10

This file replicates the numerical experiments in Section 4.1 of the following paper:

ℓ_0 Trend Filtering by Canhong Wen, Xueqin Wang, Aijun Zhang.

1 Load Packages and Functions Needed

```
library(AMIAS)
library(ggplot2)
library(ggpubr)
```

```
## Warning: package 'ggpubr' was built under R version 3.6.3
```

```
library("gridExtra")
```

```
## Warning: package 'gridExtra' was built under R version 3.6.3
```

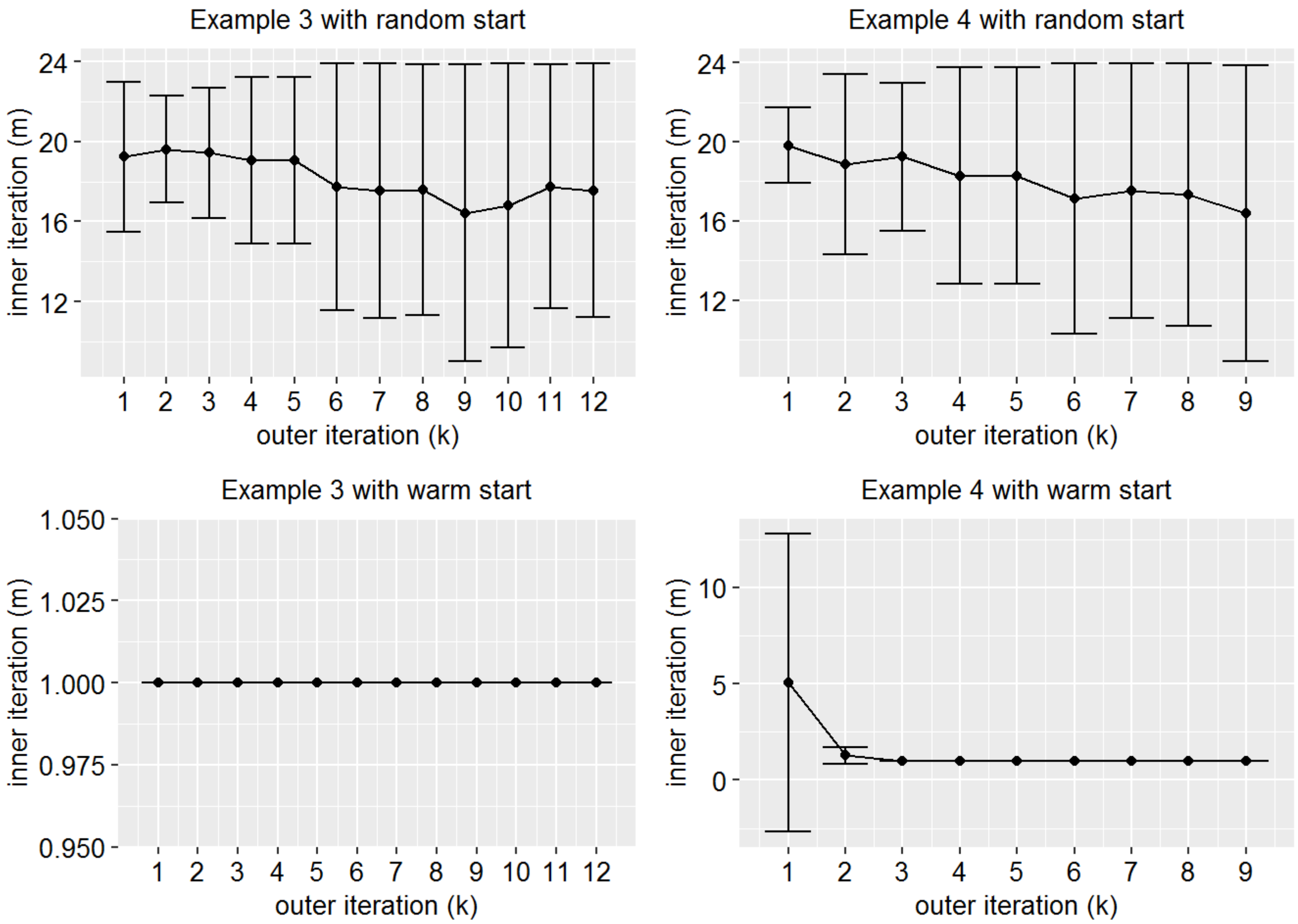
```
source("utils.R")
source("amiasutils.R")
```

