UCLA Samueli Computer Science

CS145 Discussion: Week 9

Frequent Pattern Mining, Sequential Pattern Mining & DTW

Junheng Hao

Friday, 12/04/2020

Roadmap

- Part 1: Frequent Pattern Mining and Association Rules
  - Apriori
  - FP-Growth
  - Association Rules & Pattern Evaluation
- Part 2: Sequential Pattern Mining
  - GSP
  - PrefixSpan
- Part 3: Time Series
  - DTW

Announcements

Homework #5	Due on 11:59 PM, Dec 4 (Friday, Week 9)
Homework #6	Will be released on Dec 7 (Monday Week 10), due on 11:59 PM, Dec 14 (Monday, Final week)
Project: Kaggle Deadline	Due on 11:59 PM, Dec 6 (Sunday, Week 9)
Project: Final Report	Due on 11:59 PM, Dec 18 (Friday, Final week)
Final Exam (100 minutes via CCLE)	Morning Session: Around 8:00-9:40 AM, Dec 16 (Wednesday, Final Week) Evening Session: Around 6:00-7:40 PM, Dec 16 (Wednesday, Final Week)

Roadmap

- Frequent Pattern Mining and Association Rules
  - Apriori
  - FP-Growth
  - Association Rules & Pattern Evaluation
- Sequential Pattern Mining
  - GSP
  - PrefixSpan
- Time Series
  - DTW

Basic Concepts

- Given a transactional database, two itemsets X and Y, and an association rule X -> Y
  - Absolute/relative support of X: absolute/relative frequency of X
  - A frequent itemset X (pattern X): support of X is no less than a minsup threshold
  - Support of X -> Y: probability that a transection contains X U Y
  - Confidence of X -> Y: conditional probability that a transection with X also contains Y

Closed Patterns and Max-Patterns

- An itemset X is closed if X is *frequent* and there exists *no super-pattern* Y ɔ X, *with the same support* as X (proposed by Pasquier, et al. @ ICDT'99)
- An itemset X is a max-pattern if X is frequent and there exists no frequent super-pattern Y ɔ X (proposed by Bayardo @ SIGMOD'98)
- Closed pattern is a lossless compression of freq. patterns
  - Reducing the # of patterns and rules

Apriori

- Apriori pruning principle: If there is any itemset which is infrequent, its superset should not be generated/tested! (Agrawal & Srikant @VLDB'94, Mannila, et al. @ KDD' 94)
- Method:
  - Initially, scan DB once to get frequent 1-itemset
  - Generate length k candidate itemsets from length k-1 frequent itemsets
  - Test the candidates against DB
  - Terminate when no frequent or candidate set can be generated

https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html/2
https://edumine.wordpress.com/2013/09/11/apriori-algorithm-simplified-with-an-example/

Apriori: Example 1

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=2

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=2

(1)

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

(2)

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

www.T4Tutorials.com

(3)

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=2

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

3

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3

4

Items Bought
Milk, Eggs
Milk, Tea
Milk, Cold Drink
Eggs, Tea
Eggs, Cold Drink
Tea, Cold Drink

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=2

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

3

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3

4

Items Bought
Milk, Eggs
Milk, Tea
Milk, Cold Drink
Eggs, Tea
Eggs, Cold Drink
Tea, Cold Drink

5

Items Bought	Support
Milk, Eggs	1
Milk, Tea	2
Milk, Cold Drink	1
Eggs, Tea	2
Eggs, Cold Drink	3
Tea, Cold Drink	2

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=2

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

www.T4Tutorials.com

3

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3

4

Items Bought
Milk, Eggs
Milk, Tea
Milk, Cold Drink
Eggs, Tea
Eggs, Cold Drink
Tea, Cold Drink

5

Items Bought	Support
Milk, Eggs	1
Milk, Tea	2
Milk, Cold Drink	1
Eggs, Tea	2
Eggs, Cold Drink	3
Tea, Cold Drink	2

www.T4Tutorials.com

6

Items Bought	Support
Milk, Tea	2
Eggs, Tea	2
Eggs, Cold Drink	3
Tea, Cold Drink	2

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/ min_support=2

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

3

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3

4

Items Bought
Milk, Eggs
Milk, Tea
Milk, Cold Drink
Eggs, Tea
Eggs, Cold Drink
Tea, Cold Drink

5

Items Bought	Support
Milk, Eggs	1
Milk, Tea	2
Milk, Cold Drink	1
Eggs, Tea	2
Eggs, Cold Drink	3
Tea, Cold Drink	2

6

Items Bought	Support
Milk, Tea	2
Eggs, Tea	2
Eggs, Cold Drink	3
Tea, Cold Drink	2

7

Items Bought	Support
Eggs, Tea, Cold Drink	2

There is only one itemset with minimum support 2.
So only one itemset is frequent

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=2

Apriori: Example 2

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=3

(1)

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

→

(2)

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

www.T4Tutorials.com

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=3

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

3

Items Bought	Support
Eggs	3
Tea	3
Cold Drink	3

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=3

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

3

Items Bought	Support
Eggs	3
Tea	3
Cold Drink	3

4

Items Bought
Eggs, Tea
Eggs, Cold Drink
Tea, Cold Drink

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=3

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

3

Items Bought	Support
Eggs	3
Tea	3
Cold Drink	3

4

Items Bought
Eggs, Tea
Eggs, Cold Drink
Tea, Cold Drink

5

Items Bought	Support
Eggs, Tea	2
Eggs, Cold Drink	3
Tea, Cold Drink	2

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=3

1

Tid	Items Bought
1	Milk, Tea, Cake
2	Eggs, Tea, Cold Drink
3	Milk, Eggs, Tea, Cold Drink
4	Eggs, Cold Drink
5	Juice

2

Items Bought	Support
Milk	2
Eggs	3
Tea	3
Cold Drink	3
Juice	1
Cake	1

3

Items Bought	Support
Eggs	3
Tea	3
Cold Drink	3

4

Items Bought
Eggs, Tea
Eggs, Cold Drink
Tea, Cold Drink

5

Items Bought	Support
Eggs, Tea	2
Eggs, Cold Drink	3
Tea, Cold Drink	2

6

Items Bought	Support
Eggs, Cold Drink	3

https://t4tutorials.com/apriori-algorithm-in-data-mining-with-examples/

min_support=3

FP-Growth

- Grow long patterns from short ones using local frequent items only
  - "abc" is a frequent pattern
  - Get all transactions having "abc", i.e., project DB on abc: DB|abc
  - "d" is a local frequent item in DB|abc → abcd is a frequent pattern

