**EUCIC course 11-12 September 2023**

***Network reconstruction practical.***

We are going to reconstruct the within hospital network of the UMCG, based on the provided dataset (“patient location data.csv”). As explained during the lectures, such networks consist of “nodes” and “edges”. Nodes are the locations patients are admitted to, in this case wards within the hospital. Edges form the connections between the nodes, and denote the number of patients that move from one node to an other.

> As a first step, create a new R script, and load the required libraries for this exercise.

*library(data.table)*

*library(HospitalNetwork)*

*library(lubridate)*

*library(ggplot2)*

*library(dplyr)*

> Now, load the data from the above mentioned file:

*patMovDB<-read.csv("Resources/patient location data.csv",sep=";")*

> How many records do the data contain? And how many columns?

> What data does each of the columns contain?

> How many unique patients are in the data?

Let’s start building the network from these data. We will start with tracking the admissions of one example patient, then move on to all patients. To make it more interesting, we will find the patient with most admissions to different wards and use that one as the first example.

> find the patient with most admissions to different wards

*patOverview<-patMovDB%>%group\_by(case)%>%*

*summarise(nAdm=n(),*

*nWards=length(unique(careUnit)))%>%*

*as.data.frame()*

*arrange(patOverview,desc(nWards))[1,]*

> How many wards did this patients visit, during how many admissions?

> How often did the patient get moved between which pairs of wards? (Note: order by admission date)

> Draw each of the wards on a piece of paper and connect them by the patient’s movements between them.

You have just drawn a network (the single patient network) by hand. We can do this in an automated way, by using the HospitalNetwork package. This package works in two stage: first it checks the data for inconsistencies and missing data (using checkBase), then it reconstructs the network based on the checked data (using hospinet\_from\_subject\_database). We will construct the above single patient network using this package:

> Check the data:

*examplePatientChecked <- checkBase(examplePatient,*

*admDate = "admission",*

*disDate="discharge",*

*facilityID = "careUnit",*

*subjectID = "case",*

*convertDates = T,*

*dateFormat = "ymd",*

*deleteMissing = "record"*

*)*

>build the network:

*singlePat\_NW←hospinet\_from\_subject\_database(examplePatientChecked)*

> plot the network contact matrix

*plot(singlePat\_NW)*

> plot the network as a circular network

*plot(singlePat\_NW, type = "circular\_network",alphaSet=1)+coord\_fixed()*

> extra question: the patient goes home between some of their admissions, what happens if we don’t count these as movements? This can be answered using the “manual” method as well as using the hospinet\_from\_subject\_database() function.

We can now build the network using the data from all patients included in the data. Again, we first need to check the data before constructing the network.

> Check the entire dataset (named *patMovDB* above)

> Were there any errors in the data? If yes, what, and how many?

> Construct the full network

> Make the same plots as above.

Let’s see what we can say about this within-hospital-ward-based network.

> What’s the most connected ward?

> In terms of other wards it’s connected to, or...

> in terms of the total number of patient movements to other wards.

> Are certain wards more connected to each other than others, and do they form distinct groups or clusters?