Vignette moveNT

Guillaume Bastille-Rousseau

April 18, 2017

Simulating movement strategies - *sim_mov*

The function *sim_mov* generates movement trajectories including patches and movement between patches. Movement within patches can follow an Ornstein-Uhlenbeck process (based on *simm.mou* function from package *adehabitatLT*) or two-states movement model (based on *simmData* function from package *moveHMM*). Movement between patches is following a brownian bridge movement model (based on *simm.bb* function from package *adehabitatLT*). Generated outputs are of the class *ltraj* from package *adehabitatlt*.

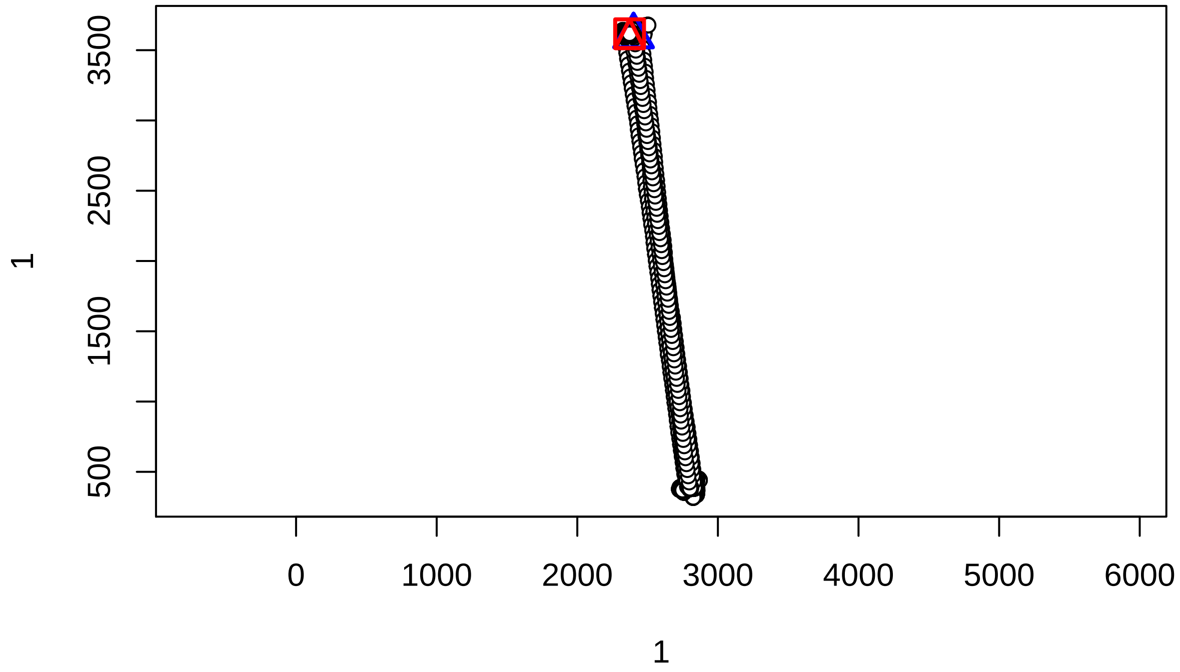
```
# Simulating migration with two-states model
mig<-sim_mov(type="2states", npatches=2, ratio=2, nswitch=25, ncore=150, grph=F)
mig
```

```
##
## *********** List of class ltraj ***********
##
## Type of the traject: Type II (time recorded)
## * Time zone: GMT *
## Regular traject. Time lag between two locs: 1 seconds
##
## Characteristics of the bursts:
##   id burst nb.reloc NAs          date.begin            date.end
## 1 id    id     4200   0 1960-01-01 00:00:01 1960-01-01 01:10:00
##
##
##  infolocs provided. The following variables are available:
## [1] "out.Corri"
```

