ĐẠI HỌC KHOA HỌC TỰ NHIÊN TP HCM
KHOA CÔNG NGHỆ THÔNG TIN

18_21
NHẬP MÔN DỮ LIỆU LỚN

BÁO CÁO

LAB 04

Lập trình Spark với Python

Mục lục

1. Thông tin thành viên nhóm:

STT	MSSV	Họ và tên
1	18120023	Nguyễn Huy Hải
2	18120058	Phạm Công Minh
3	18120533	Dương Đoàn Bảo Sơn
4	18120543	Trần Đại Tài

2. Phân công công việc và đánh giá:

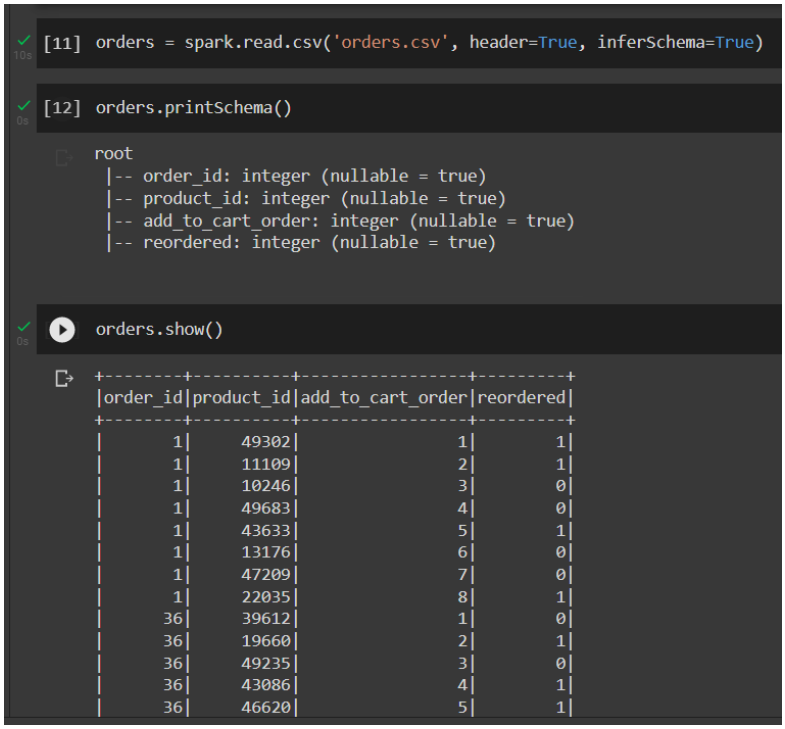
STT	Họ và tên	Phân công	Tự đánh giá
1	Nguyễn Huy Hải	Code Bài 1	100%
2	Phạm Công Minh	Tổng hợp code và quay video Viết báo cáo	100%
3	Dương Đoàn Bảo Sơn	Code Bài 2	100%
4	Trần Đại Tài	Code Bài 3	100%

3. Nội dung thực hiện:

3.1. Bài 01: Khai thác mẫu phổ biến và luật kết hợp

a) Đọc và xử lý dữ liệu:

- Đọc dữ liệu file **orders.csv**:



```
[11] orders = spark.read.csv('orders.csv', header=True, inferSchema=True)

[12] orders.printSchema()

root
 |-- order_id: integer (nullable = true)
 |-- product_id: integer (nullable = true)
 |-- add_to_cart_order: integer (nullable = true)
 |-- reordered: integer (nullable = true)

orders.show()

+--------+----------+-----------------+---------+
|order_id|product_id|add_to_cart_order|reordered|
+--------+----------+-----------------+---------+
|       1|     49302|                1|        1|
|       1|     11109|                2|        1|
|       1|     10246|                3|        0|
|       1|     49683|                4|        0|
|       1|     43633|                5|        1|
|       1|     13176|                6|        0|
|       1|     47209|                7|        0|
|       1|     22035|                8|        1|
|      36|     39612|                1|        0|
|      36|     19660|                2|        1|
|      36|     49235|                3|        0|
|      36|     43086|                4|        1|
|      36|     46620|                5|        1|
```

- Trong bài tập này chỉ sử dụng 2 cột **order_id** (mã giao dịch) và **product_id** (mã sản phẩm) vì vậy lọc ra 2 cột này:

```
[14] orders = orders.select(col('order_id').alias('id'), 'product_id')
     orders.show()

+---+----------+
| id|product_id|
+---+----------+
|  1|     49302|
|  1|     11109|
|  1|     10246|
|  1|     49683|
|  1|     43633|
|  1|     13176|
|  1|     47209|
|  1|     22035|
| 36|     39612|
| 36|     19660|
| 36|     49235|
| 36|     43086|
| 36|     46620|
| 36|     34497|
| 36|     48679|
| 36|     46979|
| 38|     11913|
| 38|     18159|
| 38|      4461|
| 38|     21616|
+---+----------+
only showing top 20 rows
```

- Đọc dữ liệu file **products.csv**

```
[15] products = spark.read.csv('products.csv', header=True, inferSchema=True)

[16] products.printSchema()

root
 |-- product_id: integer (nullable = true)
 |-- product_name: string (nullable = true)
 |-- aisle_id: string (nullable = true)
 |-- department_id: string (nullable = true)

     products.show(truncate=False)

+----------+-----------------------------------------------------------------+--------+-------------+
|product_id|product_name                                                     |aisle_id|department_id|
+----------+-----------------------------------------------------------------+--------+-------------+
|1         |Chocolate Sandwich Cookies                                       |61      |19           |
|2         |All-Seasons Salt                                                 |104     |13           |
|3         |Robust Golden Unsweetened Oolong Tea                             |94      |7            |
|4         |Smart Ones Classic Favorites Mini Rigatoni With Vodka Cream Sauce|38      |1            |
|5         |Green Chile Anytime Sauce                                        |5       |13           |
|6         |Dry Nose Oil                                                     |11      |11           |
|7         |Pure Coconut Water With Orange                                   |98      |7            |
|8         |Cut Russet Potatoes Steam N' Mash                                |116     |1            |
|9         |Light Strawberry Blueberry Yogurt                                |120     |16           |
|10        |Sparkling Orange Juice & Prickly Pear Beverage                   |115     |7            |
|11        |Peach Mango Juice                                                |31      |7            |
|12        |Chocolate Fudge Layer Cake                                       |119     |1            |
|13        |Saline Nasal Mist                                                |11      |11           |
|14        |Fresh Scent Dishwasher Cleaner                                   |74      |17           |
|15        |Overnight Diapers Size 6                                         |56      |18           |
                                                                    0s  completed at 8:48 PM
```

- Trong bài tập này chỉ sử dụng 2 cột **product_id** (mã sản phẩm) và **product_name** (tên sản phẩm) vì vậy lọc ra 2 cột này

```
products = products.select('product_id', 'product_name')
products.show(truncate=False)
```

```
+----------+--------------------------------------------------------------+
|product_id|product_name                                                  |
+----------+--------------------------------------------------------------+
|1         |Chocolate Sandwich Cookies                                    |
|2         |All-Seasons Salt                                              |
|3         |Robust Golden Unsweetened Oolong Tea                          |
|4         |Smart Ones Classic Favorites Mini Rigatoni With Vodka Cream Sauce|
|5         |Green Chile Anytime Sauce                                     |
|6         |Dry Nose Oil                                                  |
|7         |Pure Coconut Water With Orange                                |
|8         |Cut Russet Potatoes Steam N' Mash                             |
|9         |Light Strawberry Blueberry Yogurt                             |
|10        |Sparkling Orange Juice & Prickly Pear Beverage                |
|11        |Peach Mango Juice                                             |
|12        |Chocolate Fudge Layer Cake                                    |
|13        |Saline Nasal Mist                                             |
|14        |Fresh Scent Dishwasher Cleaner                                |
|15        |Overnight Diapers Size 6                                      |
|16        |Mint Chocolate Flavored Syrup                                 |
|17        |Rendered Duck Fat                                             |
|18        |Pizza for One Suprema  Frozen Pizza                           |
|19        |Gluten Free Quinoa Three Cheese & Mushroom Blend              |
|20        |Pomegranate Cranberry & Aloe Vera Enrich Drink                |
+----------+--------------------------------------------------------------+
only showing top 20 rows
```

