RenDate

Ph Dufresne

07/02/2020

Pour faire un pdf install.packages('tinytex') tinytex::install_tinytex() # Chargement des fonctions de RenDate.R ##via GitHub

```
## Loading required package: devtools
```

```
## Loading required package: usethis
```

```
## Downloading GitHub repo chrono35/RenDate@master
```

```
##
   checking for file '/private/var/folders/ms/3r6m3pqn4jq1hk94t646qdd00000gn/
T/RtmpW3HnTm/remotes53a62b9a733d/chrono35-RenDate-b7f2be4/DESCRIPTION' ...

✓  checking for file '/private/var/folders/ms/3r6m3pqn4jq1hk94t646qdd00000gn/
T/RtmpW3HnTm/remotes53a62b9a733d/chrono35-RenDate-b7f2be4/DESCRIPTION'
##

─  preparing 'RenDate':
##

✓  checking DESCRIPTION meta-information
##

─  checking for LF line-endings in source and make files and shell scripts
##

─  checking for empty or unneeded directories
## ─  looking to see if a 'data/datalist' file should be added
##

     NB: this package now depends on R (>= 3.5.0)
##

     WARNING: Added dependency on R >= 3.5.0 because serialized objects in  s
erialize/load version 3 cannot be read in older versions of R.  File(s) conta
ining such objects: 'RenDate/data/GAL2002sph2014_D.rda'  WARNING: Added depen
dency on R >= 3.5.0 because serialized objects in  serialize/load version 3 c
annot be read in older versions of R.  File(s) containing such objects:  'Ren
Date/data/GAL2002sph2014_D_BP.rda'  WARNING: Added dependency on R >= 3.5.0 b
ecause serialized objects in  serialize/load version 3 cannot be read in olde
r versions of R.  File(s) containing such objects: 'RenDate/data/GAL2002sph20
14_I.rda'  WARNING: Added dependency on R >= 3.5.0 because serialized objects
in  serialize/load version 3 cannot be read in older versions of R.  File(s)
containing such objects:  'RenDate/data/GAL2002sph2014_I_BP.rda' 'RenDate/dat
a/IntCal13.rda'  WARNING: Added dependency on R >= 3.5.0 because serialized o
bjects in  serialize/load version 3 cannot be read in older versions of R.  F
ile(s) containing such objects: 'RenDate/data/IntCal13_AD.rda'  'RenDate/data
/IntCal13_BP.rda'  WARNING: Added dependency on R >= 3.5.0 because serialized
objects in  serialize/load version 3 cannot be read in older versions of R.
File(s) containing such objects: 'RenDate/data/gwh2013uni_f.rda'  'RenDate/da
ta/gwh2013uni_f_BP.rda'
##

─  building 'RenDate_0.0.0.9000.tar.gz'
##



##
```

via source dans répertoire /R

```
# source('R/RenDate.R', echo=FALSE)
```

Memo

Quand on modifie le code , il faut penser à faire la documentation avec:

setwd("/Users/dufresne/Documents/projects/_Test_R/AMCalibrate") devtools::document()

Chargement des courbes de calibration

Chargement des fichiers de calibration AM

```
# ne pas oublier d'enlever les points de référence à la fin !!!
GAL2002sph2014_I <- read.table("GAL2002sph2014_I.csv", dec=',', sep=";",header=FALSE)
GAL2002sph2014_D <- read.table("GAL2002sph2014_D.csv", dec=',', sep=";",header=FALSE)

gwh2013uni_f <- read.table("gwh2013uni_f.csv", dec=',', sep=";",header=FALSE)
```

Chargement des fichiers de calibration 14C

```
 intCal13 <- read.csv("~/Documents/projects/_Test_R/AMCalibrate/intcal13.14c", header = FALSE,
                      comment.char="#")
```