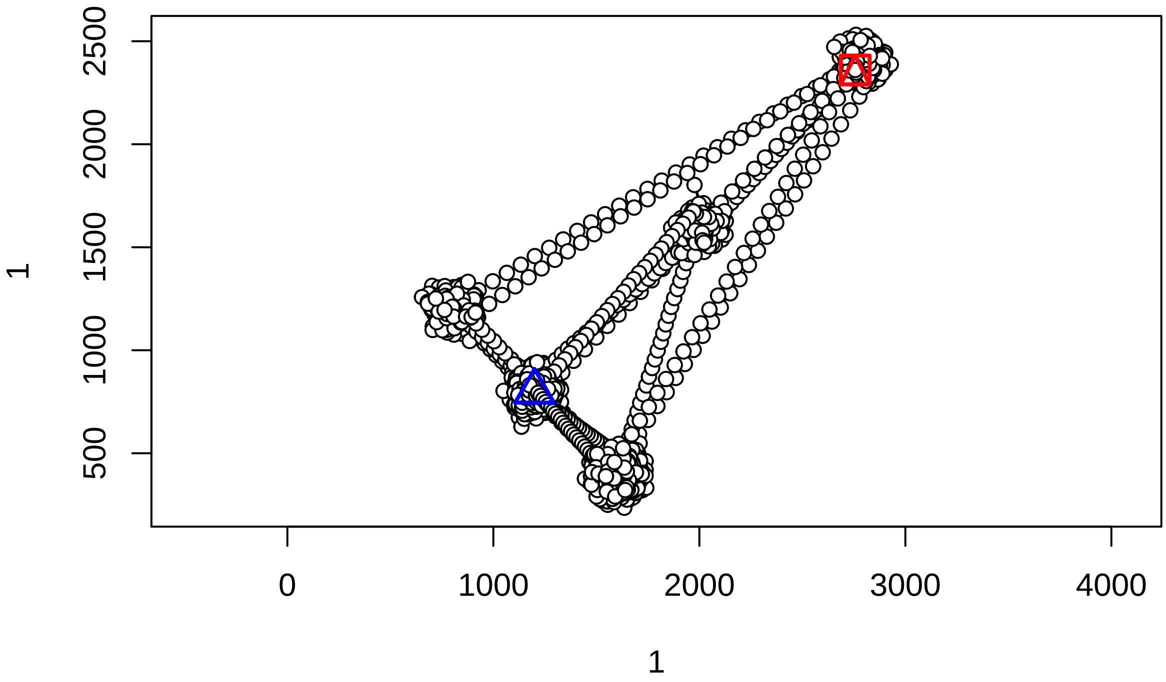
```
head(ld(mig))
```

```
##          x        y                date            dx            dy
## 1 2400.000 3600.000 1960-01-01 00:00:01 -4.296341e-04 -3.323228e-04
## 2 2400.000 3600.000 1960-01-01 00:00:02  1.215623e-01  1.431520e-01
## 3 2400.121 3600.143 1960-01-01 00:00:03 -1.163239e-05 -1.076653e-05
## 4 2400.121 3600.143 1960-01-01 00:00:04  4.211380e-03 -1.983149e-03
## 5 2400.125 3600.141 1960-01-01 00:00:05  5.131764e-04 -2.865328e-04
## 6 2400.126 3600.141 1960-01-01 00:00:06 -1.023345e+00 -2.586293e-01
##           dist dt          R2n  abs.angle   rel.angle id burst out.Corri
## 1 5.431611e-04  1 0.000000e+00 -2.4832192          NA id    id         2
## 2 1.878028e-01  1 2.950239e-07  0.8667768 -2.93318925 id    id         2
## 3 1.585026e-05  1 3.507058e-02 -2.3948318  3.02157665 id    id         2
## 4 4.654955e-03  1 3.506469e-02 -0.4400997  1.95473211 id    id         2
## 5 5.877508e-04  1 3.554011e-02 -0.5092324 -0.06913274 id    id         2
## 6 1.055521e+00  1 3.558839e-02 -2.8940468 -2.38481432 id    id         2
```

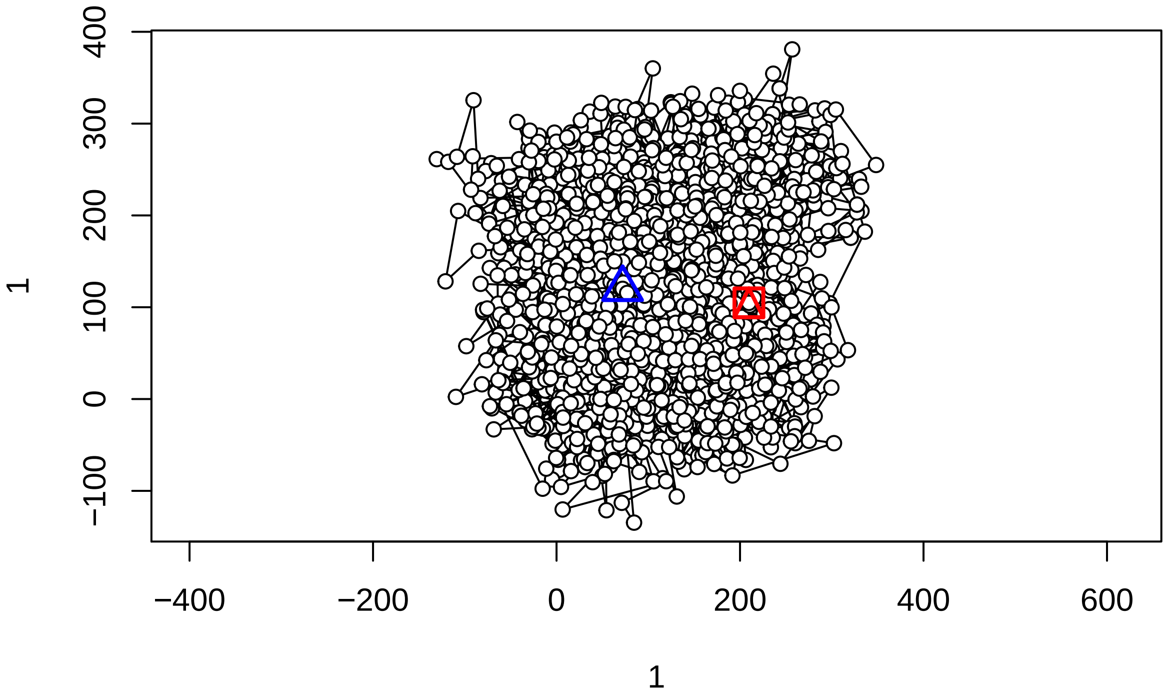
```
plot(mig)
```



```r
# Simulating multi-patches movement with Ornstein-Uhlenbeck process
patches<-sim_mov(nswitch=25, ncore=150, ratio=5, type="OU", npatches=5, grph=T)
```



```
# Simulating sedentary movement
seden<-sim_mov(type="OU", npatches=10, spacecore=12, ratio=3, nswitch=150, ncore=20, grph=T)
```