Chuyển dữ liệu thành dạng phù hợp để áp dụng các giải thuật khai thác mẫu phổ biến và luật kết hợp: bảng dữ liệu mới bao gồm hai cột theo đúng thứ tự là mã giao dịch (order_id) và danh sách tên các sản phẩm thuộc giao dịch đó như hình dưới đây

id	products
1	['Beef', 'Chicken', 'Milk']

- Đầu tiên, join bảng **orders** với bảng **product** bằng **product_id** để lấy được tên các sản phẩm

```
transaction = orders.join(products, orders.product_id == products.product_id)
transaction.show(truncate=False)
```

```
+---+----------+----------+----------------------------------------+
|id |product_id|product_id|product_name                            |
+---+----------+----------+----------------------------------------+
|1  |49302     |49302     |Bulgarian Yogurt                        |
|1  |11109     |11109     |Organic 4% Milk Fat Whole Milk Cottage Cheese|
|1  |10246     |10246     |Organic Celery Hearts                   |
|1  |49683     |49683     |Cucumber Kirby                          |
|1  |43633     |43633     |Lightly Smoked Sardines in Olive Oil    |
|1  |13176     |13176     |Bag of Organic Bananas                  |
|1  |47209     |47209     |Organic Hass Avocado                    |
|1  |22035     |22035     |Organic Whole String Cheese             |
|36 |39612     |39612     |Grated Pecorino Romano Cheese           |
|36 |19660     |19660     |Spring Water                            |
|36 |49235     |49235     |Organic Half & Half                     |
|36 |43086     |43086     |Super Greens Salad                      |
|36 |46620     |46620     |Cage Free Extra Large Grade AA Eggs     |
|36 |34497     |34497     |Prosciutto, Americano                   |
|36 |48679     |48679     |Organic Garnet Sweet Potato (Yam)       |
|36 |46979     |46979     |Asparagus                               |
|38 |11913     |11913     |Shelled Pistachios                      |
|38 |18159     |18159     |Organic Biologique Limes                |
|38 |4461      |4461      |Organic Raw Unfiltered Apple Cider Vinegar|
|38 |21616     |21616     |Organic Baby Arugula                    |
+---+----------+----------+----------------------------------------+
only showing top 20 rows
```

- Groupby theo **id** và sử dụng hàm **collect_list** để thu được cột chứa danh sách các sản phẩm trong từng giao dịch sẽ thu được bảng dữ liệu mong muốn

```
transaction = transaction.groupBy('id').agg(f.collect_list('product_name').alias('products'))
transaction.show(truncate=False)
```

```
+----+--------------------------------------------------------------------------------------------------
|id  |products
+----+--------------------------------------------------------------------------------------------------
|1   |[Bulgarian Yogurt, Organic 4% Milk Fat Whole Milk Cottage Cheese, Organic Celery Hearts, Cucu
|96  |[Roasted Turkey, Organic Cucumber, Organic Grape Tomatoes, Organic Pomegranate Kernels, Orgar
|112 |[Fresh Cauliflower, I Heart Baby Kale, Sea Salt Baked Potato Chips, Marinara Pasta Sauce, Org
|218 |[Natural Artisan Water, Okra, Organic Yellow Peaches, Black Plum, Citrus Mandarins Organic]
|456 |[Chorizo Pork, Petite Peas, Max Gel Clog Remover, Green Beans, Naturally Hickory Smoked Home
|473 |[Organic Whole Milk with DHA Omega-3, Banana, Unsweetened Original Almond Breeze Almond Milk
|631 |[Organic Strawberries, Uncured Genoa Salami, Organic Baby Carrots, Gluten Free Organic Taco S
|762 |[Organic Strawberries, Organic Romaine Lettuce, Celery Hearts, Organic Cucumber]
|774 |[Ice Cream Variety Pack, Nacho Cheese Sauce, Deli-Sliced Hot Jalapeño Peppers]
|844 |[Green Beans, Organic Red Radish, Bunch, Baby Spinach, Organic Shredded Carrots, Granny Smith
|904 |[Cup Noodles Chicken Flavor, Zero Calorie Cola]
|988 |[Natural Vanilla Ice Cream, Whipped Light Cream, Original, Complete ActionPacs Lemon Burst Dr
|1032|[Clover Org Greek Plain, Electrolyte Enhanced Water, Organic Raspberries, Natural Spring Wate
|1077|[Bag of Organic Bananas, Celery Sticks, Sparkling Water, Organic Strawberries]
|1119|[Boneless Skinless Chicken Breast, Large Lemon, Organic Grade A Free Range Large Brown Eggs,
|1139|[Banana, Organic Strawberries, Red Vine Tomato, Organic Bakery Hamburger Buns Wheat - 8 CT, C
|1143|[Natural Premium Coconut Water, Calming Lavender Body Wash, Unscented Long Lasting Stick Deod
|1145|[Banana, Original French Toast Sticks, Healthy Multi Grain Bread, Harvest Best in 100% Fruit
|1275|[Boneless Skinless Chicken Breast, Organic Garnet Sweet Potato (Yam), Small Hass Avocado, Ext
|1280|[Lactose Free Half & Half, Organic Half & Half, Vanilla Soy Milk, Organic Whole Milk, French
+----+--------------------------------------------------------------------------------------------------
```

b) Áp dụng giải thuật khai thác mẫu phổ biến và luật kết hợp:

- Tìm các mẫu phổ biến với min Support = 0.01. Mặc dù thử nghiệm trên các chỉ số min Support và min Confidence khác nhau nhưng mẫu phổ biến vẫn giống nhau, chỉ tăng thêm nếu min Support nhỏ đi nên chỉ thực hiện tìm các mẫu phổ biến 1 lần

```
fpGrowth = FPGrowth(itemsCol="products", minSupport=0.01, minConfidence=0.9)
model = fpGrowth.fit(transaction)
patternsDF = model.freqItemsets
patternsDF.orderBy('freq', ascending = False).show(truncate=False)
```

```
/usr/local/lib/python3.7/dist-packages/pyspark/sql/context.py:127: FutureWarning: D
  FutureWarning
+------------------------+-----+
|items                   |freq |
+------------------------+-----+
|[Banana]                |18726|
|[Bag of Organic Bananas]|15480|
|[Organic Strawberries]  |10894|
|[Organic Baby Spinach]  |9784 |
|[Large Lemon]           |8135 |
|[Organic Avocado]       |7409 |
|[Organic Hass Avocado]  |7293 |
|[Strawberries]          |6494 |
|[Limes]                 |6033 |
|[Organic Raspberries]   |5546 |
|[Organic Blueberries]   |4966 |
|[Organic Whole Milk]    |4908 |
|[Organic Cucumber]      |4613 |
|[Organic Zucchini]      |4589 |
|[Organic Yellow Onion]  |4290 |
|[Organic Garlic]        |4158 |
|[Seedless Red Grapes]   |4059 |
|[Asparagus]             |3868 |
|[Organic Grape Tomatoes]|3823 |
|[Organic Red Onion]     |3818 |
+------------------------+-----+
only showing top 20 rows
```

- Tìm các luật kết hợp với các chỉ số min Support và min Confidence khác nhau:

