Tour Version

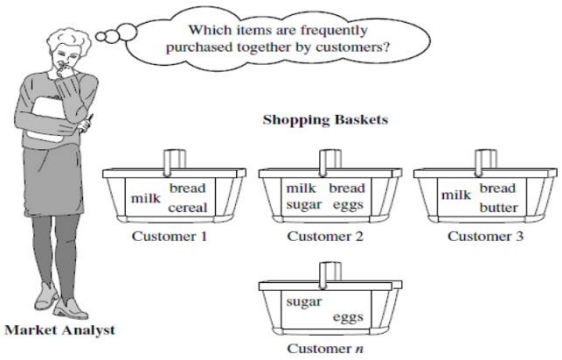
Data Mining

Tour 3

Mining Frequent Pattern

Mining Frequent Pattern, Association& Correlation

Frequent Pattern



- **a** pattern **(a set of items, subsequences, substructures, etc.) that occurs** frequently **in a data set**
- **A Big role for mining association and correlation**
- **بس ليه بدور على الأشكال دى في الداتا أيه الحافز؟؟**
- Motivation: Finding inherent regularities in data
- **لما تلاقى ان السؤال بتاعك شبه الأسئلة الجاية يبقى فعلا محتاج الموضوع ده**
    - What products were often purchased together?— Beer and diapers?!
    - What are the subsequent purchases after buying a PC?
    - What kinds of DNA are sensitive to this new drug?
    - Can we automatically classify web documents?
- **و بيخش في مجموعة من التطبيقات اللى في الحياة**
- Applications: **Basket data analysis, cross-marketing, catalog design, sale campaign analysis, Web log (click stream) analysis, and DNA sequence analysis**
- Frequent Pattern are itemsets that appear frequently in a data set (e.g. Transaction record)
- Items that are frequently associated (e.g purchased) together can be represented as association rules**.Association Rule like =>**
- Computer → antivirus_SW [Support = 2% , Confidence =60%]
- **معنى كده ان شراء الكمبيوتر بيأثر على شراء الانتى فيروس** باجمالى 2% من بياعة المنتجات **و** نسبة 60% من اللى اشتروا كمبيوترات اشتروا كمان انتى فيروس بس مش العكس ؛؛ احتمال شرطي أفتكر دكتور شريف
- **Support** and **Confidence** are measures of rule interestingness
- **و يعتبروا هم اللى بقارن بيهم الثريشولد بتوعى و بختبر بيهم مدى القرار على المنتجات و هو اللى بيعرفنى الباترن**
- **طبعا فيه ثريشولد أكتر من دول بس المحاضرة بنتكلم عن 2**
- 2% Support means 2% of Transactions Show that computers and antivirus_SW are bought Together
- 60% Confidence means 60 % of customers who bought a computer also bought antivirus_SW

- Basics About **Association Rules**:
  - If frequency of itemset I satisfies min_support count then I is a frequent itemset
  - Support(x) >= min_support => x is frequent item set
  - If a rule satisfies min_support and min_confidence thresholds,it is said to be strong
    - problem of mining association rules reduced to mining frequent itemsets
  - Support(X)>= min_support && Conf(X)>= min_confidence => strong Association
  - Association rules mining becomes a two-step process:
    - Find all frequent itemsets that occur at least as frequently as a predetermined min_support count
    - Generate strong association rules from the frequent itemsets that satisfy min_support and min_confidence

- Itemset X = {x1, …, xk} ex: X={A, B, C, D, E, F}
- Find all the rules X → Y with minimum support and confidence
  - support, s, probability that a transaction contains $X \cup Y$
  - confidence, c, conditional probability that a transaction having X also contains Y

$$support\ X \rightarrow Y = P(X \cup Y) = \frac{n(X \cup Y)}{N}$$

$$confidence\ (X \rightarrow Y) = P(Y|X) = \frac{n(X \cup Y)}{n(X)}$$

Ex: Let min_Sup. = 50%, min_conf. = 50%

Frequent Patterns:

{A:3, B:3, D:4, E:3, AD:3}

Association rules:

A → D (60%, 100%)

D → A (60%, 75%)

$$conf\ (A \rightarrow D) = \frac{3}{3} = 100\ \%$$

$$conf\ (D \rightarrow A) = \frac{3}{4} = 75\ \%$$

Transaction-id	Items bought
10	A, B, D
20	A, C, D
30	A, D, E
40	B, E, F
50	B, C, D, E, F

- نبدأ نحل المسألة دى ب 2 الجوريزم و هختبر الحلين بالبايثون كمان الله المستعان
  - Aprioiri
  - FP-Groth

Apriori Algorithm

Transactional data example
N=10, min_supp count=2

TID	List of items
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3
T1000	I1, I2

Apriori Algorithm

أول حاجة محتاجها إنه نشوف إيه هي

Frequent Itemsets?

