|  |
| --- |
| **Documentation for Jeff base table mapping to parent-ASIN** |

|  |  |  |
| --- | --- | --- |
| **Item** | **Description** | **Comments** |
| Current Scenario | In the Jeff base table, inventory status and action item bucket are mapped on child-ASIN x MP level and not on parent-ASIN x MP level |  |
| Goal | Create an output table which has ASIN x MP x week-year from the Jeff table, mapped to the corresponding parent-ASIN, dominant inventory status and dominant action item bucket | Dominant inventory status and domination action item bucket act as proxies for inventory status and action item bucket of parent-ASIN |
| Input tables | rgbit\_coupon\_jeff\_base\_v2 | Jeff base table |
| child\_parent\_asin\_mapping | Parent-Child ASIN mapping |
| temp\_action\_item\_mixed\_bucket\_mapping | Used to obtain action bucket |
| tech\_tables.tech\_asin\_country\_orders\_marketing\_data\_fbmfba\_final | Net revenue used in logic for identifying dominant inventory status and dominant action item |
| Output table | temp\_parent\_dom\_child\_inv\_and\_action\_updated\_tester |  |
| Github | [Link](https://github.com/ravindra-sagar-razor/dominant-child-mapper/blob/aa5a1e6634d94e37b3bf8e300d2e69efc14e7e47/Dom_child_script_updated.sql) |  |

**Key terms**

1. Inventory status: Obtained from the Jeff base table in the field ‘inventory\_bucket’. Indicates the inventory health on ASIN X MP level. Can be “OOS”/“Overstock”/”Healthy Stock”/”Low Stock”.
2. Action item bucket: Obtained by mapping ‘mixed\_bucket’ field in Jeff base table to ‘temp\_action\_item\_mixed\_bucket\_mapping’ table. Indicates the ASIN nature on ASIN X MP level. Can be “Cash\_in”/“Margin %”/”Margin ABS”.
3. Dominant inventory status: Acts as a proxy for the parent-ASIN inventory status. A detailed explanation of the logic used for identifying the dominant inventory status is given in the ‘Notes’ section.
4. Dominant action item bucket: Acts as a proxy for the parent-ASIN action item bucket. A detailed explanation of the logic used for identifying the dominant action item bucket is given in the ‘Notes’ section.

**Notes**

1. Input ASINs: Only those ASIN x MP x week-year combinations in the Jeff base table, that have week-year as the latest week-year in the Jeff base table are considered for the mapping
2. Logic for identifying dominant inventory :
   1. For identifying the dominant inventory status, weighted net revenue (Net revenue earned recently will have a higher weightage than net revenue earned at an older point in time) is used.
   2. The current iteration of the code uses linear weights for calculating weighted net revenue i.e. net revenue earned within the L30D has a weightage of 1, and this weightage decreases linearly every 30 days, till the weightage of net revenue earned 360 days before the ‘final\_date’ month becomes zero.
   3. The weighted net revenue on a week-year x parent-ASIN x MP x child-ASIN-inventory-status (The child-ASIN-inventory-status is the set of all inventory statuses of the child-ASINs for a given week-year x parent-ASIN x MP combination) level is calculated
   4. The week-year x parent-ASIN x MP x child-ASIN-inventory-status which has the highest corresponding weighted net revenue is taken as the dominant inventory status

Exceptions for OOS:

* + 1. In case the inventory status with the highest weighted net revenue is ‘OOS’ for a given week-year x parent-ASIN x MP combination, and there are other inventory statuses available for the combination, the inventory status having the next highest weighted net revenue is considered as the dominant inventory status
    2. In case the inventory status with the highest weighted net revenue is ‘OOS’ for a given week-year x parent-ASIN x MP combination and it is the only available inventory status for the combination, ‘OOS’ is taken as the dominant inventory status
    3. In case there is a tie in weighted net revenue between two or more inventory statuses for a given week-year x parent-ASIN x MP combination, inventory status hierarchy (Over stock > Healthy stock > Low stock > OOS ) is used to identify the dominant inventory status