Création et sauvegarde des fichiers rda. utilisés dans RenDate dans le répertoire courant

```
#library(Bchron)
# création courbe en AD/BC
createCalCurve("GAL2002sph2014_I", GAL2002sph2014_I$V1, GAL2002sph2014_I$V2,
GAL2002sph2014_I$V3 )
```

```
## Completed!
```

```
createCalCurve("GAL2002sph2014_D", GAL2002sph2014_D$V1, GAL2002sph2014_D$V2,
GAL2002sph2014_D$V3)
```

```
## Completed!
```

```
createCalCurve("gwh2013uni_f", gwh2013uni_f$V1, gwh2013uni_f$V2, gwh2013uni_
f$V3)
```

```
## Completed!
```

```
# création courbe en BP
createCalCurve("GAL2002sph2014_I_BP",GAL2002sph2014_I$V1 - 1950, GAL2002sph20
14_I$V2, GAL2002sph2014_I$V3)
```

```
## Completed!
```

```
createCalCurve("GAL2002sph2014_D_BP",GAL2002sph2014_D$V1 - 1950, GAL2002sph20
14_D$V2, GAL2002sph2014_D$V3)
```

```
## Completed!
```

```
createCalCurve("gwh2013uni_f_BP", gwh2013uni_f$V1 - 1950, gwh2013uni_f$V2, gw
h2013uni_f$V3)
```

```
## Warning in Ops.factor(gwh2013uni_f$V1, 1950): '-' not meaningful for facto
rs
```

```
## Completed!
```

```
intCal13_AD <- intCal13
intCal13_AD$V1 <- 1950 - intCal13_AD$V1
createCalCurve("IntCal13_AD", intCal13_AD$V1, intCal13_AD$V2, intCal13_AD$V3)
```

```
## Completed!
```

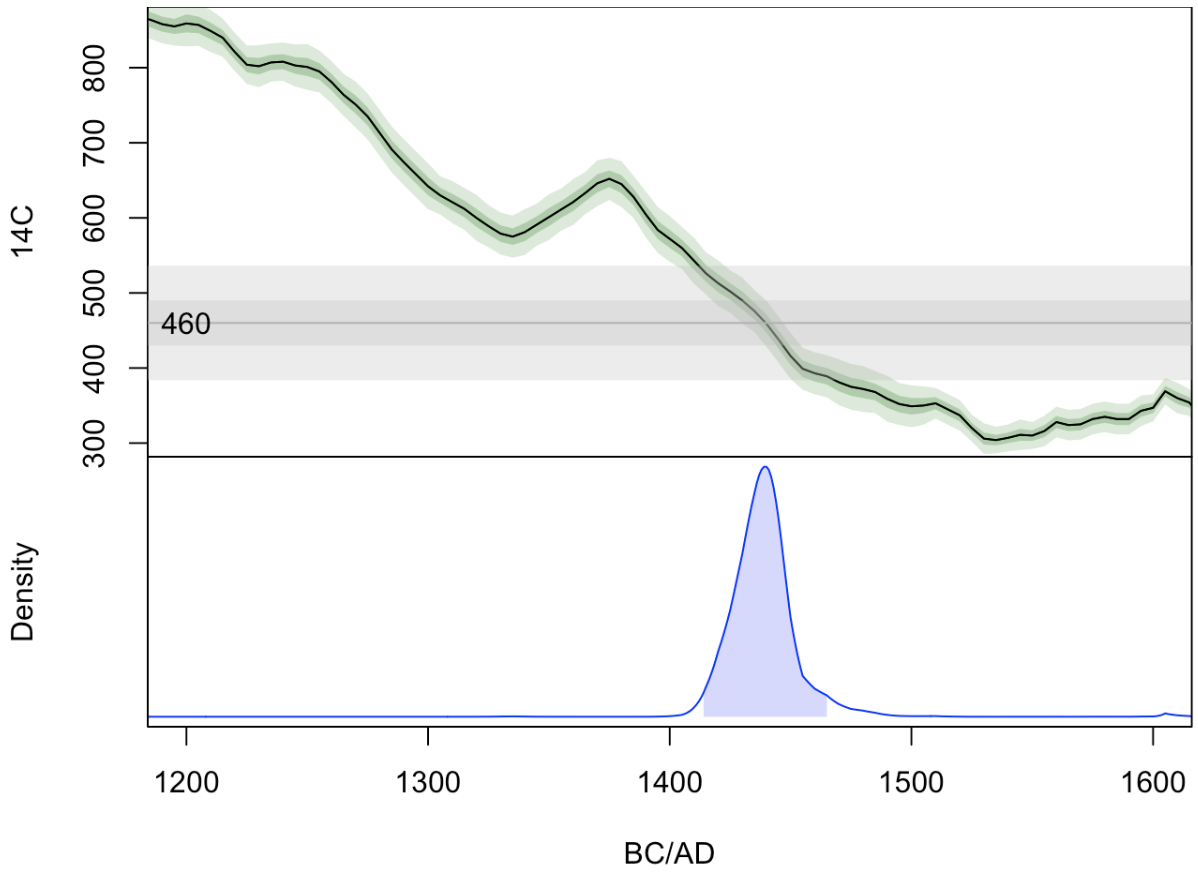
```
createCalCurve("IntCal13_BP", intCal13$V1, intCal13$V2, intCal13$V3)
```

```
## Completed!
```