دي بتتعرفها عن طريق min-Support threshold وبعد كده
تشوف Association Rules اللي من خلالها هتشوف

وهتختار كل Pattern تبع min-Confidence Threshold

واحد إبدأ بـ items وعددهم

min-Support = 2 min-Confidence = %70

C_1

L_1 Frequent itemsets هيبقى دول كلهم

Itemset	# Support
{I1}	7 ≥ 2 ✓
{I2}	8 ✓
{I3}	6 ✓
{I4}	2 ✓
{I5}	2 ✓

هنقارنهم بال min-Sup

Itemset	# Support
{I1}	7
{I2}	8
{I3}	6
{I4}	2
{I5}	2

هنعمل بقى Sets الـ Compose
ونعدهم حصلوا كام مرة

C_2

Itemsets	# Support	
$[I_1, I_2]$	$5 \geq 2$	✓
$[I_1, I_3]$	4	✓
$[I_1, I_4]$	1	✗
$[I_1, I_5]$	3	✓
$[I_2, I_3]$	4	✓
$[I_2, I_4]$	2	✓
$[I_2, I_5]$	2	✓
$[I_3, I_4]$	0	✗
$[I_3, I_5]$	1	✗
$[I_4, I_5]$	0	✗

منضتبش الـ min-sup ⇐

L_2

Itemsets	# Support
$[I_1, I_2]$	5
$[I_1, I_3]$	4
$[I_1, I_5]$	3
$[I_2, I_3]$	4
$[I_2, I_4]$	2
$[I_2, I_5]$	2

ذعمل تاني بس Compose
ناخد بالنا من الـ Pruning إن
لو فيه Subset مابتحققوش الـ min-Sup يبقى
الـ Superset كمان منه هتحقق

C_3

Itemsets	# Support	
$[I_1, I_2, I_3]$	$2 \geq 2$	✓
$[I_1, I_2, I_5]$	2	✓

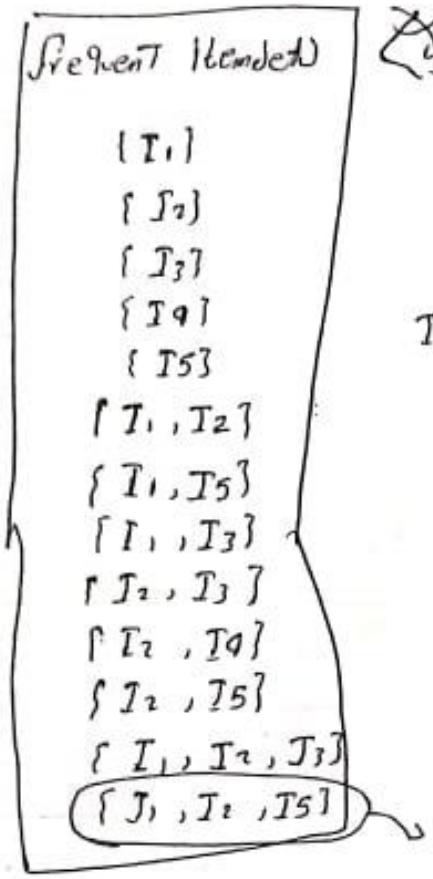
⇒

L_3

Itemsets	# Support
$[I_1, I_2, I_3]$	2
$[I_1, I_2, I_5]$	2

وطبعاً لو عملنا Compose مش هيتبقى حاجة لأنه هيحصل
مابيحققوش Subset هم $[I_3, I_5]$ لأن Pruning

لو جالك سؤال عن مين هم الـ Frequent Set بيبقى كل اللى حققوا



ItemSet يعني تعالى ناخد مثلاً منهم عشان نعمل فيهم Association Rules

نبدأ نطالع منه الـ Subsets عشان نشوف الـ Pattern

إحتمال شرطى

Association Rules	Confidence
$[I_1] \rightarrow [I_2, I_5]$	28% ✗
$[I_2] \rightarrow [I_1, I_5]$	25% ✗
$[I_5] \rightarrow [I_1, I_2]$	100% ✓
$[I_1, I_2] \rightarrow [I_5]$	40% ✗
$[I_1, I_5] \rightarrow [I_2]$	100% ✓
$[I_2, I_5] \rightarrow [I_1]$	100% ✓

