UNIVERSITÉ DE NANTES

POLYTECH NANTES

Discovery of Frequent Sequences in usage logs

Presented by: Aujasvi
Supervised by: Prof. Fabrice Gulliet

Master in Data Science (2017-2018)

Contents

1.	Project Description
2.	Problem Statement
3.	Sequential Pattern Mining
4.	GSP-Algorithm
5.	Results
6.	GSP-Algorithm Limitations
7.	Conclusion
8.	References

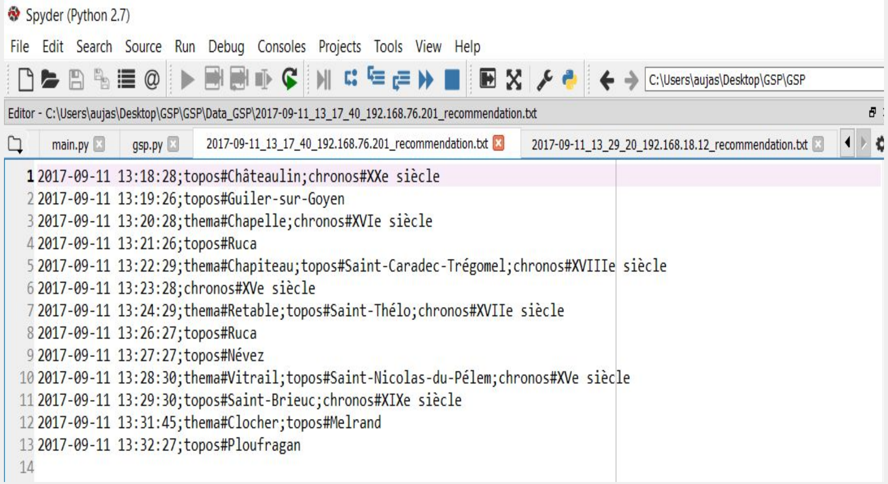
Project Description

- The work consists of in **discovering frequent** (sequential patterns) into the data of user trace
- The data is collected from user traces: the actions of several users visiting a virtual and 3D museum
- Objects in the museum are grouped in categories:
    - Churchs
    - Farms
    - Castles
    - Houses
    - Paintings
- Objects are described by their location, data and usage

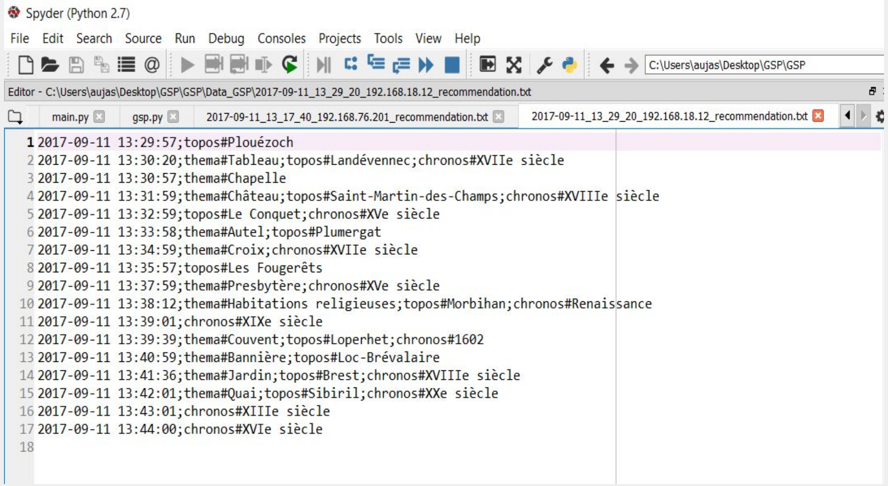
Contd.

- Data Files: “**all-backup trace**” & “**all-backup recommendation**”
- 45 Participants traces

Example trace: 2017-09-11 13: 18: 28; topos # Châteaulin; lap # twentieth century , date_time_recommendation; dimension concept #; # dimension concept; dimensional concept #


Spyder (Python 2.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\aujas\Desktop\GSP\GSP
Editor - C:\Users\aujas\Desktop\GSP\GSP\Data_GSP\2017-09-11_13_17_40_192.168.76.201_recommendation.txt
main.py
gsp.py
2017-09-11_13_17_40_192.168.76.201_recommendation.txt
2017-09-11_13_29_20_192.168.18.12_recommendation.txt

```
2017-09-11 13:18:28;topos#Châteaulin;chronos#XXe siècle
2017-09-11 13:19:26;topos#Guiler-sur-Goyen
2017-09-11 13:20:28;thema#Chapelle;chronos#XVIe siècle
2017-09-11 13:21:26;topos#Ruca
2017-09-11 13:22:29;thema#Chapiteau;topos#Saint-Caradec-Trégomel;chronos#XVIIIe siècle
2017-09-11 13:23:28;chronos#XVe siècle
2017-09-11 13:24:29;thema#Retable;topos#Saint-Thélo;chronos#XVIIe siècle
2017-09-11 13:26:27;topos#Ruca
2017-09-11 13:27:27;topos#Névez
2017-09-11 13:28:30;thema#Vitrail;topos#Saint-Nicolas-du-Pélem;chronos#XVe siècle
2017-09-11 13:29:30;topos#Saint-Brieuc;chronos#XIXe siècle
2017-09-11 13:31:45;thema#Clocher;topos#Melrand
2017-09-11 13:32:27;topos#Ploufragan
```


