stampr analysis

Colin Robertson

2016-10-26

Introduction to analysis of moving polygons in space-time using the `stampr` package.

```
library(stampr)
library(sp)
library(igraph)
```

```
##
## Attaching package: 'igraph'

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

```
data('katrina')
class(katrina)
```

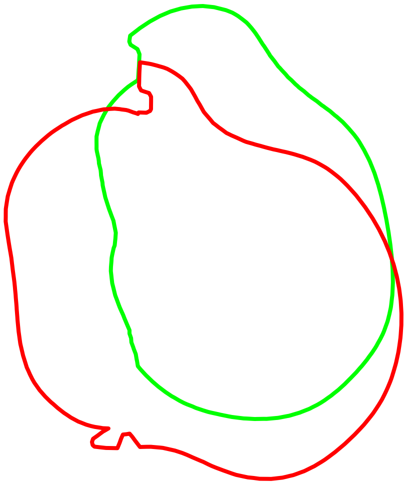
```
## [1] "SpatialPolygonsDataFrame"
## attr(,"package")
## [1] "sp"
```

```
head(katrina@data)
```

```
##   Id            DateTime
## 0  0 2005-08-25 21:00:00
## 1  1 2005-08-26 00:00:00
## 2  2 2005-08-26 03:00:00
## 3  3 2005-08-26 06:00:00
## 4  4 2005-08-26 09:00:00
## 5  5 2005-08-26 12:00:00
```

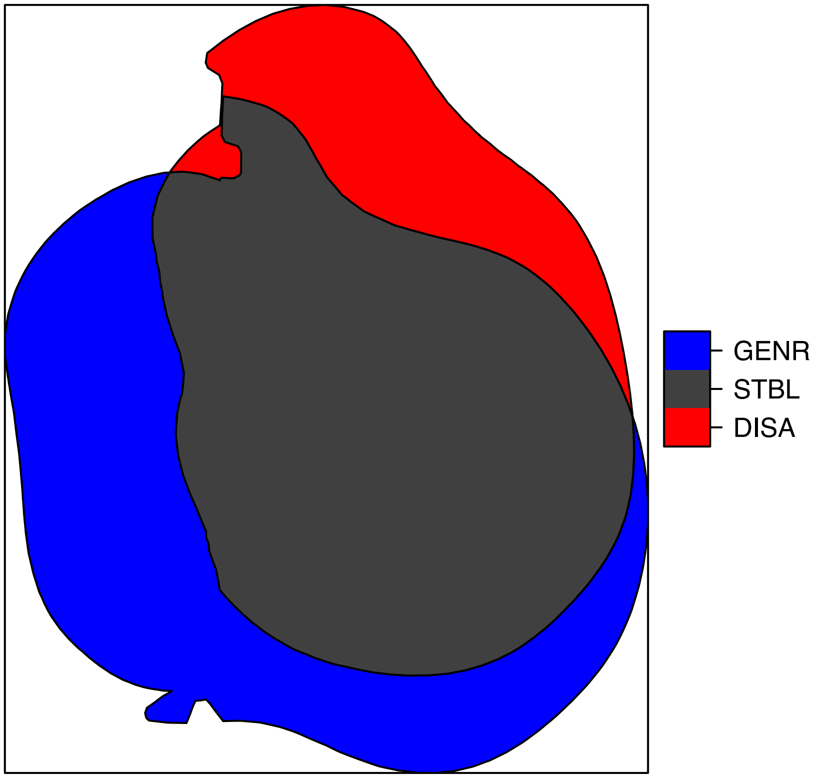
Simple two-time period change detection using overlay

```
T1 <- katrina[1, ]
T2 <- katrina[2, ]
T1$ID <- T1$Id
T2$ID <- T2$Id
plot(T1, col = NA, border = "green", lwd = 2, xlim = c(min(c(bbox(T1)[1, 1],
    bbox(T2)[1, 1])), max(c(bbox(T1)[1, 2], bbox(T2)[1, 2]))), ylim = c(min(c(bbox(T1)[2,
    1], bbox(T2)[2, 1])), max(c(bbox(T1)[2, 2], bbox(T2)[2, 2]))))
plot(T2, col = NA, border = "red", add = TRUE, lwd = 2)
```



The change we want to detect is the proportion of overlap, green only, and red only areas, representing stability, contraction and expansion events respectively.