2 Figure 8: Error bars of inner iterations against the outer iterations in the sequential AMIAS algorithm

```
c3 <- c4 <- c5 <- c6 <- c()
for(i in 1:100){
  if(i%%20==0) print(i)
  data3 = SimuEx(n=300, sigma=0.1, q=0, nknot=8, seed=i)
  data4 = SimuEx(n=300, sigma=0.1, q=1, nknot=5, seed=i)
  res1 = ramias(data3, kmax = data3$nknot+4)
  res2 = ramias(data4, kmax = data4$nknot+4)
  res3 = samias_R(y = data3$y, D = DiffMat(data3$n, data3$q+1), kmax = data3$nknot+4, rho = data3$n**(data3$q+1), q =
data3$q)
  res4 = samias_R(y = data4$y, D = DiffMat(data4$n, data4$q+1), kmax = data4$nknot+4, rho = data4$n**(data4$q+1), q =
data4$q)
  c3 = rbind(c3, res1$iters)
  c4 = rbind(c4, res2$iters)
  c5 = rbind(c5, res3$iters)
  c6 = rbind(c6, res4$iters)
}
p1 = inout_loop(c3, "Example 3 with random start", 'outer iteration (k)', 'inner iteration (m)')
p2 = inout_loop(c4, "Example 4 with random start", 'outer iteration (k)', 'inner iteration (m)')
p3 = inout_loop(c5, "Example 3 with warm start", 'outer iteration (k)', 'inner iteration (m)')
p4 = inout_loop(c6, "Example 4 with warm start", 'outer iteration (k)', 'inner iteration (m)')

save.image("samias_conv_inout.RData")
```

```
load("samias_conv_inout.RData")
# png("figs/samias_conv_inout.png", pointsize=8, width=850, height=850, res=120)
ggarrange(p1, p2, p3, p4, nrow=2, ncol = 2)
```




```
# dev.off()
```