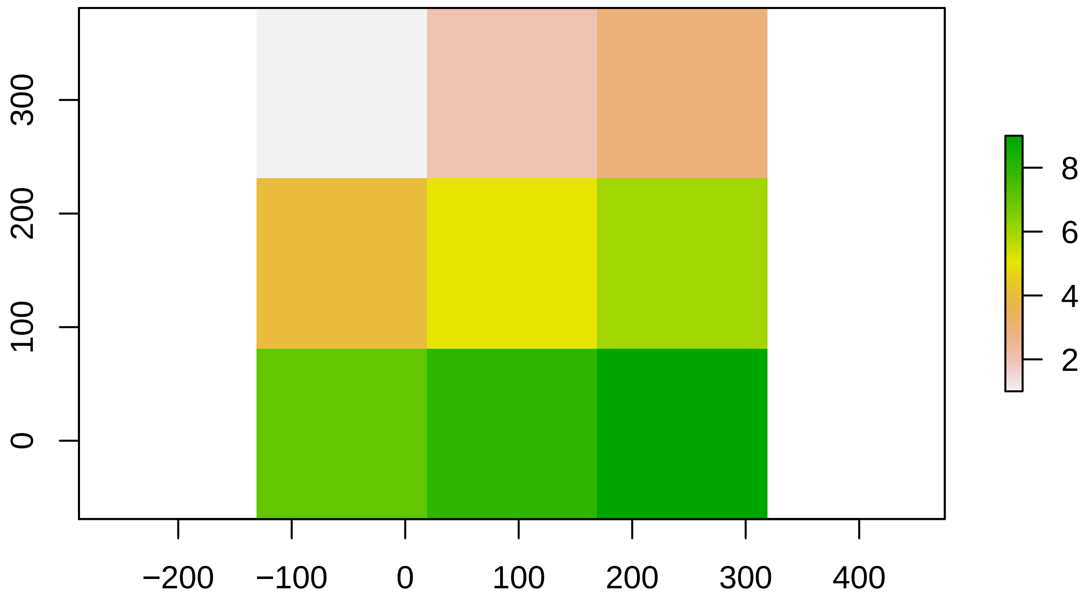
Converting movement to adjacency matrix - *traj2adj*

The function *traj2adj* converts a trajectory object of class *ltraj* to an adjacency matrix. This is done by overlapping a grid over the relocation data and tallying the number of transitions among each pixel. Users need to specify the grid size, which can be based on distance travelled. The function *quant* is a wrapper that allows to sample a quantile of step length distribution from a *ltraj* object. Output produced by *traj2adj* is a list containing the adjacency matrix, the grid used (raster format), and a raster indicating pixel numbers that are occupied. These rasters are used by other functions such as *adj2stack* and *clustnet*.

```
# Using sedentary movement and user specific grid-size
adj_seden<-traj2adj(seden, res=150) #Pixel size of 150m
adj_seden[[1]] # Adjency matrix
```

```
##       [,1] [,2] [,3] [,4] [,5] [,6] [,7] [,8] [,9]
##  [1,]   20    6    0   14    9    0    0    0    0
##  [2,]    7   90   15    5   68   14    0    0    0
##  [3,]    0   19  129    0   14   51    0    0    0
##  [4,]   10    9    0  141   69    0   16   14    0
##  [5,]   12   66   17   67  965   96   12   81   14
##  [6,]    0    9   54    0   94  178    0   14   48
##  [7,]    0    0    0   24   12    0   72   53    0
##  [8,]    0    0    0    8   81   19   58  463   76
##  [9,]    0    0    0    0   17   38    0   85  256
```