$$Conf = \frac{P([I_1] \cup [I_2, I_5])}{P([I_1])} = \frac{2}{7} = 28\%$$

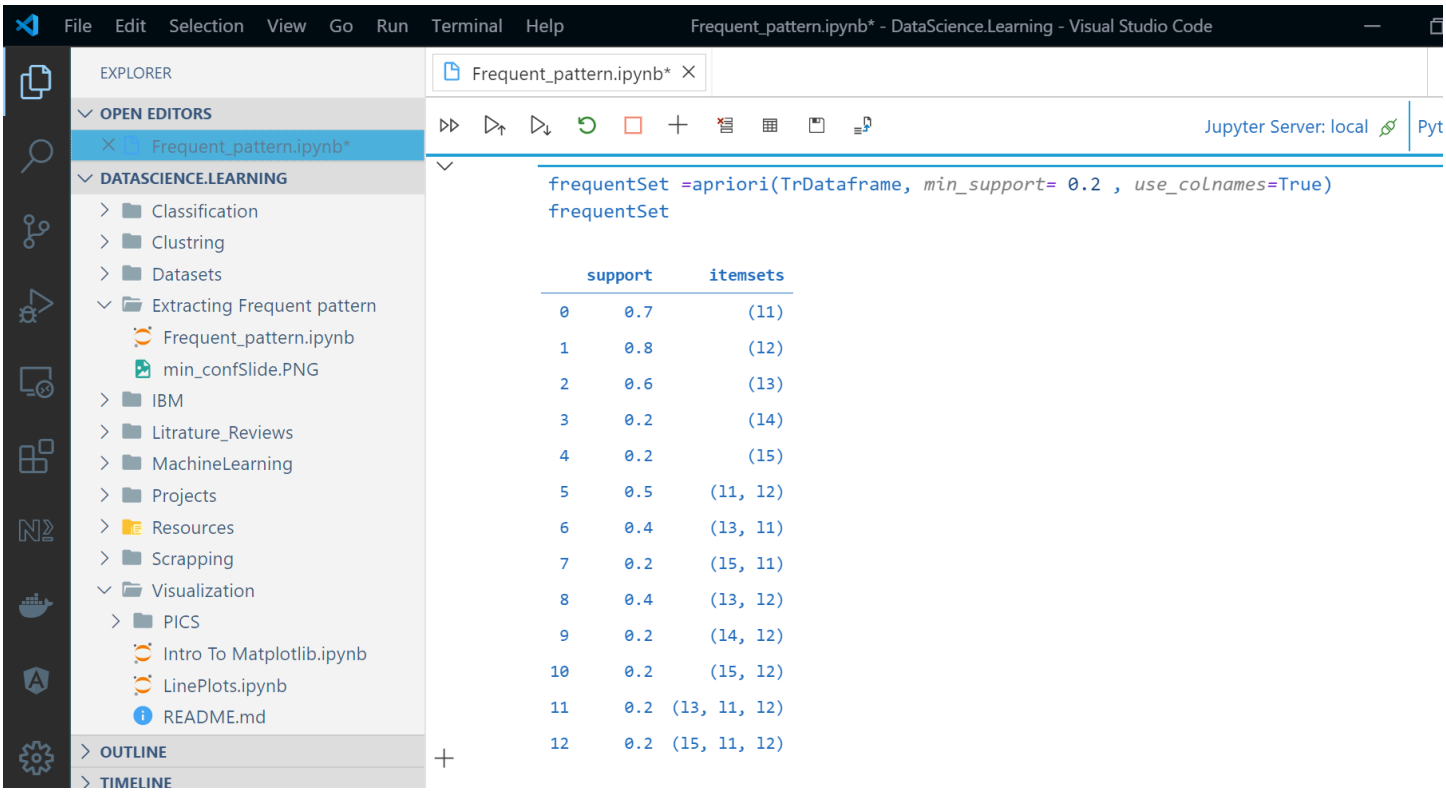
وهكذا

يبقى دول الـ Strong Association

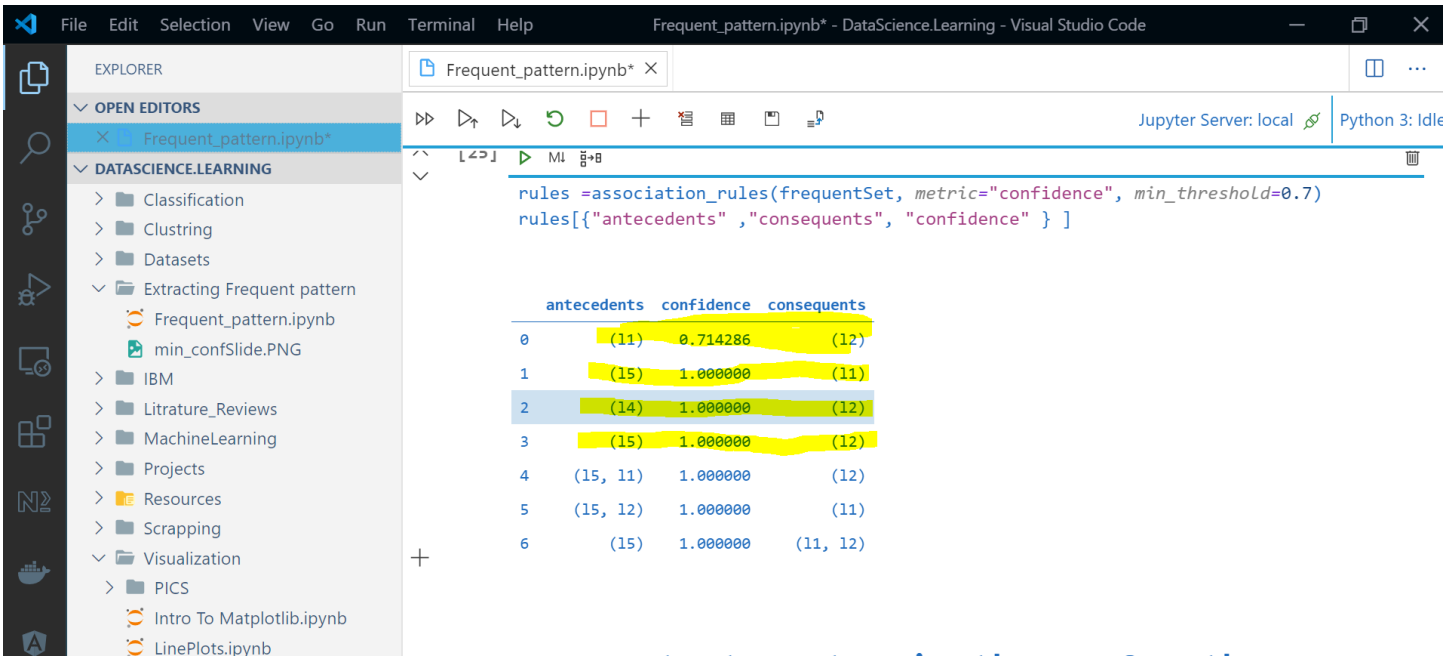
يبقى احنا اخترنا على Frequent Itemset واحدة بس

- أنا اسف خطى وحش جدا و حاسس بالموقف اللى انت فيه بس ربنا يستربس،، ده كله عشان ما بتقرأش المصادر يا عزيزى
- و بعد ما اتحلت منكم نحلها بايثون تمام و هنشوف ايه اللى فاضل من السترونج
