Algorithm for defining hospital stays

Katrine Damgaard[1*], Jon Helgeland[1],

1. Norwegian Knowledge Centre for the Health Services
*Contact author: kda@nokc.no

Keywords: Healthcare, hospitals, Multicore

Objective: The objective was to link hospital stays which constituted a complete chain of admissions for patients transferred between wards/department within/between hospitals, in order to compare mortality between hospitals

Material and methods: The Norwegian Knowledge Centre for the Health Services has developed a system for retrieving data from the patient administrative systems (PAS) at all Norwegian hospitals. By use of this system, we retrieved PAS data from each hospital for patients discharged during 2005-2009.

Each record corresponds to one admission at one ward within a department in a hospital. To define a complete patient stay, we have to aggregate the admissions; stays at one department, hospital stays and stays involving more than one hospital. The latter is important for patients transferred between hospitals to obtain their complete medical treatment history. All permanent residents in Norway have a personal identification number (PIN) which enables linking between hospitals.

We have developed an algorithm that concatenates the ward admissions to a chain of admissions. The function gives you all the levels of stays described above. The input to the function is a serial number, PIN, date of admission, date of discharge, hospital, department and further optional parameters. The function uses the package **multicore** in *R*, for the ability to use more than one processor for large datasets. One of the properties of the algorithm is that you can choose different time windows, or tolerances, for concatenating the ward admissions. We chose 24 hours as tolerance for our purpose.

Results: With the choice of 24 hours time window for difference between time of discharge and next admission, we found 10 485 022 hospital stays involving one or more hospitals out of 16 370 163 ward admissions from 3 304 546 patients.

Conclusion: The algorithm can be used to aggregate large patient administrative data sets, with acceptable running times. The tolerance limit can be adjusted to suit the purpose and the data at hand.