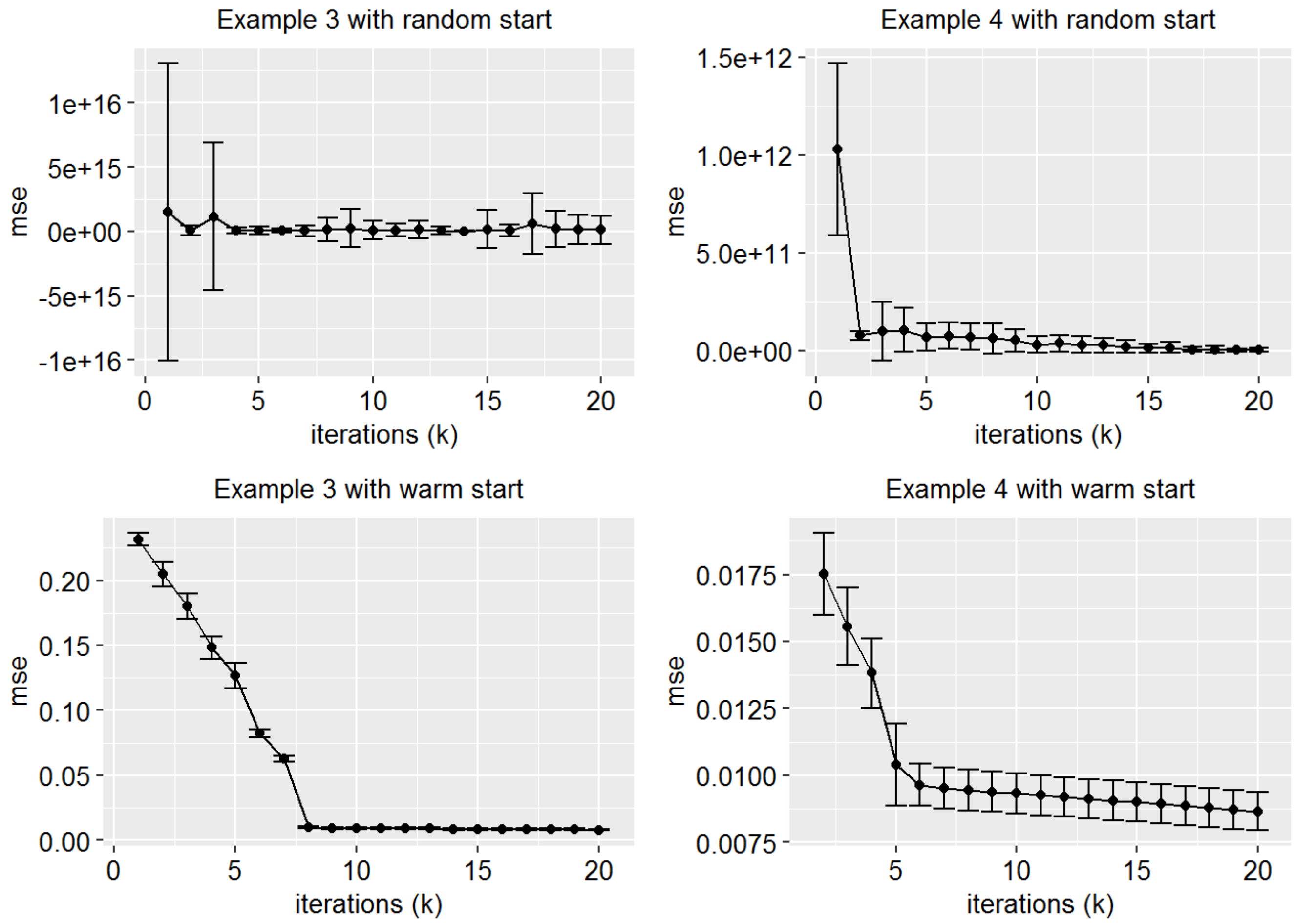
3 Figure 9: Error bars of MSE over cardinality parameters

```
c3 <- c4 <- c5 <- c6 <- c()
for(i in 1:100){
  if(i%%20==0) print(i)
  data3 = SimuEx(n=300, sigma=0.1, q=0, nknot=8, seed=i)
  data4 = SimuEx(n=300, sigma=0.1, q=1, nknot=5, seed=i)
  res1 = ramias(data3, kmax = data3$nknot+12)
  res2 = ramias(data4, kmax = data4$nknot+15)
  res3 = samias_R(y = data3$y, D = DiffMat(data3$n, data3$q+1), kmax = data3$nknot+12, rho = data3$n**(data3$q+1), q =
data3$q)
  res4 = samias_R(y = data4$y, D = DiffMat(data4$n, data4$q+1), kmax = data4$nknot+15, rho = data4$n**(data4$q+1), q =
data4$q)
  c3 = rbind(c3, res1$mse)
  c4 = rbind(c4, res2$mse)
  c5 = rbind(c5, res3$mse)
  c6 = rbind(c6, res4$mse)
}

p1 = convsteps(t(c3), 1:(data3$nknot+12), "Example 3 with random start", "iterations (k)", "mse")
p2 = convsteps(t(c4), 1:(data4$nknot+15), "Example 4 with random start", "iterations (k)", "mse")
p3 = convsteps(t(c5), 1:(data3$nknot+12), "Example 3 with warm start", "iterations (k)", "mse")
p4 = convsteps(t(c6[,-1]), 2:(data4$nknot+15), "Example 4 with warm start", "iterations (k)", "mse") # cut first col

save.image("samias_mse_conv.RData")
```

```
load("samias_mse_conv.RData")
# png("figs/samias_mse_conv.png", pointsize=8, width=850, height=850, res=120)
ggarrange(p1, p2, p3, p4, nrow=2, ncol = 2)
```




```
# dev.off()
```