Spyder (Python 2.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\aujas\Desktop\GSP\GSP
Editor - C:\Users\aujas\Desktop\GSP\GSP\Data_GSP\2017-09-11_13_29_20_192.168.18.12_recommendation.txt
main.py
gsp.py
2017-09-11_13_17_40_192.168.76.201_recommendation.txt
2017-09-11_13_29_20_192.168.18.12_recommendation.txt

```
2017-09-11 13:29:57;topos#Plouézoch
2017-09-11 13:30:20;thema#Tableau;topos#Landévennec;chronos#XVIIe siècle
2017-09-11 13:30:57;thema#Chapelle
2017-09-11 13:31:59;thema#Château;topos#Saint-Martin-des-Champs;chronos#XVIIIe siècle
2017-09-11 13:32:59;topos#Le Conquet;chronos#XVe siècle
2017-09-11 13:33:58;thema#Autel;topos#Plumergat
2017-09-11 13:34:59;thema#Croix;chronos#XVIIe siècle
2017-09-11 13:35:57;topos#Les Fougerêts
2017-09-11 13:37:59;thema#Presbytère;chronos#XVe siècle
2017-09-11 13:38:12;thema#Habitations religieuses;topos#Morbihan;chronos#Renaissance
2017-09-11 13:39:01;chronos#XIXe siècle
2017-09-11 13:39:39;thema#Couvent;topos#Loperhet;chronos#1602
2017-09-11 13:40:59;thema#Bannière;topos#Loc-Brévalaire
2017-09-11 13:41:36;thema#Jardin;topos#Brest;chronos#XVIIIe siècle
2017-09-11 13:42:01;thema#Quai;topos#Sibiril;chronos#XXe siècle
2017-09-11 13:43:01;chronos#XIIIe siècle
2017-09-11 13:44:00;chronos#XVIe siècle
```

Problem Statement

The objective is to find the subsequences of object visited, such as a church is often visited by a Castle using **sequence discovery algorithm GSP** (Generalised Sequential Pattern Mining) using Python

Sequential Pattern Mining

- **Sequence Databases-** consists of sequences of ordered elements or events (with or without time)
- **Sequential Pattern Mining-** mining of frequently occurring ordered events or subsequences as patterns
  - Example : First buying computer, then laptop and then, digital camera within first 3 months
- Usually, **categorical** and **symbolic** data
- Numeric data analysis and Time series analysis

Sequential Pattern Mining

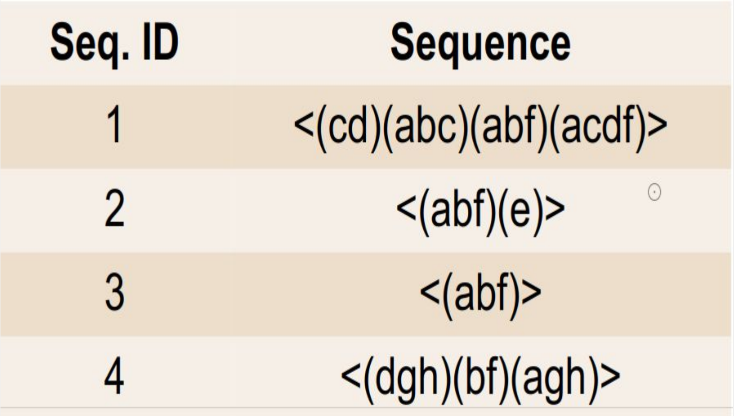
- $I = \{I1, I2, ...Ip\}$ – Set of items
- Sequence $s = <e1\ e2\ e3...\ el>$
- Ordered list of events : Each event is an element of the sequence
- Item can occur at most once in an event, but several times in a sequence
- Sequence with length l : **l-sequence**
- A sequence $\alpha = <a1a2...an>$ is a **subsequence** of $\beta = <b1b2..bm>$ denoted as $\alpha \subseteq \beta$ if there exists integers j1, j2, ... jn between 1 and m such that $a1 \subseteq bj1$, $a2 \subseteq bj2$,... $an \subseteq bjn$
- **Example,** $\alpha = <(ab),d>$ and $\beta = <(abc),(de)>$, **α is a sub-sequence of β**
- **Sequence Database:** A sequence database, S, is a set of tuples, <SID,s>, where SID is a sequence ID and s is a sequence
- A tuple <SID, s> is said to contain a sequence a, if a is a subsequence of s
- The support of a sequence α in a sequence database S is the number of tuples in the database containing α
- Given the minimum support threshold, a sequence a is frequent in sequence database S if support S(a) >= min sup

