# **BÁO CÁO THỰC HÀNH**

# **Supermarket**

### Mô tả bài toán

Bài toán khai phá dạng mẫu trong giỏ hàng để tìm ra mặt hàng hay được mua kèm với nhau

### Mô tả dữ liệu

1. Tên dữ liệu: supermarket
2. Nguồn dữ liệu: có sẵn trong data của weka
3. Số lượng mẫu dữ liệu: 4627
4. Số thuộc tính 105
5. Đặc điểm tập dữ liệu: đa biến

### Mô tả thuật toán sẽ sử dụng

Sử dụng thuật toán **Apriori** là **thuật toán** khả sinh được đề xuất bởi R. Agrawal và R. Srikant vào năm 1993 để khai thác các tập item đối với các luật kết hợp kiểu bool. ... **Apriori** dùng cách tiếp cận lặp được biết đến như tìm kiếm level-wise, với các tập k item được dùng để thăm dò các tập (k+1) item

**Bước 1**: Đếm số support cho mỗi tập một phần tử và xem chúng như một Large itemset. Support của chúng là minsup

**Bước 2**: Với mỗi tập Large item bổ sung các item vào và tạo một Large itemset mới, tập này được gọi là tập ứng viên( Candidate - C). Đếm số support cho mỗi tập C trên cơ sở dữ liệu, từ đó quyết định tập C nào là Large Item thực sự, và ta dùng cho bước kế tiếp

**Bước 3**: Lặp lại bước 2 cho đến khi không tìm thấy thêm một tập Large itemset nữa

### Các tham số

lowerBoundMinSupport: 0,3

minMetric: 0,8

Số lần thực hiện: 14

Kích thước của bộ itemsets L (1): 22

Kích thước của bộ itemsets L (2): 61

Kích thước của bộ itemsets L (3): 20

### Kết quả và đánh giá

1. ***Kết quả***

***Ta có 10 luật tốt nhất được sinh ra bởi thuật toán Apriori trên dữ liệu* supermarket*.arff như sau (sắp xếp giảm dần theo confidence)***

1. biscuits=t vegetables=t 1764 ==> bread and cake=t 1487 <conf:(0.84)> lift:(1.17) lev:(0.05) [217] conv:(1.78)

2. biscuits=t milk-cream=t 1767 ==> bread and cake=t 1485 <conf:(0.84)> lift:(1.17) lev:(0.05) [213] conv:(1.75)

3. biscuits=t fruit=t 1837 ==> bread and cake=t 1541 <conf:(0.84)> lift:(1.17) lev:(0.05) [218] conv:(1.73)

4. biscuits=t frozen foods=t 1810 ==> bread and cake=t 1510 <conf:(0.83)> lift:(1.16) lev:(0.04) [207] conv:(1.69)

5. frozen foods=t fruit=t 1861 ==> bread and cake=t 1548 <conf:(0.83)> lift:(1.16) lev:(0.05) [208] conv:(1.66)

6. frozen foods=t milk-cream=t 1826 ==> bread and cake=t 1516 <conf:(0.83)> lift:(1.15) lev:(0.04) [201] conv:(1.65)

7. baking needs=t milk-cream=t 1907 ==> bread and cake=t 1580 <conf:(0.83)> lift:(1.15) lev:(0.04) [207] conv:(1.63)

8. milk-cream=t fruit=t 2038 ==> bread and cake=t 1684 <conf:(0.83)> lift:(1.15) lev:(0.05) [217] conv:(1.61)

9. baking needs=t biscuits=t 1764 ==> bread and cake=t 1456 <conf:(0.83)> lift:(1.15) lev:(0.04) [186] conv:(1.6)

10. baking needs=t fruit=t 1900 ==> bread and cake=t 1564 <conf:(0.82)> lift:(1.14) lev:(0.04) [196] conv:(1.58)

1. ***Đánh giá***

Với độ tin cậy = 80% thì theo kết quả trên ta có:

1. Bánh quy và rau thường mua kèm với bánh mì và bánh ngọt
2. Bánh quy và kem sữa thường mua kèm với bánh mì và bánh ngọt
3. Bánh quy và trái cây thường mua kèm với bánh mì và bánh ngọt
4. Bánh quy và thực phẩm đông lạnh thường mua kèm với bánh mì và bánh ngọt
5. Trứng và trái cây thường mua kèm với bánh mì và bánh ngọt

