Exercise 12

Real user Session

S1: <P1,P2,P5>
S2: <P1,P2,P7,P3>
S3: <P1,P4,P5,P1,P6,P7>
S4: <P1,P6,P1,P4,P5,P4,P1,P6,P7>
S5: <P5,P3,P1,P2,P7,P3>
S6: <P2,P7>
S7: <P1,P2,P4,P1,P2,P6,P7,P3>

Reconstructed Session	**Sorted Reconstructed Session**
S1: <P1,P2,P5,P2,P7,P3>	<P1, P2, P2, P3, P5, P7>
S2: <P1,P4,P5,P1,P6,P7>	<P1, P1, P4, P5, P6, P7>
S3: <P1,P6,P1,P4>	<P1, P1, P4, P6>
S4: <P5,P4,P1,P6,P7>	<P1, P4, P5, P6, P7>
S5: <P5,P3>	<P3, P5>
S6: <P1,P2,P7,P3>	<P1, P2, P3, P7>
S7: <P2,P7>	<P2, P7>
S8: <P1,P2,P4,P1,P2,P6,P7,P3>	<P1, P1, P2, P2, P3, P4, P6, P7>

A real session is completely reconstructed if all its elements are contained in the same constructed session.

To determine whether a real session is contained in a constructed session keep in mind that ***"being contained in" maintains order and adjacency***. Also all elements of the real session must be present in the constructed session. So for example a real session cannot be contained in a constructed session if the real session is longer than the constructed session.

Real Session S1 is contained in Constructed Session S1
Real Session S2 is contained in Constructed Session S6
Real Session S3 is contained in Constructed Session S2
Real Session S4 is not contained in any Constructed Session
Real Session S5 is not contained in any Constructed Session
Real Session S6 is contained in Constructed Session S7
Real Session S7 is contained in Constructed Session S8

Completely Reconstructed Session = {S1, S2, S3, S6, S7}

M_cr

The number of completely reconstructed sessions divided by the total number of real sessions:
M_cr = 5 / 7 = ***0.71***

M_crs

The number of completely reconstructed sessions whose first element is also the first element of the constructed session divided by the total number of real sessions:

All of the real sessions contained in a constructed session share the first element with the constructed session.

M_crs = 5 / 7 = ***0.71***

M_cre
The number of completely reconstructed sessions whose last element is also the last element of the constructed session divided by the total number of real sessions:

Of these 5 real sessions only S2, S3, S6, and S7 share the last element with the constructed session.

M_cre = 4 / 7 = ***0.57***

M_crse
The number of identically reconstructed sessions divided by the total number of real sessions:
Of these 5 sessions only S2, S3, S6, and S7 are identical with the constructed session.

M_crse = 4 / 7 = ***0.57***

M_o
First compute the degree of overlap between each real session and the 8 constructed sessions. This is computed by dividing the Largest Continuous Intersection (LCI) by the number of elements of the real session. LCI can be determined the same way as "contained in" except that not all elements from the real session must be in the reconstructed session:

Degree of Overlap

$$deg_o(r, c) = \frac{|r \sqcap c|}{|r|}$$

S1: <P1,P2,P5>
deg(r, c) = 3/3, 1/3, 1/3, 1/3, 1/3, 2/3, 1/3, 2/3
max = 1

S2: <P1,P2,P7,P3>
deg(r, c) = 3/4, 1/4, 1/4, 1/4, 1/4, 4/4, 2/4, 2/4
max = 1

S3: <P1,P4,P5,P1,P6,P7>
deg(r, c) = 1/6, 6/6, 2/6, 3/6, 1/6, 1/6, 1/6, 2/6
max = 1

S4: <P1,P6,P1,P4,P5,P4,P1,P6,P7>
deg(r, c) = 1/9, 3/9, 4/9, 5/9, 1/9, 1/9, 1/9, 2/9
max = 5/9

S5: <P5,P3,P1,P2,P7,P3>
deg(r, c) = 3/6, 1/6, 1/6, 1/6, 2/6, 4/6, 2/6, 2/6
max = 4/6

S6: <P2,P7>
deg(r, c) = 2/2, 1/2, 0/2, 1/2, 0/2, 2/2, 2/2, 1/2
max = 1

S7: <P1,P2,P4,P1,P2,P6,P7,P3>
deg(r, c) = 2/8, 2/8, 1/8, 2/8, 1/8, 2/8, 1/8, 8/8
max = 1

M_o = AVG(max) = (1 + 1 + 1 + 5/9 + 4/6 + 1 + 1) / 7 = 6.22 / 7 = ***0.89***

M_s
First compute the degree of similarity between each real session and the 8 constructed sessions. This is computed by dividing the largest continuous intersection by the number of elements of the real session and constructed session.

Degree of Similarity

$$deg_s(r, c) = \frac{|r \sqcap c|}{|r \cup c|}$$

X is a member of the union if it is an element present in set A or in set B, or both.

S1: <P1,P2,P5>
deg(r, c) = 3/6, 1/7, 1/6, 1/6, 1/4, 2/5, 1/4, 2/9
max = 3/6 = ***0.5***

S2: <P1,P2,P7,P3>, Sorted = <P1, P2, P3, P7>
deg(r, c) = 3/6, 1/8, 1/7, 1/7, 1/5, 4/4, 2/4, 2/8
max = 4/4 = ***1.0***

S3: <P1,P4,P5,P1,P6,P7>, Sorted = <P1, P1, P4, P5, P6, P7>
deg(r, c) = 1/9, 6/6, 2/6, 3/6, 1/7, 1/8, 1/7, 2/9
max = 6/6 = ***1.0***

S4: <P1,P6,P1,P4,P5,P4,P1,P6,P7>, Sorted = <P1, P1, P1, P4, P4, P6, P6, P6, P7>
deg(r, c) = 1/13, 3/10, 4/9, 5/10, 1/11, 1/11, 1/10, 2/12
max = 5/10 = ***0.5***

S5: <P5,P3,P1,P2,P7,P3>, Sorted = <P1, P2, P3, P3, P5, P7>
deg(r, c) = 3/7, 1/9, 1/9, 1/8, 2/6, 4/6, 2/6, 2/10
max = 4/6 = ***0.67***

S6: <P2,P7>

deg(r, c) = 2/6, 1/7, 0/6, 1/6, 0/4, 2/4, 2/2, 1/8
max = 2/2 = ***1.0***

S7: <P1,P2,P4,P1,P2,P6,P7,P3>, Sorted = <P1, P1, P2, P2, P3, P4, P6, P7>
deg(r, c) = 2/9, 2/9, 1/8, 2/9, 1/9, 2/8, 1/8, 8/8
max = 8/8 = ***1.0***

M_s = AVG(max) = (0.5 + 1 + 1 + 0.5 + 0.67 + 1 + 1) / 7 = 6.22 / 7 = ***0.81***