FP-Growth

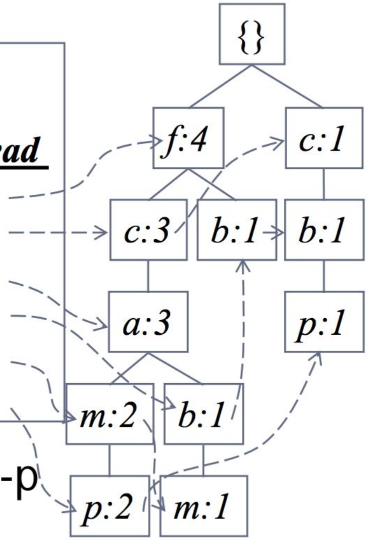
TID	*Items bought*	*(ordered) frequent items*
100	{*f, a, c, d, g, i, m, p*}	{*f, c, a, m, p*}
200	{*a, b, c, f, l, m, o*}	{*f, c, a, b, m*}
300	{*b, f, h, j, o, w*}	{*f, b*}
400	{*b, c, k, s, p*}	{*c, b, p*}
500	{*a, f, c, e, l, p, m, n*}	{*f, c, a, m, p*}

min_support = 3

1. Scan DB once, find frequent 1-itemset (single item pattern)
2. Sort frequent items in frequency descending order, f-list
3. Scan DB again, construct FP-tree

Header Table

Item	*frequency*	*head*
f	*4*	
c	*4*	
a	*3*	
b	*3*	
m	*3*	
p	*3*	



F-list = f-c-a-b-m-p

FP-Growth: Example

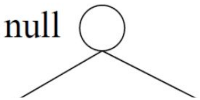
FP-Growth

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

Transaction Database

min support = 2

F-list = a-b-c-d-e



Header table

Item	Pointer
a	
b	
c	
d	
e	

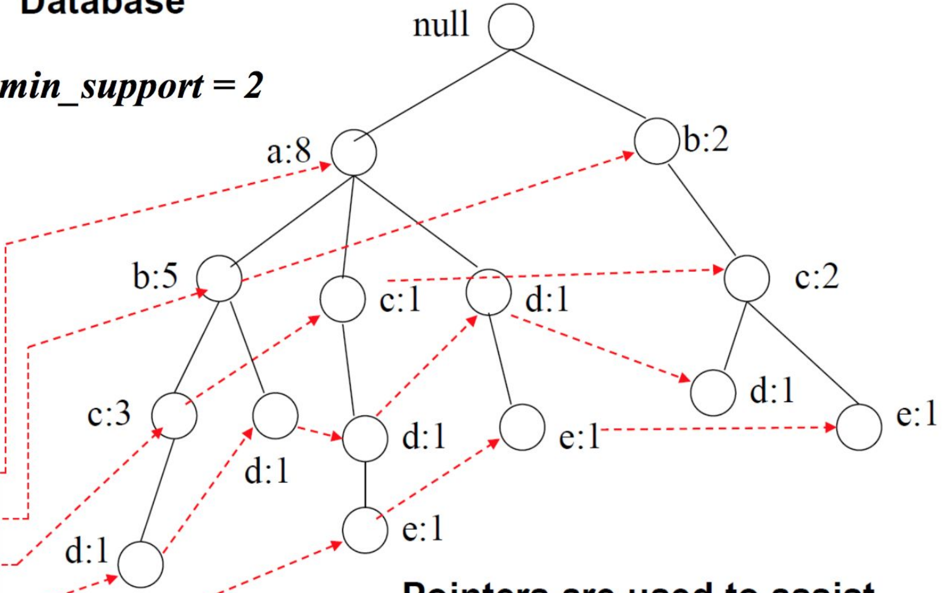
FP-Growth

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

Transaction Database

min_support = 2

F-list = a-b-c-d-e



Header table

Item	Pointer
a	
b	
c	
d	
e	

Pointers are used to assist frequent itemset generation

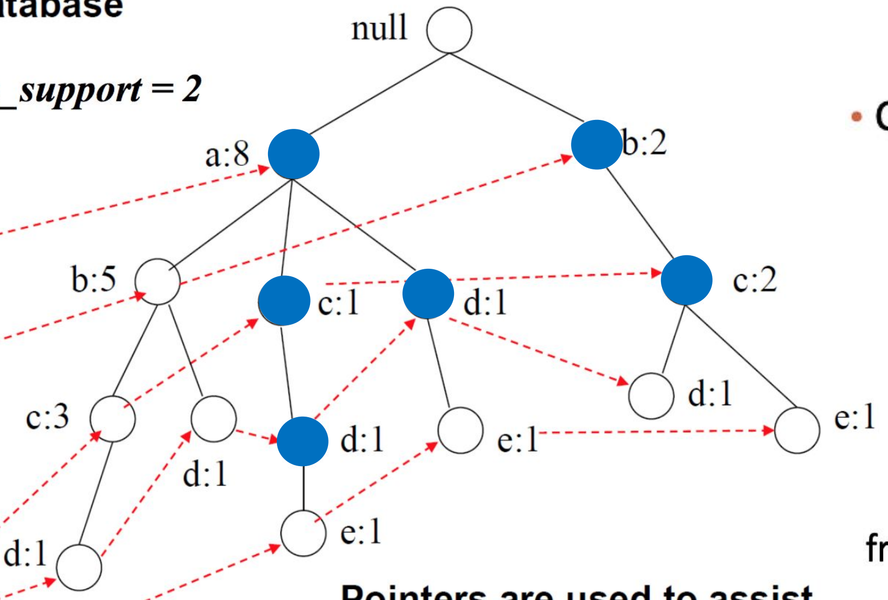
FP-Growth

TID	Items
1	{a,b}
2	{b,c,d}
3	{a,c,d,e}
4	{a,d,e}
5	{a,b,c}
6	{a,b,c,d}
7	{a}
8	{a,b,c}
9	{a,b,d}
10	{b,c,e}

Transaction Database

min_support = 2

F-list = a-b-c-d-e



Header table

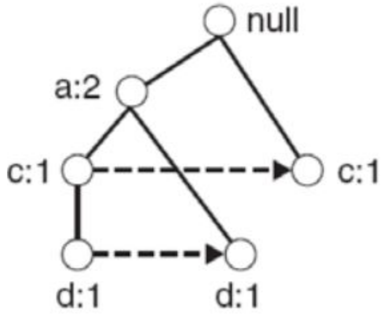
Item	Pointer
a	
b	
c	
d	
e	

Pointers are used to assist frequent itemset generation

- Conditional pattern base for e:

{acd:1; ad:1; bc:1}

- Conditional FP-tree for e:



frequent patterns with e are: {ade:2, de:2, ce:2, ae:2, e:3}

Association Rules & Pattern Evaluation

- (Relative) support

Support {🍎} = $\frac{4}{8}$

- Confidence

Confidence {🍎 → 🍺} = $\frac{\text{Support }\{🍎, 🍺\}}{\text{Support }\{🍎\}}$

- **Strong association rules**
  - Satisfying minimum support and minimum confidence
  - Recall: $Confidence(A \Rightarrow B) = P(B|A) = \frac{support(A \cup B)}{support(A)}$