Các luật kết hợp với min Support = 0.001 và min Confidence = 0.5

```
fpGrowth = FPGrowth(itemsCol="products", minSupport=0.001, minConfidence=0.5)
model = fpGrowth.fit(transaction)
rulesDF = model.associationRules
rulesDF.show(truncate=False)
```

```
/usr/local/lib/python3.7/dist-packages/pyspark/sql/context.py:127: FutureWarning: Deprecated in 3.0.0. Use SparkSession.builder.getOrCreate() instead.
  FutureWarning
+------------------------------------------------------------+----------------------+------------------+------------------+----------------------+
|antecedent                                                  |consequent            |confidence        |lift              |support               |
+------------------------------------------------------------+----------------------+------------------+------------------+----------------------+
|[Organic Whole String Cheese, Organic Hass Avocado]         |[Bag of Organic Bananas]|0.5314685314685315|4.504745125675359 |0.0011584571180330617|
|[Organic Broccoli, Organic Hass Avocado]                    |[Bag of Organic Bananas]|0.5048231511254019|4.278897986822536 |0.001196564260073623 |
|[Organic Navel Orange, Organic Raspberries]                 |[Bag of Organic Bananas]|0.5412186379928315|4.587387356098284 |0.0011508356896249496|
|[Organic Kiwi, Organic Hass Avocado]                        |[Bag of Organic Bananas]|0.5459770114942529|4.627719489738336 |0.001448071397541327 |
|[Organic D'Anjou Pears, Organic Hass Avocado]               |[Bag of Organic Bananas]|0.5170454545454546|4.3824946411792345|0.0013870999702764292|
|[Organic Raspberries, Organic Hass Avocado]                 |[Bag of Organic Bananas]|0.521099116781158 |4.416853618458589 |0.004046978484707604 |
|[Organic Navel Orange, Organic Hass Avocado]                |[Bag of Organic Bananas]|0.5283018867924528|4.477904539027839 |0.0014937999679900007|
|[Organic Raspberries, Organic Hass Avocado, Organic Strawberries]|[Bag of Organic Bananas]|0.5984251968503937|5.072272070642333 |0.0017376856770495927|
|[Organic Unsweetened Almond Milk, Organic Hass Avocado]     |[Bag of Organic Bananas]|0.5141065830721003|4.357584667849303 |0.0012499142589304088|
|[Organic Cucumber, Organic Hass Avocado, Organic Strawberries]|[Bag of Organic Bananas]|0.546875          |4.635330870478036 |0.0010669999771357147|
|[Yellow Onions, Strawberries]                               |[Banana]              |0.5357142857142857|3.7536332219526702|0.0011432142612168373|
+------------------------------------------------------------+----------------------+------------------+------------------+----------------------+
```

Các luật kết hợp với min Support = 0.001 và min Confidence = 0.4

```
fpGrowth = FPGrowth(itemsCol="products", minSupport=0.001, minConfidence=0.4)
model = fpGrowth.fit(transaction)
rulesDF = model.associationRules
rulesDF.show(truncate=False)
```

```
/usr/local/lib/python3.7/dist-packages/pyspark/sql/context.py:127: FutureWarning: Deprecated in 3.0.0. Use SparkSession.builder.getOrCreate()
  FutureWarning
+-------------------------------------------------+----------------------+-------------------+------------------+----------------------+
|antecedent                                       |consequent            |confidence         |lift              |support               |
+-------------------------------------------------+----------------------+-------------------+------------------+----------------------+
|[Apple Honeycrisp Organic, Organic Raspberries]  |[Bag of Organic Bananas]|0.4307692307692308 |3.651214470284238 |0.0010669999771357147|
|[Organic Red Bell Pepper, Organic Hass Avocado]  |[Bag of Organic Bananas]|0.4568764568764569 |3.87250019575601  |0.0014937999679900007|
|[Asparagus, Limes]                               |[Large Lemon]         |0.407673860911271  |6.575350905507923 |0.0012956428293790822|
|[Organic Fuji Apple, Large Lemon]                |[Banana]              |0.42702702702702705|2.992085292597949 |0.0012041856884817351|
|[Organic Peeled Whole Baby Carrots, Organic Avocado]|[Banana]              |0.4005681818181818 |2.806693931869156 |0.001074621405543827 |
|[Organic Yellow Squash]                          |[Organic Zucchini]    |0.46770025839793283|13.372517586431547|0.0013794785418683169|
|[Organic Whole String Cheese, Organic Hass Avocado]|[Bag of Organic Bananas]|0.5314685314685315 |4.504745125675359 |0.0011584571180330617|
|[Organic Broccoli, Organic Hass Avocado]         |[Bag of Organic Bananas]|0.5048231511254019 |4.278897986822536 |0.001196564260073623 |
|[Strawberries, Organic Avocado]                  |[Banana]              |0.4643478260869565 |3.2535839962108017|0.0020349213849659704|
|[Asparagus, Organic Raspberries]                 |[Bag of Organic Bananas]|0.42138364779874216|3.5716619537483956|0.0010212714066870413|
|[Organic Navel Orange, Organic Raspberries]      |[Bag of Organic Bananas]|0.5412186379928315 |4.587387356098284 |0.0011508356896249496|
|[Organic Raspberries, Organic Baby Spinach]      |[Bag of Organic Bananas]|0.40706806282722513|3.450322574644534 |0.002370264234922909 |
|[Honeycrisp Apple, Organic Avocado]              |[Banana]              |0.4049733570159858 |2.8375600342150205|0.0017376856770495927|
|[Organic Cucumber, Organic Raspberries]          |[Bag of Organic Bananas]|0.450530035335689  |3.8187077135891747|0.0019434642440686234|
|[Organic Kiwi, Organic Hass Avocado]             |[Bag of Organic Bananas]|0.5459770114942529 |4.627719489738336 |0.001448071397541327 |
|[Organic Cucumber, Organic Strawberries]         |[Bag of Organic Bananas]|0.4108527131782946 |3.482401398153156 |0.0032314856450395934|
|[Organic Zucchini, Organic Hass Avocado]         |[Bag of Organic Bananas]|0.40930232558139534|3.4692602608016347|0.0020120570997416336|
|[Seedless Red Grapes, Organic Avocado]           |[Banana]              |0.4117647058823529 |2.8851455353047974|0.001440449969133215 |
|[Bunched Cilantro, Large Lemon]                  |[Limes]               |0.46228710462287104|10.05407404449897 |0.001448071397541327 |
|[Organic Navel Orange, Organic Baby Spinach]     |[Bag of Organic Bananas]|0.4249201277955272 |3.6016372769976313|0.001013649978278929 |
+-------------------------------------------------+----------------------+-------------------+------------------+----------------------+
```

Các luật kết hợp với min Support = 0.005 và min Confidence = 0.3

```
fpGrowth = FPGrowth(itemsCol="products", minSupport=0.005, minConfidence=0.3)
model = fpGrowth.fit(transaction)
rulesDF = model.associationRules
rulesDF.show(truncate=False)
```

```
/usr/local/lib/python3.7/dist-packages/pyspark/sql/context.py:127: FutureWarning: Deprecated in 3.0.0. Use SparkSession.builder.getOr
  FutureWarning
+----------------------------------------+----------------------+-------------------+------------------+----------------------+
|antecedent                              |consequent            |confidence         |lift              |support               |
+----------------------------------------+----------------------+-------------------+------------------+----------------------+
|[Apple Honeycrisp Organic]              |[Bag of Organic Bananas]|0.30506216696269983|2.585717174742176 |0.0052359213163731145|
|[Organic Raspberries]                   |[Organic Strawberries]  |0.3011179228272629 |3.6267102566772844|0.012727785441547455  |
|[Organic Raspberries]                   |[Bag of Organic Bananas]|0.32095203750450774|2.720400251222801 |0.013566142566439803  |
|[Organic Lemon]                         |[Bag of Organic Bananas]|0.3044222539229672 |2.580293250321615 |0.00813206411145577   |
|[Organic Hass Avocado]                  |[Bag of Organic Bananas]|0.33182503770739064|2.8125601661853374|0.018443856747631642  |
|[Blueberries]                           |[Banana]                |0.30822212656048215|2.159645252796876 |0.00545694274020837   |
|[Organic Large Extra Fancy Fuji Apple]  |[Bag of Organic Bananas]|0.3365617433414044 |2.8527086422533805|0.0074156498410932175 |
|[Organic Fuji Apple]                    |[Banana]                |0.37150752225974826|2.60307222515109  |0.009221928373815821  |
|[Organic Navel Orange]                  |[Bag of Organic Bananas]|0.3661616161616162 |3.103598158588469 |0.0055255355958813805 |
|[Organic Hass Avocado, Organic Strawberries]|[Bag of Organic Bananas]|0.4613385315139701 |3.910320890272384 |0.005411214169759696  |
|[Cucumber Kirby]                        |[Banana]                |0.307915457936179  |2.157496492595755 |0.0056627213072274005 |
|[Honeycrisp Apple]                      |[Banana]                |0.34666291185581527|2.4289914558202317|0.009381978370386177  |
|[Broccoli Crown]                        |[Banana]                |0.3154843110504775 |2.210529796465988 |0.00704982127750383   |
+----------------------------------------+----------------------+-------------------+------------------+----------------------+
```

c) Vấn đề và hình thức của các luật

Đa số các luật đều có các hệ quả (***consequent***) là **Bag of Organic Bananas** hoặc là **Banana**.