```r
ch <- stamp(T1, T2, dc = 0, direction = TRUE, distance = TRUE)
stamp.map(ch)
```



```r
head(ch@data)
```

```
##   ID1 ID2 LEV1 LEV2 LEV3 LEV4 GROUP        AREA    CENDIR  CENDIST
## 0   0  NA DISA CONT CONT  N/A     1  4962070798   4.79028 64497.83
## 1   0   1 STBL STBL STBL  N/A     1 17555946174 184.79028 18229.88
## 2  NA   1 GENR EXPN EXPN  N/A     1 11203050305 222.57639 72247.35
```

Multiple time period polygon change analysis using overlay

```
T3 <- katrina[3, ]
T3$ID <- T3$Id
ch2 <- stamp(T2, T3, dc = 0, direction = TRUE, distance = TRUE)
head(ch2@data)
```

```
##   ID1 ID2 LEV1 LEV2 LEV3 LEV4 GROUP        AREA    CENDIR   CENDIST
## 0   1  NA DISA CONT CONT  N/A     1 11380288226  26.73386  40083.40
## 1   1   2 STBL STBL STBL  N/A     1 17378708253 206.73386  26248.25
## 2  NA   2 GENR EXPN EXPN  N/A     1 10942651871 231.35732 101808.65
```

Multiple time period polygon change analysis using overlay

Here is a brute force way to run the `stamp` function over multiple time periods using a loop;

```
outEvents <- list()
lSum <- 1
katrina$ID <- katrina$Id
for (i in 1:(nrow(katrina@data) - 1)) {
    Ti <- katrina[i, ]
    Ti_1 <- katrina[i + 1, ]
    ch <- stamp(Ti, Ti_1, dc = 0, direction = TRUE, distance = TRUE)
    xx1 <- spChFIDs(ch, paste(i, "-", as.character((lSum):(lSum + length(ch) -
        1)), sep = ""))
    lSum <- length(ch) + lSum
    outEvents[[i]] <- xx1  #@data
}

outEvents <- do.call("rbind", outEvents)
```

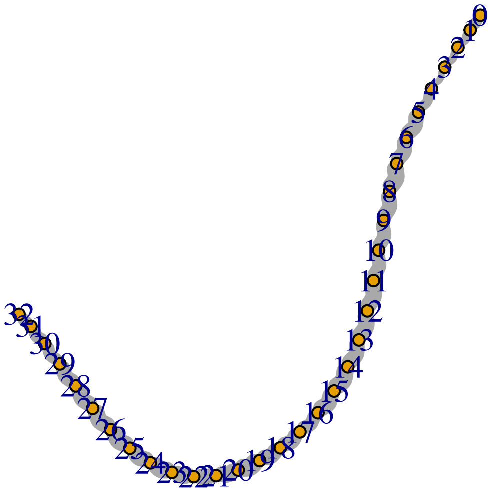
Get summaries from multiple change events

```
grps <- row.names(outEvents)  #get row names
outEvents$TGROUP <- substr(grps, 1, as.numeric(unlist(lapply(strsplit(grps,
    ""), function(x) which(x == "-")))) - 1)  #extract time period to distinguish change time periods
outEvents$STGROUP <- as.numeric(paste(outEvents$TGROUP, outEvents$GROUP, sep = ""))  #generate group ID
outSTGroup <- stamp.stgroup.summary(outEvents)
head(outSTGroup)
```

