What’s new in igraph and networks

Gábor Csárdi
2015-07-01
About igraph
About igraph

- Network analysis library, written mostly in C/C++.
- Interface to R and Python
- [https://github.com/igraph](https://github.com/igraph)
- [http://igraph.org](http://igraph.org)
- Mailing list, stack overflow help.
- Open GitHub issues for bugs
What is new?

- New ways to manipulate networks
- [ ] and [ [ ] operators
- Better function names, manipulators
- Pipe friendly
- New methods:
  - Graph layout algorithms
  - New methods for graph clustering
  - Graphlet decomposition
  - Embeddings
  - Graph matching
  - etc.
The [ operator

Imaginary adjacency matrix, queries

```
air['BOS', 'SFO']
#> [1] 6

CA <- c("LAX", "SFO", "SAN", "SMF", "SNA", "BUR", "OAK", "ONT", "SJC")
air['BOS', CA]
#> LAX SFO SAN SMF SNA BUR OAK ONT SJC
#> 7 6 1 0 0 0 0 0 0 1
```
The \[ \] operator

Imaginary adjacency matrix, manipulation

Add an edge (and potentially set its weight):

```r
air["BOS", "ANC"] <- TRUE
air["BOS", "ANC"]

#> [1] 1
```

Remove an edge:

```r
air["BOS", "ANC"] <- FALSE
air["BOS", "ANC"]

#> [1] 0
```
The \[ \[ \text{ operator} \]

Imaginary adjacency list, adjacent vertices:

\[
\text{air}[\text{"BOS"]}
\]

\[
\text{\#> } \text{BOS} \\
\text{\#> } + 269/755 \text{ vertices, named:} \\
\text{\#> } [1] \text{ BGR JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK \\
\text{\#> } [16] \text{ LAS LAS LAS MIA MIA EWR EWR EWR EWR EWR EWR EWR EWR EWR \\
\text{\#> } [31] \text{ LAX LAX LAX LAX LAX LAX PBI PBI PIT PIT PIT PIT PIT SFO \\
\text{\#> } [46] \text{ SFO SFO SFO SFO SFO IAD IAD IAD IAD IAD IAD IAD IAD IAD \\
\text{\#> } [61] \text{ BDL BDL BUF BUF BUF BWI BWI BWI BWI BWI BWI BWI BWI CAK \\
\text{\#> } [76] \text{ CLE CLE CLE CLE CLE CLT CLT CLT CLT CLT CLT CLT CLT CMH \\
\text{\#> } [91] \text{ CMH CVG CVG CVG CVG CVG CVG CVG CVG CVG DCA DCA DCA DCA \\
\text{\#> } [106] \text{ DCA DCA DCA DCA DCA DTW DTW DTW DTW DTW DTW DTW DTW DTW \\
\text{\#> } [121] \text{ DTW DTW DTW GSO IND IND LGA LGA LGA LGA LGA LGA LGA LGA MDT \\
\text{\#> } [136] \text{ MKE MKE MKE MSP MSP MSP MSP MSP MSP MSY MYR ORF PHF PHL PHL \\
\text{\#> } + \ldots \text{ omitted several vertices}
\]
The `[[` operator

Imaginary adjacency list, adjacent vertices:

```r
air[[, "BOS"]]
```

```r
#> $BOS
#> + 256/755 vertices, named:
#> #> [1] BGR JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK JFK LAS LAS
#> #> [16] LAS MIA MIA MIA EWR EWR EWR EWR EWR EWR EWR EWR LAX LAX LAX
#> #> [31] LAX LAX LAX LAX PBI PBI PIT PIT PIT PIT SFO SFO SFO SFO
#> #> [46] SFO SFO IAD IAD IAD IAD IAD IAD IAD IAD BDL BDL BDL BUF
#> #> [61] BUF BUF BUF BWI BWI BWI BWI BWI BWI CAK CAK CLE CLE CLE CLE
#> #> [76] CLE CLE CLT CLT CLT CLT CLT CLT CLT CLT CMH CMH CVG CVG CVG
#> #> [91] CVG CVG CVG DCA DCA DCA DCA DCA DCA DCA DCA DCA DTW DTW DTW
#> #> [106] DTW DTW DTW DTW DTW DTW DTW DTW DTW DTW DTW DTW DTW DTW DTW
#> #> [121] LGA LGA MDT MKE MKE MKE MSP MSP MSP MSP MSP MSP MSP MSP MSY
#> #> [136] MSY MYR PHF PHL PHL PHL PHL PHL PHL PHL PHL PHL RDU RDU RDU
#> #> + ... omitted several vertices
```
Consistent function names

make_*, sample_*, cluster_*, layout_*, etc.
Manipulators for `make_` and `sample_`

```r
ring <- make_(ring(10), with_vertex_(color = "grey", size = 25))
par(mar=c(0,0,0,0)); plot(ring)
```
Manipulators for `make_` and `sample_`

```r
rg <- sample_(degseq(c(2,4,6,8,2,2,2,2)),
              simplified())
```
g <- make_ring(5) + make_full_graph(5) + make_star(5, mode="undir")
coords <- layout_(g, in_circle(), component_wise())
par(mar=c(0,0,0,0)); plot(g)
Pipe friendly syntax

g <- make_empty_graph(10) %>%
    add_vertices(5) %>%
    set_vertex_attr("name", value = LETTERS[1:5]) %>%
    add_edges(c(1, 2, 2, 3, 3, 4, 4, 5, 5, 1)) %>%
    set_edge_attr("weight", value = runif(gsize(.))))
Easier connection to other packages

```r
library(networkD3)
d3_net <- simpleNetwork(as_data_frame(karate, what = "edges")[, 1:3])
d3_net
```
Current work

- Better connection to other packages and external software
- Inference
- Infrastructure cleanup