Microsoft

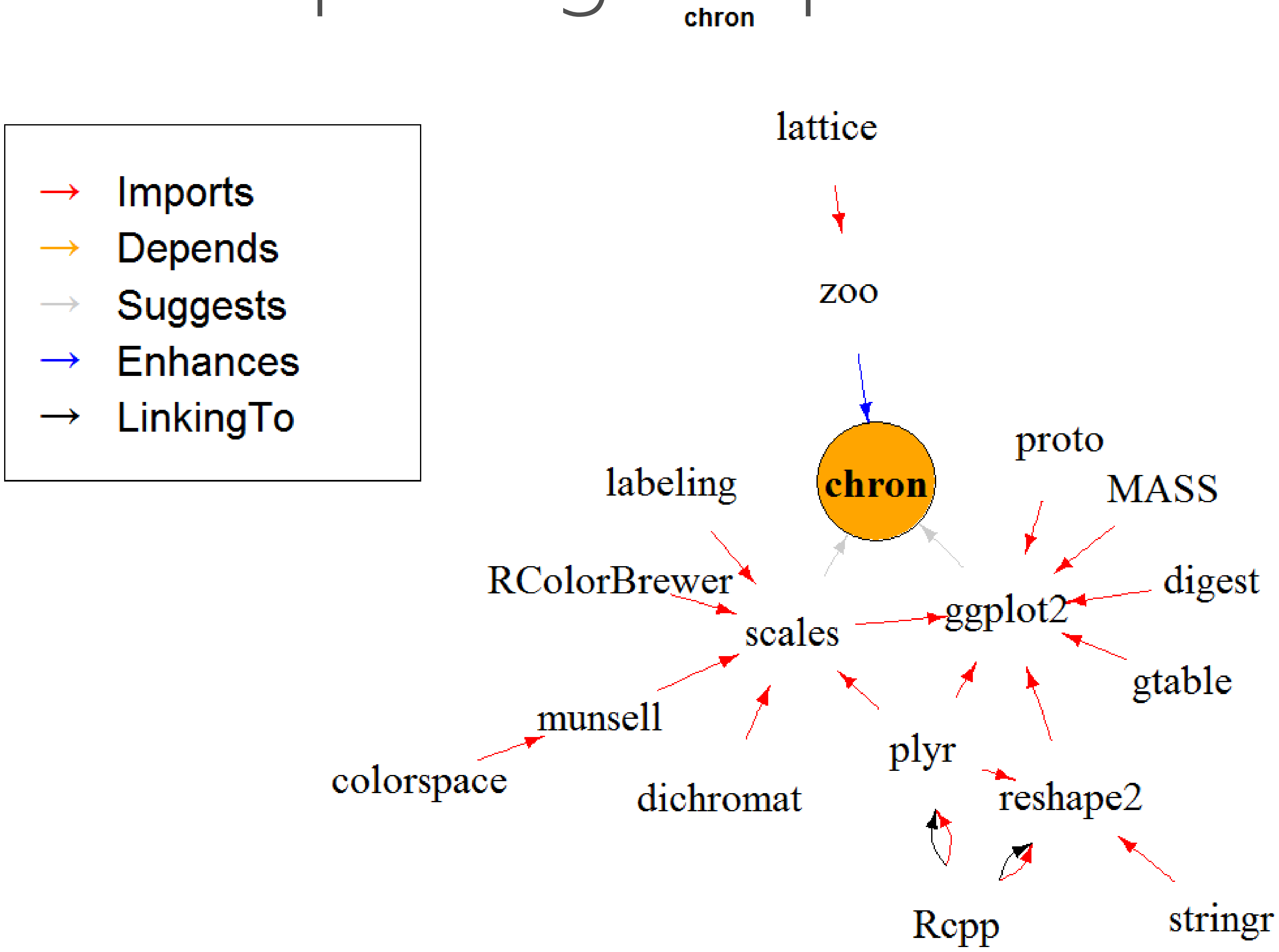
The network structure of CRAN

Andrie de Vries

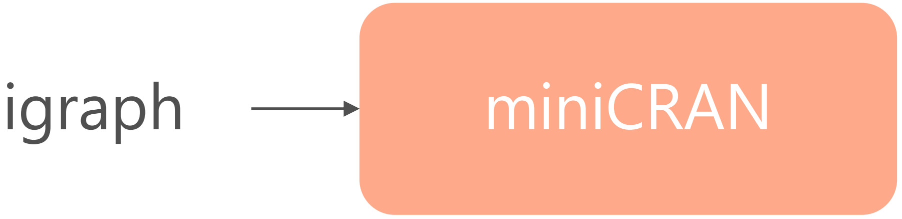
adevries@microsoft.com
@RevoAndrie

UseR!2015, Aalborg

A network of package dependencies

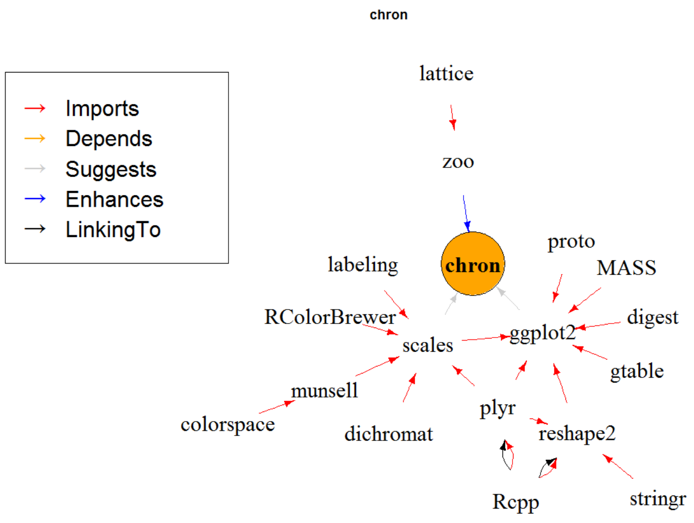


Using miniCRAN to create the network graph



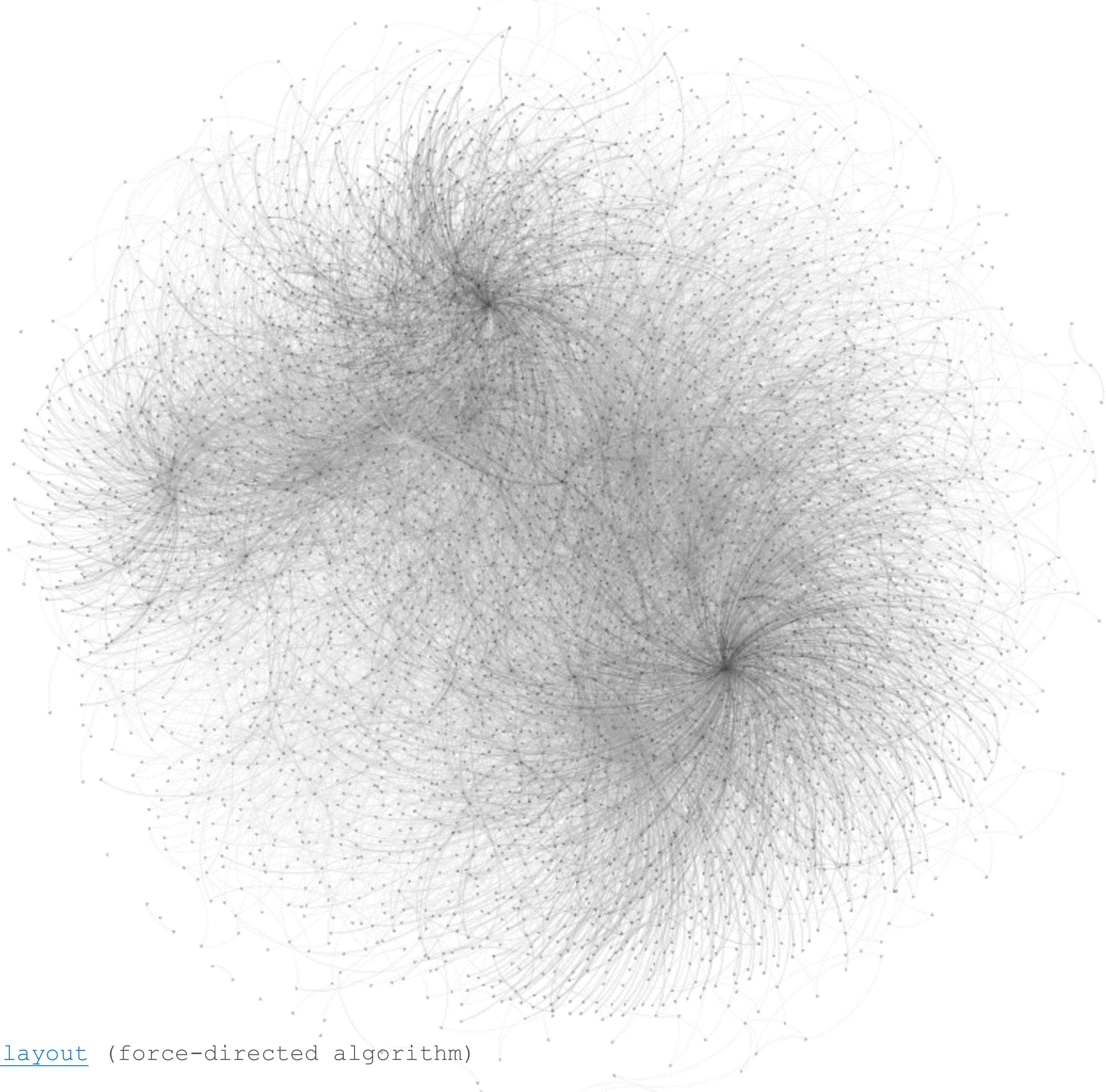
igraph → miniCRAN

```
pdb <- available.packages()
tags <- "chron"
library("miniCRAN")
dg <- makeDepGraph(tags, availPkgs = pdb)
plot(dg)
```