- وده اللينك بتاع الكود اقرا المصادر بقى

- https://github.com/AhmedKhalil777/DataScience.Learning/blob/master/Extracting%20Frequent%20pattern/Frequent_pattern.ipynb



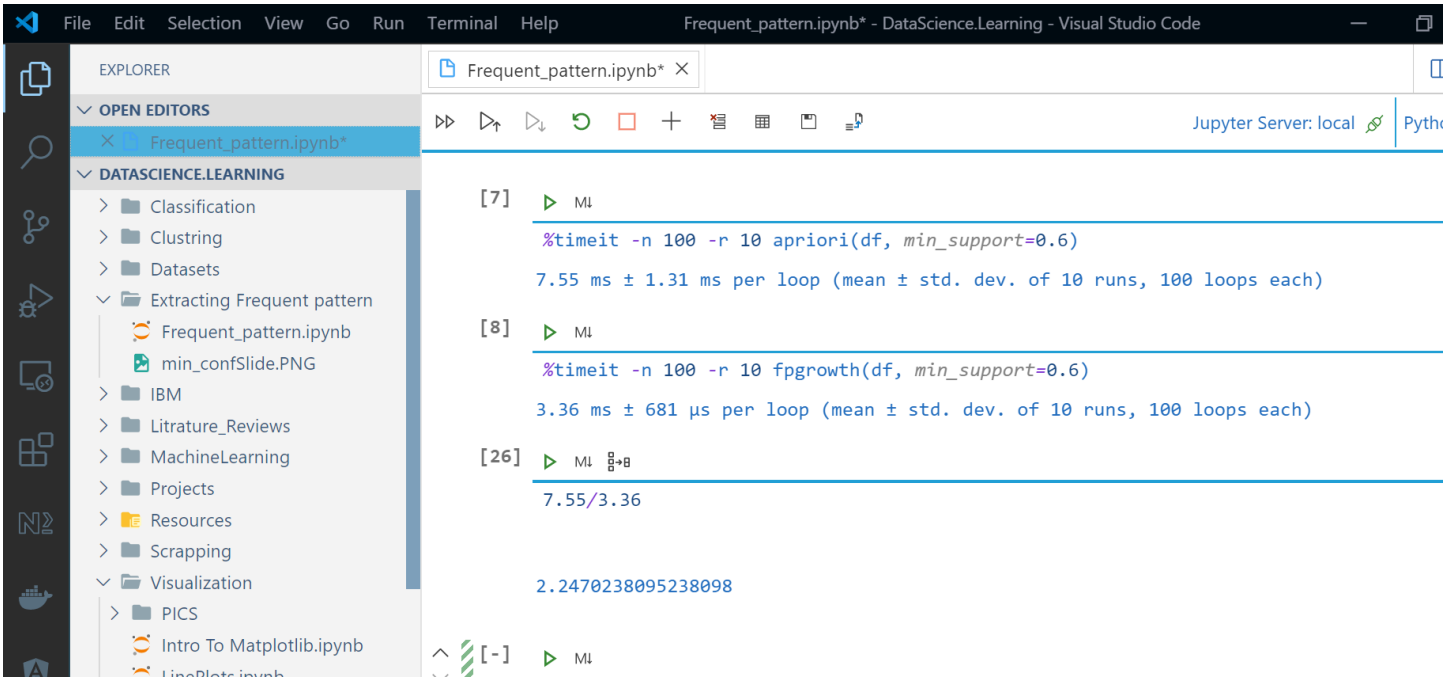
	support	itemsets
0	0.7	(11)
1	0.8	(12)
2	0.6	(13)
3	0.2	(14)
4	0.2	(15)
5	0.5	(11, 12)
6	0.4	(13, 11)
7	0.2	(15, 11)
8	0.4	(13, 12)
9	0.2	(14, 12)
10	0.2	(15, 12)
11	0.2	(13, 11, 12)
12	0.2	(15, 11, 12)