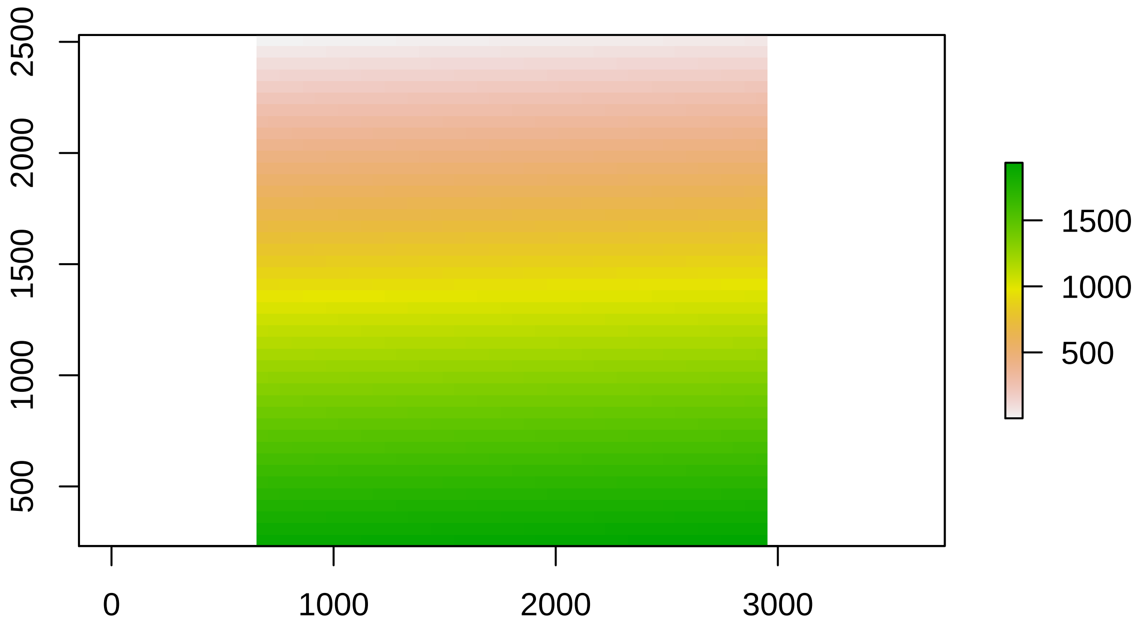
```
plot(adj_seden[[2]]) #Plot grid used
```



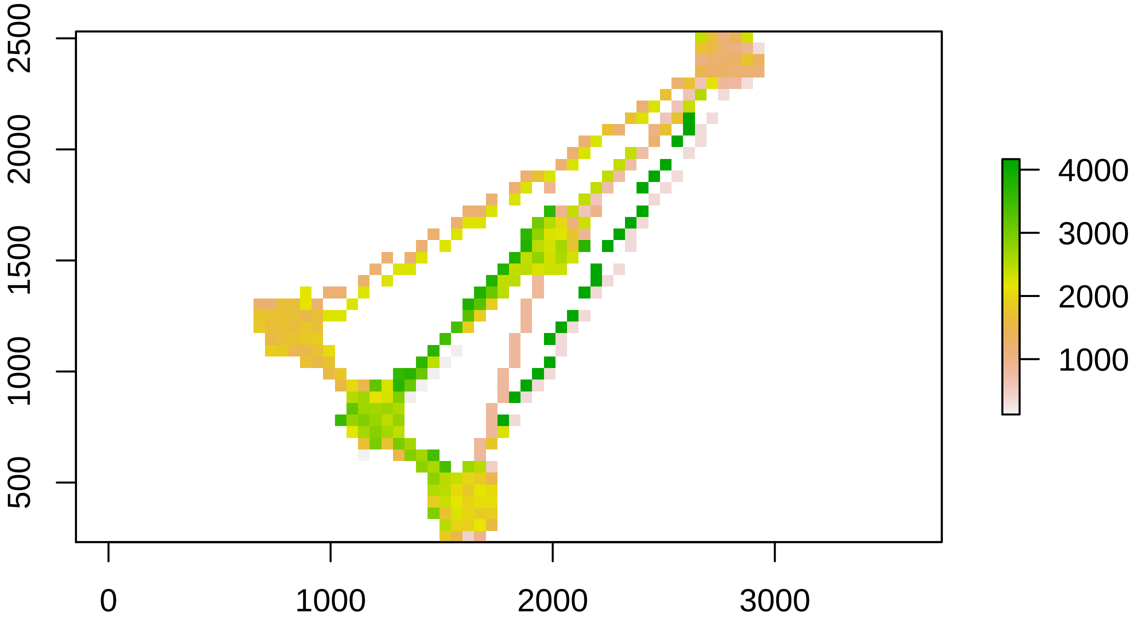
```
# Using multi-patches movement and median distance travelled
adj_patches<-traj2adj(patches, res=quant(patches, p=0.5)) #Grid size based on median
dim(adj_patches[[1]]) # Size of the adjacency matrix
```

```
## [1] 1936 1936
```

```
plot(adj_patches[[2]]) #Plot grid used
```