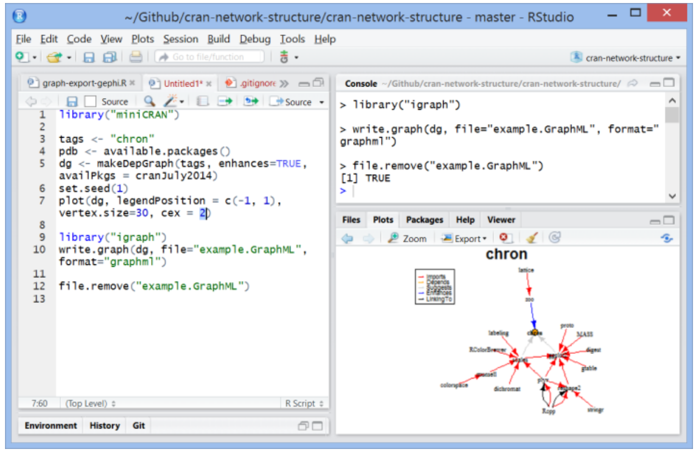
How to visualize a network of 6,791 CRAN packages?

CRAN



Note: Fruchterman Reingold layout (force-directed algorithm)

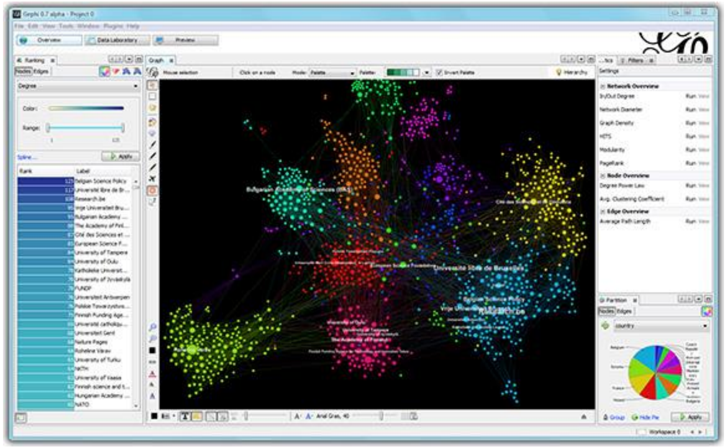
How to make this plot


~/Github/cran-network-structure/cran-network-structure - master - RStudio
File Edit Code View Plots Session Build Debug Tools Help
Go to file/function
cran-network-structure
graph-export-gephi.R
Untitled1*
.gitignore
Source
Console ~/Github/cran-network-structure/cran-network-structure/
Files Plots Packages Help Viewer
Zoom
Export
Environment History Git
7:60 (Top Level) R Script

```
library("miniCRAN")

tags <- "chron"
pdb <- available.packages()
dg <- makeDepGraph(tags, enhances=TRUE,
availPkgs = cranJuly2014)
set.seed(1)
plot(dg, legendPosition = c(-1, 1),
vertex.size=30, cex = 2)

library("igraph")
write.graph(dg, file="example.GraphML",
format="graphml")

file.remove("example.GraphML")
```

```
> library("igraph")

> write.graph(dg, file="example.GraphML", format="
graphml")

> file.remove("example.GraphML")
[1] TRUE
>
```

GraphML file



Gephi
(gephi.github.io)

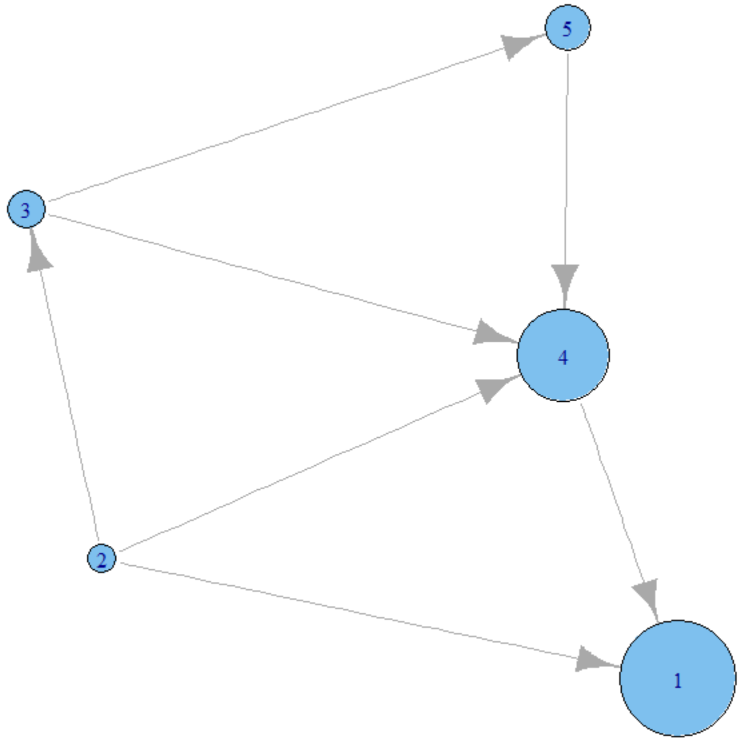
```
library("igraph")
write.graph(dg,
            file = "example.GraphML",
            format = "graphml")
```

Which nodes (packages) are more “important”?

Reverse dependencies influence importance

- Imagine that each package dependency is a vote of confidence, cast by the community of package authors

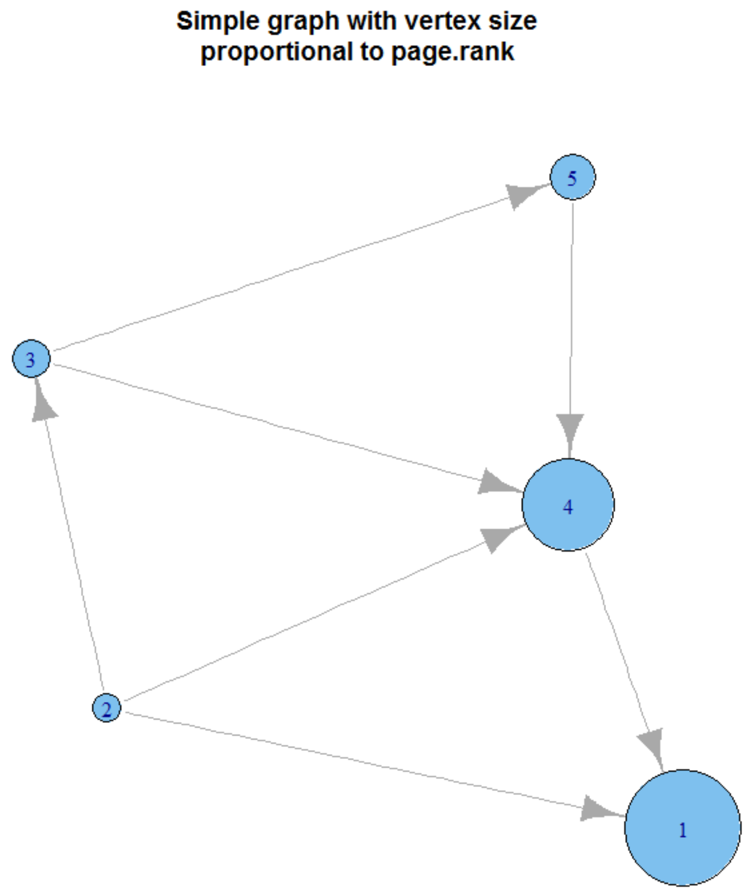
- How can we compute the importance of a package to downstream packages?