	antecedents	confidence	consequents
0	(11)	0.714286	(12)
1	(15)	1.000000	(11)
2	(14)	1.000000	(12)
3	(15)	1.000000	(12)
4	(15, 11)	1.000000	(12)
5	(15, 12)	1.000000	(11)
6	(15)	1.000000	(11, 12)

FP-Growth

- أولا انا ما اتوقعش ان هي تيجى في امتحان بس للحظر بقول افهمها من المحاضرة
- اه هي تلخبط بس اللى فاهم ال
- Merge and Conquer
- هيفهمها كويس
- بس انا هعملها كود وهحسب ال
- Time cost
- Big dataبتاع الالجوريزمين و هتلاحظ فرق الوقت لو بتتعامل مع



- بعد ال test Benchmarking
- اتضح ان في داتا بسيطة زى المسألة ان سرعة الالجوريزم ده أسرع مرتين وربع و طبعا هيختلف لو الداتا كبرت
- To avoid costly candidate generation
- Divide-and-conquer strategy:
  - Compress database representing frequent items into a frequent pattern tree (FPtree) – 2 passes over dataset
  - Divide compressed database (FP-tree) into conditional databases, then mine each for frequent itemsets – traverse through the FP-tree

دلوقتي هنتكلم على حل المسألة بطريقة FP-Growth

هنرجع لعملنا المرة اللي فاتت
هنعمل المرة دي بحس جزئيهم

C_1

Itemsets	# Support
{I1}	7
{I2}	8
{I3}	6
{I4}	2
{I5}	2

L_1

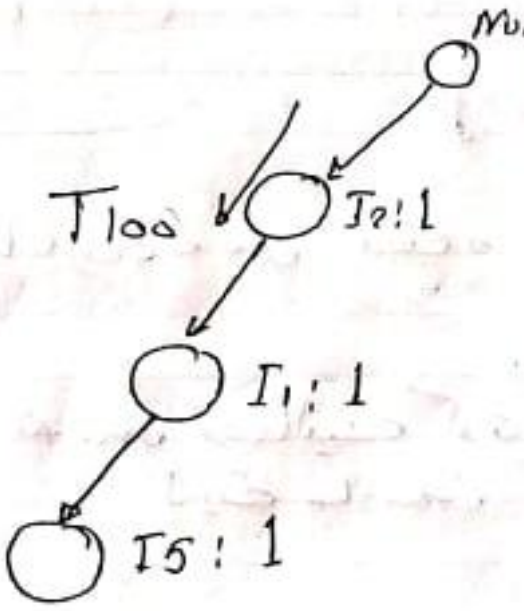
Itemsets	# Support
{I2}	8
{I1}	7
{I3}	6
{I4}	2
{I5}	2

Transactional data example
N=10, min_supp count=2

TID	List of items
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3
T1000	I1, I2