GSP-Algorithm

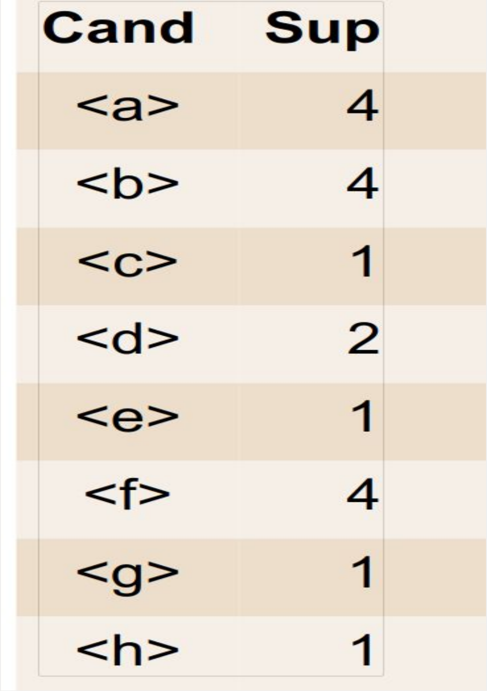
- Candidate generate and Test approach on horizontal data format
- Multi-pass, Candidate generate and test approach proposed by Agrawal and Srikant
- **Outline of the method:**
    - Initially, every item in DB is a candidate of length-1
    - for each level (i.e., sequences of length-k) do
    - Generate candidate length-(k+1) sequences from length-k frequent sequences using Apriori
    - repeat until no frequent sequence or no candidate can be found
- **Major strength:** Candidate pruning by Apriori
- **Weakness:** Generates large number of candidates

GSP-Algorithm Example

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

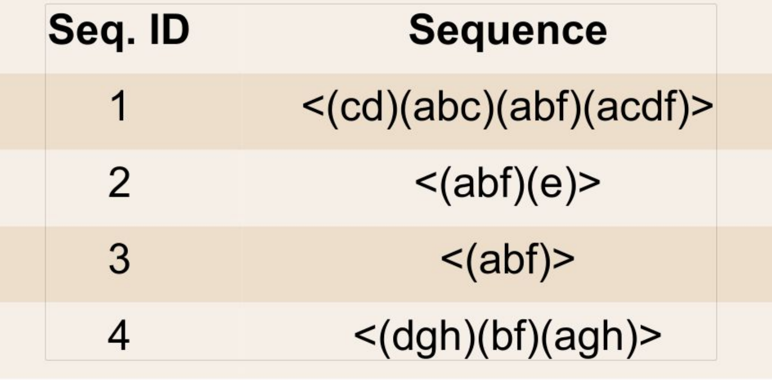


Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>

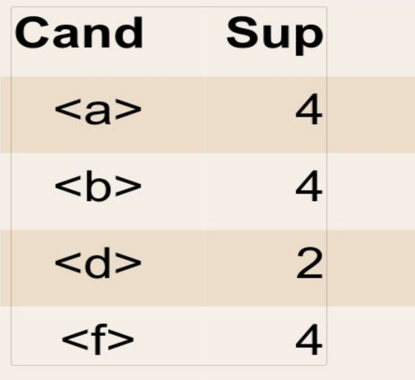


Cand	Sup
<a>	4
<b>	4
<c>	1
<d>	2
<e>	1
<f>	4
<g>	1
<h>	1

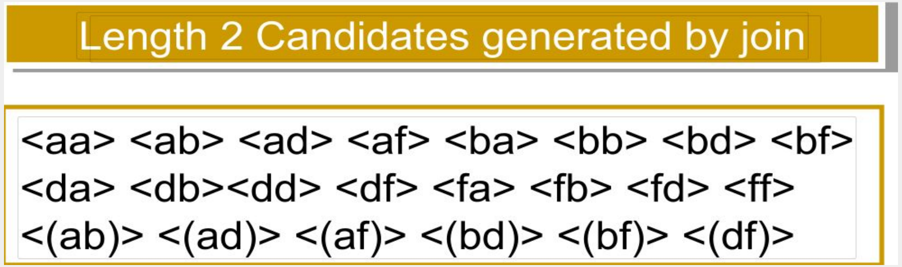
Contd.



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

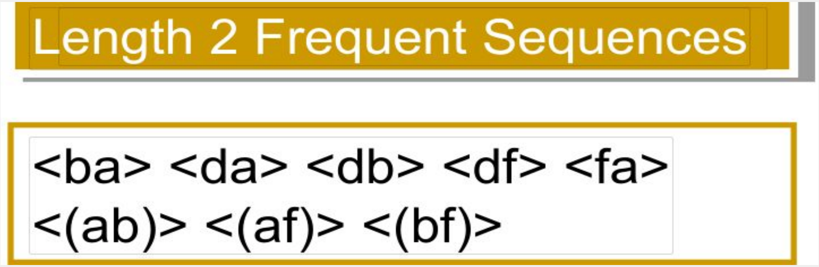


Cand	Sup
<a>	4
<b>	4
<d>	2
<f>	4



Length 2 Candidates generated by join

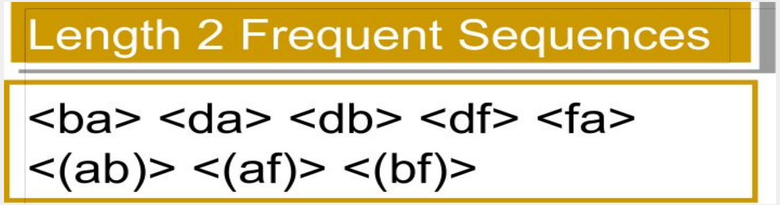
<aa> <ab> <ad> <af> <ba> <bb> <bd> <bf>
<da> <db><dd> <df> <fa> <fb> <fd> <ff>
<(ab)> <(ad)> <(af)> <(bd)> <(bf)> <(df)>