1. Logic for identifying dominant action item:
   1. Similar to dominant inventory status, for identifying the dominant action item, weighted net revenue (Net revenue earned recently will have a higher weightage than net revenue earned at an older point in time) is used.
   2. The weighted net revenue on a week-year x parent-ASIN x MP x child-ASIN-action-item-bucket (The child-ASIN-action-item-bucket is the set of all action item buckets of the child-ASINs for a given week-year x parent-ASIN x MP combination) level is calculated
   3. The week-year x parent-ASIN x MP x child-ASIN action item which has the highest corresponding weighted net revenue is taken as the dominant action item

Exceptions:

* + 1. In case there is a tie in weighted net revenue between two or more action item buckets for a given week-year x parent-ASIN x MP combination, action item bucket hierarchy (Cash-in > Margin ABS > Margin %) is used to identify the dominant action item bucket

1. Output table:
   1. Methodology: Input ASIN x MP combinations are mapped to parent-ASIN x MP from the parent-child mapping table. Then parent-ASIN x MP x week-year combinations are mapped to the corresponding dominant inventory status and dominant action item
   2. Only those ASIN x MP x Week-year combinations in the Jeff base table, that haveweek-year as the latest week-year in the Jeff base table will be available in the output table as only these combinations are considered in the input
   3. The output will be unique on ASIN x MP x week-year level
   4. The dominant inventory status and dominant action item bucket for a given parent-ASIN x MP may vary with the week-year as the weighted net revenue for a child-ASIN x MP varies with the week-year
   5. The explanation for columns in the output table is given below

|  |  |  |
| --- | --- | --- |
| **temp\_parent\_dom\_child\_inv\_and\_action\_updated\_tester** | | |
| **Field** | **Description** | **Source** |
| final\_date | Last day of week-year | Jeff base table |
| week\_year | Year and week no combination | Jeff base table |
| asin | Child-ASIN | Jeff base table |
| parent\_asin | Parent-ASIN | Parent-child ASIN mapping table |
| country\_code | Market place | Jeff base table |
| inventory\_bucket | Child-ASIN inventory status | Jeff base table |
| action\_item\_bucket | Child-ASIN action item bucket | Jeff base table |
| dominant\_inventory\_bucket | Dominant Inventory bucket with OOS exclusion\* | Calculated |
| primary\_dom\_inv\_bucket\_oos | Dominant inventory bucket without OOS exclusion\* | Calculated |
| dominant\_action\_item\_bucket | Dominant action item bucket | Calculated |
| child\_TTM\_net\_revenue | TTM net revenue of child ASIN | Orders table |

\*OOS exclusion explained in detail in ‘Notes’ section -> Logic for identifying dominant inventory -> Exception for OOS

**Working code**

**drop** **table** **if** **exists** temp\_parent\_dom\_child\_inv\_and\_action\_updated\_tester **as**

**create** **table** temp\_parent\_dom\_child\_inv\_and\_action\_updated\_tester **as**

**with** parent\_child\_map **as** (

**select** **distinct** child.final\_date

,child.week\_year

,child.**asin**

,**nvl**(parent.parent\_asin,child.**asin**) **as** parent\_asin

,child.country\_code

,child.inventory\_bucket

, **case**

**when** child.inventory\_bucket = 'Overstock' **then** 1

**when** child.inventory\_bucket = 'Healthy Stock' **then** 2

**when** child.inventory\_bucket = 'Low Stock' **then** 3

**else** 4

**end** **as** inventory\_status\_heirarchy

,child.action\_item\_bucket

, **case**

**when** child.action\_item\_bucket = 'Cash\_in' **then** 1

**when** child.action\_item\_bucket = 'Margin ABS' **then** 2

**else** 3

**end** **as** action\_item\_heirarchy

**from** (**select** a.final\_date, a.week\_year, a.**asin**, a.country\_code, a.inventory\_bucket, b.action\_item\_bucket, **dense\_rank**() **over** (**order** **by** week\_year **desc**) **as** week\_rank **from** rgbit\_coupon\_jeff\_base\_v2 **as** a **left** **join** temp\_action\_item\_mixed\_bucket\_mapping **as** b **on** a.mixed\_bucket = b.mixed\_bucket) **as** child

