**Recap**

In the last post, the discipline of event log and process mining were defined. The bupaR package was introduced as a technique to do process mining in R.

**Objectives for This Post**

1. Visualize workflow
2. Understand the concept of activity reoccurrences

We will use a pre-loaded dataset sepsis from the bupaR package. This event log is based on real life management of [sepsis](https://www.mayoclinic.org/diseases-conditions/sepsis/symptoms-causes/syc-20351214) from the point of admission to discharge.

The dataset has 15214 activity instances and 1050 cases.

library(plyr)

library(tidyverse)

library(bupaR)

sepsis

## Event log consisting of:

## 15214 events

## 846 traces

## 1050 cases

## 16 activities

## 15214 activity instances

##

## # A tibble: 15,214 x 34

## case\_id activity lifecycle resource timestamp age crp

##

## 1 A ER Regi~ complete A 2014-10-22 11:15:41 85 NA

## 2 A Leucocy~ complete B 2014-10-22 11:27:00 NA NA

## 3 A CRP complete B 2014-10-22 11:27:00 NA 210

## 4 A LacticA~ complete B 2014-10-22 11:27:00 NA NA

## 5 A ER Tria~ complete C 2014-10-22 11:33:37 NA NA

## 6 A ER Seps~ complete A 2014-10-22 11:34:00 NA NA

## 7 A IV Liqu~ complete A 2014-10-22 14:03:47 NA NA

## 8 A IV Anti~ complete A 2014-10-22 14:03:47 NA NA

## 9 A Admissi~ complete D 2014-10-22 14:13:19 NA NA

## 10 A CRP complete B 2014-10-24 09:00:00 NA 1090

## # ... with 15,204 more rows, and 27 more variables: diagnose ,

## # diagnosticartastrup , diagnosticblood , diagnosticecg ,

## # diagnosticic , diagnosticlacticacid ,

## # diagnosticliquor , diagnosticother , diagnosticsputum ,

## # diagnosticurinaryculture , diagnosticurinarysediment ,

## # diagnosticxthorax , disfuncorg , hypotensie ,

## # hypoxie , infectionsuspected , infusion ,

## # lacticacid , leucocytes , oligurie ,

## # sirscritheartrate , sirscritleucos ,

## # sirscrittachypnea , sirscrittemperature ,

## # sirscriteria2ormore , activity\_instance\_id , .order

We will subset a smaller and more granular event log to make illustrations more comprehensible.

activity\_frequency(sepsis, level = "activity") %>% arrange(relative)# least common activity

## # A tibble: 16 x 3

## activity absolute relative

##

## 1 Release E 6 0.000394

## 2 Release D 24 0.00158

## 3 Release C 25 0.00164

## 4 Release B 56 0.00368

## 5 Admission IC 117 0.00769

## 6 Return ER 294 0.0193

## 7 Release A 671 0.0441

## 8 IV Liquid 753 0.0495

## 9 IV Antibiotics 823 0.0541

## 10 ER Sepsis Triage 1049 0.0689

## 11 ER Registration 1050 0.0690

## 12 ER Triage 1053 0.0692

## 13 Admission NC 1182 0.0777

## 14 LacticAcid 1466 0.0964

## 15 CRP 3262 0.214

## 16 Leucocytes 3383 0.222

sepsis\_subset<-filter\_activity\_presence(sepsis, "Release E") # cases with least common activity to achieve smaller eventlog

**Visualizing Workflow**

In an event log, the sequence of activities are captured either using time stamps or activity instance identifier. The sequential order of activities allows us to create a workflow on how a case was managed. This workflow can be compared against theoretical workflow models to identify deviations. It can also reveal the reoccurrence of activities which will be covered later. The most intuitive approach to examine workflow is with a process map and this is achieved with the process\_map function from bupaR

The process map is either created base on frequency of activities (i.e. process\_map(type = frequency())) or base on duration of activities (i.e. process\_map(type= performance())). I have focused on the frequency aspect which is the default argument.

There are 4 arguments that you can supply to process\_map(type = frequency()) which determines the value to be displayed. The values can reflect either the number of activity instances or the number of cases. The values displayed can be in absolute or relative frequency.

