

kBestShortestPaths Package

Adrin Jalali

October 25, 2011

adrin.jalali@gmail.com

Contents

1	Licensing	1
2	Introduction	1
3	Usage	1

1 Licensing

This package is licensed under GNU General Public License version 3 (GPLv3).

2 Introduction

This package is an interface for AAAAAA's implementation of [1] algorithm for finding the k-shortest paths in a given graph in $O(m + n \times \log(n) + k)$ where m , n , and k are the number of edges, nodes and paths, respectively. TODO: add the original implementation's link and name of the author

The core function in this package is *kBestShortestPaths* which provides a *Paths* object as output. That object would contain the computed paths and the two functions *summary* and *plot* present some information about that object.

3 Usage

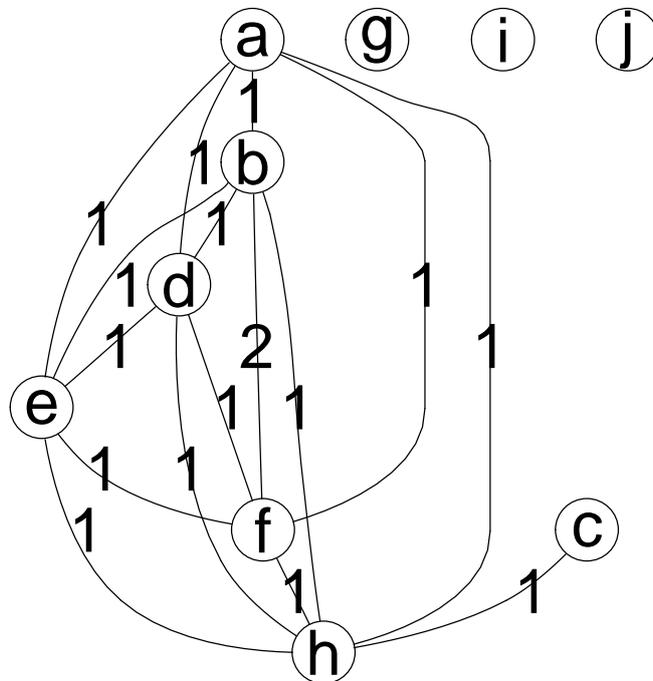
The input arguments to the *kBestShortestPaths* function are a graph, a source node, a destination node, and the number of paths. The second and third arguments are character strings representing the node names. The last argument is an integer number. The first input argument is a graphNEL object. It is under the graph package and is an S4 class. Node names, edges, and edge weights are used to find shortest paths. As an example here is illustrated how to make a sample random graph:

```
> library(graph)
> set.seed(123)
> V <- letters[1:10]
```

```
> M <- 1:4
> g1 <- randomGraph(V, M, 0.2)
```

Note that the first line loads the graph library which contains the random-Graph function and the graphNEL-class. Having Rgraphviz installed, it will plot the layed-out graph:

```
> library(Rgraphviz)
> eAttrs <- list()
> ew <- as.character(unlist(edgeWeights(g1)))
> ew <- ew[setdiff(seq(along = ew), removedEdges(g1))]
> names(ew) <- edgeNames(g1)
> eAttrs$label <- ew
> plot(g1, edgeAttrs = eAttrs)
```



Using the *kBestShortestPaths* function the *k* shortest paths in *g1* can be identified: This makes the *z* variable, a Paths class object. The summary of this object can be obtained with the *summary* function.

```
> library(kBestShortestPaths)
> z <- kBestShortestPaths(g1, "a", "h", 5)
> summary(z)
```

A set of 5 shortest paths

```
> z
```

An object of class "Paths"
Slot "best.paths":

```
[[1]]
[1] "a~h"

[[2]]
[1] "a~b" "b~h"

[[3]]
[1] "a~d" "d~h"

[[4]]
[1] "a~e" "e~h"

[[5]]
[1] "a~f" "f~h"
```

Slot "path.scores":

```
[[1]]
[1] 1

[[2]]
[1] 1 1

[[3]]
[1] 1 1

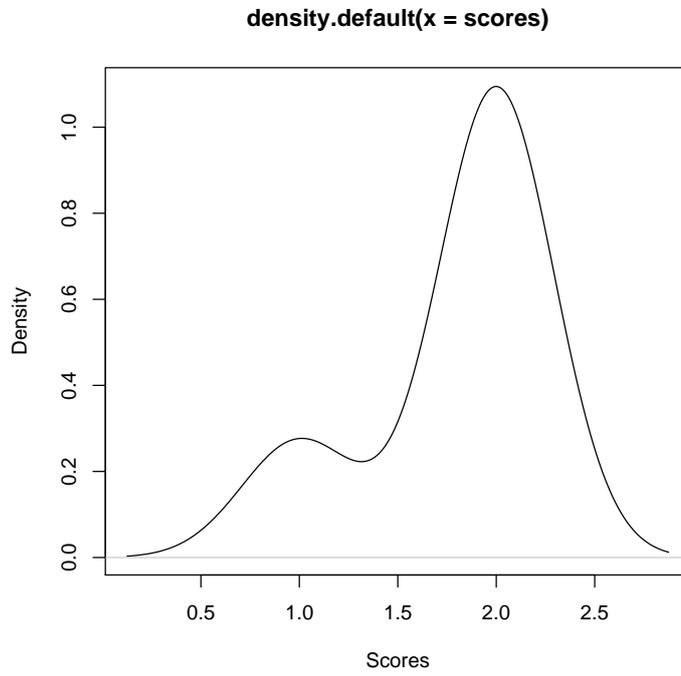
[[4]]
[1] 1 1

[[5]]
[1] 1 1
```

TODO: Talk about Z

And the plot function can be used to visualize the density of the found path weights.

```
> plot(z)
```



References

- [1] D. Eppstein. Finding the k shortest paths. *SIAM J. Comput.*, 28(2):652–673, 1998.