**left** **join**(

**select** "child asin" **as** child\_asin

, "parent asin" **as** parent\_asin

, marketplace

**from** child\_parent\_asin\_mapping) **as** parent

**on** child.**asin** = parent.child\_asin **and** child.country\_code = parent.marketplace

**where** week\_rank =1

)

, dominant\_inventory\_status **as**(

**select** week\_year

, parent\_asin

, country\_code

, inventory\_bucket

, inventory\_status\_rank

, lag\_inventory\_status

, inventory\_weighted\_revenue

**from**(

**select** week\_year

, parent\_asin

, country\_code

, inventory\_bucket

, inventory\_weighted\_revenue

, inventory\_status\_rank

, lead\_inventory\_status

, lag\_inventory\_status

, **case**

**when** inventory\_status\_rank = 1 **and** inventory\_bucket!= 'OOS' **then** 1

**when** inventory\_status\_rank = 1 **and** inventory\_bucket = 'OOS' **and** lead\_inventory\_status **is** **null** **then** 1

**when** inventory\_status\_rank = 2 **and** inventory\_bucket!= 'OOS' **and** lag\_inventory\_status = 'OOS' **then** 1

**else** 0

**end** **as** dominant\_inventory\_status\_flag

**from**(

**select** \*

, **rank**() **over** (**partition** **by** week\_year, country\_code, parent\_asin **order** **by** inventory\_weighted\_revenue, -1\*inventory\_status\_heirarchy **desc**) **as** inventory\_status\_rank

, **lead** (inventory\_bucket) **over** (**partition** **by** week\_year, country\_code, parent\_asin **order** **by** inventory\_weighted\_revenue, -1\*inventory\_status\_heirarchy **desc**) **as** lead\_inventory\_status

, **lag** (inventory\_bucket) **over** (**partition** **by** week\_year, country\_code, parent\_asin **order** **by** inventory\_weighted\_revenue, -1\*inventory\_status\_heirarchy **desc**) **as** lag\_inventory\_status

**from**(

**select** week\_year

, parent\_asin

, country\_code

, inventory\_bucket

, inventory\_status\_heirarchy

, **sum**(net\_revenue\*weight) **as** inventory\_weighted\_revenue

**from**(

**select** a.week\_year

, a.parent\_asin

, a.country\_code

, a.inventory\_bucket

, a.inventory\_status\_heirarchy

, a.action\_item\_bucket

, a.action\_item\_heirarchy

, b.net\_revenue

, 1 - **round**((DATEDIFF(**day**, b.final\_date, a.final\_date)/30),1) **AS** weight

**from** parent\_child\_map **as** a

**left** **join** tech\_tables.tech\_asin\_country\_orders\_marketing\_data\_fbmfba\_final **as** b

**on** a.**asin** = b.**asin** **and** a.country\_code = b.country\_code

**where** b.final\_date >= DATEADD(**day**, -360, a.final\_date) **and** DATEDIFF(**day**, b.final\_date, a.final\_date) >=0)

**group** **by** week\_year

, parent\_asin

, country\_code

, inventory\_bucket

, inventory\_status\_heirarchy

)

)

)

**where** dominant\_inventory\_status\_flag = 1

)

, dominant\_action\_item **as** (

**select**\*

**from**(

**select** \*

, **rank**() **over** (**partition** **by** week\_year, country\_code, parent\_asin **order** **by** action\_item\_weighted\_revenue, -1\*action\_item\_heirarchy **desc**) **as** dominant\_action\_rank