Nguyên nhân có thể là do 2 sản phẩm này có tần suất xuất hiện rất cao và vượt trội hẳn so với các sản phẩm khác. Theo mẫu phổ biến thu được ở phần trên thì **Bag of Organic Bananas** đứng thứ 2 với tần suất 15480 và **Banana** đứng 1 với tần suất 18726 chênh lệch khá lớn so với sản phẩm đứng 3 là **Organic Strawberries** với tần suất 10894. Điều quan trọng là các luật này chưa hẳn là đã đem lại giá trị cao cần thêm thông số để đánh giá các luật này

Phương pháp khắc phục:

- Cần biết là đa phần các luật sinh ra ở trên đều có **lift** không quá cao (đa phần là dưới 10)

- Về độ đo **lift**: với luật X => Y

  + Nếu **lift** bằng 1 thì X, Y độc lập nhau => luật không tốt, không đáng tin cậy

  + **lift** dương càng lớn thì X, Y càng tương quan dương với nhau => luật càng chính xác

  + **lift** âm càng lớn thì X, Y càng tương quan dương với nhau

  * tương quan dương thể hiện là nếu X xuất hiện thì khả năng cao Y cũng xuất hiện và ngược lại với tương quan âm

- Sử dụng thêm độ đo **lift** để làm ngưỡng lọc ra các luật. Ở đây sẽ chọn ra các luật có **lift** > 10

Tạo các luật kết hợp với ngưỡng thấp min Support = 0.001 và min Confidence = 0.3

```
fpGrowth = FPGrowth(itemsCol="products", minSupport=0.001, minConfidence=0.3)
model = fpGrowth.fit(transaction)
rulesDF = model.associationRules
rulesDF.show(truncate=False)
```

```
/usr/local/lib/python3.7/dist-packages/pyspark/sql/context.py:127: FutureWarning: Deprecated in 3.0.0. Use SparkSession.builder.getOrCrea
  FutureWarning
+----------------------------------------------+----------------------+------------------+------------------+----------------------+
|antecedent                                    |consequent            |confidence        |lift              |support               |
+----------------------------------------------+----------------------+------------------+------------------+----------------------+
|[Yellow Onions, Large Lemon]                  |[Banana]              |0.34404283801874164|2.410633169582456 |0.001958707100884848 |
|[Organic Kiwi, Bag of Organic Bananas]        |[Organic Strawberries]|0.3562691131498471 |4.290959616970652 |0.0017757928190901538|
|[Organic Grape Tomatoes, Organic Strawberries]|[Bag of Organic Bananas]|0.3463611859838275 |2.9357690472708025|0.001958707100884848 |
|[Apple Honeycrisp Organic, Organic Raspberries]|[Bag of Organic Bananas]|0.4307692307692308 |3.651214470284238 |0.0010669999771357147|
|[Yellow Bell Pepper]                          |[Orange Bell Pepper]  |0.33962264150943394|23.906409425864442|0.0030180856496124504|
|[Organic Lemon]                               |[Bag of Organic Bananas]|0.3044222539229672 |2.580293250321615 |0.00813206411145577  |
|[Blueberries, Strawberries]                   |[Banana]              |0.3693467336683417 |2.587932050512092 |0.0011203499759925004|
|[Organic Red Bell Pepper, Organic Hass Avocado]|[Organic Strawberries]|0.32400932400932403|3.9024177890526337|0.0010593785487276026|
|[Organic Red Bell Pepper, Organic Hass Avocado]|[Bag of Organic Bananas]|0.4568764568764569 |3.87250019575601  |0.0014937999679900007|
|[Organic Yellow Onion, Organic Baby Spinach]  |[Bag of Organic Bananas]|0.3114973262032086 |2.640261800632868 |0.0017757928190901538|
|[Asparagus, Limes]                            |[Large Lemon]         |0.407673860911271  |6.575350905507923 |0.0012956428293790822|
|[Organic Garlic, Organic Avocado]             |[Large Lemon]         |0.3415492957746479 |5.5088311677069175|0.0014785571111737763|
|[Organic Garlic, Organic Avocado]             |[Banana]              |0.34683098591549294|2.430169114118654 |0.0015014213963981129|
|[Organic Lemon, Organic Garlic]               |[Bag of Organic Bananas]|0.33624454148471616|2.850020028660732 |0.0011736999748492862|
|[Yellow Onions, Organic Avocado]              |[Banana]              |0.35514018691588783|2.488389874241521 |0.001448071397541327 |
|[Organic Cilantro, Banana]                    |[Limes]               |0.3513513513513137|7.6413823072202   |0.0015852571088873476|
|[Organic Avocado, Organic Baby Spinach]       |[Banana]              |0.34522111269614836|2.4188890834000283|0.0036887713495263284|
|[Organic Fuji Apple, Large Lemon]             |[Banana]              |0.42702702702702705|2.992085292597949 |0.0012041856884817351|
|[Organic Bosc Pear]                           |[Bag of Organic Bananas]|0.3101983002832861 |2.6292512132990753|0.0016690928213765825|
Executing (12s) Cell > evaluate() > _evaluate() > __call__() > send_command() > send_command() > readinto()
```