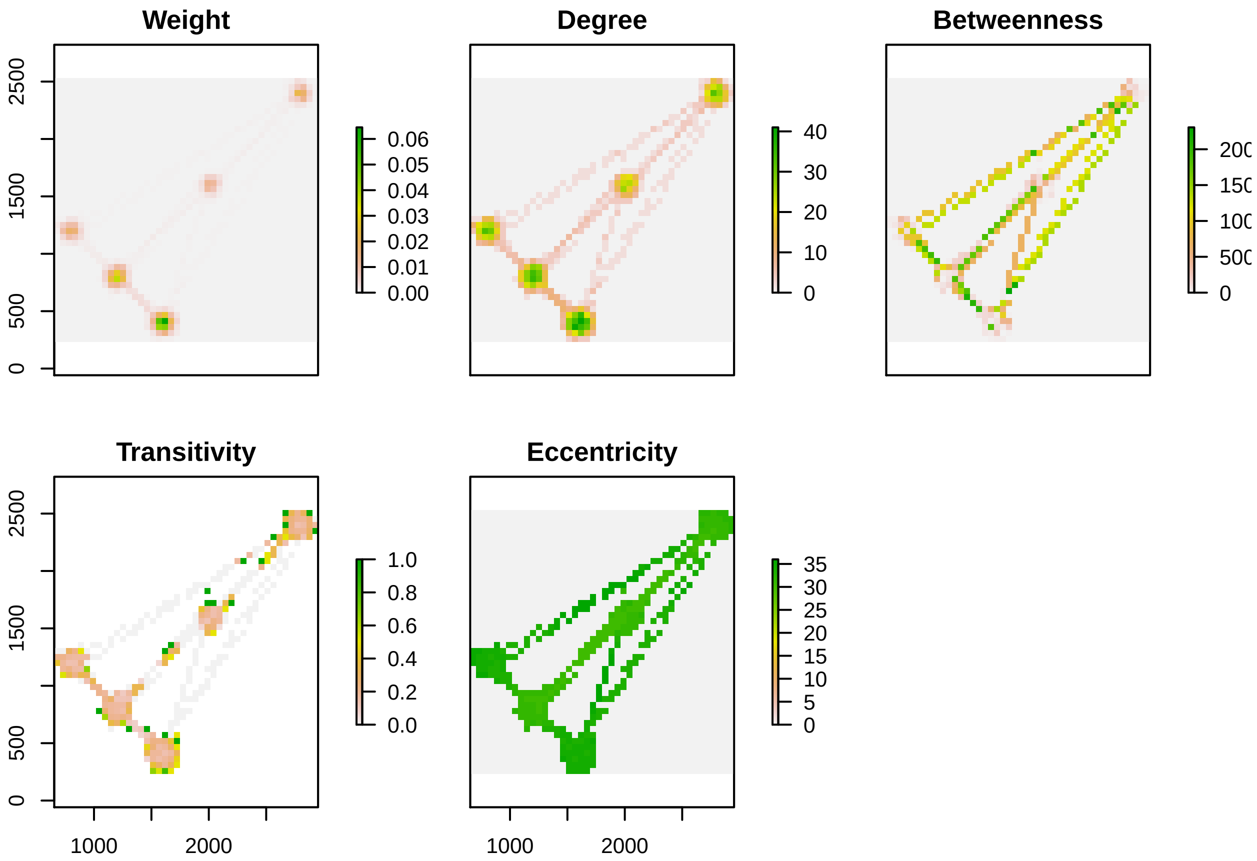
```r
plot(adj_patches[[3]]) #Plot occupied pixels
```



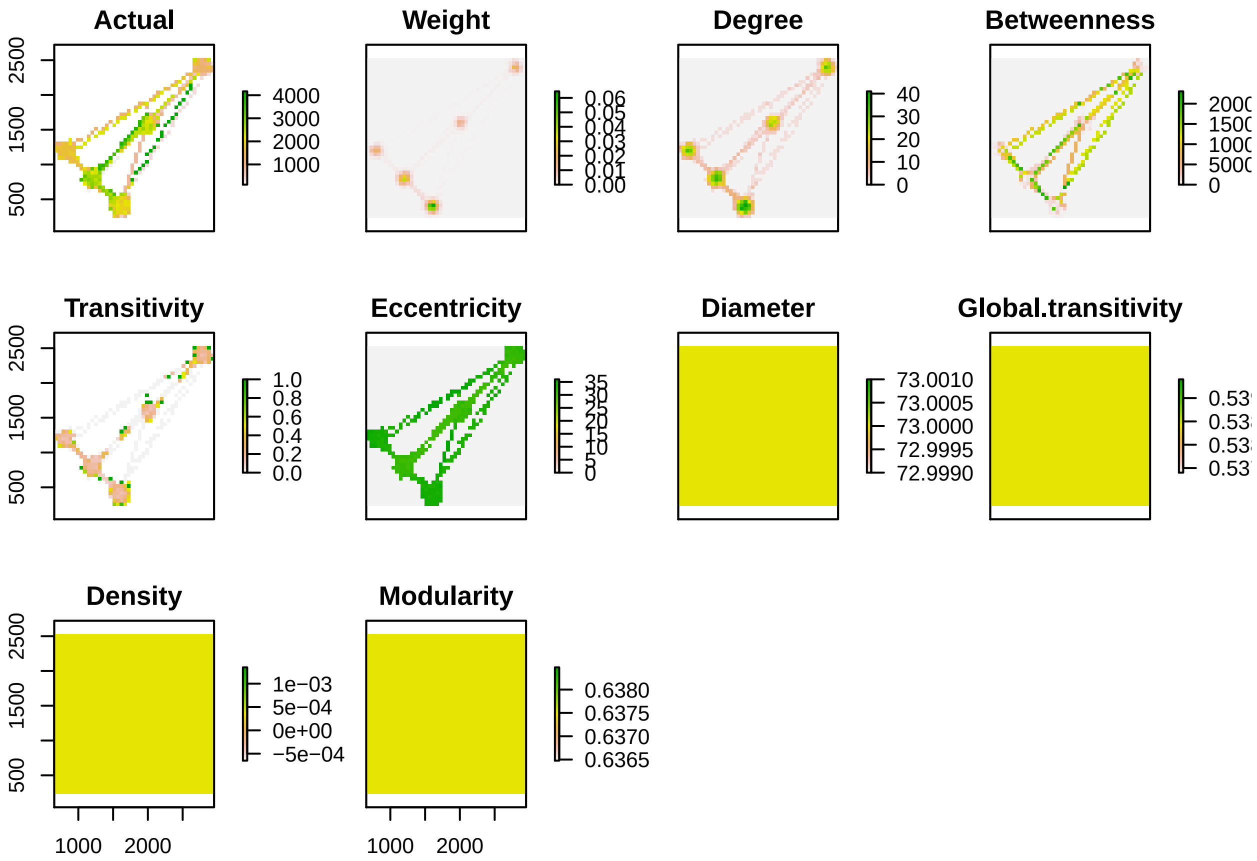
Calculation of network metrics - *adj2stack*

