CS6220 Data Mining Techniques – Spring 2015
Assignment 5
Yaxin Huang

Question 1

In the GSP algorithm, suppose we have the length-3 frequent pattern set L 3 as follows:
< {2} {3} {4} >
< {2 5} {3} >
< {3} {4} {5} >
< {1} {2} {3} >
< {1} {2 5} >
< {1} {5} {3} >
< {5} {3 4} >
Generate length-4 candidates set C 4 and frequent pattern set L 4 . Show your work by writing down the details of the join and prune steps.

Join Step:
(1) S1 = <{2} {3} {4}>, dropping the first item resulting in <{3} {4}>.

S2	Drop last item	Match <{3} {4}> ?
<{25} {3}>	<{25}>	false
<{3} {4} {5}>	**<{3} {4}>**	**true**
<{1} {2} {3}>	<{1} {2}>	false
<{1} {25}>	<{1} {2}>	false
<{1} {5} {3}>	<{1} {5}>	false
< {5} {3 4} >	<{5} {3}>	false

So we can combine <{2} {3} {4}> with <{3} {4} {5}> and get <{2} {3} {4} {5}> into C4.

(2) S1 = <{25} {3}>, dropping the first item resulting in <{5} {3}>.

S2	Drop last item	Match <{5} {3}> ?
< {2} {3} {4} >	<{2} {3}>	false
<{3} {4} {5}>	<{3} {4}>	false
<{1} {2} {3}>	<{1} {2}>	false
<{1} {25}>	<{1} {2}>	false
<{1} {5} {3}>	<{1} {5}>	false
< {5} {3 4} >	**<{5} {3}>**	**true**

So we can combine <{25} {3}> with <{5} {34}> ant get <{25} {34}> into C4.

(3) S1 = < {3} {4} {5} > , dropping the first item resulting in < {4} {5} > .

S2	Drop last item	Match < {4} {5} > ?
< {2} {3} {4} >	<{2} {3}>	false
<{25} {3}>	<{25}>	false
<{1} {2} {3}>	<{1} {2}>	false
<{1} {25}>	<{1} {2}>	false
<{1} {5} {3}>	<{1} {5}>	false
< {5} {3 4} >	<{5} {3}>	false

It cannot combine with any other subsequences to create a length-4 frequent sequence.

(4) S1 = <{1} {2} {3}>, dropping the first item resulting in <{2} {3}>.

S2	Drop last item	Match <{2} {3}> ?
< {2} {3} {4} >	**<{2} {3}>**	**true**
<{25} {3}>	<{25}>	false
< {3} {4} {5} >	<{3} {4}>	false
<{1} {25}>	<{1} {2}>	false
<{1} {5} {3}>	<{1} {5}>	false
< {5} {3 4} >	<{5} {3}>	false

So we can combine <{1} {2} {3}> with <{2} {3} {4}> and get <{1} {2} {3} {4}> into C4.

(5) S1 = < {1} {2 5} > , dropping the first item resulting in <{25}>.

S2	Drop last item	Match <{25}> ?
< {2} {3} {4} >	<{2} {3}>	false
<{25} {3}>	**<{25}>**	**true**
< {3} {4} {5} >	<{3} {4}>	false
< {1} {2} {3} >	<{1} {2}>	false
<{1} {5} {3}>	<{1} {5}>	false
< {5} {3 4} >	<{5} {3}>	false

So we can combine <{1} {25}> with <{25} {3}> and get <{1} {25} {3}> into C4.

(6) S1 = <{1} {5} {3}>, dropping the first item resulting in <{5} {3}>.

S2	Drop last item	Match <{5} {3}> ?
< {2} {3} {4} >	<{2} {3}>	false
<{25} {3}>	<{25}>	true
< {3} {4} {5} >	<{3} {4}>	false
< {1} {2} {3} >	<{1} {2}>	false
< {1} {2 5} >	<{1} {2}>	false
< {5} {3 4} >	**<{5} {3}>**	**true**

So we can combine <{1} {5} {3}> with <{5} {34}> and get <{1} {5} {34}> into C4.

(7) S1 = < {5} {3 4} > and dropping the first item results in <{34}>.