+ Các luật thu được

Lọc bỏ các luật có lift nhỏ hơn 10

```
rulesDF.filter('lift > 10').show(truncate=False)
```

```
+----------------------------------------------+----------------------------------------------+------------------+------------------+----------------------+
|antecedent                                    |consequent                                    |confidence        |lift              |support               |
+----------------------------------------------+----------------------------------------------+------------------+------------------+----------------------+
|[Yellow Bell Pepper]                          |[Orange Bell Pepper]                          |0.33962264150943394|23.906409425864442|0.0030180856496124504|
|[Total 2% Lowfat Greek Strained Yogurt With Blueberry]|[Total 2% with Strawberry Lowfat Greek Strained Yogurt]|0.3616692426584235 |48.77107878722414 |0.0017834142474982661|
|[Organic Yellow Squash]                       |[Organic Zucchini]                            |0.4677002583979328 |13.372517586431547|0.0013794785418683169|
|[Non Fat Raspberry Yogurt]                    |[Icelandic Style Skyr Blueberry Non-fat Yogurt]|0.3819444444444444 |71.08446611505121 |0.0016767142497846946|
|[Bunched Cilantro, Large Lemon]               |[Limes]                                       |0.4622871046228710 |10.05407404449897 |0.001448071397541327 |
|[Lemon Sparkling Water]                       |[Grapefruit Sparkling Water]                  |0.3130434782608696 |65.19701863354038 |0.0010974856907681638|
|[Nonfat Icelandic Style Strawberry Yogurt]    |[Icelandic Style Skyr Blueberry Non-fat Yogurt]|0.4226519337016574 |78.66062066533442 |0.0011660785464411174 |
|[Blueberry Yoghurt]                           |[Strawberry Rhubarb Yoghurt]                  |0.3102766798418972 |80.29801358062227 |0.0011965642600736 23 |
|[Organic Italian Parsley Bunch, Organic Yellow Onion]|[Organic Garlic]                              |0.353887399463807  |11.167198604195924|0.0010060285498708168|
|[Icelandic Style Skyr Blueberry Non-fat Yogurt]|[Non Fat Raspberry Yogurt]                    |0.3120567375886525 |71.08446611505121 |0.0016767142497846946|
|[Total 2% Lowfat Greek Strained Yogurt with Peach]|[Total 2% with Strawberry Lowfat Greek Strained Yogurt]|0.3524844720496894 |47.5325129426184  |0.0017300642486414804|
|[Strawberry Rhubarb Yoghurt]                  |[Blueberry Yoghurt]                           |0.3096646942800789 |80.29801358062228 |0.0011965642600736 23 |
|[Organic Red Onion, Limes]                    |[Organic Cilantro]                            |0.31666666666666665|11.783754017772736|0.00159287853729546  |
|[Sparkling Lemon Water, Sparkling Water Grapefruit]|[Lime Sparkling Water]                        |0.44984802431610943|30.02243612537762 |0.0011279714044006128|
|[Sparkling Lemon Water, Lime Sparkling Water] |[Sparkling Water Grapefruit]                  |0.45121951219512196|17.62550192783857 |0.0011279714044006128|
|[Non Fat Acai & Mixed Berries Yogurt]         |[Icelandic Style Skyr Blueberry Non-fat Yogurt]|0.4023809523809524 |74.88794663964876 |0.00128802140097097  |
|[Zero Calorie Cola]                           |[Soda]                                        |0.3919308357348703 |34.12399006366065 |0.0010365142635032657|
+----------------------------------------------+----------------------------------------------+------------------+------------------+----------------------+
```

Đánh giá kết quả thu được sao khi áp dụng biện pháp chọn các luật theo **lift**:

- Có thể thấy các luật mới thì gồm nhiều loại sản phẩm khác nhau

- Chỉ số **lift** là rất cao, có những luật lên đến hơn 80

- Không chỉ đúng về mặt số học, về mặt ngữ nghĩa cũng rất hợp lý. Vd: ở dòng đầu **Yellow Bell Pepper** => **Orange Bell Pepper** (mua một loại ớt này thì khả năng cao sẽ mua thêm loại ớt khác); nhiều luật theo dạng **... Yogurt** => **... Yogurt**

=> Phương pháp dùng **lift** để đánh giá luật và chọn luật đem lại hiệu quả và tạo ra được các luật đúng đắn hơn

3.2. Bài 02: Phân lớp

a) Đọc và xử lý dữ liệu:

- Schema của dữ liệu – **mushrooms.csv**

```
[111] df = spark.read.csv('mushrooms.csv', header=True, inferSchema=True)

df.printSchema()

root
 |-- class: string (nullable = true)
 |-- cap-shape: string (nullable = true)
 |-- cap-surface: string (nullable = true)
 |-- cap-color: string (nullable = true)
 |-- bruises: string (nullable = true)
 |-- odor: string (nullable = true)
 |-- gill-attachment: string (nullable = true)
 |-- gill-spacing: string (nullable = true)
 |-- gill-size: string (nullable = true)
 |-- gill-color: string (nullable = true)
 |-- stalk-shape: string (nullable = true)
 |-- stalk-root: string (nullable = true)
 |-- stalk-surface-above-ring: string (nullable = true)
 |-- stalk-surface-below-ring: string (nullable = true)
 |-- stalk-color-above-ring: string (nullable = true)
 |-- stalk-color-below-ring: string (nullable = true)
 |-- veil-type: string (nullable = true)
 |-- veil-color: string (nullable = true)
 |-- ring-number: string (nullable = true)
 |-- ring-type: string (nullable = true)
 |-- spore-print-color: string (nullable = true)
 |-- population: string (nullable = true)
 |-- habitat: string (nullable = true)
```

- Dữ liệu đọc được

```
df.show()
```

```
+-----+---------+-----------+---------+-------+----+--------------+------------+---------+----------+----------+----------+------------------------+---
|class|cap-shape|cap-surface|cap-color|bruises|odor|gill-attachment|gill-spacing|gill-size|gill-color|stalk-shape|stalk-root|stalk-surface-above-ring|sta
+-----+---------+-----------+---------+-------+----+--------------+------------+---------+----------+----------+----------+------------------------+---
|    p|        x|          s|        n|      t|   p|             f|           c|        n|         k|         e|         e|                       s|
|    e|        x|          s|        y|      t|   a|             f|           c|        b|         k|         e|         c|                       s|
|    e|        b|          s|        w|      t|   l|             f|           c|        b|         n|         e|         c|                       s|
|    p|        x|          y|        w|      t|   p|             f|           c|        n|         n|         e|         e|                       s|
|    e|        x|          s|        g|      f|   n|             f|           w|        b|         k|         t|         e|                       s|
|    e|        x|          y|        y|      t|   a|             f|           c|        b|         n|         e|         c|                       s|
|    e|        b|          s|        w|      t|   a|             f|           c|        b|         g|         e|         c|                       s|
|    e|        b|          y|        w|      t|   l|             f|           c|        b|         n|         e|         c|                       s|
|    p|        x|          y|        w|      t|   p|             f|           c|        n|         p|         e|         e|                       s|
|    e|        b|          s|        y|      t|   a|             f|           c|        b|         g|         e|         c|                       s|
|    e|        x|          y|        y|      t|   l|             f|           c|        b|         g|         e|         c|                       s|
|    e|        x|          y|        y|      t|   a|             f|           c|        b|         n|         e|         c|                       s|
|    e|        b|          s|        y|      t|   a|             f|           c|        b|         w|         e|         c|                       s|
|    p|        x|          y|        w|      t|   p|             f|           c|        n|         k|         e|         e|                       s|
|    e|        x|          f|        n|      f|   n|             f|           w|        b|         n|         t|         e|                       s|
|    e|        s|          f|        g|      f|   n|             f|           c|        n|         k|         e|         e|                       s|
|    e|        f|          f|        w|      f|   n|             f|           w|        b|         k|         t|         e|                       s|
|    p|        x|          s|        n|      t|   p|             f|           c|        n|         n|         e|         e|                       s|
|    p|        x|          y|        w|      t|   p|             f|           c|        n|         n|         e|         e|                       s|
|    p|        x|          s|        n|      t|   p|             f|           c|        n|         k|         e|         e|                       s|
+-----+---------+-----------+---------+-------+----+--------------+------------+---------+----------+----------+----------+------------------------+---
only showing top 20 rows
```

- Xét số giá trị phân biệt của từng thuộc tính

```
[114] df.agg(*(f.countDistinct(col(c)).alias(c) for c in df.columns)).show()
```

```
+-----+---------+-----------+---------+-------+----+--------------+------------+-
|class|cap-shape|cap-surface|cap-color|bruises|odor|gill-attachment|gill-spacing|g
+-----+---------+-----------+---------+-------+----+--------------+------------+-
|    2|        6|          4|       10|      2|   9|             2|           2|
+-----+---------+-----------+---------+-------+----+--------------+------------+-
```

=> dữ liệu có 2 class, các thuộc tính còn lại đều có ít giá trị phân biệt, tối đa là 12

- Chia dữ liệu thành tập huấn luyện và tập kiểm thử theo tỉ lệ 80:20

```
train_data, val_data = df.randomSplit([0.8, 0.2], seed=31)
```

b) Huấn luyện với mô hình Decision Tree và Random Forest:

- Chuẩn bị các **Transformer**:

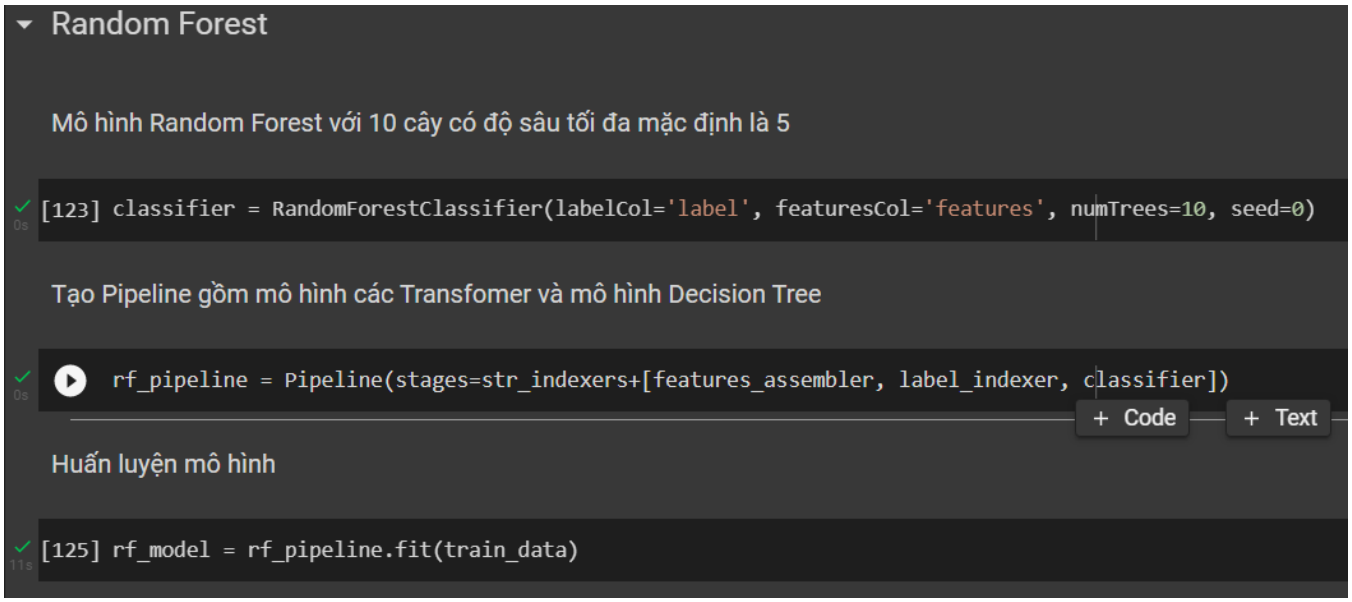
```
Lấy tên các cột feature và cột label

[116] feature_cols = df.columns[1:]
      label_col = df.columns[0]

Sử dụng StringIndexer để đánh index các thuộc tính kiểu categorical (trong dữ liệu này tất cả các thuộc tính đều là categorical)

[117] str_indexers = [StringIndexer(inputCol=c, outputCol=c+'_idx') for c in feature_cols]

Sử dụng VectorAssembler để gôm các thuộc tính thành một vector duy nhất

[118] indexed_cols = [c+'_idx' for c in feature_cols]
      features_assembler = VectorAssembler(inputCols=indexed_cols, outputCol='features')

Sử dụng StringIndexer để đánh index cho label

[119] label_indexer = StringIndexer(inputCol=label_col, outputCol='label')
```

+ Vì tất cả các cột đều là các thuộc tính categorical và các thuật toán phân lớp được cài đặt chỉ áp dụng trên số nên cần sử dụng **StringIndexer** để chuyển các thuộc tính này về số (đánh index cho các giá trị của các thuộc tính này)

+Do cách thức cài đặt các thuật toán phân lớp của **PySpark** nên cần sử dụng thêm **VectorAssembler** để gom các thuộc tính thành một vector duy nhất

- Tạo pipeline gồm các **transformer** ở trên và mô hình **Decision Tree** sau đó huấn luyện trên tập huấn luyện

- Tạo pipeline gồm các **transformer** ở trên và mô hình **Random Forest** sau đó huấn luyện trên tập huấn luyện



c) Đánh giá các mô hình:

- Dữ đoán trên tập kiểm thử

Dự đoán trên tập validation

```
[126] dt_predictions = dt_model.transform(val_data)
      rf_predictions = rf_model.transform(val_data)
```

- Tính toán các độ đo: **MulticlassMetrics** sẽ tính nhiều độ đo khác nhau dựa trên 2 cột **prediction** (dự đoán) và **label** (thực tế), **MulticlassMetrics** áp dụng trên **RDD** nên cần chuyển về **RDD**

- Tính toán các độ đo

```
[ ] dt_pred_label = dt_predictions.select(['prediction','label']).orderBy('prediction')
    dt_metrics = MulticlassMetrics(dt_pred_label.rdd.map(tuple))
    rf_pred_label = rf_predictions.select(['prediction','label']).orderBy('prediction')
    rf_metrics = MulticlassMetrics(rf_pred_label.rdd.map(tuple))
```

- Kết quả đối với mô hình **Decision Tree**:

Decision Tree

```
print("Decision Tree - Test Accuracy = %g" % (dt_metrics.accuracy))
print("Decision Tree - Test Error = %g" % (1.0 - dt_metrics.accuracy))
print("Decision Tree - Test F1 score = %g" % (dt_metrics.weightedFMeasure()))
```

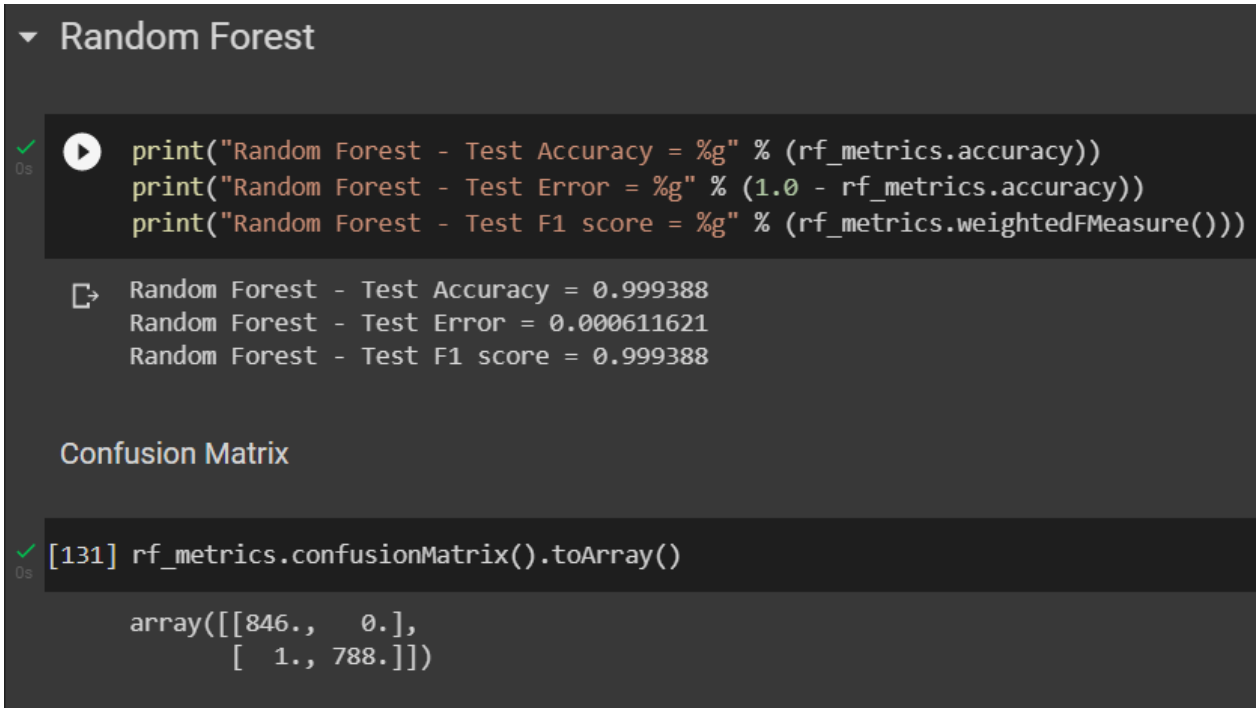
```
Decision Tree - Test Accuracy = 0.998165
Decision Tree - Test Error = 0.00183486
Decision Tree - Test F1 score = 0.998165
```