Page rank

- Page rank is famous as one of the criteria in google search results
- You can use page rank on any graph to compute “importance”

```
igraph::page.rank()
```



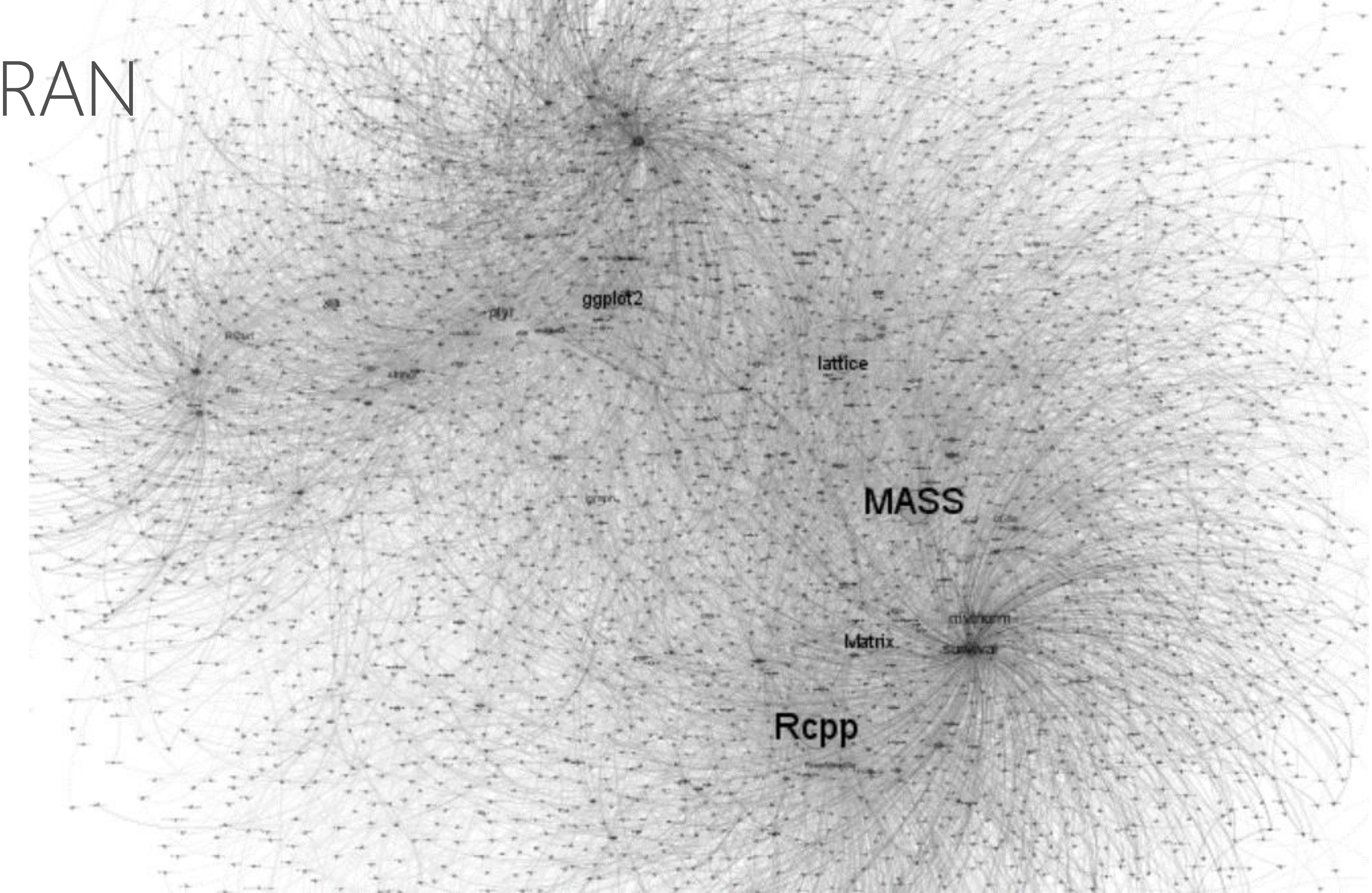
Simple graph with vertex size proportional to page.rank

Pagerank results

Package	PageRank (Dec 2014)	PageRank (Jun 2015)	Description
Rcpp	0.0166	0.0197	Interface to use C++ code in R
MASS	0.021	0.0196	Functions and datasets to support Venables and Ripley, 'Modern Applied Statistics with S' (4th edition, 2002).
Matrix	0.01	0.0095	Sparse matrix engine
ggplot2	0.0073	0.0086	Graphics engine
lattice	0.0096	0.0085	Base R package for lattice (trellis) graphics
mvtnorm	0.0088	0.0083	Multivariate normal distributions
survival	0.0083	0.0079	Time-to-event analysis
plyr	0.0067	0.0072	Group-by operations
igraph	0.0047	0.0049	Analyse graph structures
XML	0.0047	0.0047	Parse and manipulate documents in XML format

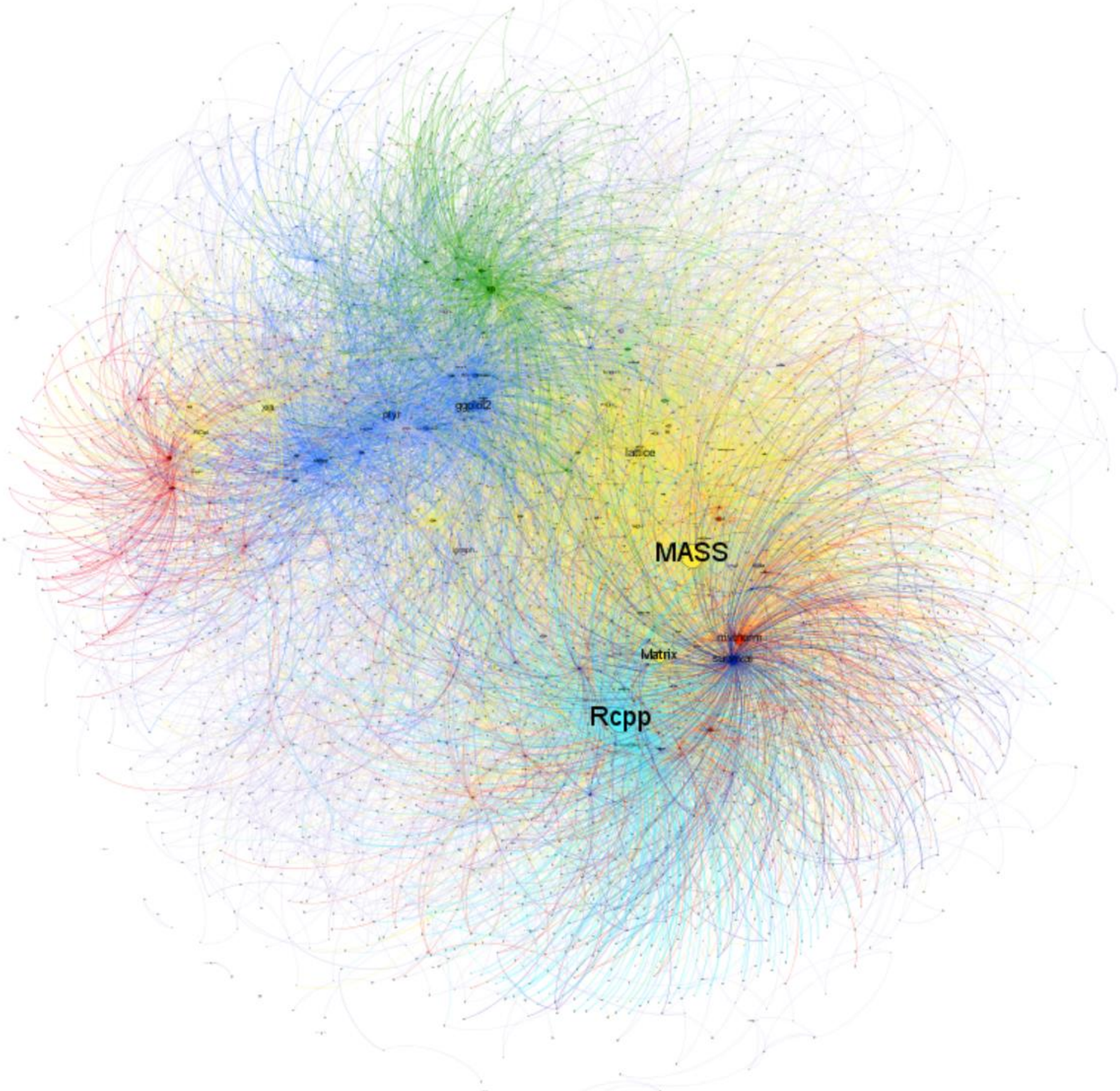
http://blog.revolutionanalytics.com/2014/12/a-reproducible-r-example-finding-the-most-popular-packages-using-the-pagerank-algorithm.html