…

# **Khai phá dữ liệu hỗ trợ sinh viên lập kế hoạch học tập**

### Mô tả bài toán

Tìm ra các môn học nếu sinh viên học tốt môn này thì sẽ có khả năng học tốt môn kia

### Mô tả dữ liệu

Link dữ liệu: [Dữ liệu hỗ trợ sinh viên lập kế hoạch học tập](https://docs.google.com/spreadsheets/d/19-iXjRunZHgHTtLxoRvrUCKscekrg0ECp8BKQ8nPqDk/edit?usp=sharing)

* Dữ liệu thu thập thông qua website daotao.vnua.edu.vn. Lấy điểm TBC cuối kỳ trong 6 môn học của 48 sinh viên

1. Số lượng mẫu dữ liệu: 48
2. Số thuộc tính: 6( Cấu trúc dữ liệu & giải thuật, Xác suất thống kê, Toán giải tích, Phương pháp tính, Toán rời rạc, Trí tuệ nhân tạo)

* Chuẩn hoá dữ liệu về dạng nhị phân với 1 là học tốt và 0 là không học tốt. Chuẩn hoá dựa trên điểm sinh viên nếu >=5 là 1; <5 là 0

### Mô tả thuật toán sử dụng

Sử dụng thuật toán **Apriori** là **thuật toán** khả sinh được đề xuất bởi R. Agrawal và R. Srikant vào năm 1993 để khai thác các tập item đối với các luật kết hợp kiểu bool. ... **Apriori** dùng cách tiếp cận lặp được biết đến như tìm kiếm level-wise, với các tập k item được dùng để thăm dò các tập (k+1) item

**Bước 1**: Đếm số support cho mỗi tập một phần tử và xem chúng như một Large itemset. Support của chúng là minsup

**Bước 2**: Với mỗi tập Large item bổ sung các item vào và tạo một Large itemset mới, tập này được gọi là tập ứng viên( Candidate - C). Đếm số support cho mỗi tập C trên cơ sở dữ liệu, từ đó quyết định tập C nào là Large Item thực sự, và ta dùng cho bước kế tiếp

**Bước 3**: Lặp lại bước 2 cho đến khi không tìm thấy thêm một tập Large itemset nữa

### Các tham số

lowerBoundMinSupport: 0,4

minMetric: 0,7

Số lần thực hiện: 12

Kích thước của bộ itemsets L (1): 9

Kích thước của bộ itemsets L (2): 6

### Kết quả và đánh giá

1. XacSuatThongKe=1 22 ==> TriTueNhanTao=1 22 <conf:(1)> lift:(1.2) lev:(0.08) [3] conv:(3.67)

1. conf = 1 quy tắc này là đúng 100% sinh viên học tốt XacSuatThongKe thì sẽ học tốt môn TriTueNhanTao
2. lift = 1,2 > 1 nó là tỷ lệ hữu ích khi học tốt môn XacSuatThongKe sẽ có ảnh hưởng tích cực đến việc học tốt môn TriTueNhanTao
3. lev = 0,08 tỷ lệ khi hai môn này được học cùng nhau sẽ tốt hơn 8%
4. conv = 3.67 cho biết tần suất học tốt môn xác suất thống kê mà không cần học tốt môn TriTueNhanTao

2. PhuongPhapTinh=1 28 ==> TriTueNhanTao=1 26 <conf:(0.93)> lift:(1.11) lev:(0.06) [2] conv:(1.56)

1. conf = 0,93 tỷ lệ 93% sinh viên học tốt PhuongPhapTinh sẽ học tốt môn AI.
2. lift = 1,11 > 1 nó là tỷ lệ của việc nếu học tốt môn PhuongPhapTinh thì sẽ có ảnh hưởng tích cực đến việc học tốt môn TriTueNhanTao.
3. lev: 0,06 => là tỷ lệ khi học 2 môn cùng nhau sẽ hiệu quả hơn 6%
4. conv: 1.56 => là tỷ lệ của việc học tốt môn PhuongPhapTinh mà không cần học tốt môn TriTueNhanTao