Confusion Matrix

```
[129] dt_metrics.confusionMatrix().toArray()
```

```
array([[846.,   0.],
       [  3., 786.]])
```

- Kết quả đối với mô hình **Random Forest**:



- Nhận xét:

+ Cả 2 mô hình đều cho kết quả rất cao

+ Mô hình **Random Forest** cho kết quả tốt hơn một chút so với **Decision Tree**

3.3. Bài 3 – Gom cụm

a) Đọc và xử lý dữ liệu:

- Đọc file **plans.data** vào **RDD** để xử lý:

- Đọc dữ liệu vào RDD

```
[49] rdd = spark.sparkContext.textFile('plants.data')
```

- Mỗi dòng dữ liệu sẽ là mỗi row của RDD

```
[50] rdd.take(5)
```

```
['abelia,fl,nc',
 'abelia x grandiflora,fl,nc',
 'abelmoschus,ct,dc,fl,hi,il,ky,la,md,mi,ms,nc,sc,va,pr,vi',
 'abelmoschus esculentus,ct,dc,fl,il,ky,la,md,mi,ms,nc,sc,va,pr,vi',
 'abelmoschus moschatus,hi,pr']
```

- Tách mỗi dòng của **RDD** thành 1 tuple gồm 2 phần tử: tên plant và một list các state

- Tách các dòng thành 2 phần: tên plants và list các state

```
[51] splited_rdd = rdd.map(lambda line: line.split(',')).map(lambda line: (line[0], line[1:]))
     splited_rdd.take(3)
```

```
[('abelia', ['fl', 'nc']),
 ('abelia x grandiflora', ['fl', 'nc']),
 ('abelmoschus',
  ['ct',
   'dc',
   'fl',
   'hi',
   'il',
   'ky',
   'la',
   'md',
   'mi',
   'ms',
   'nc',
   'sc',
   'va',
   'pr',
   'vi'])]
```

- Chuyển về dạng **Dataframe**

```
[52] df = splited_rdd.toDF(['plant', 'states'])

df.printSchema()

root
 |-- plant: string (nullable = true)
 |-- states: array (nullable = true)
 |    |-- element: string (containsNull = true)

[54] df.show(truncate=False)

+------------------------------+------------------------------------------------------------------------
|plant                         |states
+------------------------------+------------------------------------------------------------------------
|abelia                        |[fl, nc]
|abelia x grandiflora          |[fl, nc]
|abelmoschus                   |[ct, dc, fl, hi, il, ky, la, md, mi, ms, nc, sc, va, pr, vi]
|abelmoschus esculentus        |[ct, dc, fl, il, ky, la, md, mi, ms, nc, sc, va, pr, vi]
|abelmoschus moschatus         |[hi, pr]
|abies                         |[ak, az, ca, co, ct, ga, id, in, ia, me, md, ma, mi, mn, mt, nv, nh, nm
|abies alba                    |[nc]
|abies amabilis                |[ak, ca, or, wa, bc]
|abies balsamea                |[ct, in, ia, me, md, ma, mi, mn, nh, ny, oh, pa, ri, vt, va, wv, wi, ab
|abies balsamea var. balsamea  |[ct, in, ia, me, md, ma, mi, mn, nh, ny, oh, pa, ri, vt, va, wv, wi, ab
|abies balsamea var. phanerolepis|[me, nh, vt, va, wv, nb, nf, ns, on, pe, qc]
|abies bracteata               |[ca]
|abies concolor                |[az, ca, co, id, me, ma, nv, nm, or, ut, wy]
|abies concolor var. concolor  |[az, co, id, me, ma, nv, nm, or, ut, wy]
|abies concolor var. lowiana   |[ca, nv, or]
|abies fraseri                 |[ga, nc, tn, va]
|abies grandis                 |[ca, id, mt, or, wa, bc]
|abies homolepis               |[ny]
|abies lasiocarpa              |[ak, az, ca, co, id, mt, nv, nm, or, ut, wa, wy, ab, bc, nt, yt]
|abies lasiocarpa var. arizonica |[az, co, nm]
```

Các bước tiếp theo sẽ chuyển dữ liệu về dạng tên cột là tên tất cả các state: giá trị của mỗi dòng (tương ứng với mỗi plant) tại mỗi cột sẽ là 0 hoặc 1 (cột ứng với state có trong cột states - list các state, sẽ có giá trị 1)

- Đầu tiên, sử dụng hàm **explode** với cột state để tách list các tách thành nhiều dòng, mỗi dòng là 1 state

- Explode cột states

```
exploded_df = df.withColumn('state', f.explode(df.states))
exploded_df = exploded_df.select('plant', 'state')
exploded_df.show()
```

```
+--------------------+-----+
|               plant|state|
+--------------------+-----+
|              abelia|   fl|
|              abelia|   nc|
|abelia x grandiflora|   fl|
|abelia x grandiflora|   nc|
|         abelmoschus|   ct|
|         abelmoschus|   dc|
|         abelmoschus|   fl|
|         abelmoschus|   hi|
|         abelmoschus|   il|
|         abelmoschus|   ky|
|         abelmoschus|   la|
|         abelmoschus|   md|
|         abelmoschus|   mi|
|         abelmoschus|   ms|
|         abelmoschus|   nc|
|         abelmoschus|   sc|
|         abelmoschus|   va|
|         abelmoschus|   pr|
|         abelmoschus|   vi|
|abelmoschus escul...|   ct|
+--------------------+-----+
only showing top 20 rows
```

- **Groupby** cột **plant** và sử dụng **pivot** sẽ thu được bảng dữ liệu gồm tên cột là tất cả các state và mỗi dòng nhận giá trị 1 tại 1 tại state mà plant đó xuất hiện

```
pivot_df = exploded_df.groupBy("plant").pivot("state").count()
pivot_df.show()
```

```
+--------------------+----+----+----+----+----+----+----+----+----+----+----+-----+----+------+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|               plant|  ab|  ak|  al|  ar|  az|  bc|  ca|  co|  ct|  dc|  de|dengl|  fl|fraspm|  ga|  gl|  hi|  ia|  id|  il|  in|  ks|  ky|  la|  lb|  ma|  mb|  md|  me|  mi|  mn|
+--------------------+----+----+----+----+----+----+----+----+----+----+----+-----+----+------+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
|           achnella|   1|null|null|null|null|   1|null|null|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
|     cannabis sativa|null|null|   1|   1|   1|null|   1|   1|   1|   1|   1| null|   1|  null|   1|null|   1|   1|   1|   1|   1|   1|   1|null|   1|null|   1|null|   1|null|   1|null|
|   carex heleonastes|   1|   1|null|null|null|   1|null|null|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|   1|null|null|null|null|null|null|null|
|chaenactis suffru...|null|null|null|null|null|null|   1|null|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
|   crassula solierii|null|null|null|null|null|null|   1|null|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
|        gymnocladus|null|null|   1|   1|null|null|null|null|   1|null|   1| null|null|  null|   1|null|null|   1|null|   1|   1|   1|   1|null|null|   1|null|null|null|null|null|null|
|houstonia canadensis|null|null|null|null|null|null|null|null|null|null|null| null|null|  null|   1|null|null|null|null|   1|   1|null|   1|null|null|null|null|null|null|null|null|
|     hypericum majus|   1|null|null|null|null|   1|null|   1|   1|null|   1| null|null|  null|null|null|null|   1|   1|   1|   1|null|null|null|   1|null|null|null|null|null|null|
|   ipomoea turbinata|null|null|null|   1|null|null|null|null|null|null|null| null|   1|  null|   1|null|null|null|null|null|null|null|null|   1|null|null|null|null|null|null|null|
|juniperus communi...|   1|   1|   1|null|   1|   1|   1|   1|null|   1|null| null|null|     1|   1|null|null|   1|   1|   1|   1|null|   1|null|   1|   1|   1|   1|   1|   1|   1|
|   abutilon parvulum|null|null|null|null|   1|null|   1|   1|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
|           chilopsis|null|null|null|null|   1|null|   1|null|null|null|null| null|null|  null|   1|null|null|null|null|null|null|null|null|   1|null|null|null|null|null|null|null|
|           boerhavia|null|null|   1|   1|   1|null|   1|   1|null|null|null| null|   1|  null|   1|null|   1|null|null|null|null|null|null|   1|null|null|null|null|null|null|null|
|  cenchrus echinatus|null|null|   1|null|   1|null|   1|null|null|   1|null| null|   1|  null|   1|null|   1|null|null|null|null|null|null|   1|null|null|null|null|null|null|null|
|evolvulus sericeu...|null|null|null|   1|   1|null|null|null|null|null|null| null|   1|  null|   1|null|null|null|null|null|null|null|null|   1|null|null|null|null|null|null|null|
|allium schoenoprasum|   1|   1|null|null|null|   1|null|   1|   1|null|null| null|null|  null|null|null|null|null|null|   1|   1|   1|null|null|   1|   1|   1|   1|   1|   1|   1|
|anemone multifida...|null|null|null|null|   1|null|null|null|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
|anemone narcissif...|null|   1|null|null|null|   1|null|   1|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
|             cojoba|null|null|null|null|null|null|null|null|null|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
|      angelica grayi|null|null|null|null|null|null|null|null|   1|null|null| null|null|  null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|null|
+--------------------+----+----+----+----+----+----+----+----+----+----+----+-----+----+------+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+----+
```