```
##   STGROUP nEVENTS         AREA nCONV nCONC nCONT nDISP1 nDISA nSTBL nEXPN
## 1      11       3  33721067276     0     0     1      0     0     1     1
## 2      21       3  39701648350     0     0     1      0     0     1     1
## 3      31       3  37586522910     0     0     1      0     0     1     1
## 4      41       3  67472954921     0     0     1      0     0     1     1
## 5      51       3 107975540054     0     0     1      0     0     1     1
## 6      61       3 112036033057     0     0     1      0     0     1     1
##   nFRAG nDIVR nDISP2 nGENA aCONV aCONC       aCONT aDISP1 aDISA
## 1     0     0      0     0     0     0  4962070798      0     0
## 2     0     0      0     0     0     0 11380288226     0     0
## 3     0     0      0     0     0     0 13307530739     0     0
## 4     0     0      0     0     0     0  1320338249     0     0
## 5     0     0      0     0     0     0 11840390904     0     0
## 6     0     0      0     0     0     0 10405856472     0     0
##          aSTBL       aEXPN aFRAG aDIVR aDISP2 aGENA
## 1 17555946174 11203050305     0     0      0     0
## 2 17378708253 10942651871     0     0      0     0
## 3 15013829385  9265162785     0     0      0     0
## 4 22958653921 43193962751     0     0      0     0
## 5 54312225768 41822923382     0     0      0     0
## 6 85729292679 15900883906     0     0      0     0
```

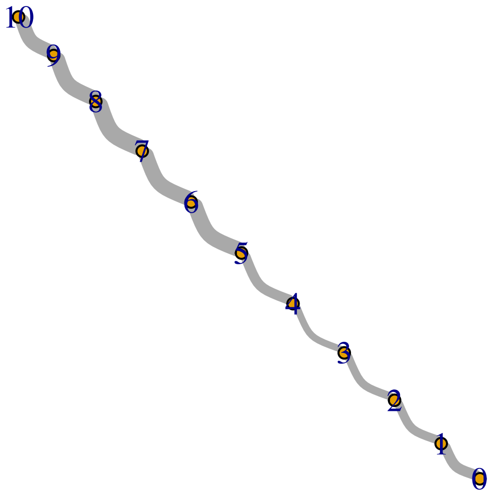
Do some more graphing of the topological relationships

```
df <- data.frame(from = outEvents$ID1, to = outEvents$ID2, stg = outEvents$STGROUP)
df <- df[complete.cases(df), ]
df <- merge(outSTGroup, df, by.x = "STGROUP", by.y = "stg")
df$weight <- (df$aSTBL/df$AREA) * 10
df <- data.frame(from = df$from, to = df$to, weight = df$weight)
g <- graph_from_data_frame(df, directed = TRUE, vertices = df$ID1)
E(g)$weight <- df$weight
plot(g, edge.width = E(g)$weight, layout = layout.fruchterman.reingold, edge.curved = TRUE,
    vertex.size = 5, edge.arrow.mode = "-")
```

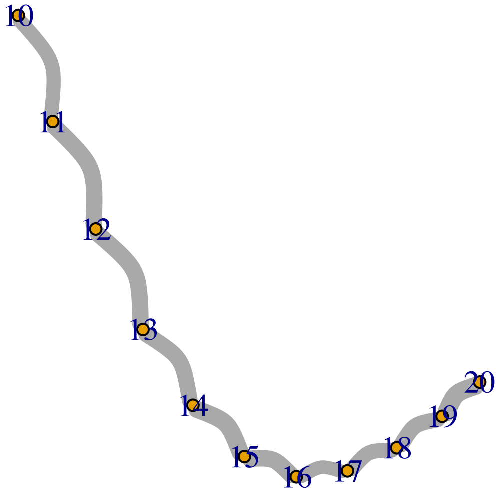


Lets look at T=1:10, and T=11:20, and T=21:32 separately to see if we can see any changes in space-time structure…

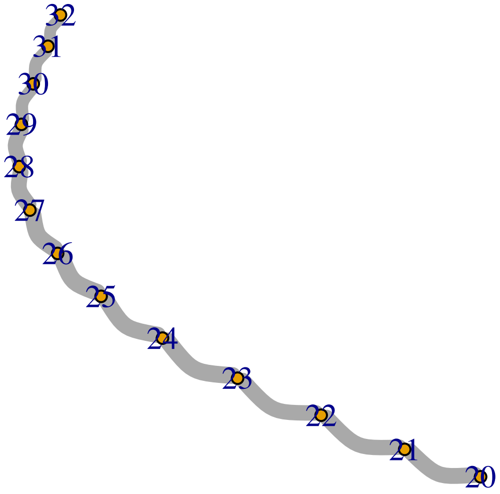
```
df2 <- df[1:10, ]
g <- graph_from_data_frame(df2, directed = TRUE, vertices = df2$ID1)
E(g)$weight <- df2$weight
plot(g, edge.width = E(g)$weight, layout = layout.fruchterman.reingold, edge.curved = TRUE,
    vertex.size = 5, edge.arrow.mode = "-")
```