**from**(

**select** week\_year

, parent\_asin

, country\_code

, action\_item\_bucket

, action\_item\_heirarchy

, **sum**(net\_revenue\*weight) **as** action\_item\_weighted\_revenue

**from**(

**select** a.week\_year

, a.parent\_asin

, a.country\_code

, a.inventory\_bucket

, a.inventory\_status\_heirarchy

, a.action\_item\_bucket

, a.action\_item\_heirarchy

, b.net\_revenue

, 1 - **cast**((DATEDIFF(**day**, b.final\_date, a.final\_date)/30) **as** **int**)/12 **AS** weight

**from** parent\_child\_map **as** a

**left** **join** tech\_tables.tech\_asin\_country\_orders\_marketing\_data\_fbmfba\_final **as** b

**on** a.**asin** = b.**asin** **and** a.country\_code = b.country\_code

**where** b.final\_date >= DATEADD(**day**, -360, a.final\_date) **and** DATEDIFF(**day**, b.final\_date, a.final\_date) >=0)

**group** **by** week\_year

, parent\_asin

, country\_code

, action\_item\_bucket

, action\_item\_heirarchy

)

)

**where** dominant\_action\_rank = 1

)

**select** g.final\_date

,g.week\_year

,g.**asin**

,g.parent\_asin

,g.country\_code

,g.inventory\_bucket

,g.action\_item\_bucket

,g.dominant\_inventory\_bucket

,**case**

**when** primary\_dom\_inv\_bucket\_OOS **is** **null** **then** g.dominant\_inventory\_bucket

**else** primary\_dom\_inv\_bucket\_OOS

**end** **as** primary\_dom\_inv\_bucket\_OOS

,h.child\_TTM\_net\_revenue

**from**(

**select** c.\*

, d.inventory\_bucket **as** dominant\_inventory\_bucket

, d.lag\_inventory\_status **as** primary\_dom\_inv\_bucket\_OOS

, d.action\_item\_bucket **as** dominant\_action\_item\_bucket

**from** parent\_child\_map **as** c

**left** **join** (

**select** a.\*

, b.inventory\_bucket

, b.inventory\_status\_rank

, b.inventory\_weighted\_revenue

, lag\_inventory\_status

**from** dominant\_action\_item **as** a

**left** **join** dominant\_inventory\_status **as** b

**on** a.parent\_asin = b.parent\_asin **and** a.country\_code = b.country\_code **and** a.week\_year = b.week\_year

) **as** d

**on** c.parent\_asin = d.parent\_asin **and** c.country\_code = d.country\_code **and** c.week\_year = d.week\_year

) **as** g

**left** **join** (

**select** e.**asin**

,e.country\_code

,e.final\_date

, **sum**(net\_revenue) **as** child\_TTM\_net\_revenue

**from** parent\_child\_map **as** e

**left** **join** tech\_tables.tech\_asin\_country\_orders\_marketing\_data\_fbmfba\_final **as** f

**on** e.**asin** = f.**asin** **and** e.country\_code = f.country\_code

**where** f.final\_date >= DATEADD(**month**, -12, e.final\_date) **and** DATEDIFF(**day**, f.final\_date, e.final\_date) >=0

**group** **by** e.**asin**

, e.country\_code

,e.final\_date

) **as** h

**on** g.**asin** = h.**asin** **and** g.final\_date = h.final\_date **and** g.country\_code = h.country\_code

**order** **by** g.week\_year, g.**asin**, g.parent\_asin, g.country\_code

**Code Explanation**

Step 1: Creating CTE ‘parent\_child\_map’**.** This gives the ASINs x MP x week-year combinations which need mapping.

Step 2: Creating CTE ‘dominant\_inventory\_status’**.** Gives the dominant inventory status for a given parent-ASINs x MP x week-year combination.

Step 3: Creating CTE ‘dominant\_action\_item’**.** Gives the dominant action item for a given parent-ASINs x MP x week-year combination.

Step 4: Creating final output table

1. Mapping ‘parent\_child\_map’ , ‘dominant\_inventory\_status’ and ‘dominant\_inventory\_action’
2. Mapping the TTM net revenue to ASINs