دلوقتي هنعمل الـ FP-Tree وهنبدأ بـ Null Sets في الأول وبترتيب العناصر

I: Support في كل Node



دلوقتي هنرجع للجدول الأساسي

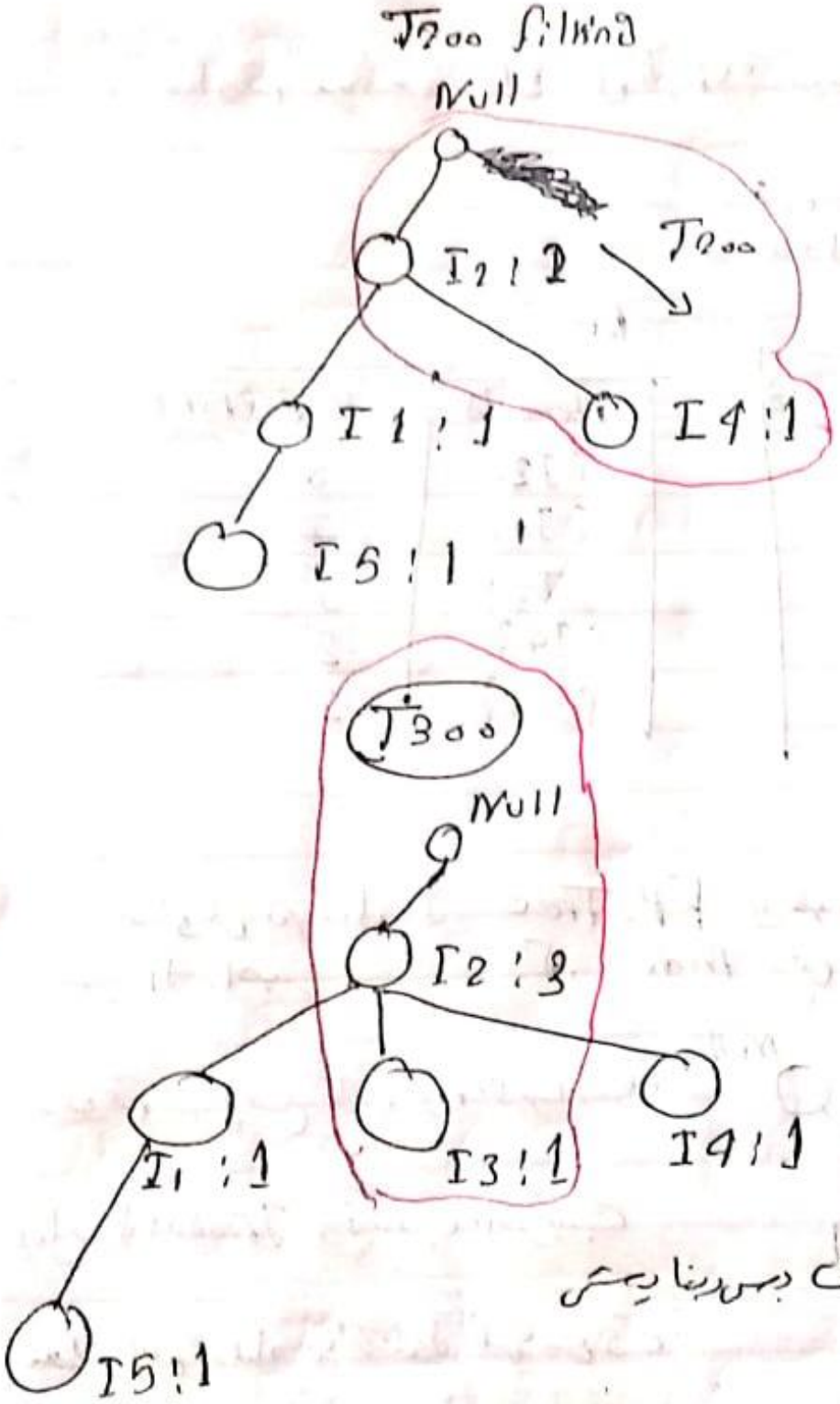
بتاع الـ Transactions وهنكتب بالترتيب

اللي عملناه في الجدول اللي فوق

يعني اللي في الـ Tree دي T100

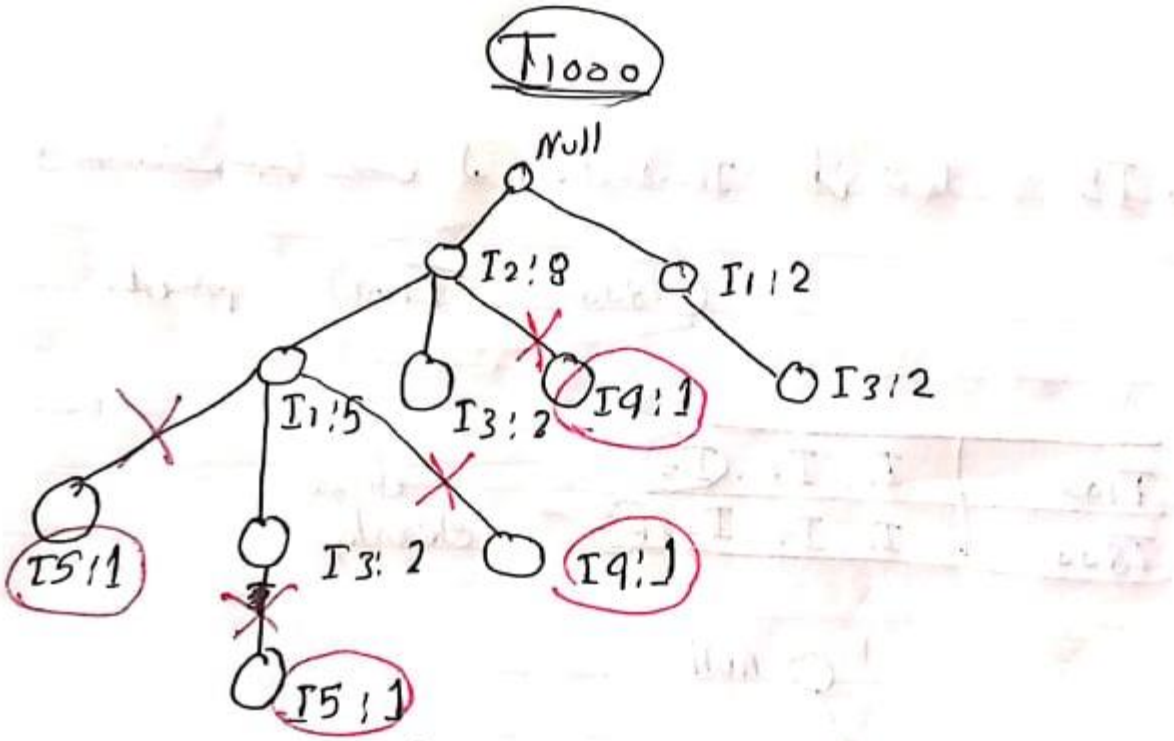
بس بترتيب الجدول L1

وحطت كل واحدة بـ 1 لأن دي Transaction1 بس



Transactional data example
N=10, min_supp count=2

TID	List of items
T100	I1, I2, I5
T200	I2, I4
T300	I2, I3
T400	I1, I2, I4
T500	I1, I3
T600	I2, I3
T700	I1, I3
T800	I1, I2, I3, I5
T900	I1, I2, I3
T1000	I1, I2



هعمل كل مسار فى الـ Tree وهشوف إذا كان محقق الـ min_Sup