Length 2 Frequent Sequences

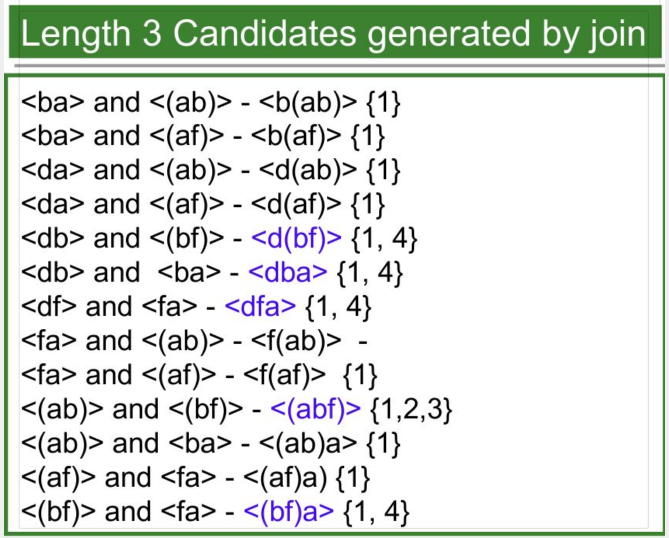
<ba> <da> <db> <df> <fa>
<(ab)> <(af)> <(bf)>

Contd.



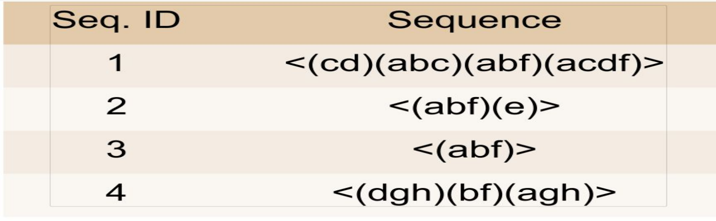
Length 2 Frequent Sequences

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

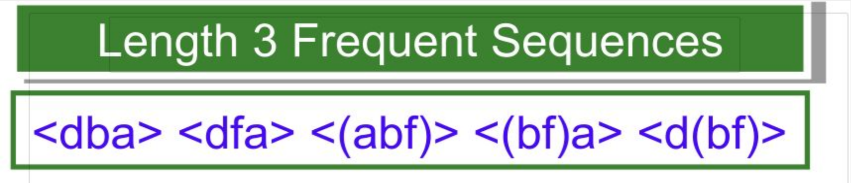


Length 3 Candidates generated by join

<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}

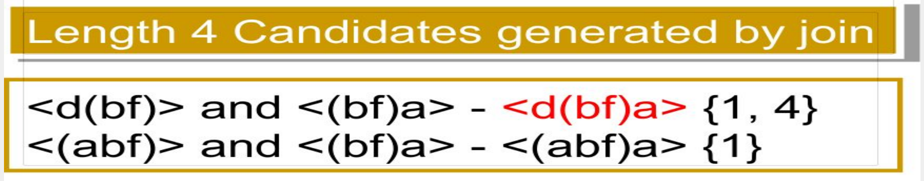


Seq. ID	Sequence
1	<(cd)(abc)(abf)(acdf)>
2	<(abf)(e)>
3	<(abf)>
4	<(dgh)(bf)(agh)>



