10 Data Mining

Solutions

SID	Sequence
1	<(dc)b(ac)>
2	<bc(bac)>
3	<(ab)a>

- Exercise 1:GSP
  - Initial step
    - All singleton sequences are <a>, <b>, <c>, <d>
  - General step, k = 1
    - <d> can't form patterns so it can be left out

Cand	Support
<a>	3
<b>	3
<c>	2
<d>	1

10 Data Mining

Solutions

– General step, k = 1, generate length 2 candidates

- First generate 2 event candidates

	<a>	<b>	<c>
<a>	<aa>	<ab>	<ac>
<b>	<ba>	<bb>	<bc>
<c>	<ca>	<cb>	<cc>

- Then generate 1 sequence candidates, each event with 2 items

	<a>	<b>	<c>
<a>		<(ab)>	<(ac)>
<b>			<(bc)>
<c>			

10 Data Mining

Solutions

– k = 2, we have 12 2-length candidates

- After the second table scan we remain with 7 2-patterns: <ba>, <bc>, <ca>, <cb>, <cc>, <(ab)>, <(ac)>

SID	Sequence
1	<(dc)b(ac)>
2	<bc(bac)>
3	<(ab)a>

Candidate	Support	SIDs
<aa>	1	3
<ab>	0	-
<ac>	0	-
<ba>	3	1, 2, 3
<bb>	1	2
<bc>	2	1, 2
<ca>	2	1, 2
<cb>	2	1, 2
<cc>	2	1, 2
<(ab)>	2	2, 3
<(ac)>	2	1, 2
<(bc)>	1	2

10 Data Mining

Solutions

- Generalization:
  - Join
    - Joining k-1 elements together to obtain k-length candidates
    - Idea by join is that two sequences, s1 and s2 can be joined if after dropping the first item from s1 and the last item from s2, we obtain the same sequence
    - E.g.:
      - » <bc> and <ca> can be joined since by dropping b from <bc> and a from <ca> we obtain <c>. The joined result is <bca>
      - » <ba> and <(ab)> can also be joined and we obtaine <b(ab)>
  - Prune
    - Is similar to the apriori algorithm
    - <bca> passes pruning only if <bc>, <ba> and <ca> $\in F2$
    - <b(ab)> passes pruning only if <ba>, <bb> and <(ab)> $\in F2$

10 Data Mining

Solutions

– k = 2, generate length 3 candidates
  - <ba>, <bc>, <ca>, <cb>, <cc>, <(ab)>, <(ac)>

	<ba>	<bc>	<ca>	<cb>	<cc>	<(ab)>	<(ac)>
<ba>	-	-	-	-	-	<b(ab)>	<b(ac)>
<bc>	-	-	<bca>	<bcb>	<bcc>		
<ca>	-	-	-	-	-	<c(ab)>	<c(ac)>
<cb>	<cba>	<cbc>	-	-	-	-	-
<cc>	-	-	<cca>	<ccb>	-	-	-
<(ab)>	<(ab)a>	<(ab)c>	-	-	-	-	-
<(ac)>	-	-	<(ac)a>	<(ac)b>	<(ac)c>	-	-

  - Now perform pruning
    – <bc>, <ba> and <ca> $\in F_2$ so <bca> is a good candidate
    – <bcb> is not, because <bb> $\notin F_2$
    – …
  - After pruning
    – C_3=<b(ac)>, <bca>, <bcc>, <c(ab)>, <c(ac)>, <cba>, <cbc>, <cca>, <ccb>

10 Data Mining

Solutions

– k = 3, we have 9 3-length candidates

- C_3=<b(ac)>, <bca>, <bcc>, <c(ab)>, <c(ac)>, <cba>, <cbc>, <cca>, <ccb>
- After table scan
  F_3 = <b(ac)>, <c(ac)>

Candidate	Support	SIDs
<b(ac)>	2	1, 2
<bca>	1	2
<bcc>	0	-
<c(ab)>	1	2
<c(ac)>	2	1, 2
<cba>	1	1
<cbc>	1	1
<cca>	0	-
<ccb>	0	-

SID	Sequence
1	<(dc)b(ac)>
2	<bc(bac)>
3	<(ab)a>

10 Data Mining

Solutions

– Build C_4 from F_3= <b(ac)>, <c(ac)>

	<b(ac)>	<c(ac)>
<b(ac)>	-	-
<c(ac)>	-	-

- We can't build any 4 length candidate so we remain with <b(ac)>, <c(ac)> as 3-patterns

10 Data Mining

Solutions

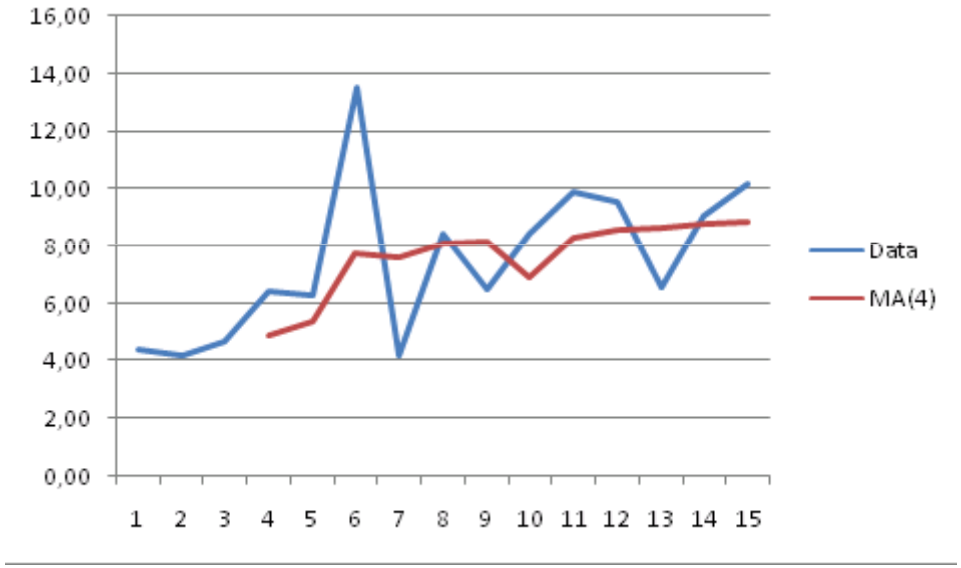
- Exercise 2.1: time-series
  - A sequences of values or events **changing with time**
  - Data is recorded at **regular intervals**

10 Data Mining

Solutions

- Exercise 2.2: MA(4)

Data	MA(4)
4,38	
4,19	
4,65	
6,40	4,905
6,26	5,375
13,51	7,705
4,19	7,59
8,41	8,0925
6,50	8,1525
8,43	6,8825
9,87	8,3025
9,56	8,59
6,57	8,6075
9,03	8,7575
10,18	8,835



10 Data Mining

Solutions

- Exercise 2.3: whole matching method
  - Index building
    - Obtain the DFT coefficients of each sequence in the database
    - Build a 2k-dimensional index using the first k Fourier coefficients (2k-dimensions are needed because Fourier coefficients are complex numbers)
  - Query processing
    - Obtain the DFT coefficients of the query sequence
    - Use the 2k-dimensional index to filter out such sequences that are at most ε distance away from the query sequence
    - Discards false alarms by computing the actual distance between two sequences