CRAN

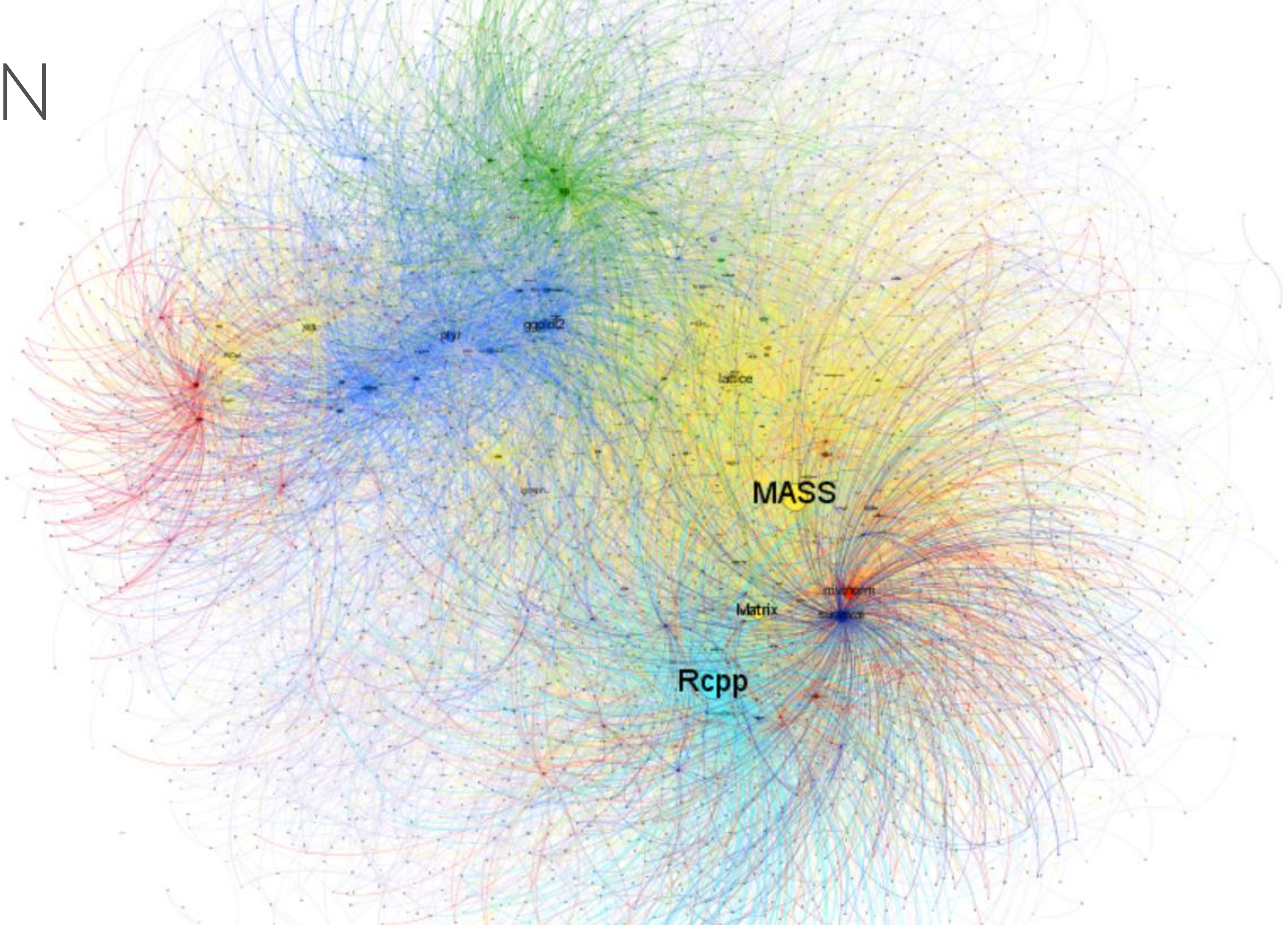

ggplot2
plyr
lattice
MASS
Matrix
survival
mvtnorm
Rcpp
igraph

Can we detect clusters of packages?

CRAN

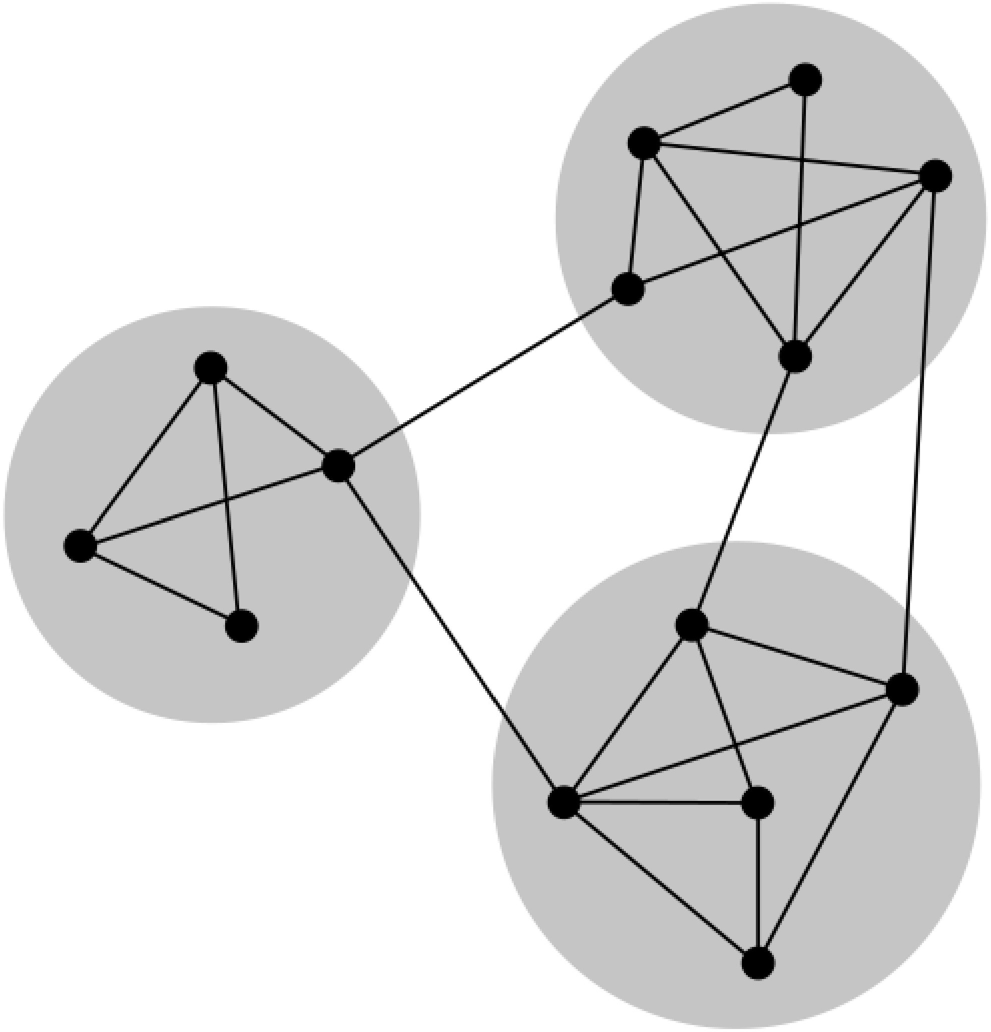

ggplot2
plyr
lattice
MASS
mvtnorm
Matrix
survival
Rcpp
igraph

CRAN


plyr
ggplot2
lattice
igraph
MASS
mvtnorm
Matrix
survival
Rcpp

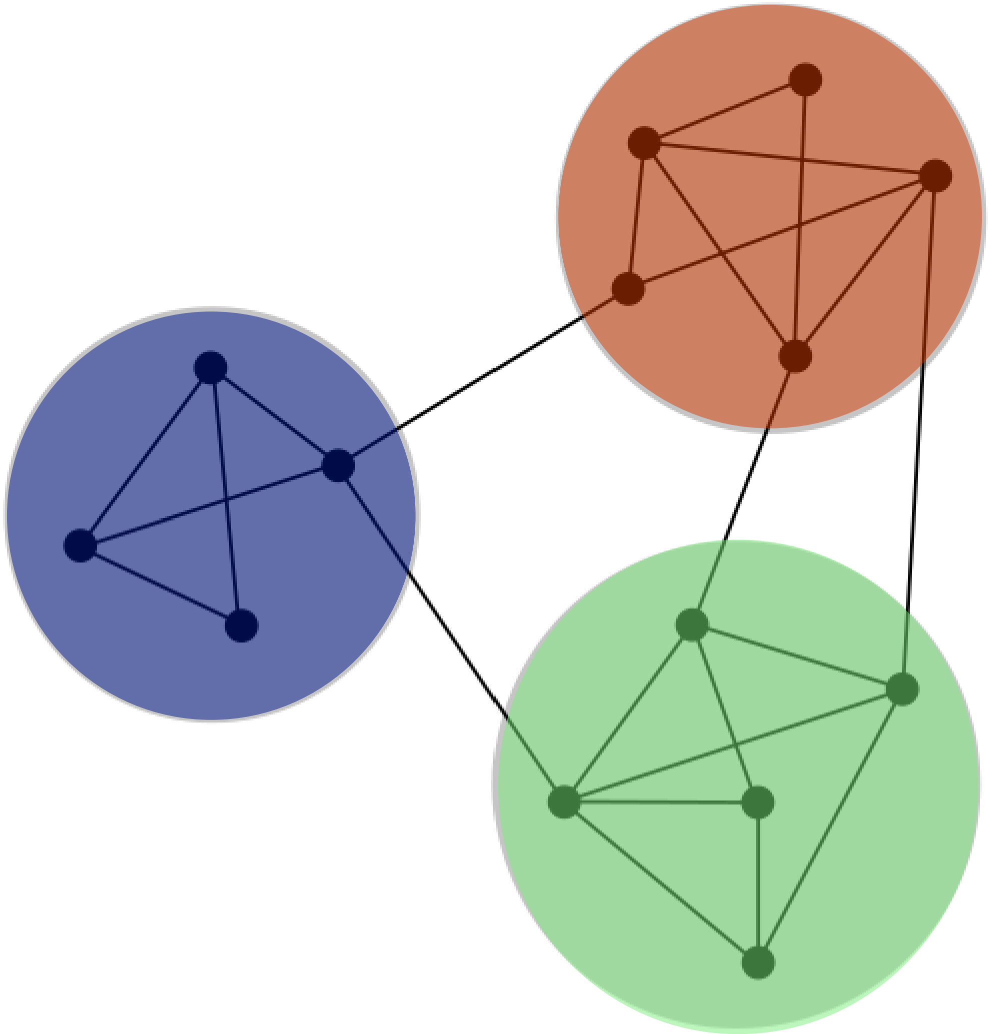
Community detection

- Communities are densely connected groups of nodes in a graph
- Several implemented algorithms in igraph:
  - Fast greedy
  - Walktrap
  - Spinglass
  - Leading eigenvector
  - Edge betweenness