The function *adj2stack* takes the output of function *traj2adj* and calculates a series of node- and graph-level metrics. Each metric is stored as a individual raster and the output is a raster stack combining each metric. Graph-level metrics are also stored as a raster, each containing an unique value. The function *graphmet* extracts graph-level metrics. The function *val* extracts only the occupied cells (remove NA) in a raster and allows the calculation of statistics from node-level metrics.

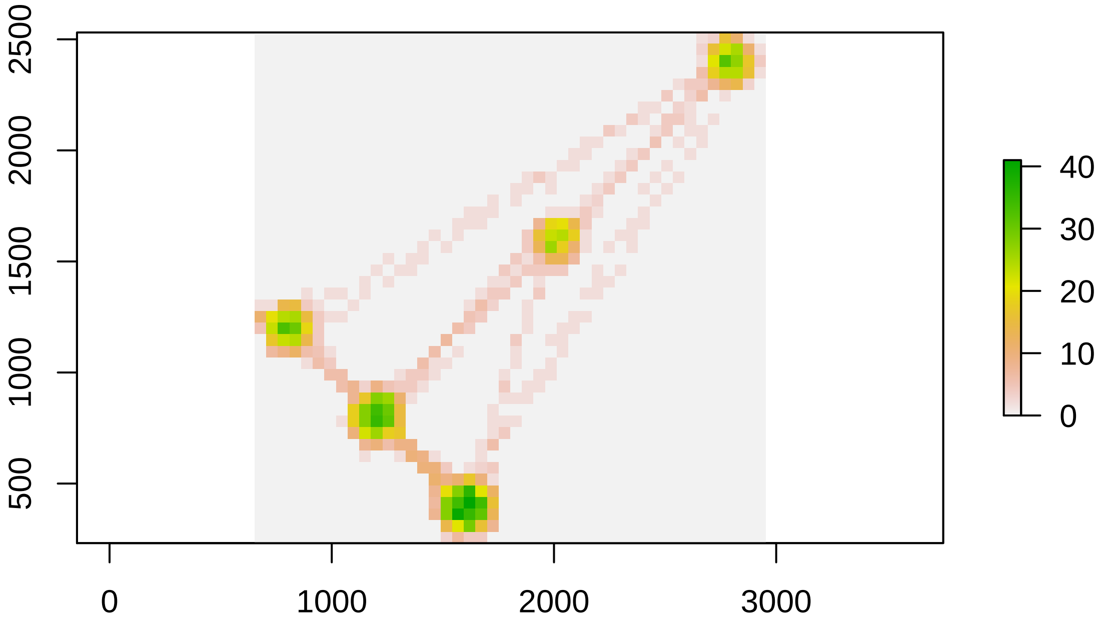
```
# Using multi-patches movement and median distance travelled
stck<-adj2stack(adj_patches,grph=T) #Plot the node-level metrics at the same time
```



```
plot(stck) #Plot also the graph-level metrics (not really useful)
```



```
plot(stck[[3]]) #Plot only one metric (degree)
```



```
graphmet(stck) # Extract graph-level metrics
```

```
##            Diameter Global.transitivity             Density
##        7.300000e+01        5.384016e-01        3.502253e-04
##          Modularity
##        6.374767e-01
```