4 Figure 10: Scatterplot of number of the detected knots against sample size *n*

```
nlist <- seq(100, 2000, 100)
re <- re_o <- re_u <- kest <- mse <- array(0, c(length(nlist),100, 2))

for(n in nlist){
  print(n)
  for(seed in 1:100){

    par(mfrow=c(1,2),mar=c(3,3,3,3))
    # Piecewise constant case
    sigma=0.1; q=0; nknot=8;
    data = SimuEx(n=n, sigma=sigma, q=q, nknot=nknot, seed=seed)
    resL0 = samias(as.numeric(data$y), D_type="tf0", kmax=nknot+4)
    #
    mse[n/100,seed,1] <- mean((as.numeric(data$y)-resL0$alpha)^2)
    kest[n/100,seed,1] <- resL0$kopt
    re[n/100,seed,1] <- nknot==resL0$kopt
    re_o[n/100,seed,1] <- nknot<resL0$kopt
    re_u[n/100,seed,1] <- nknot>resL0$kopt

    sigma=0.1; q=1; nknot=5;
    data = SimuEx(n=n, sigma=sigma, q=q, nknot=nknot, seed=seed)
    resL0 = samias(as.numeric(data$y), D_type="tfq", q=q, kmax=nknot+4, adjust = TRUE )
    mse[n/100,seed,2] <- mean((as.numeric(data$y)-resL0$alpha)^2)

    kest[n/100,seed,2] <- resL0$kopt

    re[n/100,seed,2] <- nknot==resL0$kopt
    re_o[n/100,seed,2] <- nknot<resL0$kopt
    re_u[n/100,seed,2] <- nknot>resL0$kopt
  }
}

save.image("smias_k_n.RData")
```

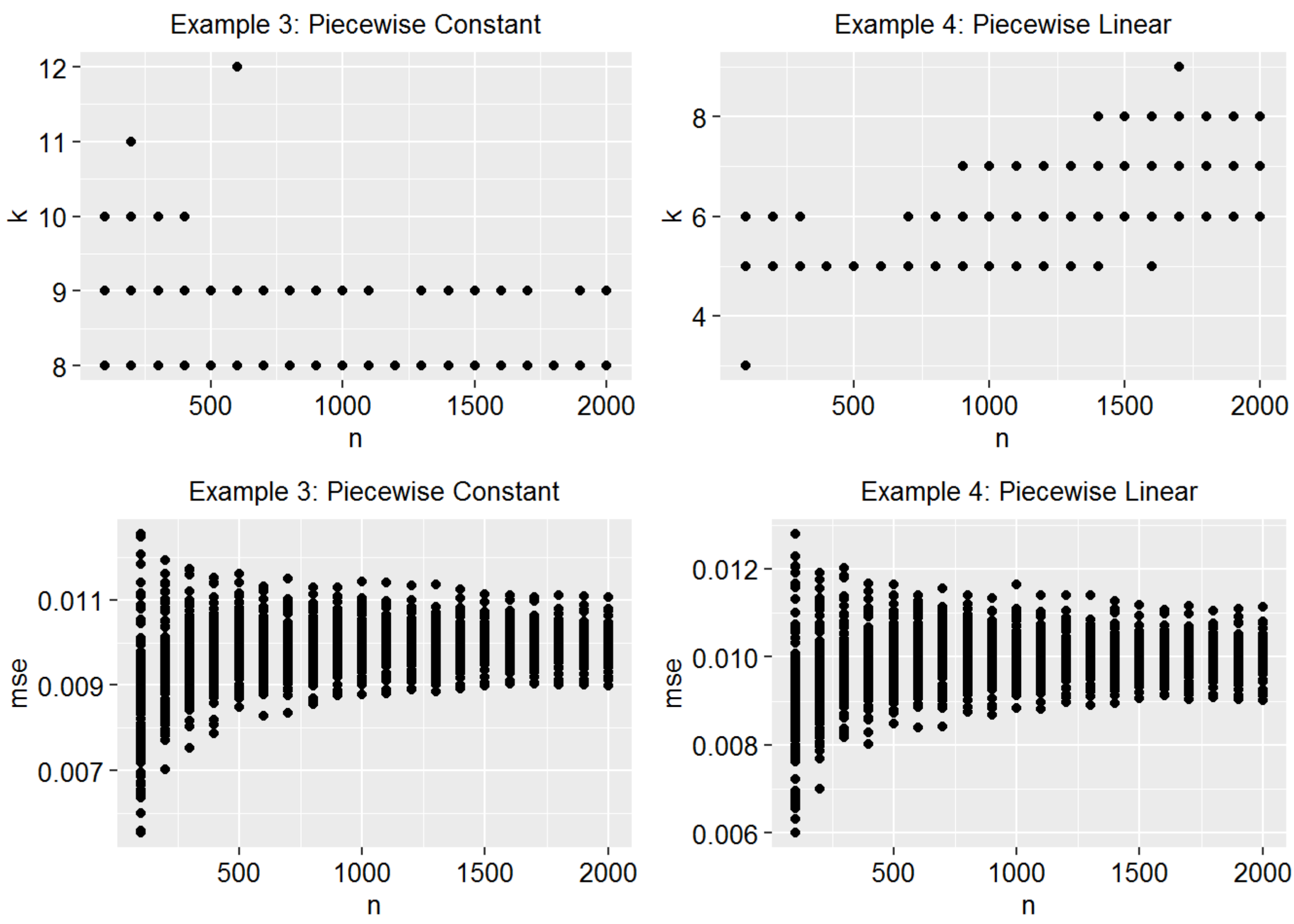
```
load("smias_k_n.RData")
library(ggplot2)
title <- c("Example 3: Piecewise Constant", "Example 4: Piecewise Linear")
p <- list()
for(j in 1:2){