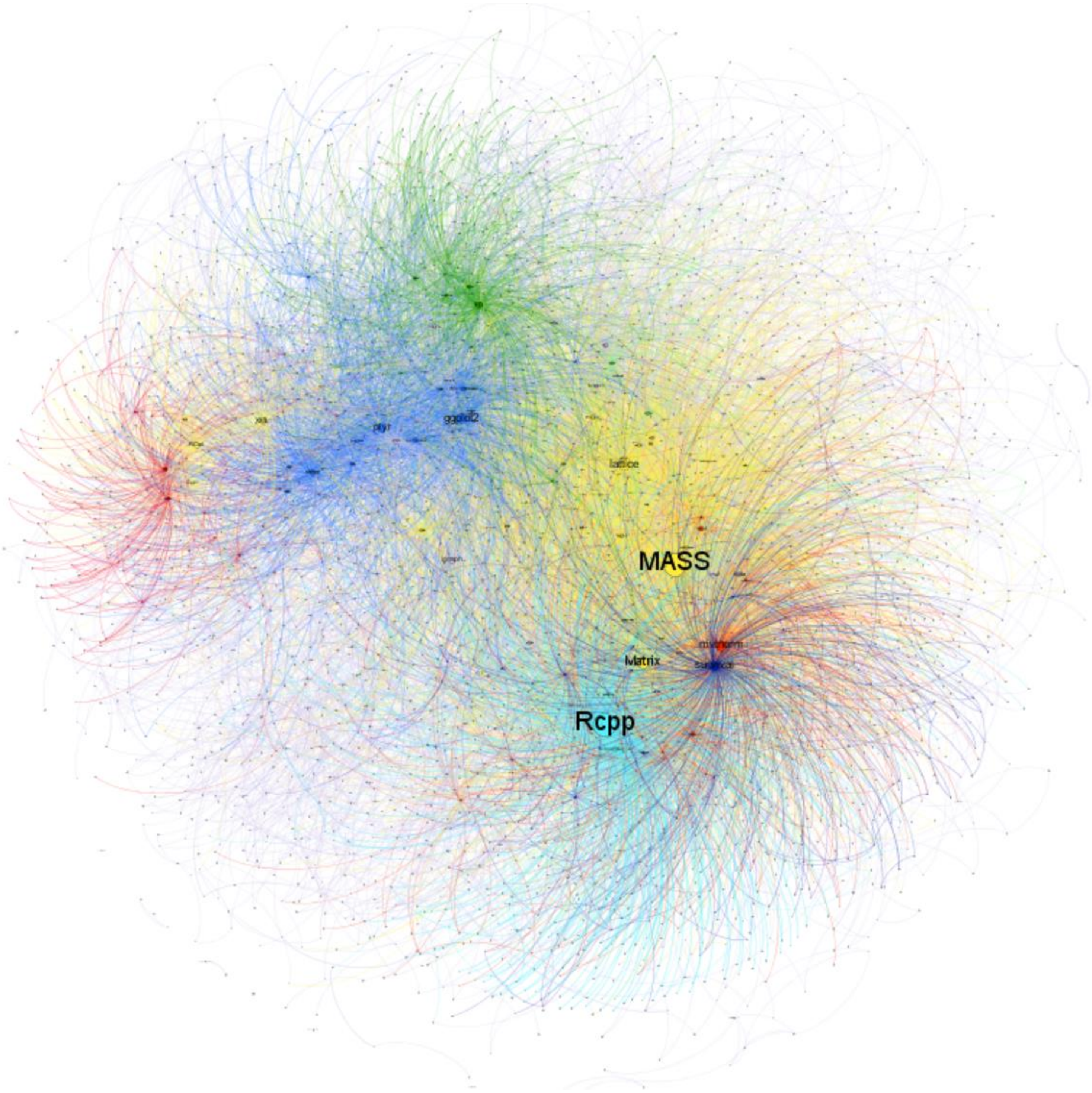
The walktrap algorithm

- `?walktrap.community`
    - This function tries to find densely connected subgraphs, also called communities in a graph via random walks. The idea is that short random walks tend to stay in the same community.
- Efficient algorithm for large, densely connected graphs*

