Adaptive temporal intensity

First example: Temporal point pattern with a hotspot

```
# Setting a simulation of temporal point pattern with a hotspot
# intensity for Nt points
Nt <- 2000
y1 <- rnorm(Nt, 10, 0.1)

# fixed bandwidth estimate
classic.dens <- density.default(y1, from = min(y1), to = max(y1))
classic.dens$y <- classic.dens$y * Nt
adapt.dens <- dens.par.temp(y1, at = "bins", dimt = 512)
true.dens <- Nt * dnorm(classic.dens$x, 10, 0.1)

##ISE
ISE.adapt <- (sum(adapt.dens$y - true.dens) ^ 2) * diff(adapt.dens$x[1:2])
ISE.classic <- (sum(classic.dens$y - true.dens) ^ 2) * diff(adapt.dens$x[1:2])

PD <- data.frame(x = rep(adapt.dens$x, 3),
                 intensity = c(classic.dens$y, adapt.dens$y, true.dens),
                 estimator = factor(rep(1:3, rep(512,3)), levels = 1:3,
                                    labels = c("classical", "adaptive", "true")))
ggplot(data = PD, aes(x = x, y = intensity, group = estimator, colour = estimator)) +
  geom_path() + theme(axis.title.x = element_blank())
```

Second example: Temporal point pattern with three black holes

```
# Setting a simulation of a high clustered temporal point pattern
# Probability density function
fdens.x <- function(x) (dbeta(x %% 4, 2, 2))
# intensity for Nt points
Nt <- 2000
x <- runif(Nt, 0, 10)
# temporal point pattern
y <- sample(x, replace = T, prob = fdens.x(x))
# fixed bandwidth estimate
classic.dens <- density.default(y, from = min(y), to = max(y))
classic.dens$y <- classic.dens$y * Nt
# Global bandwidth (we give such a refined one because of the high clustering)
bw0 <- bw.nrd0(y) / 4
# Abram's bandwidth
bw1 <- bw.abram.temp(y, h0 = bw0, trim = 2)
# Adaptive intensity
adapt.dens <- dens.par.temp(y, bw = bw1, at = "bins", dimt = 512)
true.dens <- Nt * fdens.x(adapt.dens$x)

##ISE
ISE.adapt <- (sum(adapt.dens$y - true.dens) ^ 2) * diff(adapt.dens$x[1:2])
ISE.classic <- (sum(classic.dens$y - true.dens) ^ 2) * diff(adapt.dens$x[1:2])
```

```
PD <- data.frame(x = rep(adapt.dens$x, 3),
                 intensity = c(classic.dens$y, adapt.dens$y, true.dens),
                 estimator = factor(rep(1:3, rep(512,3)), levels = 1:3,
                                    labels = c("classical", "adaptive", "true")))
ggplot(data = PD, aes(x = x, y = intensity, group = estimator, colour = estimator)) +
  geom_line() + theme(axis.title.x = element_blank())
```

Third example: Aegiss temporal intensity

```
# Load aegiss data-set and plotting
data(aegiss)
plot(aegiss, bg = rainbow(512), pch = 21, cex = 1,
     main = "Gastrointestinal desease cases in Hampshire")
```

```
# Fixed bandwidth estimate
ti <- aegiss$marks
Nt <- aegiss$n
# Classical estimate
classic.dens <- density.default(ti, from = min(ti), to = max(ti))
classic.dens$y <- classic.dens$y * Nt
# Adaptive estimate
adapt.dens <- dens.par.temp(ti, at = "bins", dimt = 512)

aegissD <- data.frame(x = rep(adapt.dens$x, 2),
                 intensity = c(classic.dens$y, adapt.dens$y),
                 estimator = factor(rep(1:2, rep(512,2)), levels = 1:2,
                                    labels = c("classical", "adaptive")))
ggplot(data = aegissD, aes(x = x, y = intensity,
                           group = estimator, colour = estimator)) +
  geom_line() + theme(axis.title.x = element_blank())
```