Transaction 1	🍎 🍺 🍚 🍗
Transaction 2	🍎 🍺 🍚
Transaction 3	🍎 🍺
Transaction 4	🍎 🍐
Transaction 5	🍼 🍺 🍚 🍗
Transaction 6	🍼 🍺 🍚
Transaction 7	🍼 🍺
Transaction 8	🍼 🍐

https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html

Association Rules & Pattern Evaluation

- Not all strong association rules are interesting

	Basketball	Not basketball	Sum (row)
Cereal	2000	1750	3750
Not cereal	1000	250	1250
Sum(col.)	3000	2000	5000

- Shall we target people who play basketball for cereal ads? *play basketball* $\Rightarrow$ *eat cereal* [40%, 66.7%]
- Hint: What is the overall probability of people who eat cereal?
  - 3750/5000 = 75% > 66.7%!
- Confidence measure of a rule could be misleading

UCLA
Engineer Change.

Association Rules & Pattern Evaluation

- Lift

Lift {🍎 → 🍺} = Support {🍎, 🍺} / (Support {🍎} x Support {🍺})

$$lift = \frac{P(A \cup B)}{P(A)P(B)}$$

1: independent
>1: positively correlated
<1: negatively correlated

Transaction 1	🍎 🍺 🍚 🍗
Transaction 2	🍎 🍺 🍚
Transaction 3	🍎 🍺
Transaction 4	🍎 🍐
Transaction 5	🍼 🍺 🍚 🍗
Transaction 6	🍼 🍺 🍚
Transaction 7	🍼 🍺
Transaction 8	🍼 🍐

https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html

Association Rules & Pattern Evaluation

Support {🍎} = $\frac{4}{8}$

Confidence {🍎 → 🍺} = $\frac{\text{Support \{🍎, 🍺\}}}{\text{Support \{🍎\}}}$

Lift {🍎 → 🍺} = $\frac{\text{Support \{🍎, 🍺\}}}{\text{Support \{🍎\} x Support \{🍺\}}}$

Rule	Support	Confidence	Lift
$A \Rightarrow D$			
$C \Rightarrow A$			
$A \Rightarrow C$			
$B\&C \Rightarrow D$			

https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html

UCLA
Engineer Change.

Association Rules & Pattern Evaluation

Support {🍎} = 4 / 8

Confidence {🍎 → 🍺} = Support {🍎,🍺} / Support {🍎}

Lift {🍎 → 🍺} = Support {🍎,🍺} / (Support {🍎} x Support {🍺})

Rule	Support	Confidence	Lift
$A \Rightarrow D$	2/5	2/3	10/9
$C \Rightarrow A$	2/5	2/4	5/6
$A \Rightarrow C$	2/5	2/3	5/6
$B\&C \Rightarrow D$	1/5	1/3	5/9

https://www.kdnuggets.com/2016/04/association-rules-apriori-algorithm-tutorial.html

Chi-Square Test

- Question: Are education level and marital status related?

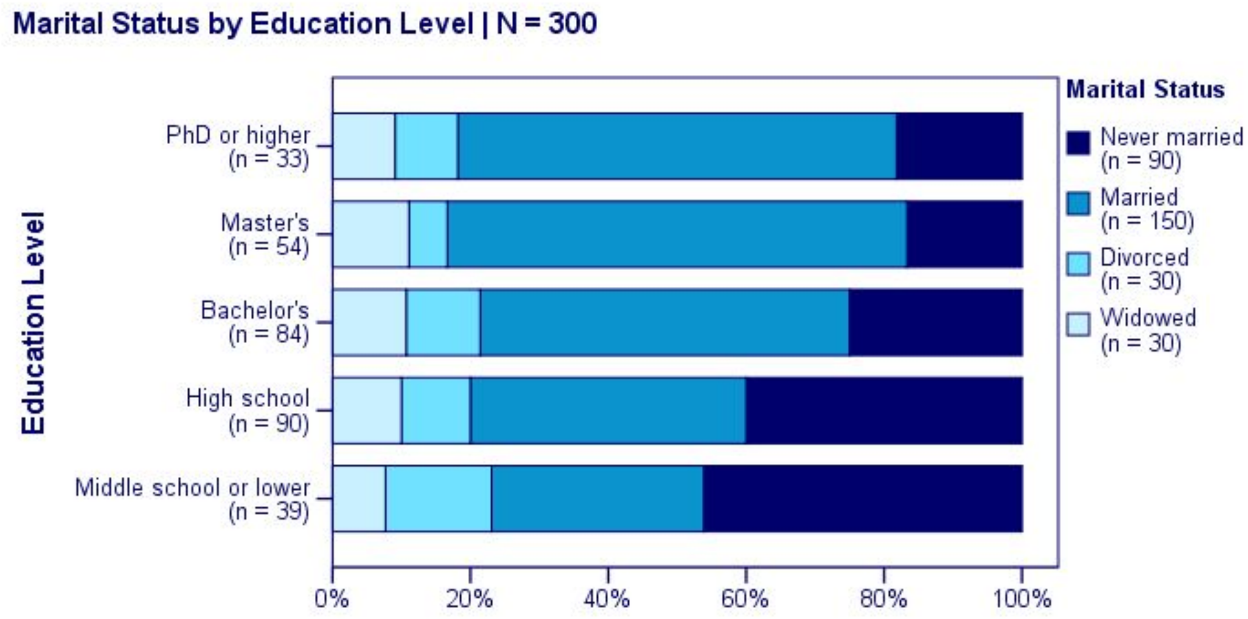
Marital Status by Education | n = 300

	Middle school or lower	High school	Bachelor's	Master's	PhD or higher	Total
Never married	18	36	21	9	6	90
Married	12	36	45	36	21	150
Divorced	6	9	9	3	3	30
Widowed	3	9	9	6	3	30
Total	39	90	84	54	33	300

https://www.spss-tutorials.com/chi-square-independence-test/

Chi-Square Test

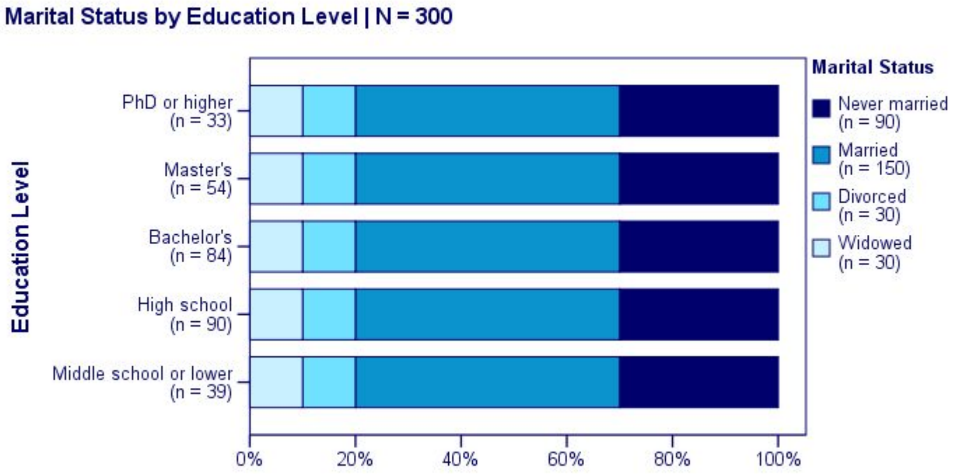
- Marital status is related to education level.