Datation

Datation 14C

```r
C14 <- 460
errC14 <- 30
date14C <- calibrate(mesures = C14 ,std = errC14, calCurves = 'IntCal13_AD', ids = 'C14', timeScale = 1)

# Tracé de la courbe 14C
 # réduction de la période
tmin <- 1200
tmax <- 1600
xlim <- c(tmin, tmax)

# Il faut mettre des valeur qui existent dans le tableau
imin <- which(intCal13_AD$V1 == tmin)
imax <- which(intCal13_AD$V1 == tmax)
ylim <- range(intCal13_AD$V2[imin:imax] )

par( fig=c(0, 1, 0.50, 1), mar=c(0, 5, 0, 1))

courbe.enveloppe(t=intCal13_AD$V1, mean=intCal13_AD$V2, intCal13_AD$V3, ylab = '14C', xlab = NA,  xaxt = "n",
                 xlim = xlim, ylim = ylim)
mesure.enveloppe(intCal13_AD$V1, mesure = C14, std = errC14)
text(intCal13_AD$V1[imin], C14, labels=as.character(C14) )

par(fig=c(0, 1, 0.0, 0.50), new= TRUE, mar=c(5, 5, 0, 1) )

plot(date14C, col = "blue", hdrCol = adjustcolor( "blue", alpha.f = 0.2), main = NA, xlab ='BC/AD',
     xlim = xlim, yaxt="n")
```



```
print('Resultat pour le 14C')
```

```
## [1] "Resultat pour le 14C"
```