ولا لأ واللى مبيحققش Eliminate

ونبدأ نكتب إيه الـ Set اللى بتحقق

وهنا هيظهر مشكلة إن الـ I3 لما ظهرت هى السبب إن

حالة الـ I5 تنقسم

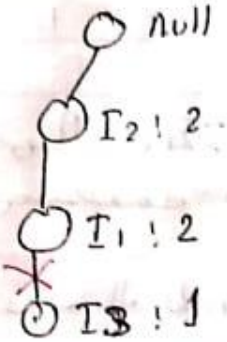
ومنبقى منه تاخد الجدول بس بطريقة الـ Conditional_FP

وهنستخدم بس الـ Transaction اللي فيها I5 مثلاً

(T800 , T100) اللي هم

T100	I1, I2, I5
T800	I1, I2, I3, I5

هنشيل I5 eleminate



تقاطع كده هنحذف I3 eliminate
I3 < min-supp

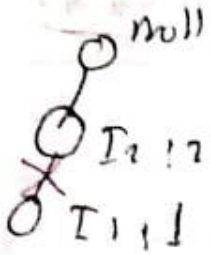
نيجي على تاني واحدة حصلها elimination برضه عشان هي كمان عددهم < 2

I4

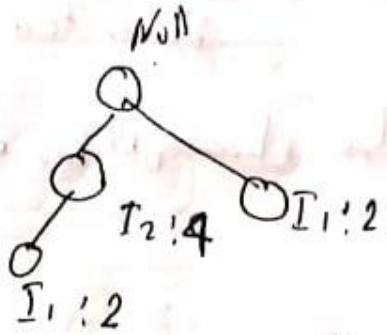
وبرضه الـ Transaction اللي فيهم I9 هنشيل فيهم زي I8

