ForceAtlas2 layout for network analysis

Adolfo Álvarez

2015-06-12

This is the R implementation of the Force Atlas 2 graph layout designed for Gephi. The algorithm is detailed in:

Jacomy M, Venturini T, Heymann S, Bastian M (2014) ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software. PLoS ONE 9(6): e98679

Installation

This package is not yet available in CRAN, so install it directly from Github with:

```r
# install.packages("devtools")
devtools::install_github("analyxcompany/ForceAtlas2")
```

Usage

After installation the package is loaded as usual with:

```r
library(ForceAtlas2)
```

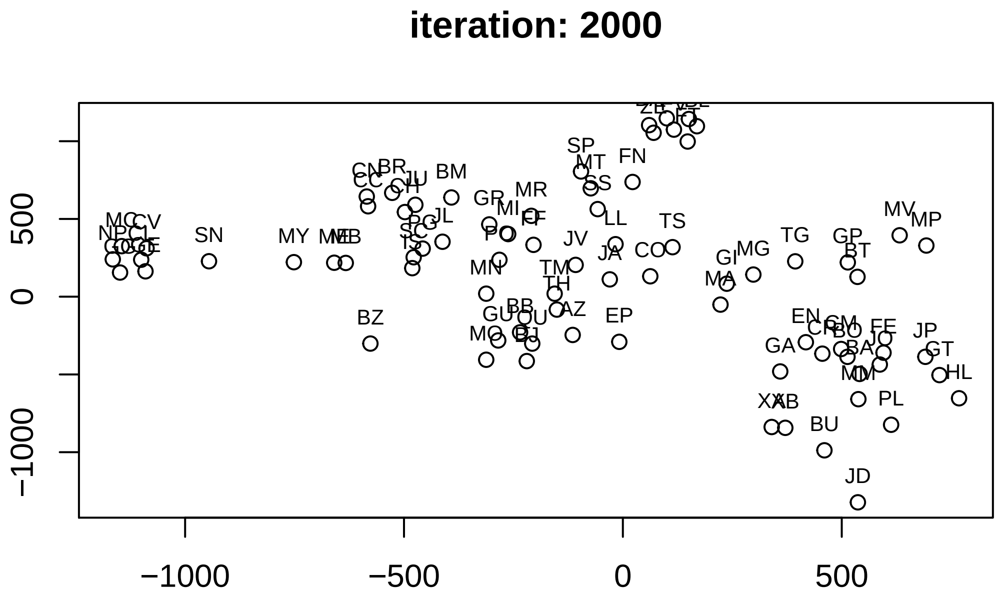
This implementation accepts as inputs an `igraph` object or a data frame. A full parameter description of the algorithm can be found in Jacomy et al, but a summary is accesible via `help("layout.forceatlas2")`.

Among those parameters you would probably like to set the number of iterations (`iterations`) and how often a plot of the positions should be generated (`plotstep`). This is particularly useful to evaluate the convergence of the algorithm. Set `plotstep=0` to suppress intermediate plots.

igraph input

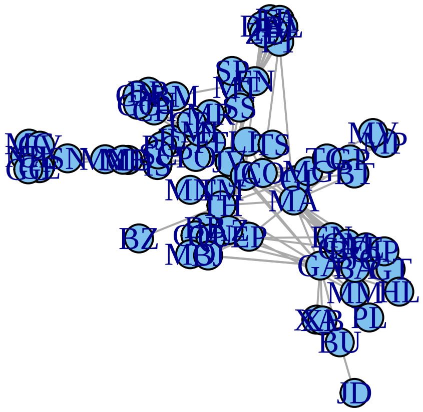
For this example I will use the coappeareance network from Les Miserables, by Victor Hugo. Get more details about this data set with `igraph::nexus.info("miserables")`

```r
library(igraph)
g <- nexus.get("miserables")
layout <- layout.forceatlas2(g, iterations=2000, plotstep=100)
```



This can be also plotted from as an igraph object:

```
plot(g, layout=layout)
```



data frame input

For those not familiar with the `igraph` package, is possible to calculate the algorithm directly from a data frame. This data frame should consist in three columns: `from`, `to`, and `weights`, indicating the corresponding nodes connections and the weights.

```
data <- get.data.frame(g)
dim(data) #The dimension of the data
```

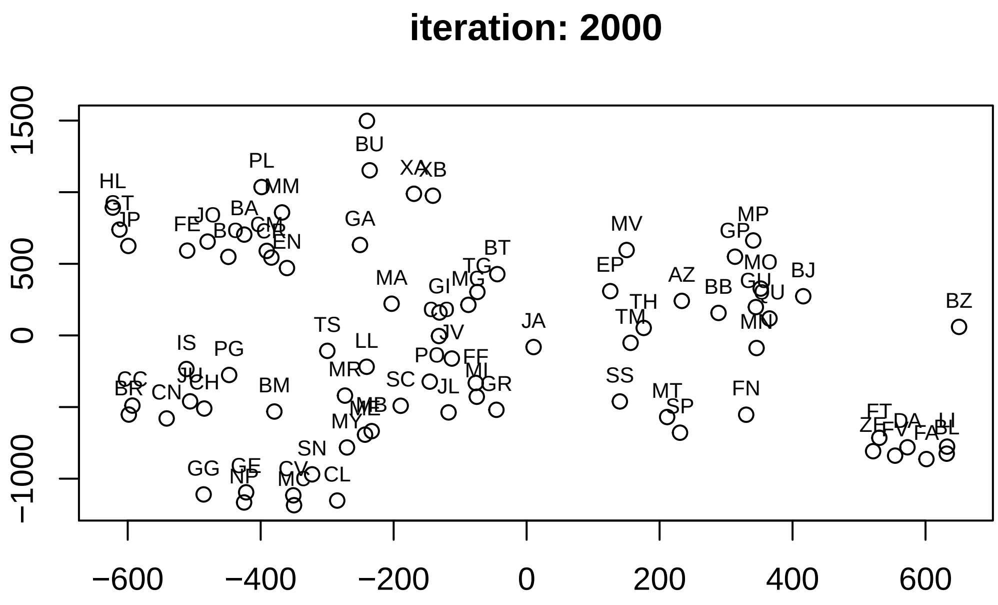
```
## [1] 254   3
```

```
head(data,10) #We show just the first 10 rows
```

```
##    from to weight
## 1    MY NP      1
## 2    MY MB      8
## 3    MY ME     10
## 4    MY CL      1
## 5    MY GE      1
## 6    MY MC      1
## 7    MY CV      1
## 8    MY SN      2
## 9    MY GG      1
## 10   MY JV      5
```

After you have your data in that format, the application of the function is equivalent to the previous one, with one exception, the parameter `directed` indicating if the network is directed or not (`directed = TRUE` by default). In this example, `the miserables` is an undirected graph so we change it accordingly.

```
layout.forceatlas2(data, directed=FALSE, iterations = 2000, plotstep = 100)
```



When the input is a data frame, the output is also a data frame including the name of the nodes and positions.

```
head(layout, 10) #Show just the 10 first rows
```

```
##    name        V1        V2
## 1    MY -270.1605 -781.8193
## 2    MB -233.0312 -667.8271
## 3    ME -243.0333 -692.6312
## 4    JL -117.4105 -537.3446
## 5    JV -112.4921 -161.2268
## 6    MT  211.3754 -569.4590
## 7    FT  530.4976 -714.4392
## 8    LI  632.4362 -776.2653
## 9    FA  601.3423 -863.0877
## 10   BL  631.8835 -825.2879
```