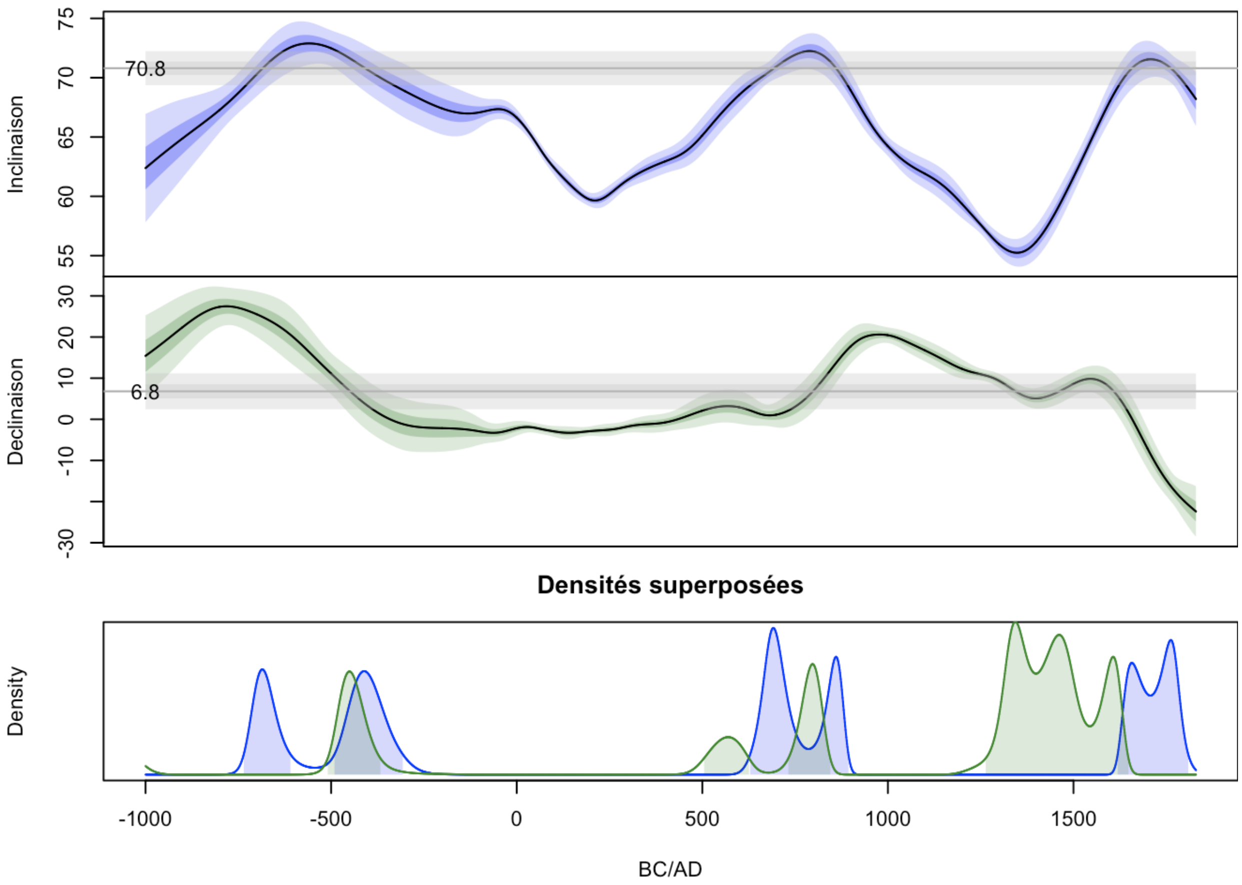
```
hpd(date14C$C14, prob = .95)
```

```
## $`94.8%`
## [1] 1414 1465
```

Datation AM

```
IParis = 70.8
DParis = 6.8
alpha95 = 1.38
# clcul des erreurs
errInc <- alpha95 /2.448
errDec <- alpha95/(2.448*cos(IParis*pi/180))

dateInc1 <- calibrate(mesures = IParis, std = errInc, calCurves='GAL2002sph2014_I', ids='Inclinaison', timeScale = 0.01)

dateDec1 <- calibrate(mesures = DParis , std = errDec, calCurves='GAL2002sph2014_D', ids='Declinaison', timeScale = 0.01)

# Exemple : plusieurs dates en même temps
#dateIncDec <- calibrate(mesures = c(IParis, DParis), std = c(errInc, errDec), calCurves = c('GAL2002sph2014_I', 'GAL2002sph2014_D'), ids = c('Inclinaison', 'Declinaison'), positions = c(1, 2) )
```