https://www.spss-tutorials.com/chi-square-independence-test/

Chi-Square Test

- The ***null hypothesis*** for a chi-square independence test is that
  - two categorical variables are ***independent*** in some population.



- ***Statistical independence*** means that
  - the frequency distribution of a variable is the same for all levels of some other variable.

Chi-Square Test

- ***Expected frequencies*** are
  - the frequencies we expect in our sample if the ***null hypothesis*** holds.

Expected Frequencies for Perfectly Independent Variables

	Middle school or lower	High school	Bachelor's	Master's	PhD or higher	Total
Never married	11.7	27.0	25.2	16.2	9.9	90.0
Married	19.5	45.0	42.0	27.0	16.5	150.0
Divorced	3.9	9.0	8.4	5.4	3.3	30.0
Widowed	3.9	9.0	8.4	5.4	3.3	30.0
Total	39.0	90.0	84.0	54.0	33.0	300.0

https://www.spss-tutorials.com/chi-square-independence-test/

Chi-Square Test

- ***Null hypothesis (independent)→expected frequencies***

P(middle,never)=P(middle)P(never)=(39/300)*(90/300)
Expected # of (middle,never) = 300*P(middle, never)=39*90/300=11.7

Expected Frequencies for Perfectly Independent Variables

	Middle school or lower	High school	Bachelor's	Master's	PhD or higher	Total
Never married	11.7	27.0	25.2	16.2	9.9	90.0
Married	19.5	45.0	42.0	27.0	16.5	150.0
Divorced	3.9	9.0	8.4	5.4	3.3	30.0
Widowed	3.9	9.0	8.4	5.4	3.3	30.0
Total	39.0	90.0	84.0	54.0	33.0	300.0

https://www.spss-tutorials.com/chi-square-independence-test/

Chi-Square Test

- Real data → ***observed frequencies***:

Marital Status by Education | n = 300

	Middle school or lower	High school	Bachelor's	Master's	PhD or higher	Total
Never married	18	36	21	9	6	90
Married	12	36	45	36	21	150
Divorced	6	9	9	3	3	30
Widowed	3	9	9	6	3	30
Total	39	90	84	54	33	300

https://www.spss-tutorials.com/chi-square-independence-test/

Chi-Square Test

- Add up the differences for each of the 5*4=20 cells
  - → χ2

- $X^2 = \sum_{i=1}^{r} \sum_{j=1}^{c} \frac{(o_{ij}-e_{ij})^2}{e_{ij}}$

$$\chi^2 = \frac{(18-11.7)^2}{11.7} + \frac{(36-27)^2}{27} + \ldots + \frac{(6-5.4)^2}{5.4} = 23.57$$

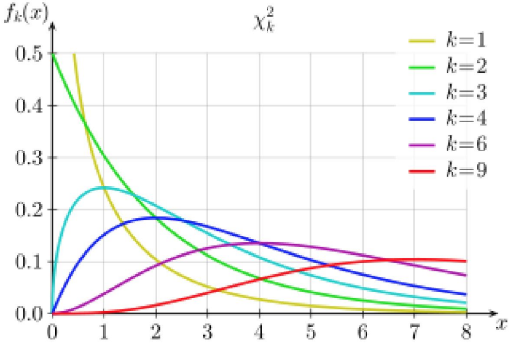
https://www.spss-tutorials.com/chi-square-independence-test/

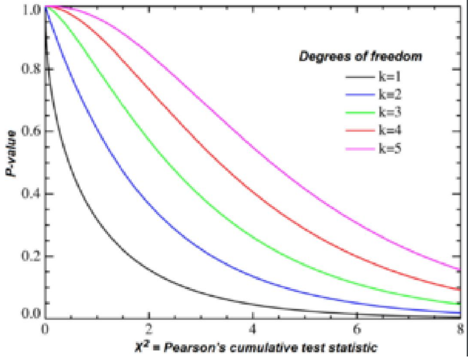
Chi-Square Test

- Is $\chi2$=23.57 a large value?
  - If yes, reject the null hypothesis → A and B are dependent
  - But how to tell if it is a large value?
    - $\chi2$

• Follows Chi-squared distribution with degree of freedom as $(r-1) \times (c-1)$

Pearson established it in 1900. See more.

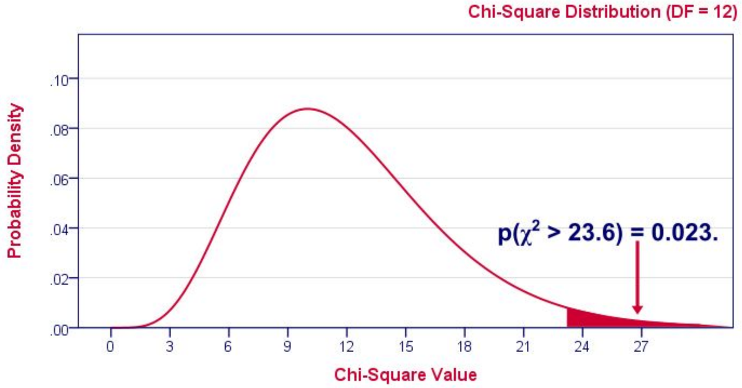




https://www.spss-tutorials.com/chi-square-independence-test/

Chi-Square Test

- In this example, $df = (5-1)\cdot(4-1) = 12.$
  - How to interpret $P(\chi^2>23.57)=0.023$?
    - The probability of ______ under ______ hypothesis is very small, 2.3%.
  - A small p-value basically means that the data are unlikely under the null hypothesis. The convention is to reject the null hypothesis if $p < 0.05$.
  - Should we reject the null hypothesis in this case? Yes!
    - ***"An association between education and marital status was observed, $\chi^2(12)$ = 23.57, p = 0.023."***



https://www.spss-tutorials.com/chi-square-independence-test/
https://www.spss-tutorials.com/statistical-significance/

Roadmap

- Frequent Pattern Mining and Association Rules
  - Apriori
  - FP-Growth
  - Association Rules & Pattern Evaluation
- Sequential Pattern Mining
  - GSP
  - PrefixSpan
- Time Series
  - DTW

Basic Concepts

- Given a set of sequences, find the complete set of *frequent* subsequences

A ***sequence*** : < (ef) (ab) (df) c b >

An element may contain a set of items. Items within an element are unordered and we list them alphabetically.