```
df2 <- df[11:20, ]
g <- graph_from_data_frame(df2, directed = TRUE, vertices = df2$ID1)
E(g)$weight <- df2$weight
plot(g, edge.width = E(g)$weight, layout = layout.fruchterman.reingold, edge.curved = TRUE,
    vertex.size = 5, edge.arrow.mode = "-")
```



```
df2 <- df[21:32, ]
g <- graph_from_data_frame(df2, directed = TRUE, vertices = df2$ID1)
E(g)$weight <- df2$weight
plot(g, edge.width = E(g)$weight, layout = layout.fruchterman.reingold, edge.curved = TRUE,
    vertex.size = 5, edge.arrow.mode = "-")
```



Space-time Graph Clustering

```
data('mpb')
mpb$ID <- 1:nrow(mpb)
T1 <- subset(mpb, as.numeric(TGROUP)==1)
T2 <- subset(mpb, as.numeric(TGROUP)==2)
ch <- stamp(T1, T2, dc=2500, direction=TRUE, distance=TRUE)
```

Lets try it out, getting multiple change events for katrina data when each row is a polygon at a separate time period

```
rm(list=ls()) #need to rerun code above to create function after this
data("katrina")
katrina$ID <- katrina$Id
chng <- stamp.multichange(katrina, changeByRow = TRUE, stampArgs = list(0, TRUE, TRUE))
outSTGroup <- stamp.stgroup.summary(chng)
head(outSTGroup)
```

```
##   STGROUP nEVENTS         AREA nCONV nCONC nCONT nDISP1 nDISA nSTBL nEXPN
## 1      11       3  33721067276     0     0     1      0     0     1     1
## 2      21       3  39701648350     0     0     1      0     0     1     1
## 3      31       3  37586522910     0     0     1      0     0     1     1
## 4      41       3  67472954921     0     0     1      0     0     1     1
## 5      51       3 107975540054     0     0     1      0     0     1     1
## 6      61       3 112036033057     0     0     1      0     0     1     1
##   nFRAG nDIVR nDISP2 nGENA aCONV aCONC       aCONT aDISP1 aDISA
## 1     0     0      0     0     0     0  4962070798      0     0
## 2     0     0      0     0     0     0 11380288226      0     0
## 3     0     0      0     0     0     0 13307530739      0     0
## 4     0     0      0     0     0     0  1320338249      0     0
## 5     0     0      0     0     0     0 11840390904      0     0
## 6     0     0      0     0     0     0 10405856472      0     0
##         aSTBL       aEXPN aFRAG aDIVR aDISP2 aGENA
## 1 17555946174 11203050305     0     0      0     0
## 2 17378708253 10942651871     0     0      0     0
## 3 15013829385  9265162785     0     0      0     0
## 4 22958653921 43193962751     0     0      0     0
## 5 54312225768 41822923382     0     0      0     0
## 6 85729292679 15900883906     0     0      0     0
```

Now lets try the `stamp.multichange` function on the `mpb` data and a sumary function `stamp.stgroup.summary` :

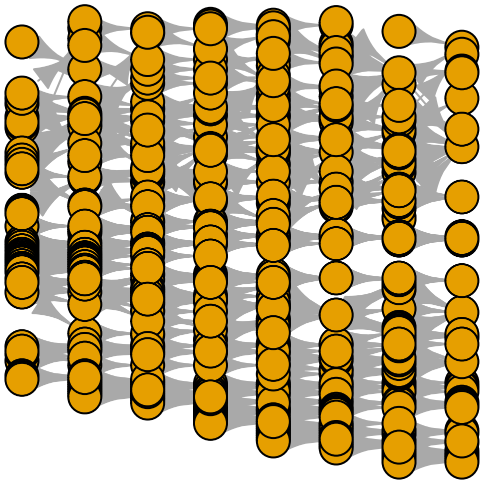
```
data("mpb")
mpb$ID <- nrow(mpb)
chng <- stamp.multichange(mpb, changeByRow = FALSE, changeByField=TRUE, changeField = "TGROUP", stampArg
outSTGroup <- stamp.stgroup.summary(chng)
head(outSTGroup)
```

