

## Frequent Pattern Mining

### FP-Growth

FP-growth operates on *itemsets*. An itemset is an unordered collection of unique items. Spark does not have a *set* type, so itemsets are represented as arrays.

spark.ml's FP-growth implementation takes the following (hyper-)parameters:

- **minSupport**: the minimum support for an itemset to be identified as frequent. For example, if an item appears 3 out of 5 transactions, it has a support of  $3/5=0.6$ .
- **minConfidence**: minimum confidence for generating Association Rule. Confidence is an indication of how often an association rule has been found to be true. For example, if in the transactions itemset X appears 4 times, X and Y co-occur only 2 times, the confidence for the rule  $X \Rightarrow Y$  is then  $2/4 = 0.5$ . The parameter will not affect the mining for frequent itemsets, but specify the minimum confidence for generating association rules from frequent itemsets.
- **numPartitions**: the number of partitions used to distribute the work. By default the param is not set, and number of partitions of the input dataset is used.

The FPGrowthModel provides:

- **freqItemsets**: frequent itemsets in the format of a DataFrame with the following columns:
  - items: array: A given itemset.
  - freq: long: A count of how many times this itemset was seen, given the configured model parameters.

- **associationRules**: association rules generated with confidence above minConfidence, in the format of a DataFrame with the following columns:
  - antecedent: array: The itemset that is the hypothesis of the association rule.
  - consequent: array: An itemset that always contains a single element representing the conclusion of the association rule.
  - confidence: double: Refer to minConfidence above for a definition of confidence.
  - lift: double: A measure of how well the antecedent predicts the consequent, calculated as  $\text{support}(\text{antecedent} \cup \text{consequent}) / (\text{support}(\text{antecedent}) \times \text{support}(\text{consequent}))$
  - support: double: Refer to minSupport above for a definition of support.
- **transform**: For each transaction in itemsCol, the transform method will compare its items against the antecedents of each association rule. If the record contains all the antecedents of a specific association rule, the rule will be considered as applicable and its consequents will be added to the prediction result. The transform method will summarize the consequents from all the applicable rules as prediction. The prediction column has the same data type as itemsCol and does not contain existing items in the itemsCol.

**Example :-**

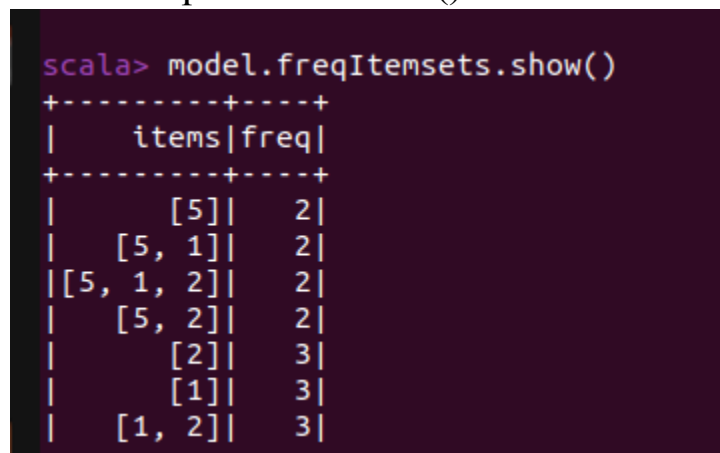
```
import org.apache.spark.ml.fpm.FPGrowth
```

```
val dataset = spark.createDataset(Seq(
  "1 2 5",
  "1 2 3 5",
  "1 2")
).map(t => t.split(" ")).toDF("items")
```

```
val fpgrowth = new FPGrowth().setItemsCol("items").
setMinSupport(0.5).setMinConfidence(0.6)
```

```
val model = fpgrowth.fit(dataset)
```

```
// Display frequent itemsets.
model.freqItemsets.show()
```



```
scala> model.freqItemsets.show()
+-----+-----+
|   items|freq|
+-----+-----+
|    [5]|  2|
|  [5, 1]|  2|
|[5, 1, 2]|  2|
|  [5, 2]|  2|
|    [2]|  3|
|    [1]|  3|
|  [1, 2]|  3|
```

```
// Display generated association rules.
model.associationRules.show()
```

```
scala> model.associationRules.show()
+-----+-----+-----+-----+
|antecedent|consequent|confidence|lift|support|
+-----+-----+-----+-----+
| [2]| [5]|0.6666666666666666|1.0|0.6666666666666666|
| [2]| [1]|1.0|1.0|1.0|
| [5, 2]| [1]|1.0|1.0|0.6666666666666666|
| [1, 2]| [5]|0.6666666666666666|1.0|0.6666666666666666|
| [5, 1]| [2]|1.0|1.0|0.6666666666666666|
| [5]| [1]|1.0|1.0|0.6666666666666666|
| [5]| [2]|1.0|1.0|0.6666666666666666|
| [1]| [5]|0.6666666666666666|1.0|0.6666666666666666|
| [1]| [2]|1.0|1.0|1.0|
+-----+-----+-----+-----+
```

```
// transform examines the input items against all the association rules
and summarize the
// consequents as prediction
model.transform(dataset).show()
```

```
scala> model.transform(dataset).show()
+-----+-----+
| items|prediction|
+-----+-----+
| [1, 2, 5]| []|
| [1, 2, 3, 5]| []|
| [1, 2]| [5]|
+-----+-----+
```



<https://spark.apache.org/docs/latest/ml-frequent-pattern-mining.html>