A ***sequence database***

SID	sequence
10	<a(abc)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(ab)(df)cb>
40	<eg(af)cbc>

<a(bc)dc> is a ***subsequence*** of <a(abc)(ac)d(cf)>

Given ***support threshold*** *min_sup* =2, <(ab)c> is a ***sequential pattern***

UCLA Engineer Change.

GSP—Generalized Sequential Pattern Mining

The Apriori Property of Sequential Patterns

- A basic property: Apriori (Agrawal & Sirkant'94)
  - If a sequence S is not frequent
  - Then none of the super-sequences of S is frequent
  - E.g, <hb> is infrequent → so do <hab> and <(ah)b>

Seq. ID	Sequence
10	<(bd)cb(ac)>
20	<(bf)(ce)b(fg)>
30	<(ah)(bf)abf>
40	<(be)(ce)d>
50	<a(bd)bcb(ade)>

Given *support threshold*
min_sup =2

UCLA
Engineer Change.

GSP

- s_1 *and* s_2 can join, if dropping first item in s_1 is the same as dropping the last item in s_2
- Examples:
  - <(12)3> join <(2)34> = <(12)34>
  - <(12)3> join <(2)(34)> = <(12)(34)>

GSP: Example

GSP

min_sup = 2

- Initial candidates: all singleton sequences
  - <a>, <b>, <c>, <d>, <e>, <f>, <g>, <h>
- Scan database once, count support for candidates

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Cand	Sup
<a>	
<b>	
<c>	
<d>	
<e>	
<f>	
<g>	
<h>	

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

GSP

min_sup = 2

- Initial candidates: all singleton sequences
  - <a>, <b>, <c>, <d>, <e>, <f>, <g>, <h>
- Scan database once, count support for candidates

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Cand	Sup
<a>	4
<b>	4
<c>	1
<d>	2
<e>	1
<f>	4
<g>	1
<h>	1

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

UCLA Engineer Change.

GSP

min_sup = 2

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Cand	Sup
<a>	4
<b>	4
<d>	2
<f>	4

Length 2 Candidates generated by join

Length 2 Frequent Sequences

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

UCLA
Engineer Change.

GSP

min_sup = 2

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Cand	Sup
<a>	4
<b>	4
<d>	2
<f>	4

Length 2 Candidates generated by join

<aa> <ab> <ad> <af> <ba> <bb> <bd> <bf>
<da> <db><dd> <df> <fa> <fb> <fd> <ff>
<(ab)> <(ad)> <(af)> <(bd)> <(bf)> <(df)>

Length 2 Frequent Sequences

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

GSP

min_sup = 2

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Cand	Sup
<a>	4
<b>	4
<d>	2
<f>	4

Length 2 Candidates generated by join

<aa> <ab> <ad> <af> <ba> <bb> <bd> <bf>
<da> <db><dd> <df> <fa> <fb> <fd> <ff>
<(ab)> <(ad)> <(af)> <(bd)> <(bf)> <(df)>

Length 2 Frequent Sequences

<ba> <da> <db> <df> <fa>
<(ab)> <(af)> <(bf)>

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

GSP

min_sup = 2

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Length 2 Frequent Sequences

<ba> <da> <db> <df> <fa>
<(ab)> <(af)> <(bf)>

Length 3 Candidates generated by join

Length 3 Frequent Sequences

Length 4 Candidates generated by join

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

GSP

min_sup = 2

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Length 2 Frequent Sequences

<ba> <da> <db> <df> <fa>
<(ab)> <(af)> <(bf)>

Length 3 Candidates generated by join

<ba> and <(ab)> - <b(ab)> {1}
<ba> and <(af)> - <b(af)> {1}
<da> and <(ab)> - <d(ab)> {1}
<da> and <(af)> - <d(af)> {1}
<db> and <(bf)> - <d(bf)> {1, 4}
<db> and <ba> - <dba> {1, 4}
<df> and <fa> - <dfa> {1, 4}
<fa> and <(ab)> - <f(ab)> -
<fa> and <(af)> - <f(af)> {1}
<(ab)> and <(bf)> - <(abf)> {1,2,3}
<(ab)> and <ba> - <(ab)a> {1}
<(af)> and <fa> - <(af)a) {1}
<(bf)> and <fa> - <(bf)a> {1, 4}

Length 3 Frequent Sequences

Length 4 Candidates generated by join

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

GSP

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

min_sup = 2

Length 2 Frequent Sequences

<ba> <da> <db> <df> <fa>
<(ab)> <(af)> <(bf)>

Length 3 Candidates generated by join

<ba> and <(ab)> - <b(ab)> {1}
<ba> and <(af)> - <b(af)> {1}
<da> and <(ab)> - <d(ab)> {1}
<da> and <(af)> - <d(af)> {1}
<db> and <(bf)> - <d(bf)> {1, 4}
<db> and <ba> - <dba> {1, 4}
<df> and <fa> - <dfa> {1, 4}
<fa> and <(ab)> - <f(ab)> -
<fa> and <(af)> - <f(af)> {1}
<(ab)> and <(bf)> - <(abf)> {1,2,3}
<(ab)> and <ba> - <(ab)a> {1}
<(af)> and <fa> - <(af)a) {1}
<(bf)> and <fa> - <(bf)a> {1, 4}

Length 3 Frequent Sequences

<dba> <dfa> <(abf)> <(bf)a> <d(bf)>

Length 4 Candidates generated by join

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

UCLA
Engineer Change.