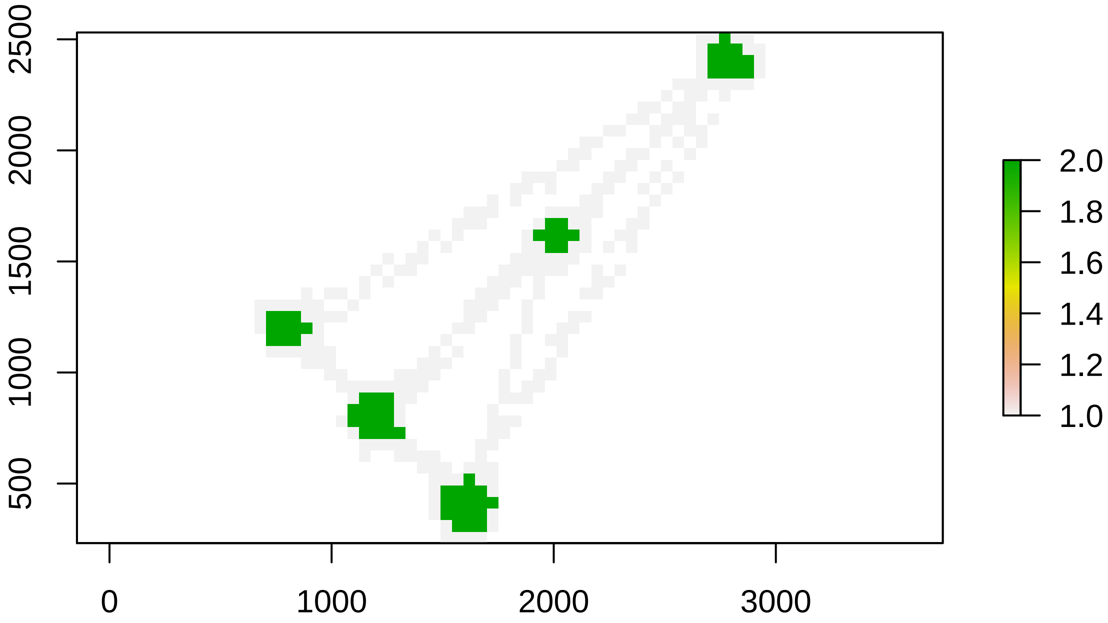
```
cv(val(stck, 4)) #Extract coefficient of variation of node-level betweenness.
```

```
## [1] 81.1104
```

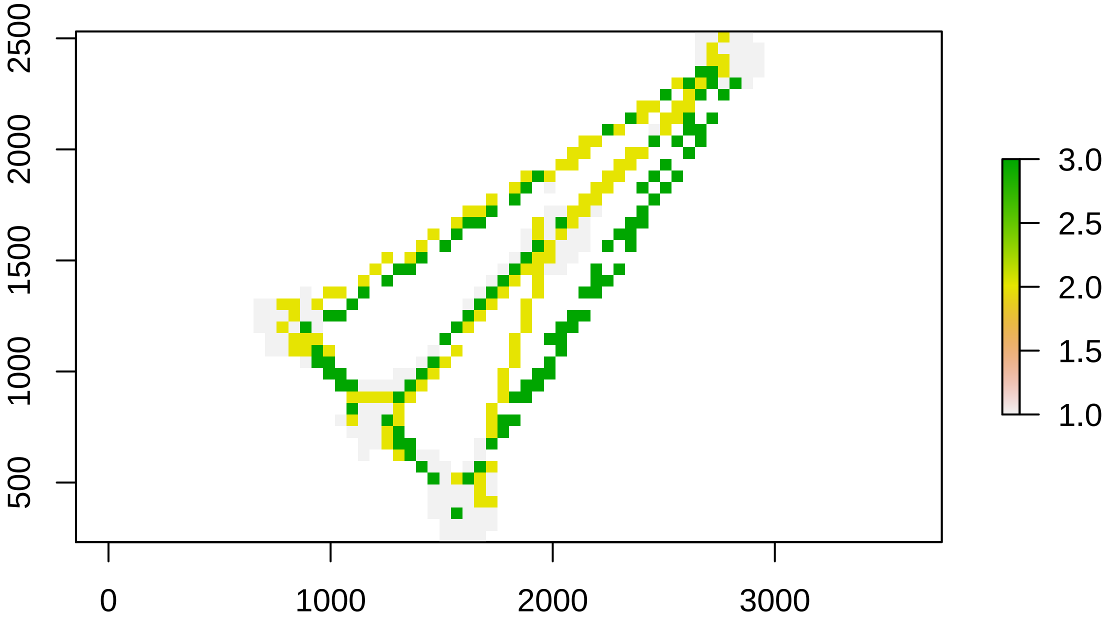
Clustering of node level metrics - *clustnet*

The function *clustnet* applies a normal mixture model to node-level metrics in order to cluster them into separate groups (default = 2). The function takes the output of function *adj2stack* with the user specifying the metric to cluster and the number of groups. Return a list containing output of function *Mclust* from package *mclust* and a raster displaying classification.

```
# Using multi-patches movement and median distance travelled
clust2<-clustnet(stck, id=3, nclust=2) # Clustering of degree in two groups
```