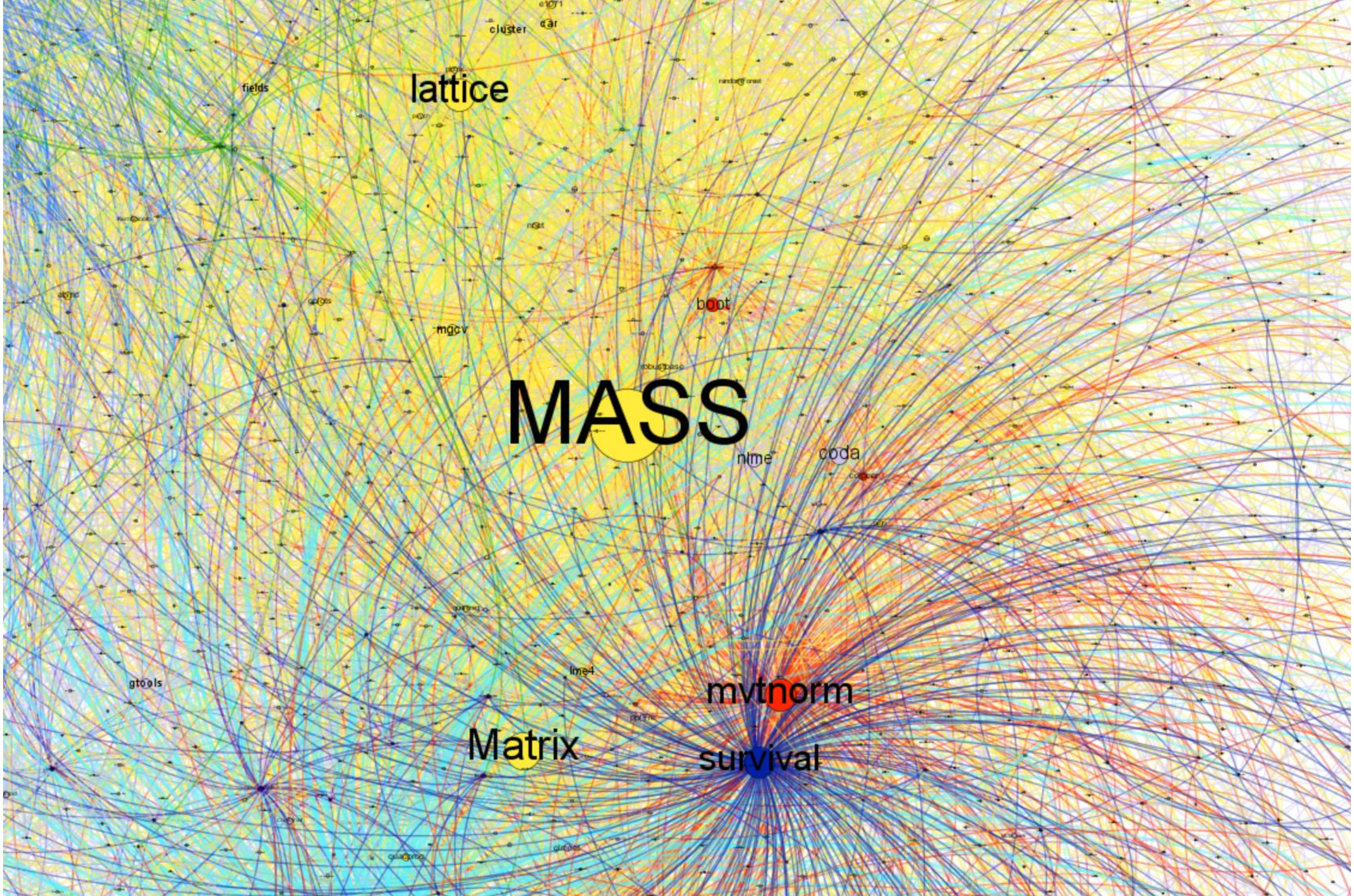


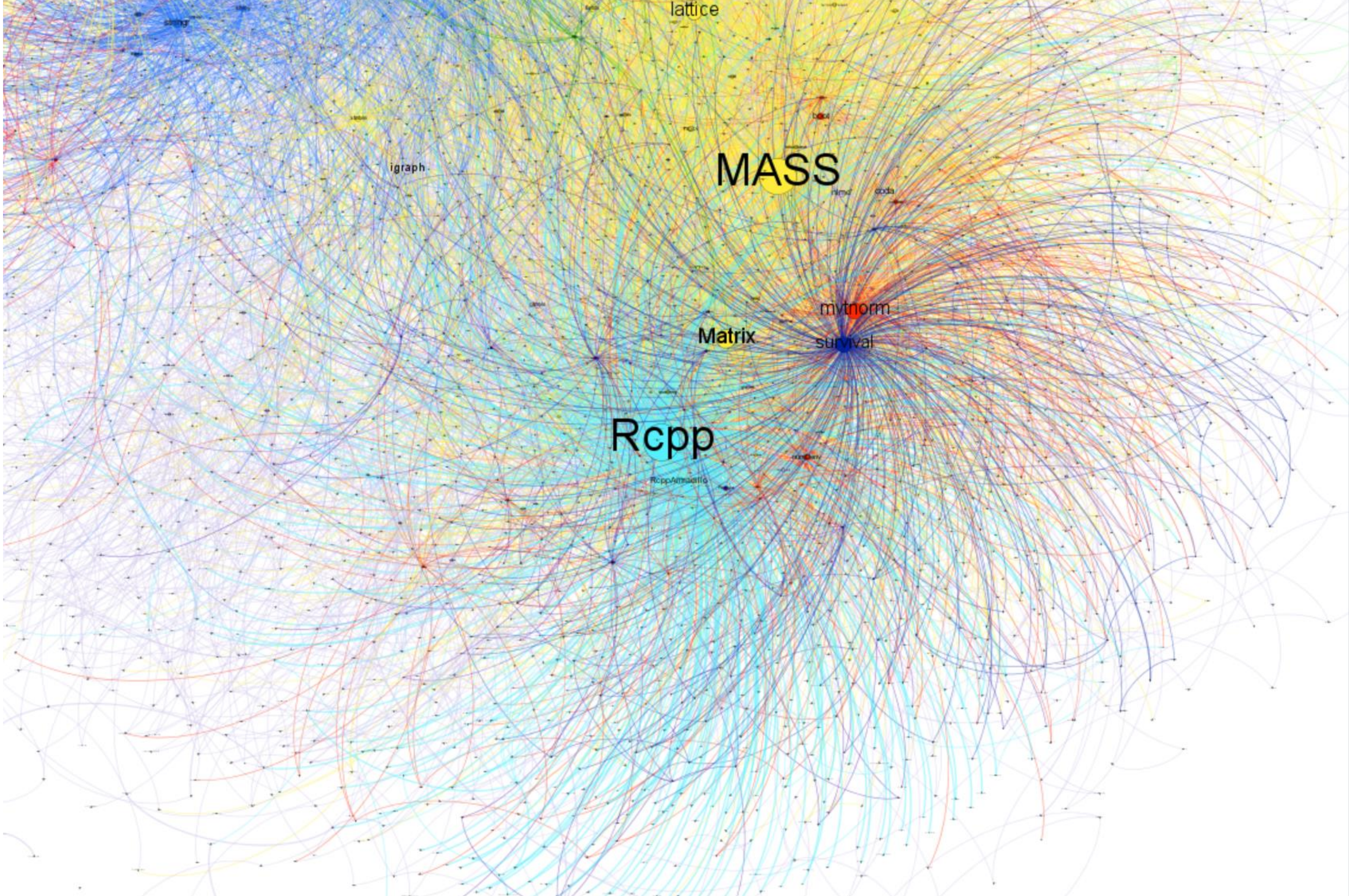
*Source: Computing communities in large networks using random walks (long version), Pascal Pons, Matthieu Latapy, http://arxiv.org/abs/physics/0512106