GSP

min_sup = 2

Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

Length 2 Frequent Sequences

<ba> <da> <db> <df> <fa>
<(ab)> <(af)> <(bf)>

Length 3 Candidates generated by join

<ba> and <(ab)> - <b(ab)> {1}
<ba> and <(af)> - <b(af)> {1}
<da> and <(ab)> - <d(ab)> {1}
<da> and <(af)> - <d(af)> {1}
<db> and <(bf)> - <d(bf)> {1, 4}
<db> and <ba> - <dba> {1, 4}
<df> and <fa> - <dfa> {1, 4}
<fa> and <(ab)> - <f(ab)> -
<fa> and <(af)> - <f(af)> {1}
<(ab)> and <(bf)> - <(abf)> {1,2,3}
<(ab)> and <ba> - <(ab)a> {1}
<(af)> and <fa> - <(af)a) {1}
<(bf)> and <fa> - <(bf)a> {1, 4}

Length 3 Frequent Sequences

<dba> <dfa> <(abf)> <(bf)a> <d(bf)>

Length 4 Candidates generated by join

<d(bf)> and <(bf)a> - <d(bf)a> {1, 4}
<(abf)> and <(bf)a> - <(abf)a> {1}

https://www.slideshare.net/Krish_ver2/53-mining-sequential-patterns

PrefixSpan

Assume a pre-specified order on items, e.g., alphabetical order

- <a>, <aa>, <a(ab)> and <a(abc)> are *prefixes* of sequence <a(abc)(ac)d(cf)>
  - Note <a(ac)> is not a prefix of <a(abc)(ac)d(cf)>
- Given sequence <a(abc)(ac)d(cf)>

Prefix	*Suffix*
<a>	<(abc)(ac)d(cf)>
<aa>	<(_bc)(ac)d(cf)>
<a(ab)>	<(_c)(ac)d(cf)>

  - (_bc) means: the last element in the prefix together with (bc) form one element

PrefixSpan

- Given a sequence, α, let α' be subsequence of α
  - α' is called a projection of α w.r.t. prefix β, if and only if
    - α' has prefix β, and
    - α' is the maximum subsequence of α with prefix β
  - Example:
    - <ad(cf)> is a projection of <a(abc)(ac)d(cf)> w.r.t. prefix <ad>

SID	sequence
10	<a(abc)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(ab)(df)cb>
40	<eg(af)cbc>

PrefixSpan: Example

PrefixSpan

- 1. Find length-1 sequential patterns:

min_sup = 2

id	Sequence
10	<a(abc)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(ab)(df)cb>
40	<eg(af)cbc>

<a>	<b>	<c>	<d>	<e>	<f>	<g>

http://home.etf.rs/~vm/os/dmsw/PrefixSpan%20algorithm.pptx

PrefixSpan

min_sup = 2

- 1. Find length-1 sequential patterns:

id	Sequence
10	<a(abc)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(ab)(df)cb>
40	<eg(af)cbc>

<a>	<b>	<c>	<d>	<e>	<f>	~~<g>~~
4	4	4	3	3	3	~~1~~

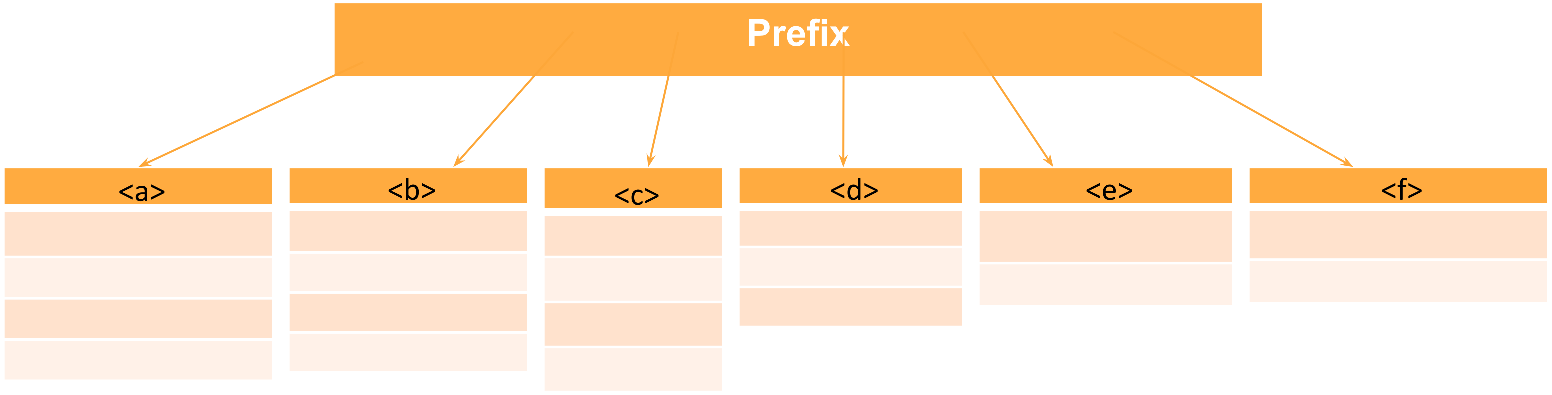
<a><b><c><d><e><f>

PrefixSpan

- 2. Divide search space

id	Sequence
10	<a(abc)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(ab)(df)cb>
40	<eg(af)cbc>

min_sup = 2



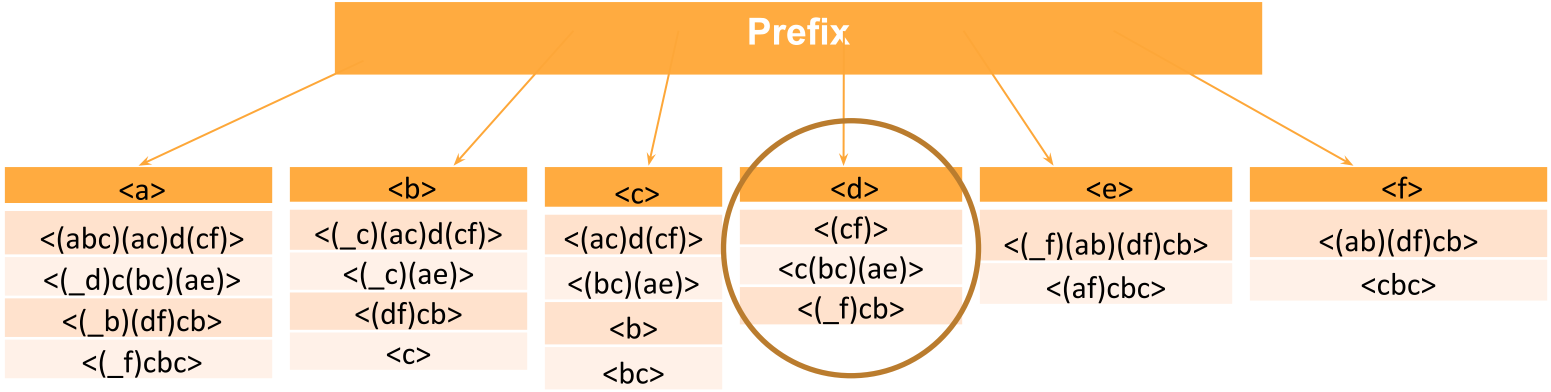
PrefixSpan

UCLA Engineer Change.