t1 <- as.numeric(t(kest[,,j]))
data <- cbind(rep(nlist, each=100), t1)
data <- as.data.frame(data)
colnames(data) <- c("n", "k")

p[[j]] <- ggplot(data = data, aes( x= n, y= k)) + geom_point() +
      # geom_smooth(method=lm, se= FALSE, size=2, color = "deepskyblue") +
      ggtitle(title[j]) +
      theme(axis.text.x = element_text(size = 10,color="black"),
            axis.title.x = element_text(size = 10,color="black"),
            axis.text.y = element_text(size = 10,color="black"),
            axis.title.y = element_text(size = 10,color="black"),
            plot.title = element_text(size = 10, hjust = 0.5))
}

p2 <- list()
for(j in 1:2){
  t1 <- as.numeric(t(mse[,,j]))
  data <- cbind(rep(nlist, each=100), t1)
  data <- as.data.frame(data)
  colnames(data) <- c("n", "mse")

  p2[[j]] <- ggplot(data = data, aes( x= n, y= mse)) + geom_point() +
       ggtitle(title[j]) +
      theme(axis.text.x = element_text(size = 10,color="black"),
            axis.title.x = element_text(size = 10,color="black"),
            axis.text.y = element_text(size = 10,color="black"),
            axis.title.y = element_text(size = 10,color="black"),
            plot.title = element_text(size = 10, hjust = 0.5))
}

# png("figs/samias_k_n.png", pointsize=8, width=850, height=850, res=120)
grid.arrange(p[[1]], p[[2]], p2[[1]], p2[[2]], ncol=2)
```