3. ToanRoiRac=1 23 ==> TriTueNhanTao=1 21 <conf:(0.91)> lift:(1.1) lev:(0.04) [1] conv:(1.28)

1. conf: 0.91 => tỷ lệ 91% sinh viên học tốt ToanRoiRac sẽ học tốt môn trí tuệ nhân tạo.
2. lift: 1.1 > 1 nó là tỷ lệ của việc nếu học tốt môn ToanRoiRac thì sẽ có ảnh hưởng tích cực đến việc học tốt môn TriTueNhanTao
3. lev: 0,04 => là tỷ lệ khi học 2 môn cùng nhau sẽ hiệu quả hơn 4%
4. conv: 1.28 => là tỷ lệ của việc học tốt môn ToanRoiRac mà không cần học tốt môn TriTueNhanTao

4. ToanGiaiTich=0 28 ==> TriTueNhanTao=1 22 <conf:(0.79)> lift:(0.94) lev:(-0.03) [-1] conv:(0.67)

1. conf = 0,79 79% sinh viên học tốt ToanGiaiTich thì sẽ học tốt môn TriTueNhanTao
2. lift = 0,94 cho thấy rằng điều này nếu học kém môn ToanGiaiTich sẽ ảnh hưởng đến việc học tốt môn TriTueNhanTao
3. lev: -0,03 => là tỷ lệ của việc học kém ToanGiaiTich ảnh hưởng đến việc học tốt AI
4. conv: 0.67 => là tỷ lệ của việc học kém môn ToanGiaiTich mà không phụ thuộc vào việc học tốt môn TriTueNhanTao.

5. ToanRoiRac=0 25 ==> XacSuatThongKe=0 19 <conf:(0.76)> lift:(1.4) lev:(0.11) [5] conv:(1.64)

1. conf: 0.76 => tỷ lệ 76% sinh viên học kém ToanRoiRac sẽ học kém môn XacSuatThongKe
2. lift: 1.4 > 1 nó là tỷ lệ của việc nếu học kém môn ToanRoiRac thì sẽ có ảnh hưởng tích cực đến việc học kém môn TriTueNhanTao
3. lev: 0.11 => là tỷ lệ khi học kém 2 môn cùng nhau là 11%
4. conv: 1.64 => là tỷ lệ của việc học kém môn ToanRoiRac mà không phụ thuộc vào việc học kém môn XacSuatThongKe.

6. ToanRoiRac=0 25 ==> TriTueNhanTao=1 19 <conf:(0.76)> lift:(0.91) lev:(-0.04) [-1] conv:(0.6)

1. conf: 0.76 => tỷ lệ 76% sinh viên học kém ToanRoiRac sẽ học tốt môn AI.
2. lift: 0.91 < 1 Điều này có nghĩa là nếu học kém môn ToanRoiRac thì sẽ ảnh hưởng xấu đến việc học tốt môn TriTueNhanTao.
3. lev: -0,04 => là tỷ lệ của việc học kém ToanRoiRac ảnh hưởng đến việc học tốt AI
4. conv: 0.6 => là tỷ lệ của việc học kém môn ToanRoiRac mà không phụ thuộc vào việc học tốt môn TriTueNhanTao

7. XacSuatThongKe=0 26 ==> ToanRoiRac=0 19 <conf:(0.73)> lift:(1.4) lev:(0.11) [5] conv:(1.56)

1. conf: 0.73 => tỷ lệ 73% sinh viên học kém XacSuatThongKe sẽ học kém môn ToanRoiRac.
2. lift: 1.4 > 1 nó là tỷ lệ của việc nếu học kém môn XacSuatThongKe thì sẽ có ảnh hưởng tích cực đến việc học kém môn ToanRoiRac.
3. lev: 0.11 => là tỷ lệ học kém 2 môn cùng nhau.
4. conv: 1.56 => là tỷ lệ của việc học kém môn XacSuatThongKe mà không phụ thuộc vào việc học kém môn ToanRoiRac.

Link tham khảo:

1. <https://en.wikipedia.org/wiki/Association_rule_learning#Conviction>
2. <https://michael.hahsler.net/research/recommender/associationrules.html>