- 2. Divide search space

id	Sequence
10	<a(abc)(ac)d(cf)>
20	<(ad)c(bc)(ae)>
30	<(ef)(ab)(df)cb>
40	<eg(af)cbc>

min_sup = 2



PrefixSpan

- 3. Find subsets of sequential patterns

min_sup = 2

<d>
<(cf)>
<c(bc)(ae)>
<(_f)cb>

<a>	<b>	<c>	<d>	<e>	<f>	<_f>

PrefixSpan

- 3. Find subsets of sequential patterns

min_sup = 2

<d>
<(cf)>
<c(bc)(ae)>
<(_f)cb>

<db>
<(_c)(ae)>

<dc>
<(bc)(ae)>
<b>

~~<a>~~	<b>	<c>	~~<d>~~	~~<e>~~	~~<f>~~	~~<_f>~~
~~1~~	2	3	~~0~~	~~1~~	~~1~~	~~1~~

<db> <dc>

PrefixSpan

min_sup = 2

- 3. Find subsets of sequential patterns

<d>
<(cf)>
<c(bc)(ae)>
<(_f)cb>

~~<db>~~
~~<(_c)(ae)>~~

<dc>
<(bc)(ae)>
<b>

~~<a>~~	<b>	<c>	~~<d>~~	~~<e>~~	~~<f>~~	~~<_f>~~
1	2	3	0	1	1	1

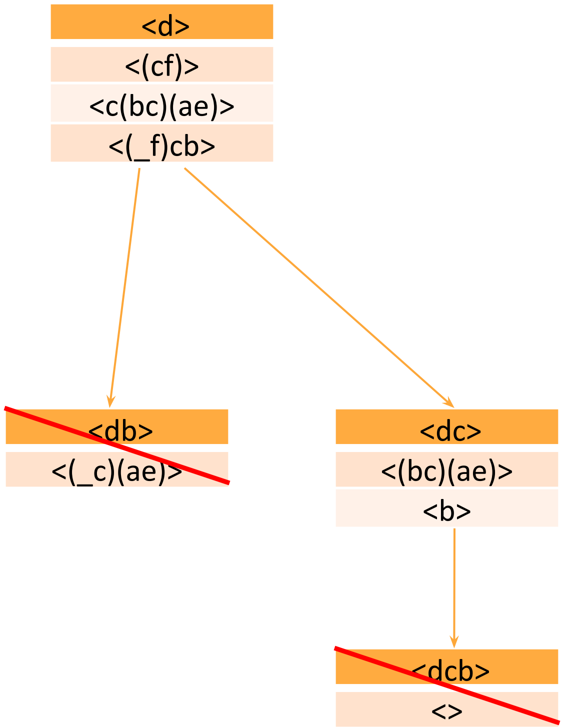
<db> <dc>

<b>	<a>	<e>	<c>

PrefixSpan

min_sup = 2

- 3. Find subsets of sequential patterns



<a>	<b>	<c>	<d>	<e>	<f>	<_f>
1	2	3	0	1	1	1

<db> <dc>

<b>	<a>	<e>	<c>
2	1	1	1

<dcb>

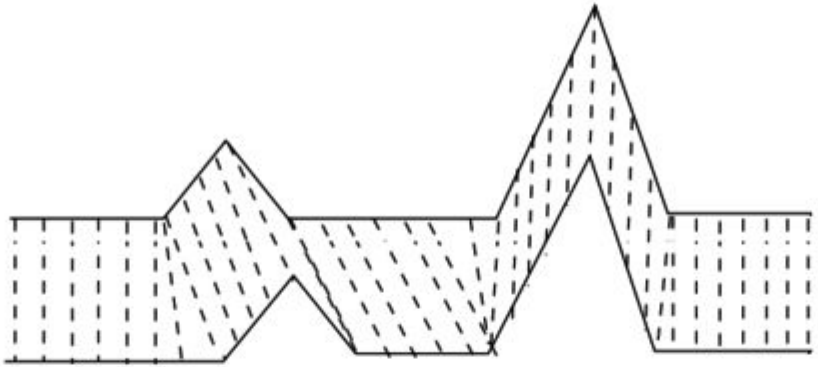
Roadmap

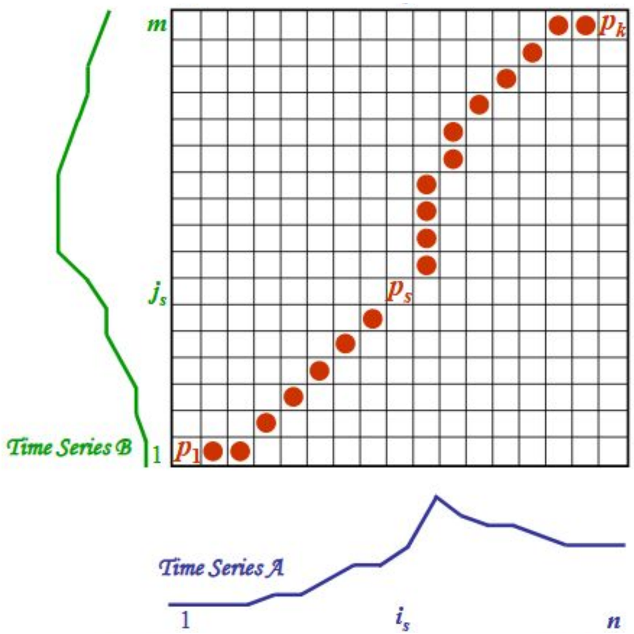
- Frequent Pattern Mining and Association Rules
  - Apriori
  - FP-Growth
  - Association Rules & Pattern Evaluation
- Sequential Pattern Mining
  - GSP
  - PrefixSpan
- Time Series
  - DTW

Goal of DTW

- Given
  - Two sequences (with possible different lengths):
    - $X = \{x_1, x_2, \dots, x_N\}$
    - $Y = \{y_1, y_2, \dots, y_M\}$
  - A local distance (cost) measure between x_n and y_m: $c(x_n, y_m)$
- Goal:
  - Find an alignment between X and Y, such that, the overall cost is minimized

DTW



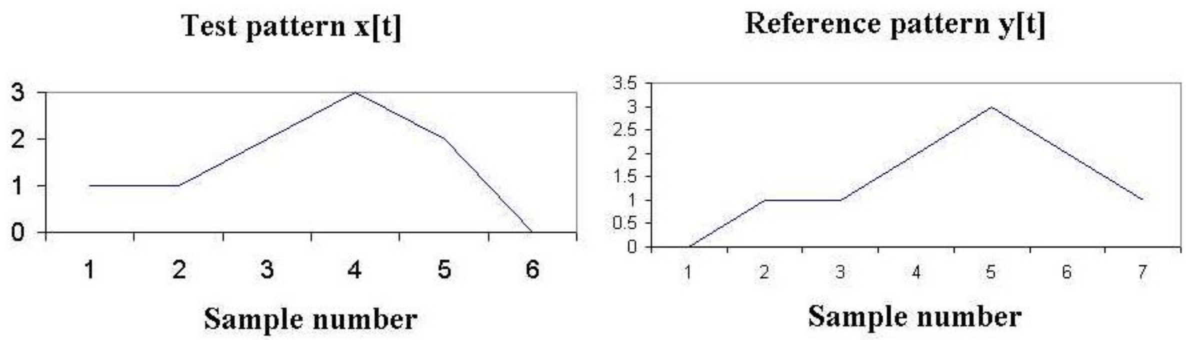


https://en.wikipedia.org/wiki/Dynamic_time_warping
https://medium.com/datadriveninvestor/dynamic-time-warping-dtw-d51d1a1e4afc