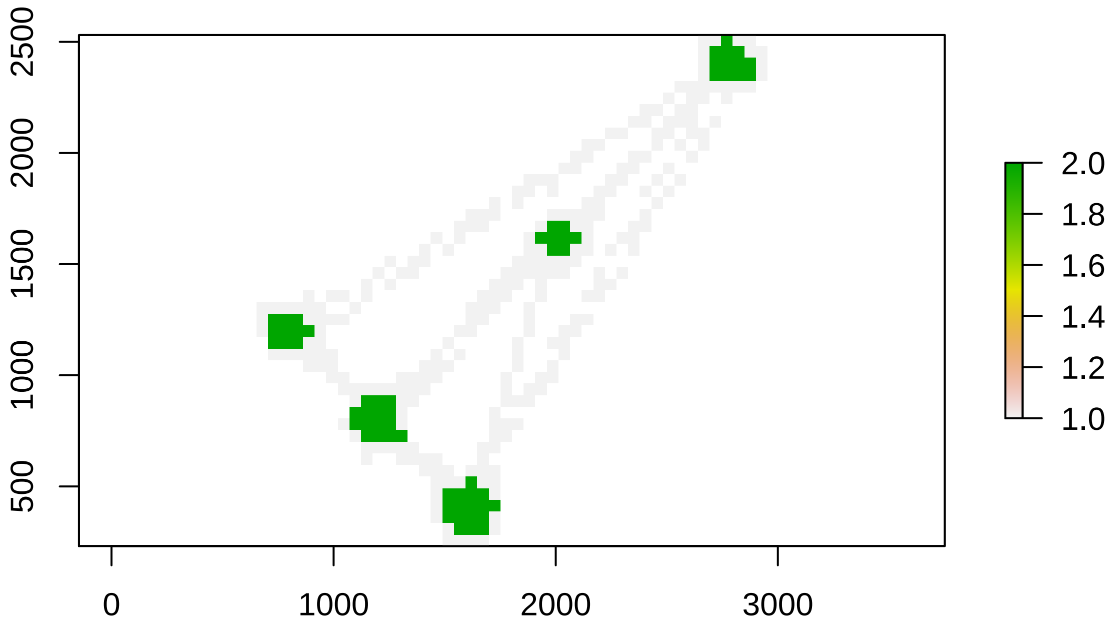
```
clust3<-clustnet(stck, id=4, nclust=3) #Clustering of betweenness in three groups
```



```
summary(clust2[[1]])
```

```
## ----------------------------------------------------
## Gaussian finite mixture model fitted by EM algorithm
## ----------------------------------------------------
##
## Mclust E (univariate, equal variance) model with 2 components:
##
##  log.likelihood   n df       BIC       ICL
##       -1079.416 326  4 -2181.98 -2197.655
##
## Clustering table:
##   1   2
## 264  62
```

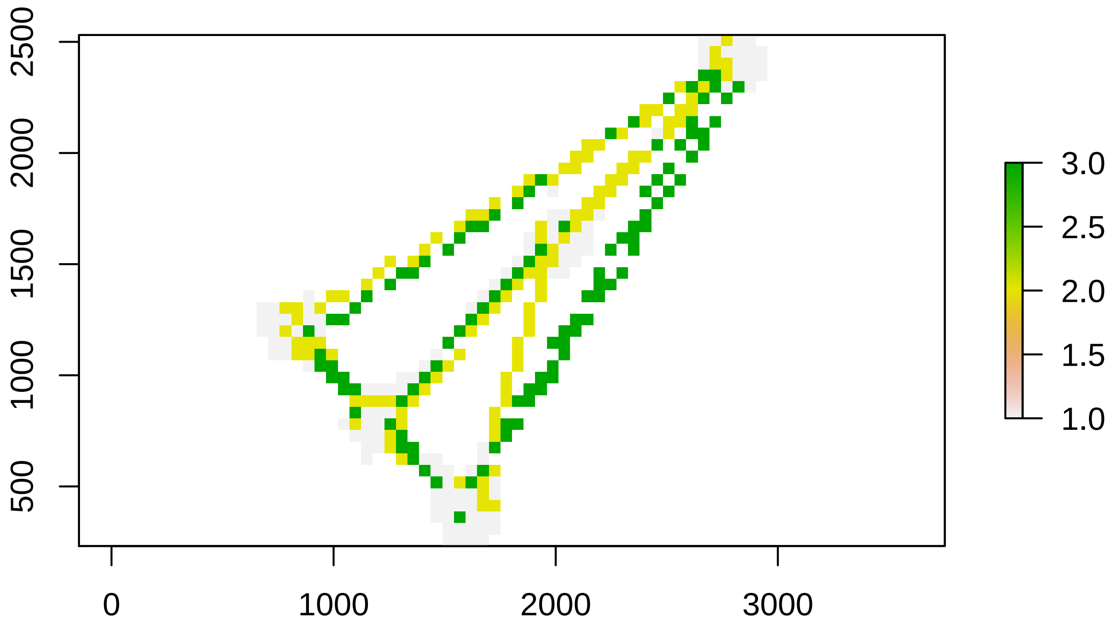
```
plot(clust2[[2]])
```



```
summary(clust3[[1]])
```

```
## ----------------------------------------------------
## Gaussian finite mixture model fitted by EM algorithm
## ----------------------------------------------------
##
## Mclust V (univariate, unequal variance) model with 3 components:
##
##  log.likelihood   n df       BIC       ICL
##       -3181.441 326  8 -6409.177 -6550.915
##
## Clustering table:
##   1   2   3
## 112 110 104
```

```
plot(clust3[[2]])
```


2500
2000
1500
1000
500
0
1000
2000
3000
3.0
2.5
2.0
1.5
1.0