Length 3 Frequent Sequences

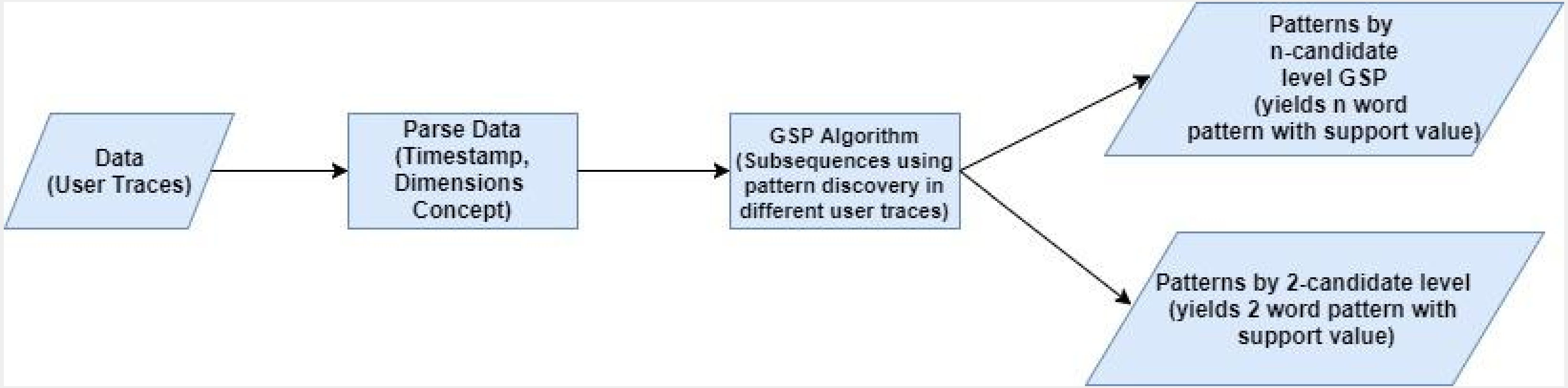
<dba> <dfa> <(abf)> <(bf)a> <d(bf)>



Length 4 Candidates generated by join

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

GSP-Algorithm Workflow



Data
(User Traces)

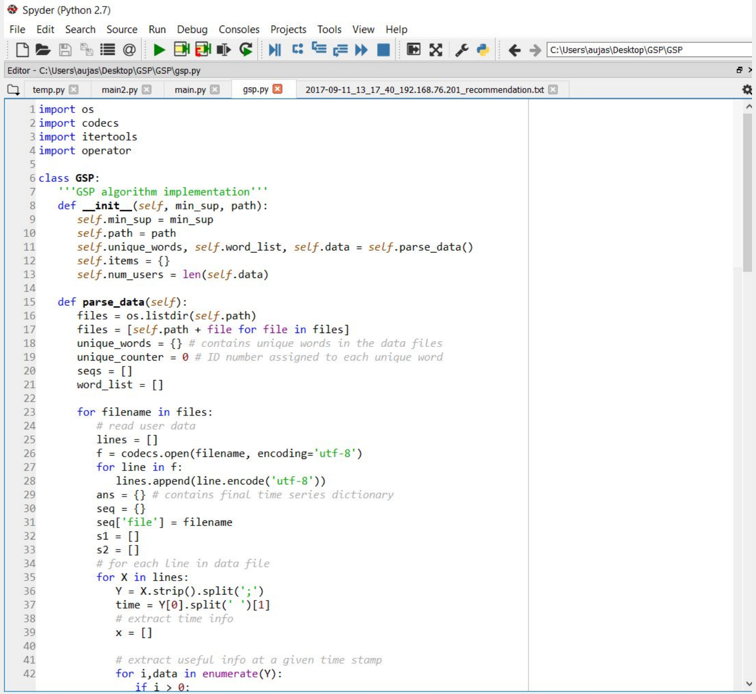
Parse Data
(Timestamp,
Dimensions
Concept)

GSP Algorithm
(Subsequences using
pattern discovery in
different user traces)

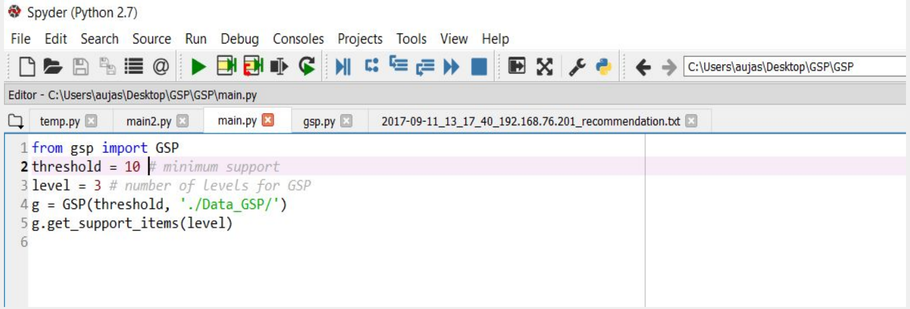
Patterns by
n-candidate
level GSP
(yields n word
pattern with support value)

Patterns by 2-candidate level
(yields 2 word pattern with
support value)