```
##   STGROUP nEVENTS     AREA nCONV nCONC nCONT nDISP1 nDISA nSTBL nEXPN
## 1      11       1   740000     0     0     0      0     1     0     0
## 2      12       2  8810000     0     0     0      1     0     0     0
## 3      13       1 10880000     0     0     0      0     1     0     0
## 4      14       1  6880000     0     0     0      0     1     0     0
## 5      15       3  8515000     0     0     1      0     0     1     1
## 6      16       3  7906267     0     0     1      0     0     1     1
##   nFRAG nDIVR nDISP2 nGENA aCONV aCONC       aCONT  aDISP1    aDISA   aSTBL
## 1     0     0      0     0     0     0      0.000       0   740000       0
## 2     0     0      1     0     0     0      0.000 4615000        0       0
## 3     0     0      0     0     0     0      0.000       0 10880000       0
## 4     0     0      0     0     0     0      0.000       0  6880000       0
## 5     0     0      0     0     0     0 745000.000       0        0 1190000
## 6     0     0      0     0     0     0   1266.667       0        0 4658733
##     aEXPN aFRAG aDIVR  aDISP2 aGENA
## 1       0     0     0       0     0
## 2       0     0     0 4195000     0
## 3       0     0     0       0     0
## 4       0     0     0       0     0
## 5 6580000     0     0       0     0
## 6 3246267     0     0       0     0
```

Lets try our previous s-t topology graphing with the more complex `mpb` dataset

```
df <- data.frame(from=chng$ID1,to=chng$ID2, stg = chng$STGROUP, tg=chng$TGROUP)
df <- df[complete.cases(df),]
df <- merge(outSTGroup, df, by.x="STGROUP", by.y="stg")
df$weight <- (df$aSTBL / df$AREA) * 10
df <- data.frame(from=df$from,to=df$to, weight = df$weight, tg = df$tg)
fromIDs <- data.frame(names=as.character(unique(df$from)))
toIDs <- data.frame(names=as.character(unique(df$to)))
vertNames <- rbind(fromIDs, toIDs)

g <- graph_from_data_frame(df, directed = TRUE, vertices=unique(vertNames))

E(g)$weight <- df$weight
#replace with function
v_layers_df <- unique( rbind(
  expand.grid( ID = df$from[df$tg==1], Layer = 1),
```

```r
  expand.grid( ID = df$to[df$tg==1], Layer = 2),
  expand.grid( ID = df$from[df$tg==2], Layer = 2),
  expand.grid( ID = df$to[df$tg==2], Layer = 3),
  expand.grid( ID = df$from[df$tg==3], Layer = 3),
  expand.grid( ID = df$to[df$tg==3], Layer = 4),
  expand.grid( ID = df$from[df$tg==4], Layer = 4),
  expand.grid( ID = df$to[df$tg==4], Layer = 5),
  expand.grid( ID = df$from[df$tg==5], Layer = 5),
  expand.grid( ID = df$to[df$tg==5], Layer = 6),
  expand.grid( ID = df$from[df$tg==6], Layer = 6),
  expand.grid( ID = df$to[df$tg==6], Layer = 7),
  expand.grid( ID = df$from[df$tg==7], Layer = 7),
  expand.grid( ID = df$to[df$tg==7], Layer = 8)
  ))

v_layers <- setNames( v_layers_df$Layer, v_layers_df$ID)
V(g)$layer <- v_layers[V(g)$name]


layout.k_partite <- function(g) {
  l <- layout.sugiyama(g)$layout[,2:1]
  l[,1] <- V(g)$layer
  l[,2] <- - l[,2] + 1 + max(l[,2])
  l
}

plot(g, layout = layout.k_partite(g), vertex.label=NA)
```



```r
plot(g)
```