S2	Drop last item	Match <{34}> ?
< {2} {3} {4} >	<{2} {3}>	false
<{25} {3}>	<{25}>	false
< {3} {4} {5} >	<{3} {4}>	false
< {1} {2} {3} >	<{1} {2}>	false
< {1} {2 5} >	<{1} {2}>	false
<{1} {5} {3}>	<{1} {5}>	false

It cannot combine with any other subsequences to create a length-4 frequent sequence.

Thus, the resulting C4 is:
<{2} {3} {4} {5}>,
<{25} {34}>,
<{1} {2} {3} {4}>,
<{1} {25} {3}>,
<{1} {5} {34}>.

Prune Step:
(1) Length-3 subsequences of <{2} {3} {4} {5}>:

Subsequence	Frequent? (In L3?)
<{3} {4} {5}>	true
<{2} {4} {5}>	false
<{2} {3} {5}>	false
<{2} {3} {4}>	true

Because there are some infrequent subsequences, according to the Apriori property, <{2} {3} {4} {5}> will not be a frequent sequence. It should be pruned.

(2) Length-3 subsequences of <{25} {34}>:

Subsequence	Frequent? (In L3?)
<{5} {34}>	true
<{2} {34}>	false
<{25} {4}>	false
<{25} {3}>	true

Because there are some infrequent subsequences, according to the Apriori property, <{25} {3} {4}> will not be a frequent sequence. It should be pruned.

(3) Length-3 subsequences of <{1} {2} {3} {4}>:

Subsequence	Frequent? (In L3?)
<{2} {3} {4}>	true
<{1} {3} {4}>	false
<{1} {2} {4}>	false
<{1} {2} {3}>	true

Because there are some infrequent subsequences, according to the Apriori property, <{1} {2} {3} {4}> will not be a frequent sequence. It should be pruned.

(4) Length-3 subsequences of <{1} {25} {3}>:

Subsequence	Frequent? (In L3?)
<{25} {3}>	true
<{1} {5} {3}>	true
<{1} {2} {3}>	true
<{1} {25} >	true

Because all the length-3 subsequences of <{1} {25} {3}> are frequent, we keep it.

(5) Length-3 subsequence of <{1} {5} {34}>:

Subsequence	Frequent? (In L3?)
<{5} {34}>	true
<{1} {34}>	false
<{1} {5} {4}>	false
<{1} {5} {3}>	true

Because there are some infrequent subsequences, according to the Apriori property, <{1} {2} {3} {4}> will not be a frequent sequence. It should be pruned.

Frequent Length-4 sequence:
<{1} {25} {3}>.
Actually we need to scan the DB again to decide whether it is a frequent sequence.

Question 2

In the PrefixSpan algorithm, given sequence database:

TID	Items
1	<a(abc)(ac)d(cf)>
2	<(ad)c(bc)(ae)>
3	<(ef)(ab)(df)cb>
4	<eg(af)cbc>

(a) Find <b> projected database :

1	<(_c)(ac)d(cf)>
2	<(_c)(ae)>
3	<(df)cb>
4	<c>

(b) Find all length-2 sequential pattern having prefix <b>:
First we count the occurrence of every item appearing in <b> projected database:
<a>: 2
<b>: 1
<c>: 3
<d>: 2
<e>: 1
<f>: 2
<_c>: 2
Therefore, the length-2 sequetial patterns having prefix <b> are:
<ba>, <bc>, <bd>, <bf>, <(bc)>.

Question 3

(a) The Hidden Markov Model:

Alphabet: $\Psi=\{R,G,B\}$.

Set of states: $Q=\{1,\ 2\}$,

where state 1 means drawing from urn A, and state B means drawing from urn B.

Transition probabilities:

$a_{11}=9/10$

$a_{12}=1/10$

$a_{21}=1/10$

$a_{22}=9/10$

Start probabilities:

$a_{01}=4/5$

$a_{02}=1/5$

Emission probabilities:

$e_1(R)=1/3,\ e_1(G)=1/3, e_1(B)=1/3.$

$e_2(R)=1/4,\ e_2(G)=1/2, e_2(B)=1/4.$

(b) Using the forward algorithm:

$$f_0(0)=1$$

$$f_1(\mathrm{G})=f_0(0)\cdot a_{01}\cdot e_1(G)=1\cdot\frac{4}{5}\cdot\frac{1}{3}=4/15$$

$$f_2(\mathrm{G})=f_0(0)\cdot a_{02}\cdot e_2(G)=1\cdot\frac{1}{5}\cdot\frac{1}{2}=1/10$$

$$f_1(\mathrm{GG})=f_1(G)\cdot a_{11}\cdot e_1(G)+f_2(G)\cdot a_{21}\cdot e_1(G)=4/15\cdot 9/10\cdot 1/3+1/10\cdot 1/10\cdot 1/3=1/12$$

$$f_2(\mathrm{GG})=f_1(G)\cdot a_{12}\cdot e_2(G)+f_2(G)\cdot a_{22}\cdot e_2(G)=7/120$$

$$f_1(\mathrm{GGG})=f_1(\mathrm{GG})\cdot a_{11}\cdot e_1(G)+f_2(\mathrm{GG})\cdot a_{21}\cdot e_1(G)=97/3600\approx 0.0269$$

$$f_2(\mathrm{GGG})=f_1(\mathrm{GG})\cdot a_{12}\cdot e_2(G)+f_2(\mathrm{GG})\cdot a_{22}\cdot e_2(G)=73/2400\approx 0.0304$$

$$Prob(\mathrm{GGG})=f_1(\mathrm{GGG})+f_2(\mathrm{GGG})\approx 0.0573$$

(c) Using the Viterbi algorithm, the most likely path in the model to get “GGG” can be computed in this way:

$$V_0(0)=1$$

$$V_1(G)=e_1(G)\cdot max_k\{V_k(0)\cdot a_{k1}\}=e_1(G)\cdot V_0(0)\cdot a_{01}=\frac{1}{3}\cdot 1\cdot\frac{4}{5}=\frac{4}{15}$$

$$V_2(G)=e_2(G)\cdot max_k\{V_k(0)\cdot a_{k2}\}=e_2(G)\cdot V_0(0)\cdot a_{02}=\frac{1}{2}\cdot 1\cdot\frac{1}{5}=\frac{1}{10}$$

$$V_1(\mathrm{GG})=e_1(G)\cdot max_k\{V_k(G)\cdot a_{k1}\}$$

Because:

$V_1(G)\cdot a_{11}=(4/15)\cdot(9/10)=6/25(larger)$

$V_2(G)\cdot a_{21}=(1/10)\cdot(1/10)=1/100$

Thus, $V_1(\mathrm{GG})=e_1(G)\cdot V_1(G)\cdot a_{11}=(1/3)\cdot(6/25)=2/25$

$$V_2(\mathrm{GG})=e_2(G)\cdot max_k\{V_k(G)\cdot a_{k2}\}$$

Because:

$V_1(G)\cdot a_{12}=(4/15)\cdot(1/10)=2/75$

$V_2(G)\cdot a_{22}=(1/10)\cdot(9/10)=9/100(larger)$

Thus, $V_2(\mathrm{GG})=e_2(G)\cdot V_2(G)\cdot a_{22}=(1/2)\cdot(9/100)=9/200$

$$V_1(\mathrm{GGG})=e_1(G)\cdot max_k\{V_k(\mathrm{GG})\cdot a_{k1}\}$$

Because:

$V_1(\mathrm{GG})\cdot a_{11}=(2/25)\cdot(9/10)=9/125(larger)$

$V_2(\mathrm{GG})\cdot a_{21}=(9/200)\cdot(1/10)=9/2000$

Thus, $V_1(\mathrm{GGG})=e_1(G)\cdot V_1(\mathrm{GG})\cdot a_{11}=(1/3)\cdot(9/125)=3/125$

$$V_2(\mathrm{GGG})=e_2(G)\cdot max_k\{V_k(\mathrm{GG})\cdot a_{k2}\}$$

Because:

$V_1(\mathrm{GG})\cdot a_{12}=(2/25)\cdot(1/10)=2/250$

$V_2(\mathrm{GG})\cdot a_{22}=(9/200)\cdot(9/10)=81/2000(larger)$

Thus, $V_2(\mathrm{GGG})=e_2(G)\cdot V_2(\mathrm{GG})\cdot a_{22}=(1/2)\cdot(81/2000)=81/4000$

Because $V_1(\mathrm{GGG})>V_2(\mathrm{GGG})$,

from *the computation of* $V_1(\mathrm{GGG})$ *we know that the most probable path is*:

urn_1 urn_1 urn_1

Question 4

For the following two time series:
X = [39 44 43 39 46 38 39 43]
Y = [37 44 41 44 39 39 39 40]
(a) Calculate the L1 norm between X and Y
L1 norm:

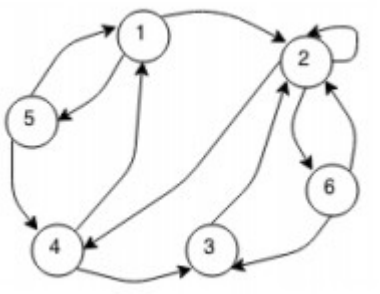
$$\sum_{i=1}^{8} |X_i - Y_i| = 2+0+2+5+7+1+0+3 = 20$$

(b) Calculate the DTW distance between X and Y and point out the optimal warping path. (The local cost function is defined as the absolute difference of the two values, e.g., c(x1, y1) = d(39, 37) = 2)

The cost matrix and the warping path (highlighted):

Start	X 39	44	43	39	46	38	39	43
Y 37	**2**	9	15	17	26	27	29	35
44	7	**2**	**3**	8	10	16	21	22
41	9	5	4	**5**	10	13	15	17
44	14	5	5	9	**7**	13	18	16
39	14	10	9	5	12	**8**	8	12
39	14	15	13	5	12	9	**8**	12
39	14	19	17	5	12	10	**8**	12
40	15	18	20	6	11	12	9	**11**

Question 5



(a) Write down the adjacency matrix A for G
A:

	From 1	2	3	4	5	6
To 1	0	0	0	1	1	0
2	1	1	1	0	0	1
3	0	0	0	1	0	1
4	0	1	0	0	1	0
5	1	0	0	0	0	0
6	0	1	0	0	0	0

(b) Write down the column stochastic matrix M for G
M:

	From 1	2	3	4	5	6
To 1	0	0	0	1/2	1/2	0
2	1/2	1/3	1	0	0	1/2
3	0	0	0	1/2	0	1/2
4	0	1/3	0	0	1/2	0
5	1/2	0	0	0	0	0
6	0	1/3	0	0	0	0

(c) Use the damping factor p=0.15, calculate the PageRank scores of nodes in G
First constructs the new matrix H = (1-p)M + pB:

```
> M
     [,1]      [,2] [,3] [,4] [,5] [,6]
[1,]  0.0 0.0000000    0  0.5  0.5  0.0
[2,]  0.5 0.3333333    1  0.0  0.0  0.5
[3,]  0.0 0.0000000    0  0.5  0.0  0.5
[4,]  0.0 0.3333333    0  0.0  0.5  0.0
[5,]  0.5 0.0000000    0  0.0  0.0  0.0
[6,]  0.0 0.3333333    0  0.0  0.0  0.0

> B = matrix(rep(1/6,36),6,6)
> B
           [,1]      [,2]      [,3]      [,4]      [,5]
[1,] 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667
[2,] 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667
[3,] 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667
[4,] 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667
[5,] 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667
[6,] 0.1666667 0.1666667 0.1666667 0.1666667 0.1666667
           [,6]
[1,] 0.1666667
[2,] 0.1666667
[3,] 0.1666667
[4,] 0.1666667
[5,] 0.1666667
[6,] 0.1666667

> p = 0.15
> p
[1] 0.15

> H = (1-p)*M + p*B
> H
      [,1]      [,2]  [,3]  [,4]  [,5]  [,6]
[1,] 0.025 0.0250000 0.025 0.450 0.450 0.025
[2,] 0.450 0.3083333 0.875 0.025 0.025 0.450
[3,] 0.025 0.0250000 0.025 0.450 0.025 0.450
[4,] 0.025 0.3083333 0.025 0.025 0.450 0.025
[5,] 0.450 0.0250000 0.025 0.025 0.025 0.025
[6,] 0.025 0.3083333 0.025 0.025 0.025 0.025

> V1 = eigen(H)$vectors[, 1]
> V1
[1] 0.2730902+0i 0.7755506+0i 0.3171525+0i 0.3458453+0i 0.1699434+0i 0.2736194+0i
> H %*% V1  # Let's check if this is the correct eigenvector we are looking for:
             [,1]
[1,] 0.2730902+0i
[2,] 0.7755506+0i
[3,] 0.3171525+0i
[4,] 0.3458453+0i
[5,] 0.1699434+0i
[6,] 0.2736194+0i

> From the results above we could see that the V1 is the PageRank vector which converges to the
equilibrium state. So the scores of each page is the elements in the V1 vector.
```