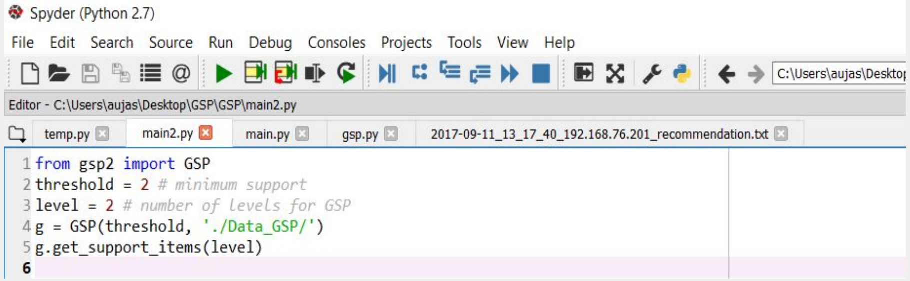
GSP-Python Implementation


Spyder (Python 2.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\aujas\Desktop\GSP\GSP
Editor - C:\Users\aujas\Desktop\GSP\GSP\gsp.py
temp.py
main2.py
main.py
gsp.py
2017-09-11_13_17_40_192.168.76.201_recommendation.txt

```python
import os
import codecs
import itertools
import operator

class GSP:
   '''GSP algorithm implementation'''
   def __init__(self, min_sup, path):
      self.min_sup = min_sup
      self.path = path
      self.unique_words, self.word_list, self.data = self.parse_data()
      self.items = {}
      self.num_users = len(self.data)

   def parse_data(self):
      files = os.listdir(self.path)
      files = [self.path + file for file in files]
      unique_words = {} # contains unique words in the data files
      unique_counter = 0 # ID number assigned to each unique word
      seqs = []
      word_list = []

      for filename in files:
         # read user data
         lines = []
         f = codecs.open(filename, encoding='utf-8')
         for line in f:
            lines.append(line.encode('utf-8'))
         ans = {} # contains final time series dictionary
         seq = {}
         seq['file'] = filename
         s1 = []
         s2 = []
         # for each line in data file
         for X in lines:
            Y = X.strip().split(';')
            time = Y[0].split(' ')[1]
            # extract time info
            x = []

            # extract useful info at a given time stamp
            for i,data in enumerate(Y):
               if i > 0:
```

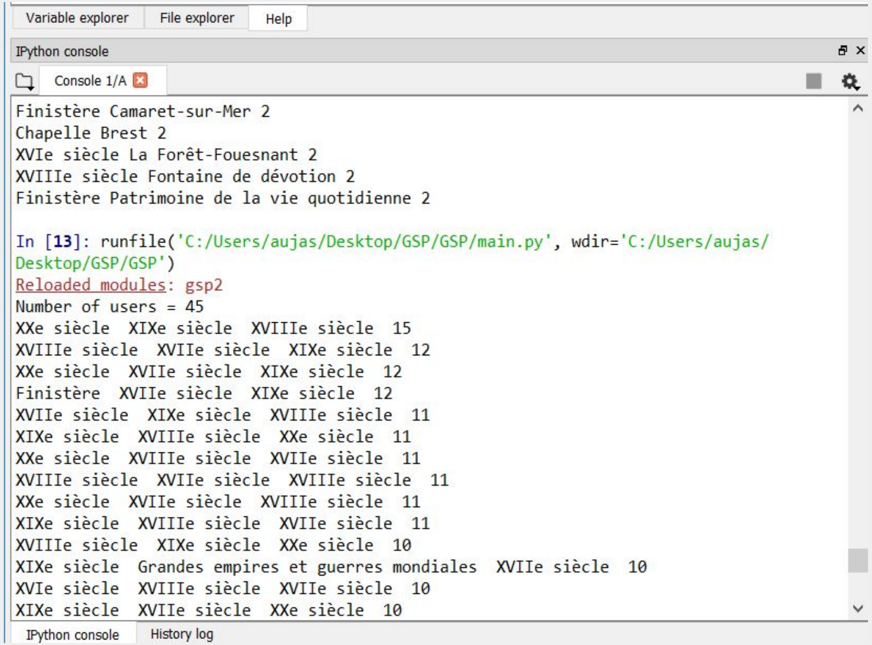

Spyder (Python 2.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\aujas\Desktop\GSP\GSP
Editor - C:\Users\aujas\Desktop\GSP\GSP\main.py
temp.py
main2.py
main.py
gsp.py
2017-09-11_13_17_40_192.168.76.201_recommendation.txt

```python
from gsp import GSP
threshold = 10 # minimum support
level = 3 # number of levels for GSP
g = GSP(threshold, './Data_GSP/')
g.get_support_items(level)
```

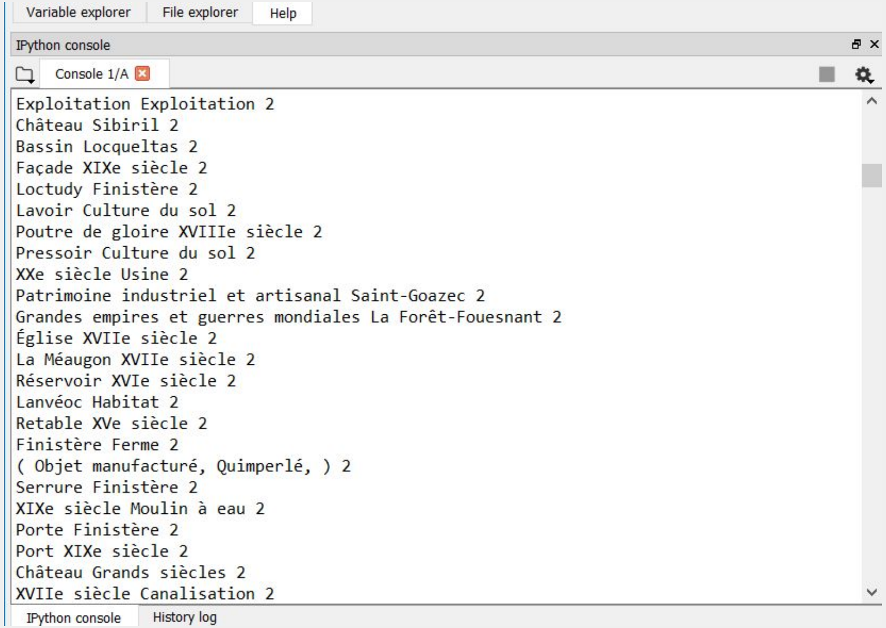

Spyder (Python 2.7)
File Edit Search Source Run Debug Consoles Projects Tools View Help
C:\Users\aujas\Desktop
Editor - C:\Users\aujas\Desktop\GSP\GSP\main2.py
temp.py
main2.py
main.py
gsp.py
2017-09-11_13_17_40_192.168.76.201_recommendation.txt

```python
from gsp2 import GSP
threshold = 2 # minimum support
level = 2 # number of levels for GSP
g = GSP(threshold, './Data_GSP/')
g.get_support_items(level)
```