T200	I2, I9
T400	I1, I2, I9

eleminate



For I3



T300	I2 , I3
T500	I1 , I3
T600	I2 , I3
T700	I1 , I3
T800	I1 , I2 , I3 , I5
T900	I1 , I2 , I3

هنا بنعمل Cond-Tree اللي بتنتهي بـ I3 ويبقى مش ضروري I5 تاني

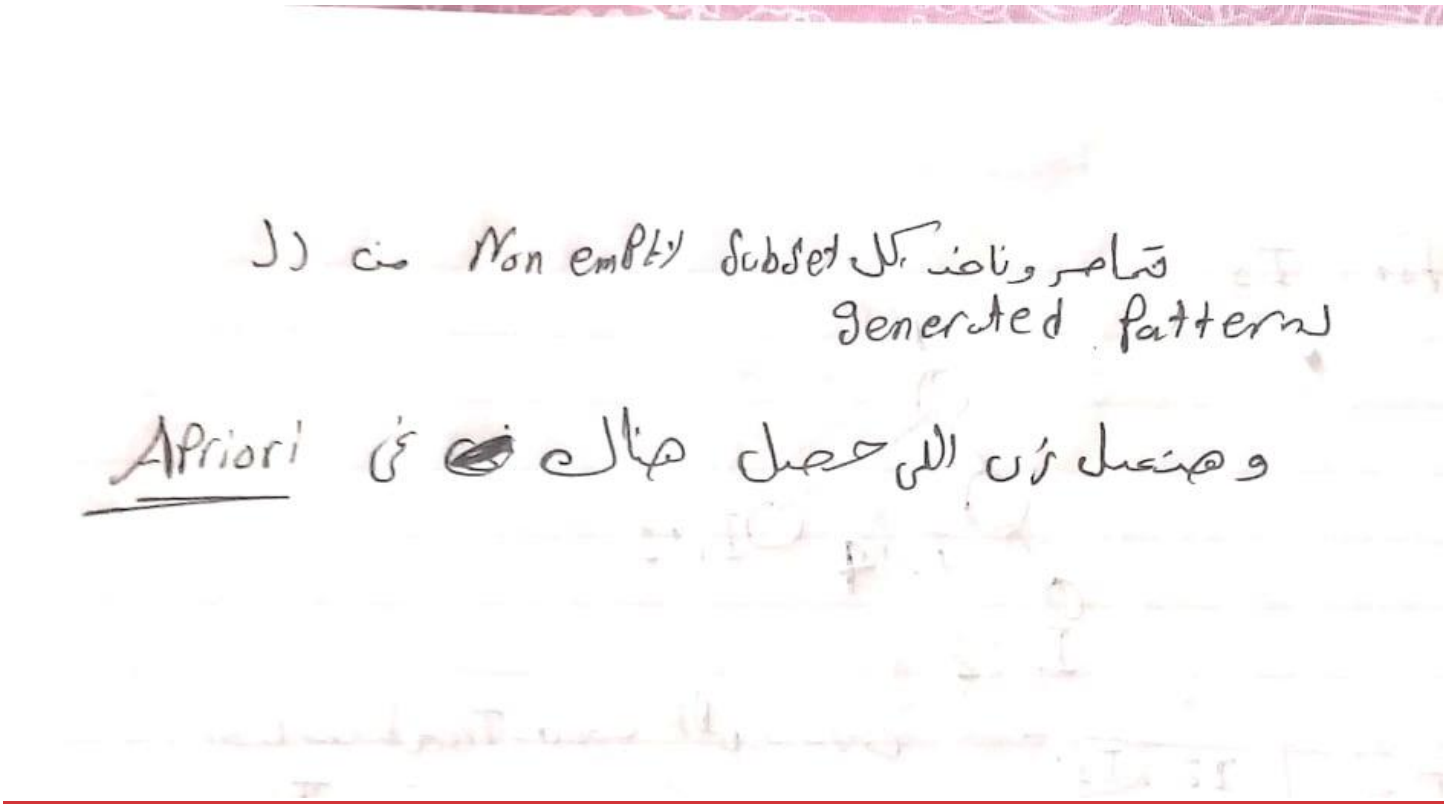
TID	List of items
T100	~~I1~~, I2, I5
~~T200~~	~~I2, I4~~
~~T300~~	~~I2, I3~~
T400	~~I1~~, I2, I4
T500	~~I1~~, I3
~~T600~~	~~I2, I3~~
T700	~~I1~~, I3
T800	~~I1~~, I2, I3, I5
T900	~~I1~~, I2, I3
T1000	~~I1~~, I2

For I1

وبرضوا هنعمل Elemination للباقي
زي ما احنا عارفين

طبعاً الـ (Conditional-FP) Trees هنبدأ نجيب منهم الـ Frequent Itemsets ونحطهم في جدول

Item	Conditional Pattern Base	Conditional FP-tree	Frequent Patterns Generated
I5	{{I2, I1: 1}, {I2, I1, I3: 1}}	<I2:2, I1:2>	{I2, I5: 2}, {I1, I5: 2}, {I2, I1, I5: 2}
I4	{{I2, I1: 1}, {I2: 1}}	<I2:2>	{I2, I4: 2}
I3	{{I2, I1: 2}, {I2: 2}, {I1: 2}}	<I2:4, I1:2>, <I1:2>	{I2, I3: 4}, {I1, I3: 4}, {I2, I1, I3: 2}
I1	{{I2: 5}}	<I2:5>	{I2, I1: 5}



Pattern Evaluation Method

- Not all association rules are interesting
  - Buys(X,"Computer games" → buys(X,"Videos") [40%, 66%]
  - P("videos") is 75% > 66%
  - The two items are
    - negatively associated means buying one decreases the likelihood of buying the other
  - We need to measure "real strength" of rule
  - Correlation analysis

- A→ B [support , confidence , correlation]

- يعنى نضيف قاعدة زيادة عشان نشوف اذا كان نشوف هو ايجابى او سلبى

1. Lift = $\frac{P(A \cup B)}{P(A)P(B)}$
   - A and B are independent if P(A U B) = P(A)P(B)
   - Otherwise, dependent and correlated occurrence
   - If lift <1, A is Negatively correlated with B
   - If lift >1, A is Positively correlated with B …… A's occurrence "lifts" the occurrence of B