**Process mapping based on absolute activity instances**

sepsis\_subset %>% process\_map(type = frequency("absolute"))

**Process mapping based on absolute number of cases for the activity**

sepsis\_subset %>% process\_map(type = frequency("absolute\_case"))

The darker the colour of the boxes and the darker and thicker the arrows, the higher the value. From the process map, activities leading up to “Release E” were “CRP”, “Leucocytes” and “Lactic Acid”. “CRP” contributed to half the activity instances and cases leading up to “Release E”.

**Reoccurence of Activities**

In the above process map, the top of the boxes for “CRP”, “Leucocytes”, “Admission NC” and “Lactic Acid” has an arrow head and its tail belonging to the same activity. These arrows indicate reoccurrence of the activities for the same case. Reoccurrence of activities can suggest inefficiency or interruptions or disruptions which may warrant further investigation to optimise workflow.

bupaR has 2 constructs to define reoccurrence of activities for the same case.

**Construct 1 (resource reduplicating the activity)**

If the same resource reduplicates the activity for a particular case, it is known as “repeat” in bupaR’s terminology. If a different resource reduplicates the activity, it is known as “redo”.

**Construct 2 (activity instances when the activity reoccured)**

When the activity for a specific case reoccurs as consecutive activity instances, it is term as “self loop”. This is an example of “1 self loop”

| **Activity** | **Self Loop** |
| --- | --- |
| ER Triage | – |
| CRP | 1 |
| CRP | 1 |
| Release A | – |

When the activity reoccurs as non consecutive activity instances, it is known as “repetition”. In other words, there are other activities that occur before that specific activity is replicated. This is an example of “1 repetition”

| **Activity** | **Repetition** |
| --- | --- |
| ER Triage | – |
| CRP | 1 |
| Leucocytes | – |
| CRP | 1 |

The permutation of these constructs result in these 4 types of activity reoccurrences:

1. Redo self-loop
2. Redo repetition
3. Repeat self-loop
4. Repeat repetition

Let’s look at some examples using bupaR

**Which activities reoccurred consecutively in a case?**

number\_of\_repetitions(sepsis\_subset, level="activity", type="all")

## # Description: activity\_metric [12 x 3]

## activity absolute relative

##

## 1 Admission IC 0 0

## 2 Admission NC 2 0.167

## 3 CRP 6 0.188

## 4 ER Registration 0 0

## 5 ER Sepsis Triage 0 0

## 6 ER Triage 0 0

## 7 IV Antibiotics 0 0

## 8 IV Liquid 0 0

## 9 LacticAcid 3 0.158

## 10 Leucocytes 6 0.136

## 11 Release E 0 0

## 12 Return ER 0 0

**Which activities reoccurred consecutively in a case where the same resource repeated the activities?**

number\_of\_repetitions(sepsis\_subset, level="activity", type="repeat")

## # Description: activity\_metric [12 x 3]

## activity absolute relative

##

## 1 Admission IC 0 0

## 2 Admission NC 0 0

## 3 CRP 6 0.188

## 4 ER Registration 0 0

## 5 ER Sepsis Triage 0 0

## 6 ER Triage 0 0

## 7 IV Antibiotics 0 0

## 8 IV Liquid 0 0

## 9 LacticAcid 3 0.158

## 10 Leucocytes 6 0.136

## 11 Release E 0 0

## 12 Return ER 0 0

**Which cases had activities duplicated? The duplicated activities were interrupted by other activities and these duplicated activities were redone by a different resource.**

number\_of\_selfloops(sepsis\_subset, level="case", type = "redo")

## # Description: case\_metric [6 x 3]

## case\_id absolute relative

##

## 1 BCA 0 0

## 2 CY 1 0.0303

## 3 JAA 0 0

## 4 JM 1 0.0769

## 5 LG 1 0.0244

## 6 SAA 0 0

**Summing Up**

In this post, we learnt how to visualize the workflow registered in event logs with a process map. We also learnt bupaR’s constructs of activity reoccurrences and therefore the types of activity reoccurrences. If you want to learn more about bupaR.