- Thay các giá trị null bằng 0 sẽ thu được bảng dữ liệu theo mong muốn

```
pivot_df = pivot_df.na.fill(0)
pivot_df.show()
```

```
+--------------------+---+---+---+---+---+---+---+---+---+---+---+-----+---+------+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|               plant| ab| ak| al| ar| az| bc| ca| co| ct| dc| de|dengl| fl|fraspm| ga| gl| hi| ia| id| il| in| ks| ky| la| lb| ma| mb| md| me| mi| mn|
+--------------------+---+---+---+---+---+---+---+---+---+---+---+-----+---+------+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|           achnella|  1|  0|  0|  0|  0|  1|  0|  0|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  0|
|     cannabis sativa|  0|  0|  1|  1|  1|  0|  1|  1|  1|  1|  1|    0|  1|     0|  1|  0|  1|  1|  1|  1|  1|  1|  0|  1|  1|  1|  1|  1|  1|  1|
|   carex heleonastes|  1|  1|  0|  0|  0|  1|  0|  0|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  1|  0|  1|  0|  0|  1|  0|  0|
|chaenactis suffru...|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|   crassula solierii|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|        gymnocladus|  0|  0|  1|  1|  0|  0|  0|  0|  1|  0|  1|    0|  0|     0|  1|  0|  0|  1|  0|  1|  1|  1|  1|  0|  0|  1|  0|  1|  1|  1|  1|
|houstonia canadensis|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|    0|  0|     0|  1|  0|  0|  0|  0|  1|  1|  0|  1|  0|  0|  0|  0|  0|  1|  1|  1|
|     hypericum majus|  1|  0|  0|  0|  0|  1|  0|  1|  1|  0|  1|    0|  0|     0|  0|  0|  0|  0|  1|  1|  1|  1|  1|  0|  0|  0|  1|  1|  0|  1|  1|
|   ipomoea turbinata|  0|  0|  0|  1|  0|  0|  0|  0|  0|  0|  0|    0|  1|     0|  1|  0|  0|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  0|  0|  0|  0|
|juniperus communi...|  1|  1|  1|  0|  1|  1|  1|  1|  1|  0|  1|    0|  0|     1|  1|  0|  0|  1|  1|  1|  1|  0|  1|  0|  1|  1|  1|  1|  1|  1|  1|
|   abutilon parvulum|  0|  0|  0|  0|  1|  0|  1|  1|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|           chilopsis|  0|  0|  0|  0|  1|  0|  1|  0|  0|  0|  0|    0|  0|     0|  1|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|           boerhavia|  0|  0|  1|  1|  1|  0|  1|  1|  0|  0|  0|    0|  1|     0|  1|  0|  1|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  1|  0|  0|  0|
|  cenchrus echinatus|  0|  0|  1|  0|  1|  0|  1|  0|  0|  1|  0|    0|  1|     0|  1|  0|  1|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  1|  0|  0|  0|
|evolvulus sericeu...|  0|  0|  0|  1|  1|  0|  0|  0|  0|  0|  0|    0|  1|     0|  1|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|allium schoenoprasum|  1|  1|  0|  0|  0|  1|  0|  1|  1|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  1|  1|  1|  0|  0|  0|  1|  1|  1|  1|  1|  1|
|anemone multifida...|  0|  0|  0|  0|  1|  0|  0|  0|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|anemone narcissif...|  0|  1|  0|  0|  0|  1|  0|  1|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|             cojoba|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|    0|  1|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|
|      angelica grayi|  0|  0|  0|  0|  0|  0|  0|  1|  0|  0|  0|    0|  0|     0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|  0|
+--------------------+---+---+---+---+---+---+---+---+---+---+---+-----+---+------+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
```

- Xuất ra file csv: **file plants.csv sẽ được lưu trong thư mục Data khi nộp**

- Xuất ra thành file plants.csv

```
[58] pivot_df.toPandas().to_csv('plants.csv', index=False)
```

b) Áp dụng thuật toán K-Means và đánh giá kết quả:

- Tiền xử lý dữ liệu: gôm các cột state thành một vector duy nhất bằng **VectorAssembler**

```
• Dùng VectorAssembler để gom tất cả các thuộc tính thành 1 vector

[59] feature_cols = pivot_df.columns[1:]

[60] features_assembler = VectorAssembler(inputCols=feature_cols, outputCol='features')

[61] km_df = features_assembler.transform(pivot_df).select('plant', 'features')
```

- Chạy thuật toán **K-Means** với số cụm khác nhau và đánh giá kết quả bằng chỉ số **Silhouette**

```
• Chạy và đánh giá thuật toán KMeans với số cụm khác nhau

ks = [2, 3, 4, 5, 10, 20, 50, 200]

for k in ks:
  kmeans = KMeans(featuresCol='features', k=k)
  model = kmeans.fit(km_df)
  predictions = model.transform(km_df)
  evaluator = ClusteringEvaluator()
  silhouette = evaluator.evaluate(predictions)
  print(f"Silhouette Score với {k} cụm = {silhouette}")

Silhouette Score với 2 cụm = 0.7181780188904239
Silhouette Score với 3 cụm = 0.5853974121439827
Silhouette Score với 4 cụm = 0.5244290483240194
Silhouette Score với 5 cụm = 0.5057257370896932
Silhouette Score với 10 cụm = 0.27407395108643184
Silhouette Score với 20 cụm = 0.2324589364475111
Silhouette Score với 50 cụm = 0.3072368177634605
Silhouette Score với 200 cụm = 0.40052822119475195
```

- Nhận xét:

+ Về cách đánh giá: sử dụng chỉ số **Silhouette**: nhận giá trị trong khoảng [-1, 1], chỉ số **Silhouette** càng gần 1 thì chất lượng của các cụm càng tốt

+ Trong các thử nghiệm, chỉ số **Silhouette** đạt cao nhất với số cụm = 2 với giá trị **0.718**, là một ngưỡng rất tốt

+ Khi số cụm tăng dần thì chất lượng các cụm cũng giảm dần cho đến khi số cụm đủ lớn (> 20) thì chất lượng được tăng lên lại

4. Tài liệu tham khảo:

1. API Reference — PySpark 3.2.0 documentation (apache.org)
2. PySpark Tutorial For Beginners | Python Examples — Spark by {Examples} (sparkbyexamples.com)