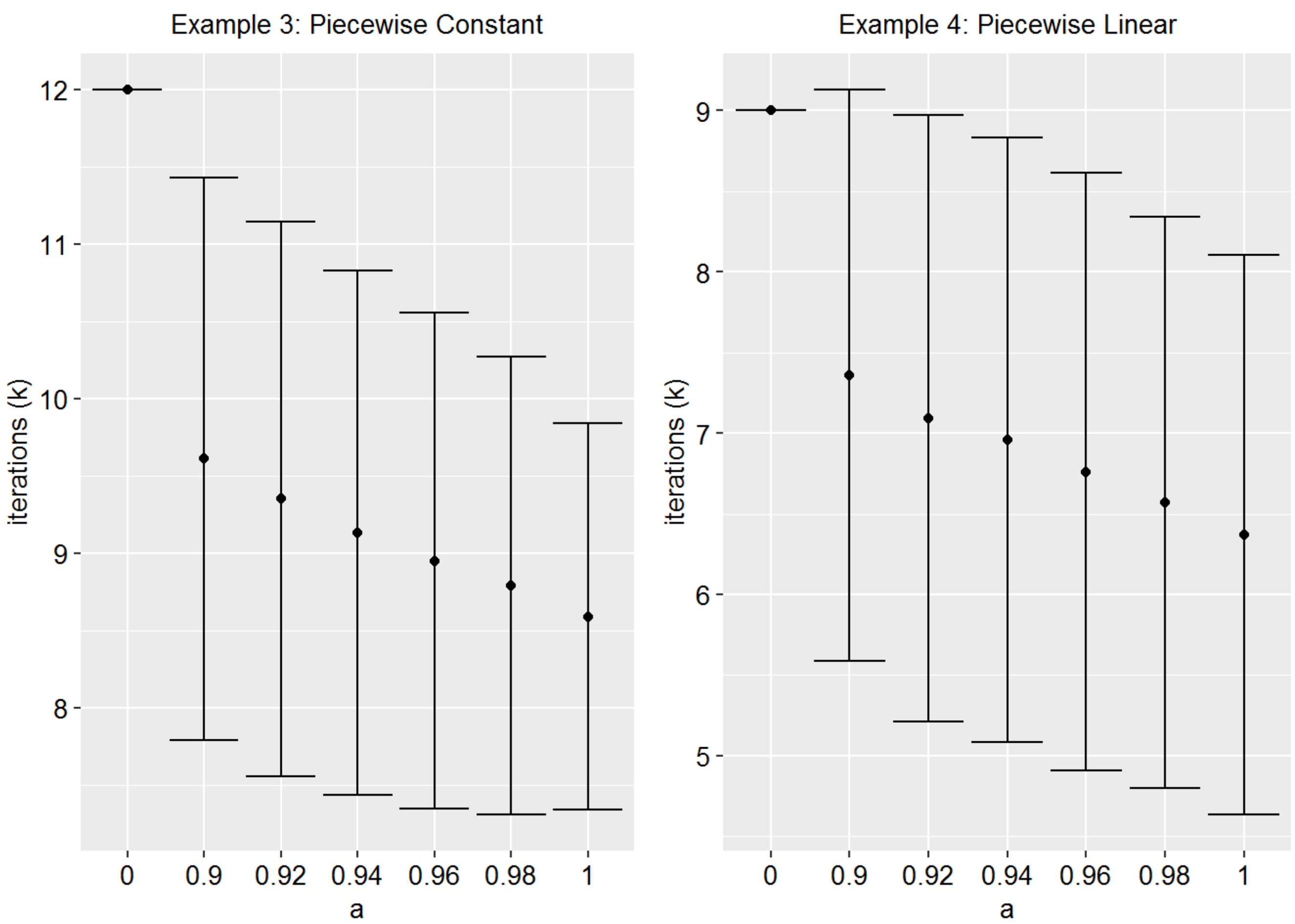

```
# dev.off()
```

5 Figure 11: Early stopping rule effect on the convergence of sequential AMIAS algorithm

```
# samias eps vs convergence
c1 = c2 = c()
for(i in 1:100){
  if(i%%20==0) print(i)
  data3 = SimuEx(n=300, sigma=0.1, q=0, nknot=8, seed=i)
  data4 = SimuEx(n=300, sigma=0.1, q=1, nknot=5, seed=i)
  r1 <- r2 <- c()
  sig1 <- median(abs(diff(data3$y,diff=1)))/(qnorm(3/4)*sqrt(choose(2,1)))
  sig2 <- median(abs(diff(data4$y,diff=2)))/(qnorm(3/4)*sqrt(choose(4,2)))

  eps1 <- c(0,seq(0.9,1,0.02))*(sig1^2)
  eps2 <- c(0,seq(0.9,1,0.02))*(sig2^2)
  for(eps in eps1){
    res1 = samias_R(y = data3$y, D = DiffMat(data3$n, data3$q+1), kmax = data3$nknot+4, rho = data3$n**(data3$q+1), q = data3$q, eps=eps)
    r1 <- c(r1, res1$outiter)
  }
  for(eps in eps2){
    res2 = samias_R(y = data4$y, D = DiffMat(data4$n, data4$q+1), kmax = data4$nknot+4, rho = data4$n**(data4$q+1), q = data4$q, eps=eps)
    r2 <- c(r2, res2$outiter)
  }
  c1 <- rbind(c1, r1)
  c2 <- rbind(c2, r2)
}
epsl <- c(0,seq(0.9,1,0.02))
p1 = convsteps(t(c1), as.character(epsl), "Example 3: Piecewise Constant", "a", "iterations (k)")
p2 = convsteps(t(c2), as.character(epsl), "Example 4: Piecewise Linear", "a", "iterations (k)")
save.image("samias_eps_conv.RData")
```

```
load("samias_eps_conv.RData")
# png("figs/samias_eps_conv.png", pointsize=8, width=850, height=400, res=120)
ggarrange(p1, p2, ncol = 2)
```


Example 3: Piecewise Constant
Example 4: Piecewise Linear
iterations (k)
iterations (k)
12
11
10
9
8
9
8
7
6
5
0
0.9
0.92
0.94
0.96
0.98
1
a
0
0.9
0.92
0.94
0.96
0.98
1
a

```
# dev.off()
```