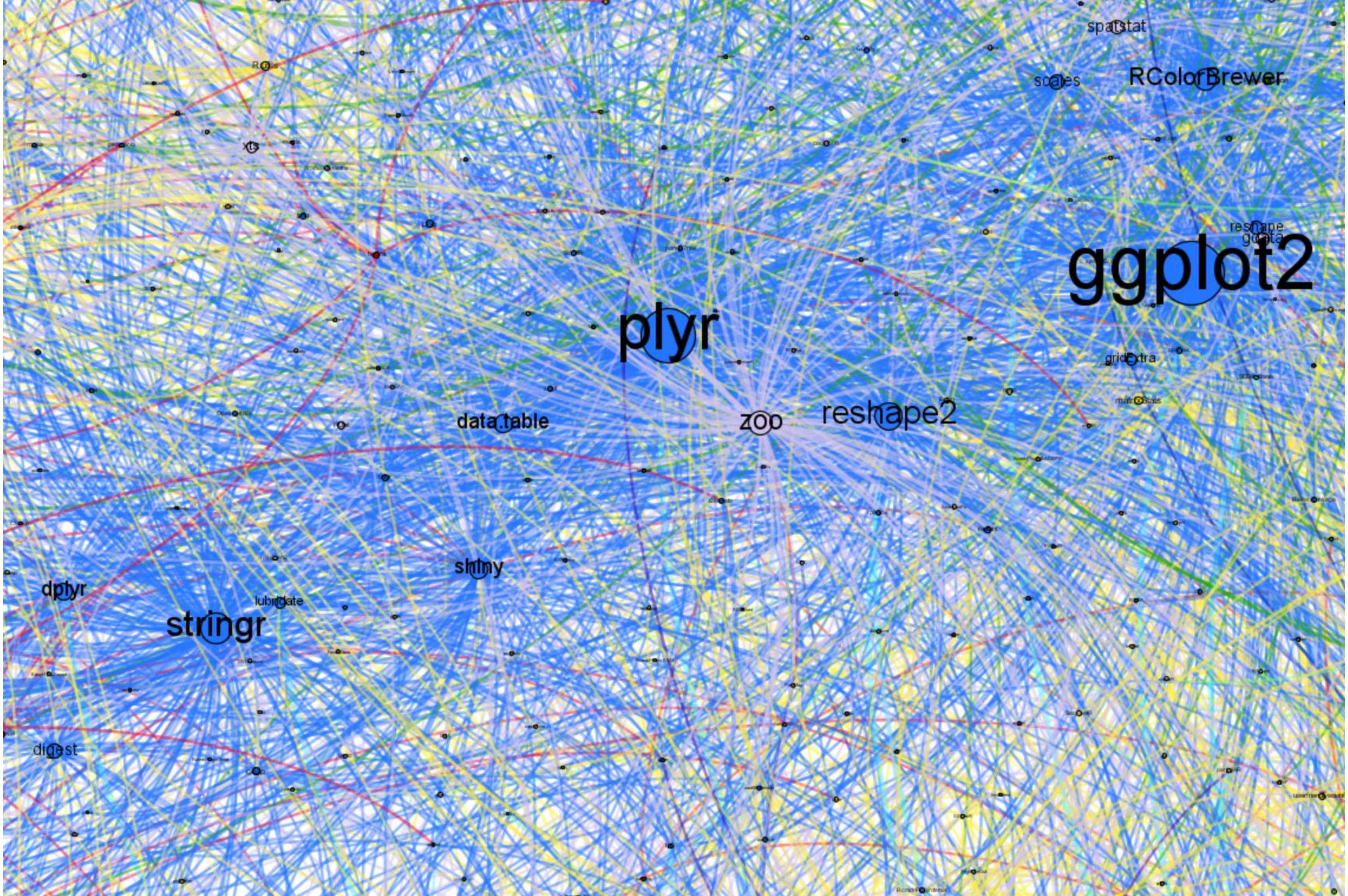
CRAN

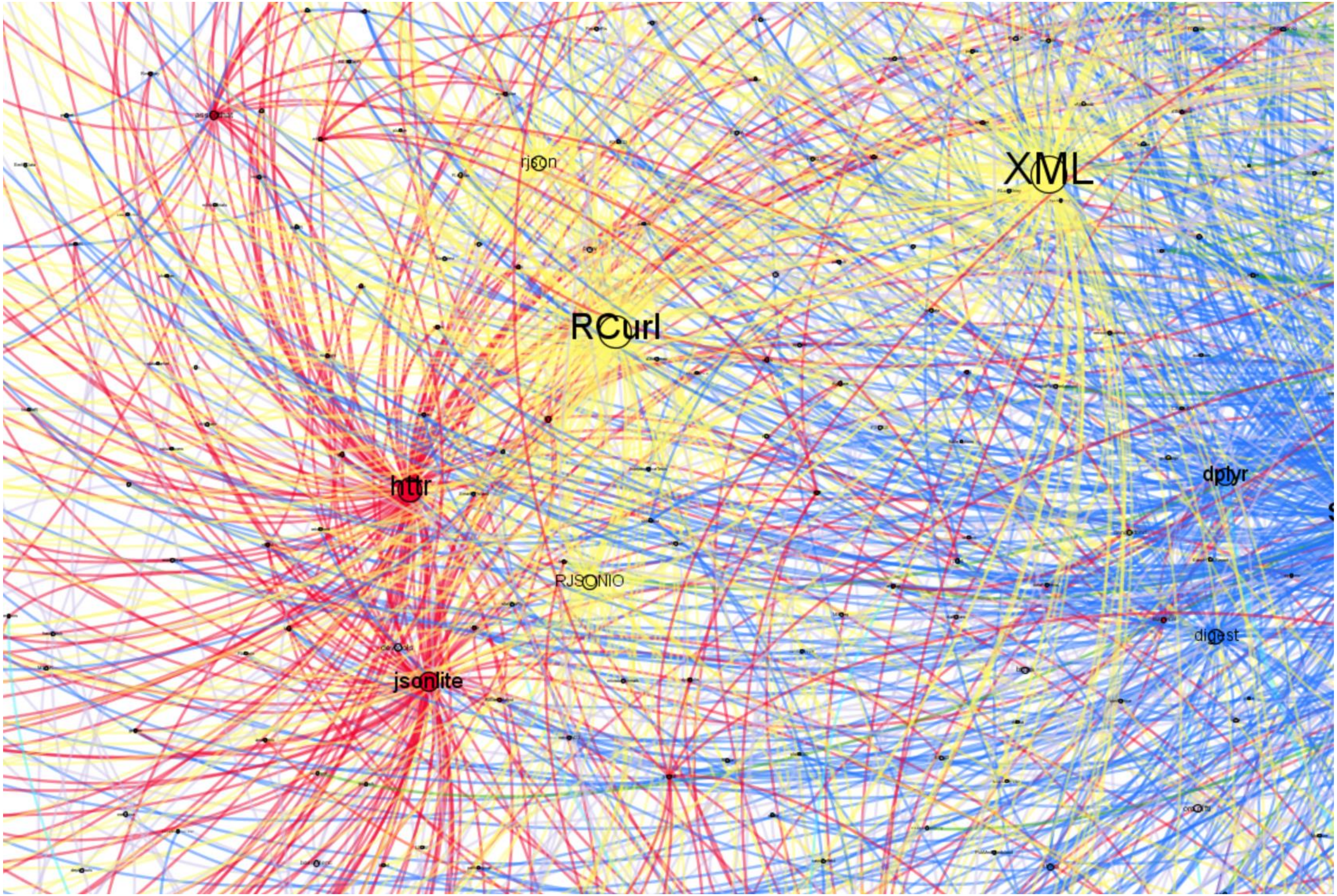

ggplot2
plyr
lattice
igraph
MASS
mvtnorm
Matrix
survival
Rcpp

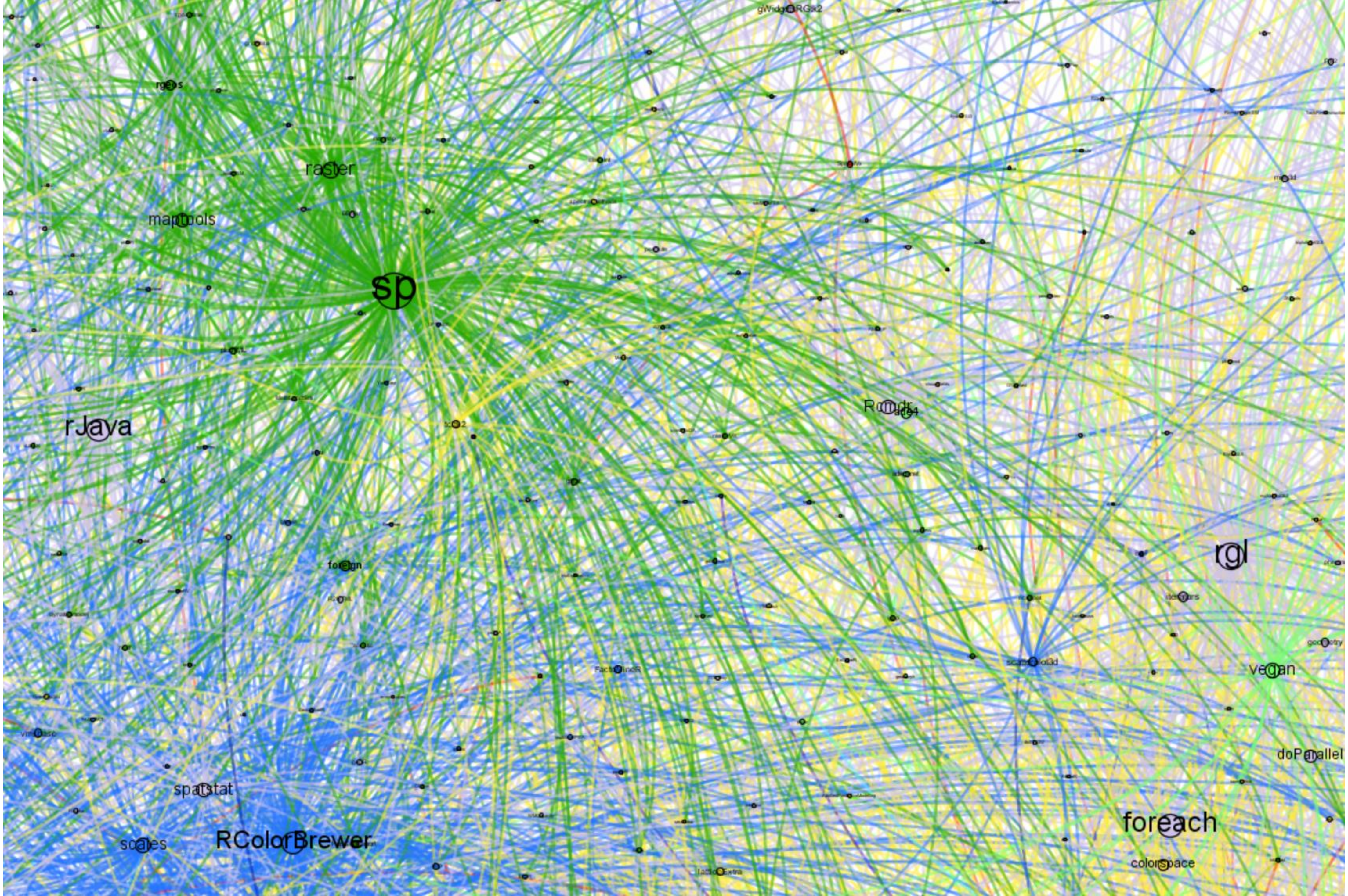
Zooming in


e1071
cluster
car
fields
lattice
randomForest
boot
mgcv
robustbase
MASS
nlme
coda
lme4
gtools
mvtnorm
Matrix
survival