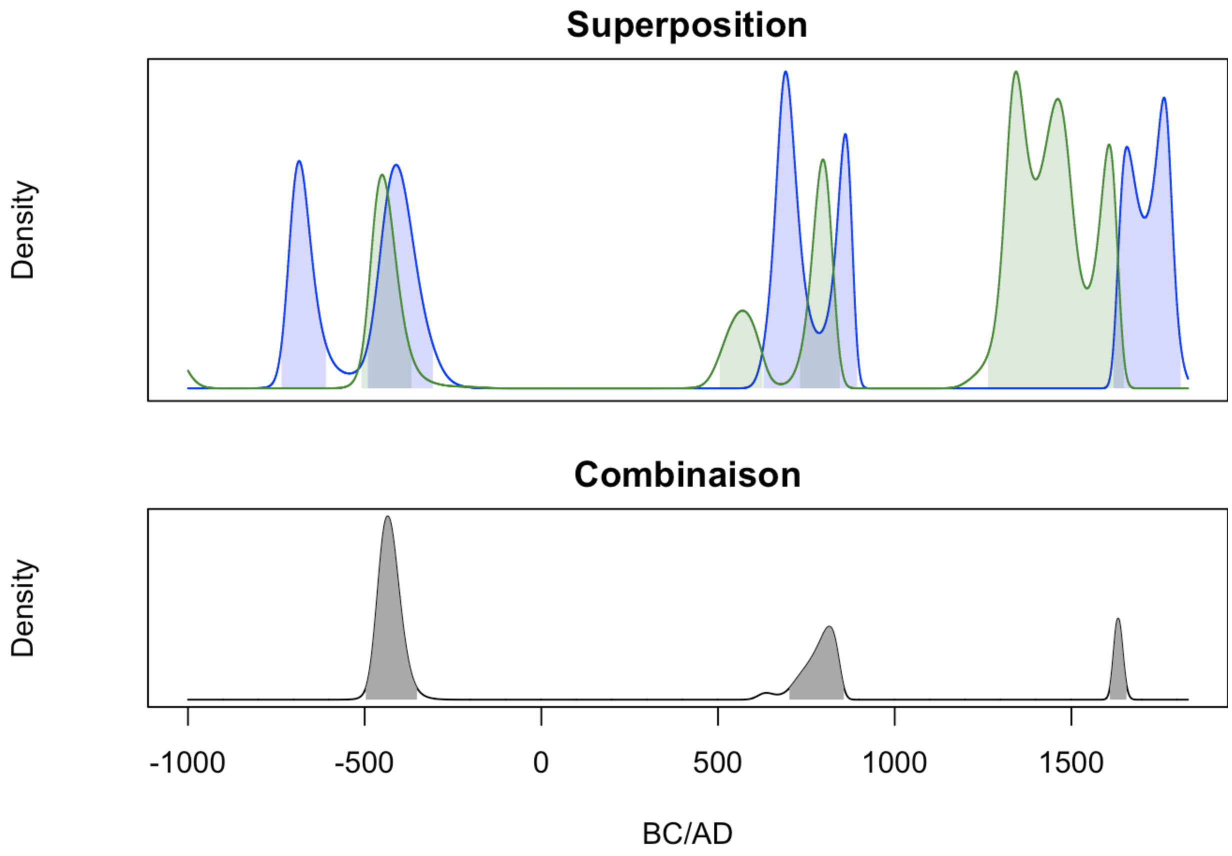
```
# Tracé de la superposition des deux densités obtenues mfrow = c(3,1),

# RenDate.plot(dateIncDec, col = "blue", hdrCol = adjustcolor( "blue", alpha.f = 0.2), main="Densités positionnées", withPosition=TRUE, fillCols = c(adjustcolor( "blue", alpha.f = 0.2), adjustcolor( "blue", alpha.f = 0.4)), dateHeight = 1 )

 par(cex=0.7)
# Tracé de la courbe Inclinaison
par( fig=c(0, 1, 0.70, 1), new=FALSE, mar=c(0, 5, 0.2, 0))
courbe.enveloppe(GAL2002sph2014_I$V1, GAL2002sph2014_I$V2, GAL2002sph2014_I$V3, ylab='Inclinaison', xlab=NA,  xaxt="n", col.env = "blue")
mesure.enveloppe(GAL2002sph2014_I$V1, mesure = IParis, std = errInc)
text(GAL2002sph2014_I$V1[1], IParis, labels=as.character(IParis) )
# Tracé de la courbe Déclinaison
par(fig=c(0, 1, 0.40, 0.70), new=TRUE, mar=c(0, 5, 0, 0))
courbe.enveloppe(GAL2002sph2014_D$V1, GAL2002sph2014_D$V2, GAL2002sph2014_D$V3, ylab='Declinaison', xlab=NA, xaxt="n" )
mesure.enveloppe(GAL2002sph2014_D$V1, mesure = DParis, std = errDec)
text(GAL2002sph2014_D$V1[1], DParis, labels=as.character(DParis) )

# Tracé de la superposition des deux densités obtenues
par(fig=c(0, 1, 0.0, 0.40), new=TRUE, mar=c(5, 5, 3, 0))
plot(dateInc1, col = "blue", hdrCol = adjustcolor( "blue", alpha.f = 0.2), main="Densités superposées", xlab='BC/AD', yaxt="n" )
lines(dateDec1, col="forestgreen", hdrCol = adjustcolor( "forestgreen", alpha.f = 0.2))
```



```
#print('Resultat pour l inclinaison')
#hpd(dateInc1$Inclinaison, prob = .95 )

#print('Resultat pour la declinaison')
#my_hdr =hpd(dateDec1$Declinaison, prob = .95)
```