Results



```
Finistère Camaret-sur-Mer 2
Chapelle Brest 2
XVIe siècle La Forêt-Fouesnant 2
XVIIIe siècle Fontaine de dévotion 2
Finistère Patrimoine de la vie quotidienne 2

In [13]: runfile('C:/Users/aujas/Desktop/GSP/GSP/main.py', wdir='C:/Users/aujas/
Desktop/GSP/GSP')
Reloaded modules: gsp2
Number of users = 45
XXe siècle  XIXe siècle  XVIIIe siècle  15
XVIIIe siècle  XVIIe siècle  XIXe siècle  12
XXe siècle  XVIIe siècle  XIXe siècle  12
Finistère  XVIIe siècle  XIXe siècle  12
XVIIe siècle  XIXe siècle  XVIIIe siècle  11
XIXe siècle  XVIIIe siècle  XXe siècle  11
XXe siècle  XVIIIe siècle  XVIIe siècle  11
XVIIIe siècle  XVIIe siècle  XVIIIe siècle  11
XXe siècle  XVIIe siècle  XVIIIe siècle  11
XIXe siècle  XVIIIe siècle  XVIIe siècle  11
XVIIIe siècle  XIXe siècle  XXe siècle  10
XIXe siècle  Grandes empires et guerres mondiales  XVIIe siècle  10
XVIe siècle  XVIIIe siècle  XVIIe siècle  10
XIXe siècle  XVIIe siècle  XXe siècle  10
```

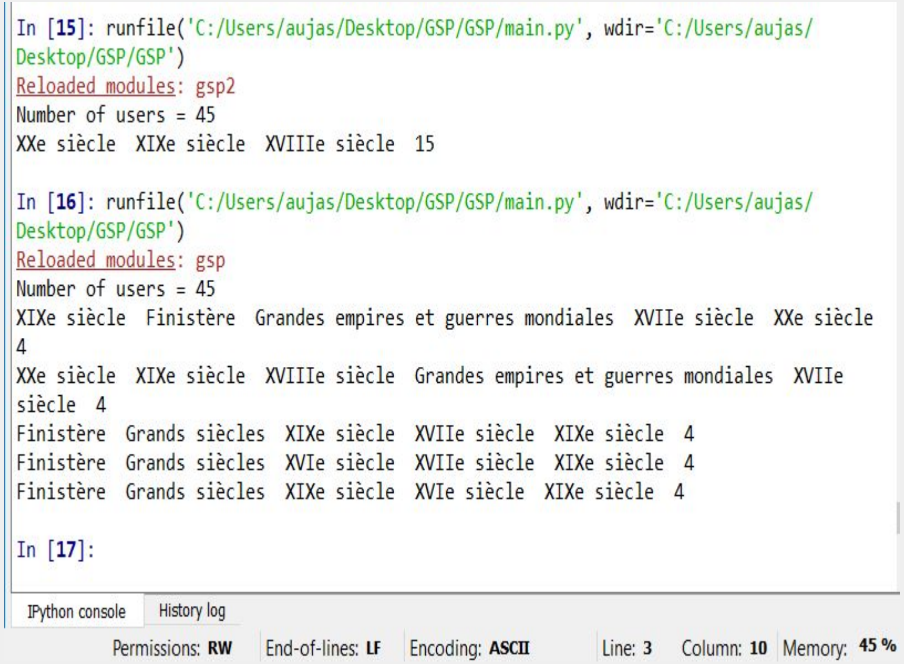


```
Exploitation Exploitation 2
Château Sibiril 2
Bassin Locqueltas 2
Façade XIXe siècle 2
Loctudy Finistère 2
Lavoir Culture du sol 2
Poutre de gloire XVIIIe siècle 2
Pressoir Culture du sol 2
XXe siècle Usine 2
Patrimoine industriel et artisanal Saint-Goazec 2
Grandes empires et guerres mondiales La Forêt-Fouesnant 2
Église XVIIe siècle 2
La Méaugon XVIIe siècle 2
Réservoir XVIe siècle 2
Lanvéoc Habitat 2
Retable XVe siècle 2
Finistère Ferme 2
( Objet manufacturé, Quimperlé, ) 2
Serrure Finistère 2
XIXe siècle Moulin à eau 2
Porte Finistère 2
Port XIXe siècle 2
Château Grands siècles 2
XVIIe siècle Canalisation 2
```