lattice
igraph
boot
MASS
nlme
coda
mvtnorm
Matrix
survival
Rcpp
RcppArmadillo

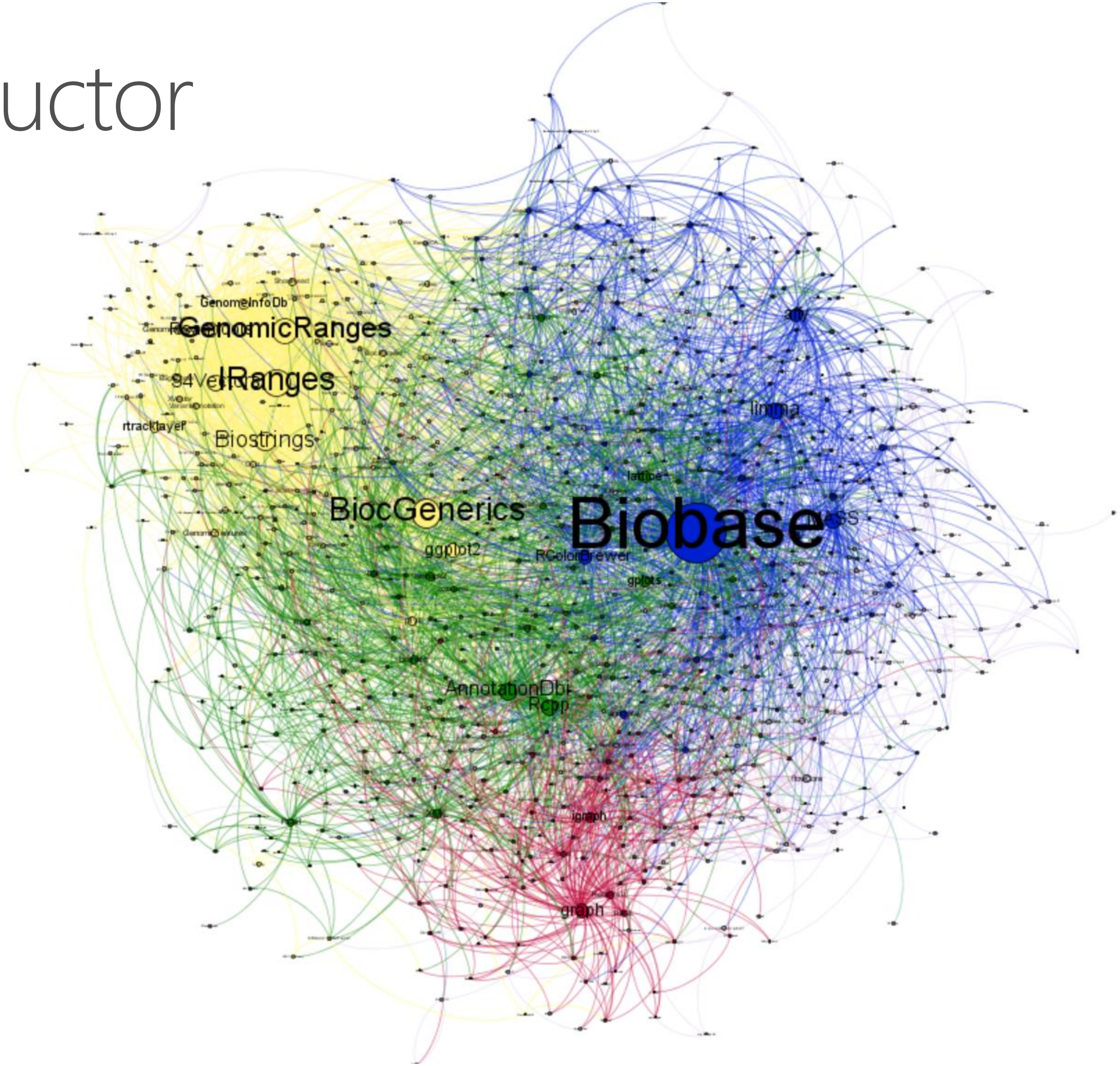

spatstat
scales
RColorBrewer
reshape
gtable
ggplot2
plyr
gridExtra
matrixStats
data.table
zoo
reshape2
shiny
dplyr
lubridate
stringr
digest


rjson
XML
RCurl
httr
dplyr
RJSONIO
digest
jsonlite


gWidgetsRGtk2
rgeos
raster
maptools
sp
rJava
Rcmdr
foreign
rgl
scatterplot3d
FactoMineR
vegan
geometry
doParallel
spatstat
scales
RColorBrewer
latticeExtra
foreach
colorspace

Is BioConductor different?

BioConductor

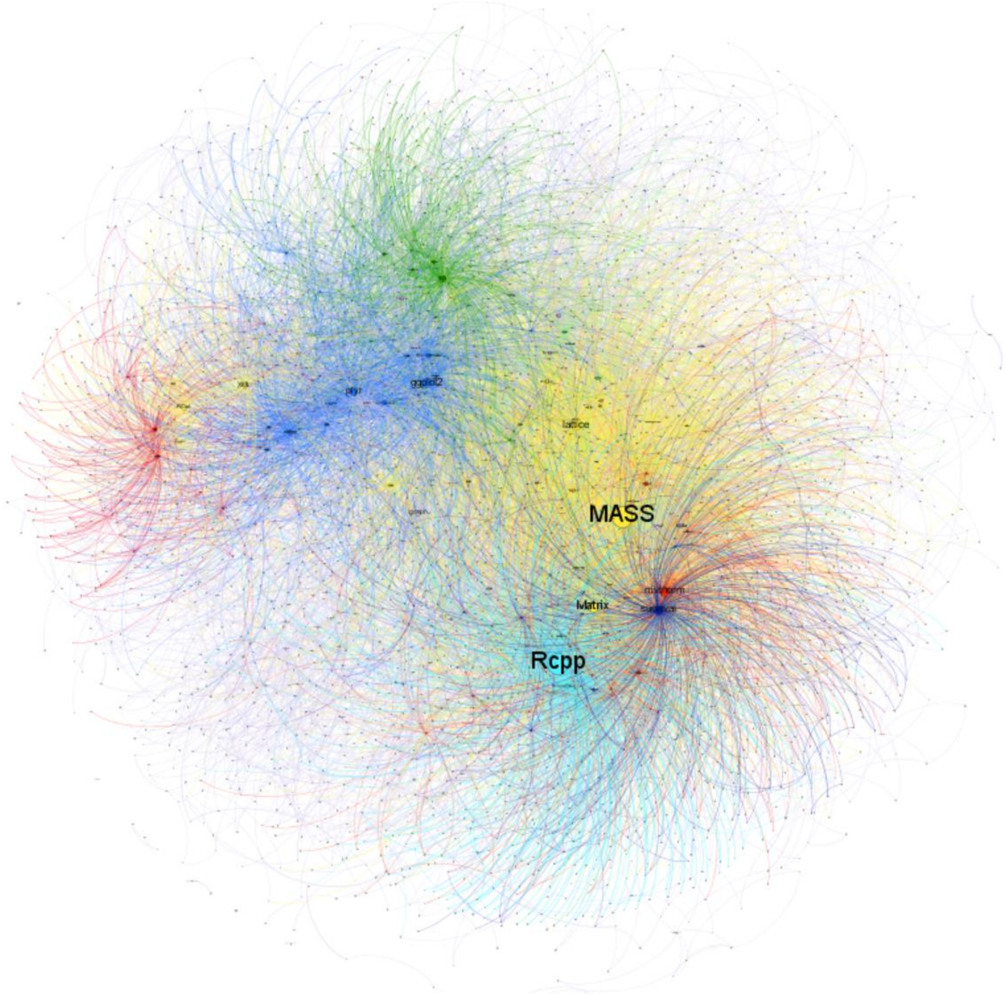


Further analysis required

Summary

Summary

```
library(miniCRAN)
makeDepGraph()

library(igraph)
page.rank()
walktrap.community()
write.graph()
```

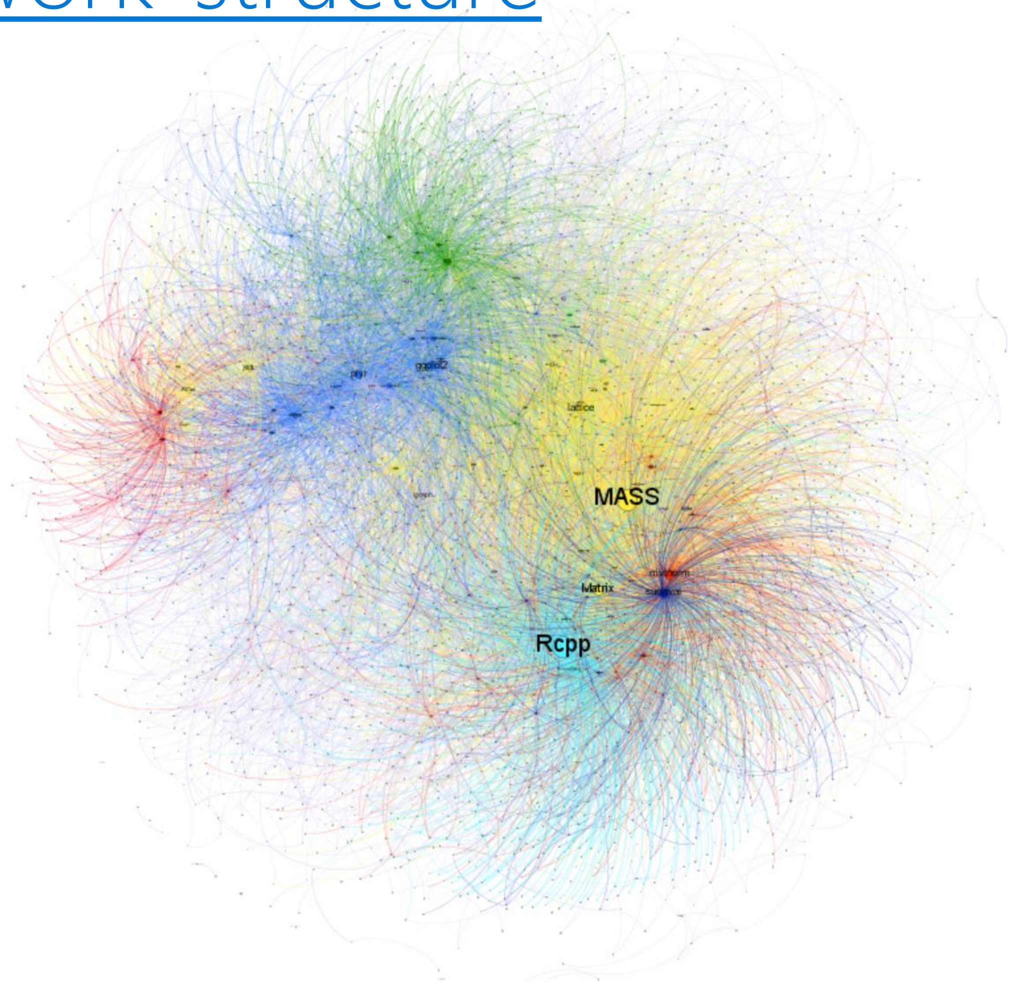


Scripts available at:

https://github.com/andrie/cran-network-structure

adevries@microsoft.com

@RevoAndrie


ggplot2
plyr
lattice
MASS
mvtnorm
Matrix
Rcpp