Combinaison AM

```
combiAM<- produit.RenDate(dateInc1, dateDec1, timeScale = 1)

par(mfrow = c(2,1), mar=c(1, 5, 2, 0))

plot(dateInc1, col = "blue", hdrCol = adjustcolor( "blue", alpha.f = 0.2), xlab=NA, main='Superposition',  normalize = TRUE,  yaxt="n", xaxt="n")
lines(dateDec1, col="forestgreen", hdrCol = adjustcolor( "forestgreen", alpha.f = 0.2),  normalize = TRUE,  yaxt="n")

par( mar=c(5, 5, 2, 0))
val.hdr <- paste(hpd(combiAM$Combinaison, prob = .95), ' à 95% BC/AD' )
plot(combiAM, withHDR = TRUE, main='Combinaison' ,  normalize = TRUE, yaxt="n", xlab='BC/AD')
```

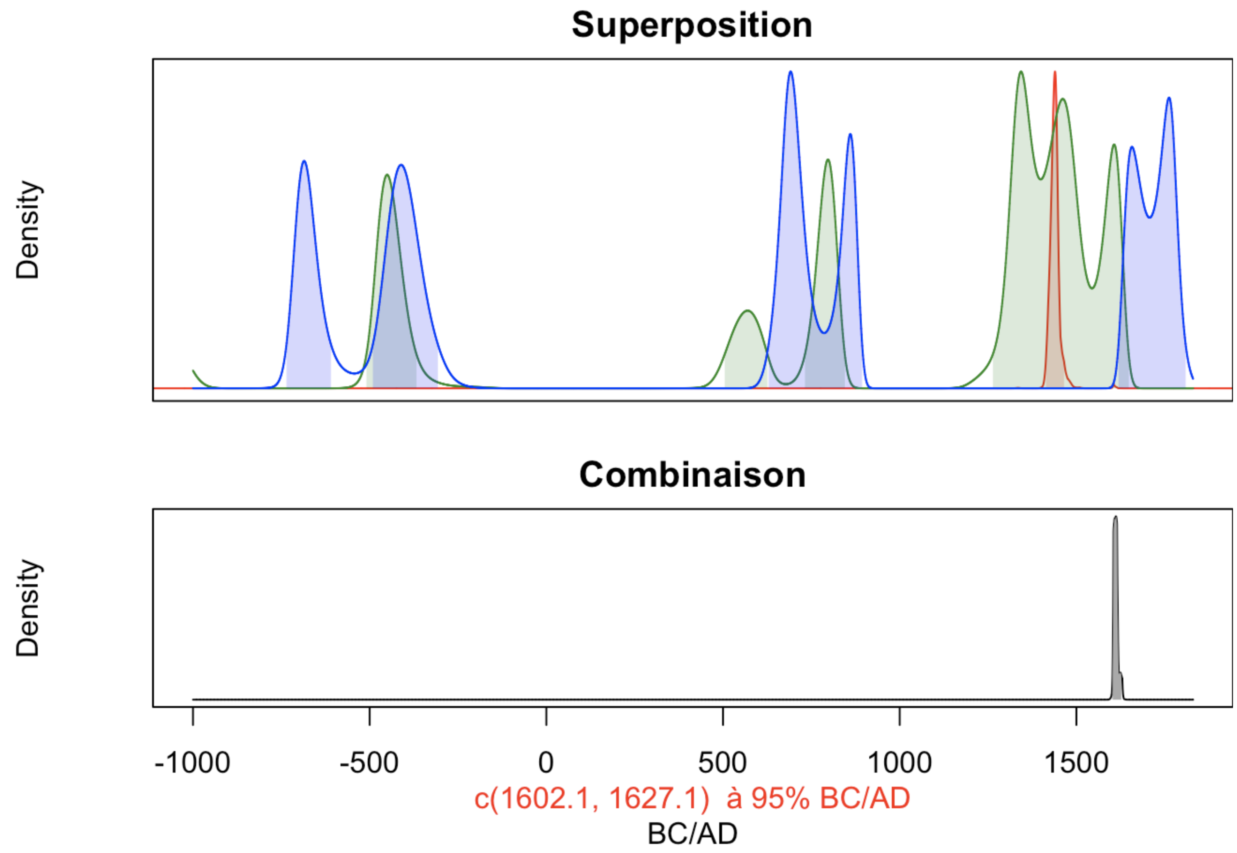


```
#mtext(val.hdr, side = 1, col = 'red')

hpd(combiAM$Combinaison, prob = .95)
```

```
## $`56.9%`
## [1] -497 -352
##
## $`28%`
## [1] 702 856
##
## $`10.1%`
## [1] 1609 1656
```

Combinaison AM et 14C

```
combiAMC14<- produit.RenDate(combiAM, date14C, timeScale = 0.1)

par(mfrow = c(2,1), mar=c(1, 5, 2, 0))

xlim <- range(combiAMC14$Combinaison$timeGrid)
plot(date14C, col = "red", hdrCol = adjustcolor( "red", alpha.f = 0.2), xlab=
NA, xlim = xlim, main='Superposition', normalize = TRUE,  yaxt="n", xaxt="n")
lines(dateDec1, col="forestgreen", hdrCol = adjustcolor( "forestgreen", alph
a.f = 0.2),  normalize = TRUE)
lines(dateInc1, col="blue", hdrCol = adjustcolor( "blue", alpha.f = 0.2),  no
rmalize = TRUE)

par( mar=c(5, 5, 2, 0))
val.hdr <- paste( hpd(combiAMC14$Combinaison, prob = .95), ' à 95% BC/AD' )
plot(combiAMC14, withHDR = TRUE, main='Combinaison', normalize = TRUE ,  yaxt
="n", xlab="BC/AD")
mtext(val.hdr, side = 1, col = 'red', line = 2)
```