DTW: Example

DTW

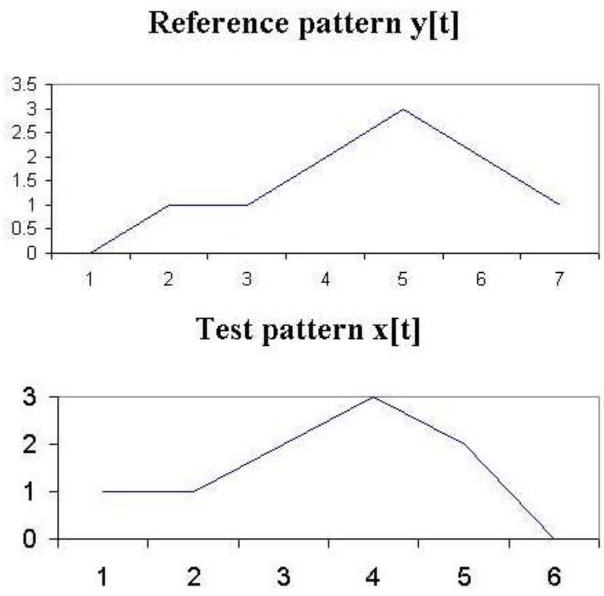
- How similar are these two peaked functions



http://www.phon.ox.ac.uk/jcoleman/old_SLP/Lecture_5/DTW_explanation.html

DTW

Distance function:
C(x, y) = (x - y) ^ 2



7						
6						
5						
4						
3						
2						
1						
	1	2	3	4	5	6

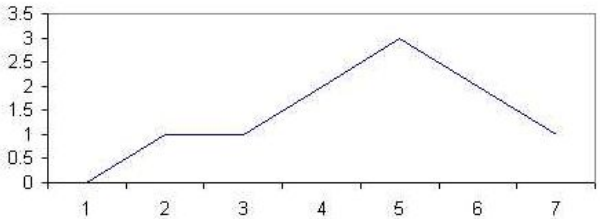
http://www.phon.ox.ac.uk/jcoleman/old_SLP/Lecture_5/DTW_explanation.html

DTW

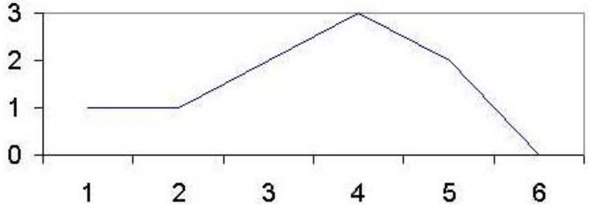
Distance function:

$C(x, y) = (x - y) ^ 2$

Reference pattern y[t]



Test pattern x[t]



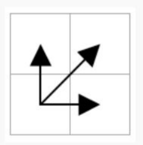
7	**0**	**0**	**1**	**4**	**1**	**1**
6	**1**	**1**	**0**	**1**	**0**	**4**
5	**4**	**4**	**1**	**0**	**1**	**9**
4	**1**	**1**	**0**	**1**	**0**	**4**
3	**0**	**0**	**1**	**4**	**1**	**1**
2	**0**	**0**	**1**	**4**	**1**	**1**
1	**1**	**1**	**4**	**9**	**4**	**0**
	1	2	3	4	5	6

http://www.phon.ox.ac.uk/jcoleman/old_SLP/Lecture_5/DTW_explanation.html

UCLA Engineer Change.

DTW

Update rule:



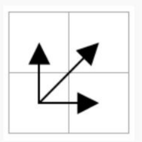
```
DTW[i, j] := cost + minimum(DTW[i-1, j  ],
                            DTW[i  , j-1],
                            DTW[i-1, j-1])
```

7	0	0	1	4	1	1
6	1	1	0	1	0	4
5	4	4	1	0	1	9
4	1	1	0	1	0	4
3	0	0	1	4	1	1
2	0	0	1	4	1	1
1	1	1	4	9	4	0
	1	2	3	4	5	6

http://www.phon.ox.ac.uk/jcoleman/old_SLP/Lecture_5/DTW_explanation.html

DTW

Update rule:



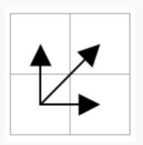
```
DTW[i, j] := cost + minimum(DTW[i-1, j  ],
                            DTW[i  , j-1],
                            DTW[i-1, j-1])
```

7	0 ->7	0 ->7	1 ->3	4 ->6	1 ->2	1 ->2
6	1 ->7	1 ->7	0 ->2	1 ->2	0 ->1	4 ->5
5	4 ->6	4 ->6	1 ->2	0 ->1	1 ->2	9 ->11
4	1 ->2	1 ->2	0 ->1	1 ->2	0 ->2	4 ->6
3	0 ->1	0 ->1	1 ->2	4 ->6	1 ->7	1 ->8
2	0 ->1	0 ->1	1 ->2	4 ->6	1 ->7	1 ->8
1	1 ->1	1 ->2	4 ->6	9 ->15	4 ->19	0 ->19
	1	2	3	4	5	6

http://www.phon.ox.ac.uk/jcoleman/old_SLP/Lecture_5/DTW_explanation.html

DTW

Update rule:



```
DTW[i, j] := cost + minimum(DTW[i-1, j  ],
                            DTW[i  , j-1],
                            DTW[i-1, j-1])
```

7	0 ->7	0 ->7	1 ->3	4 ->6	1 ->2	1 ->2
6	1 ->7	1 ->7	0 ->2	1 ->2	0 ->1	4 ->5
5	4 ->6	4 ->6	1 ->2	0 ->1	1 ->2	9 ->11
4	1 ->2	1 ->2	0 ->1	1 ->2	0 ->2	4 ->6
3	0 ->1	0 ->1	1 ->2	4 ->6	1 ->7	1 ->8
2	0 ->1	0 ->1	1 ->2	4 ->6	1 ->7	1 ->8
1	1 ->1	1 ->2	4 ->6	9 ->15	4 ->19	0 ->19
	1	2	3	4	5	6

http://www.phon.ox.ac.uk/jcoleman/old_SLP/Lecture_5/DTW_explanation.html

UCLA Samueli Computer Science

Thank you!

Q & A