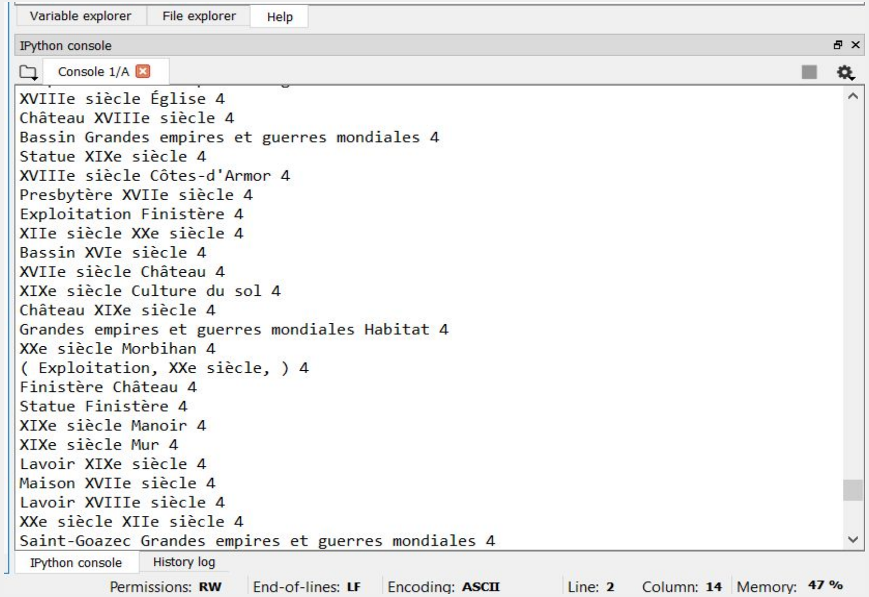
Patterns by 3-Candidate Level GSP, Sup-10

Patterns by 2-Candidate Level GSP(Tuples), Sup-2

Contd.



```
In [15]: runfile('C:/Users/aujas/Desktop/GSP/GSP/main.py', wdir='C:/Users/aujas/
Desktop/GSP/GSP')
Reloaded modules: gsp2
Number of users = 45
XXe siècle  XIXe siècle  XVIIIe siècle  15

In [16]: runfile('C:/Users/aujas/Desktop/GSP/GSP/main.py', wdir='C:/Users/aujas/
Desktop/GSP/GSP')
Reloaded modules: gsp
Number of users = 45
XIXe siècle  Finistère  Grandes empires et guerres mondiales  XVIIe siècle  XXe siècle
4
XXe siècle  XIXe siècle  XVIIIe siècle  Grandes empires et guerres mondiales  XVIIe
siècle  4
Finistère  Grands siècles  XIXe siècle  XVIIe siècle  XIXe siècle  4
Finistère  Grands siècles  XVIe siècle  XVIIe siècle  XIXe siècle  4
Finistère  Grands siècles  XIXe siècle  XVIe siècle  XIXe siècle  4

In [17]:
```



```
XVIIIe siècle Église 4
Château XVIIIe siècle 4
Bassin Grandes empires et guerres mondiales 4
Statue XIXe siècle 4
XVIIIe siècle Côtes-d'Armor 4
Presbytère XVIIe siècle 4
Exploitation Finistère 4
XIIe siècle XXe siècle 4
Bassin XVIe siècle 4
XVIIe siècle Château 4
XIXe siècle Culture du sol 4
Château XIXe siècle 4
Grandes empires et guerres mondiales Habitat 4
XXe siècle Morbihan 4
( Exploitation, XXe siècle, ) 4
Finistère Château 4
Statue Finistère 4
XIXe siècle Manoir 4
XIXe siècle Mur 4
Lavoir XIXe siècle 4
Maison XVIIe siècle 4
Lavoir XVIIIe siècle 4
XXe siècle XIIe siècle 4
Saint-Goazec Grandes empires et guerres mondiales 4
```

Patterns by 5-Candidate Level GSP, Sup - 4

Patterns by 3-Candidate Level GSP(Tuples), Sup-4

GSP-Algorithm (Bottlenecks)

- Scans the database multiple times
- Generate a huge set of candidate sequences
- Non-existent candidates
- Maintaining candidates in the memory

There is need of more efficient mining methods!

Conclusion

- Sequential Pattern Mining is useful in many applications eg. weblog analysis, financial market prediction, Bioinformatics etc.
- Useful for frequent itemsets mining, but with consideration of ordering
- Descendants of popular algorithm in mining frequent itemsets like **AprioriAll**

References

- ❑ [1] R. Srikant and R. Agrawal. Mining sequential patterns: Generalizations and performance improvements. EDBT’96.
- ❑ [2] Fournier-Viger, P., Lin, J. C.-W., Kiran, R. U., Koh, Y. S., Thomas, R. (2017). A Survey of Sequential Pattern Mining. Data Science and Pattern Recognition, vol. 1(1), pp. 54-77.
- ❑ [3] Wikipedia: "GSP Algorithm." http://en.wikipedia.org/wiki/GSP_Algorithm
- ❑ [4] Jed Isom. (2015, March 10). Generalized Sequential Pattern (GSP) Mining [Blog Post]. Retrieved from http://simpledatamining.blogspot.fr/2015/03/generalized-sequential-pattern-gsp.html/

Thank You!