Superposition

Combinaison

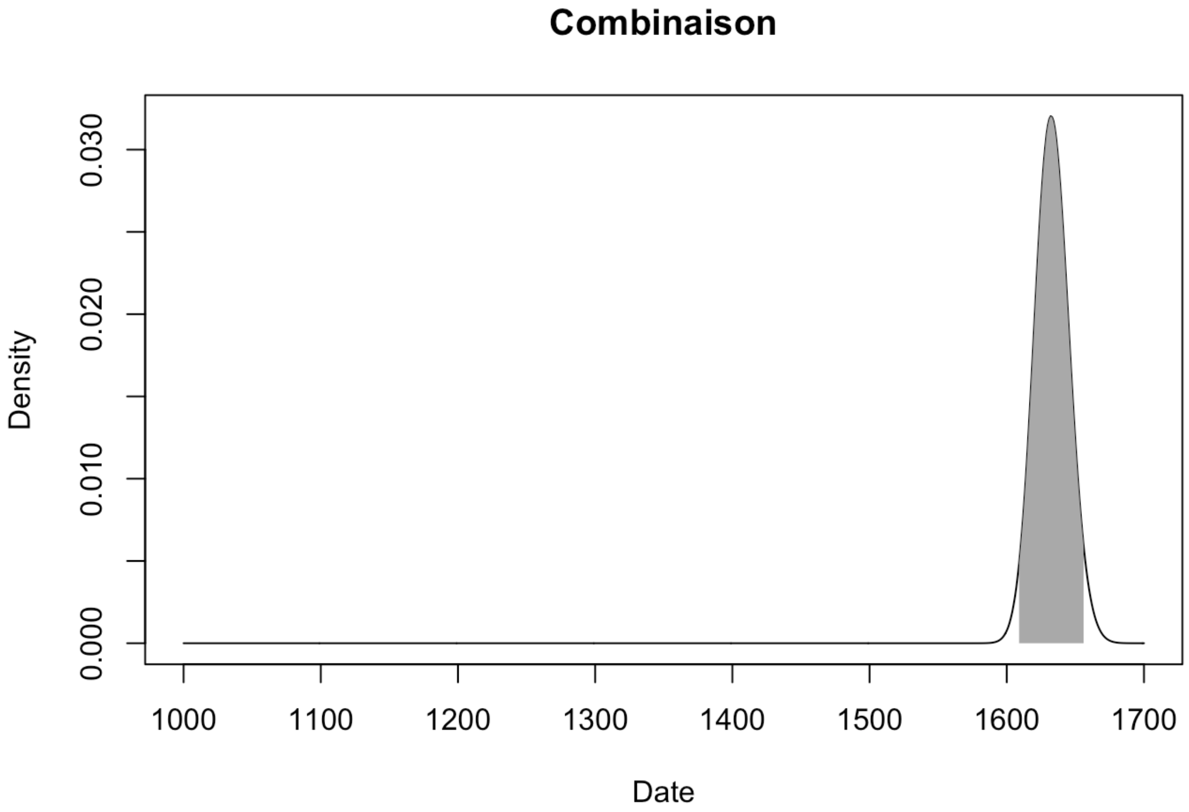
```
hpd(combiAMC14$Combinaison, prob = .95)
```

```
## $`95%`
## [1] 1602.1 1627.1
```

```
 # réduction de la période
tmin <- 1000
tmax <- 1700
# Il faut mettre des valeur qui existent dans le tableau
imin <- which(combiAM$Combinaison$timeGrid == tmin)
imax <- which(combiAM$Combinaison$timeGrid == tmax)
 tmp <- combiAM
 tmp$Combinaison$timeGrid <- tmp$Combinaison$timeGrid[imin:imax]
 tmp$Combinaison$densities <- tmp$Combinaison$densities[imin:imax]
 tmp$Combinaison$densities <-tmp$Combinaison$densities /sum(tmp$Combinaison$d
ensities)

# Affichage résultat
hpd(tmp$Combinaison, prob = .95)
```

```
## $`94.6%`
## [1] 1609 1656
```

```
plot(tmp)
```


Combinaison
Density
Date