# Forecast Disaccumulation: Step Description

Forecast Disaccumulation step is aimed to provide forecasts at the required time granularity level.

This step is needed if forecast granularity from hybridization step differs from the target granularity regarding time dimension.

Regarding Service Model this step corresponds to the 1.6.2 Forecast Disaggregation service.

## Code Realization Requirements

The code should be created in a form of a SAS macro set on SAS VIYA platform.

# Input Data

The initial data for the forecast flag definition are listed below. These tables should be present in the system before this step for example in STG or in DDS area.

* IN\_PRODUCT (see data requirements)
* IN\_LOCATION (see data requirements)
* IN\_CUSTOMER (see data requirements)
* IN\_DISTR\_CHANNEL (see data requirements)

## AGG\_HYB\_FCST\_<fcst\_type>\_<target var postfix>

Forecast values from VF forecast step that is disaggregated to product/location level and (dis)accumulated to daily granularity.

|  |  |
| --- | --- |
| **AGG\_HYB\_FCST** | |
| Column Name | Description |
| **product\_lvl\_id<m>** | ID of product group for which specific ML model will be trained, m= **out\_product\_lvl** |
| **location\_lvl\_id<n>** | ID of location group for which specific ML model will be trained, n= **out\_location\_lvl** |
| **customer\_lvl\_id<l>** | ID of customer group for which specific ML model will be trained, l= **out\_customer\_lvl** |
| **distr\_channel\_lvl\_id<k>** | ID of distr\_channel group for which specific ML model will be trained, k= **out\_distr\_channel\_lvl** |
| **PERIOD\_DT** | Date of the first sales of forecasting timestamp |
| **PERIOD\_END\_DT** | Date of the last sales of forecasting timestamp |
| **SEGMENT\_NAME** | Name of segment that was linked to a pair product/location within VF Project (can be missing) |
| **VF\_FORECAST\_VALUE** | VF forecast value |
| **DEMAND\_TYPE** | ‘promo’ or ‘regular’ – type ML model is used to forecast |
| **ASSORTMENT\_TYPE** | new, or old |
| **ML\_FORECAST\_VALUE** | ML forecasted value |
| **HYBRID\_FORECAST\_VALUE** | HYBRID forecast value |

## CONFIG\_PARAMETERS

The following config parameters are used within the step.

The following config parameters are used within the step.

|  |  |  |
| --- | --- | --- |
| **CONFIG.TGT\_VAR\_CONFIG** | | |
| Column Name | Description | Example |
| **tgt\_type** | One of 3 types of the target variable:   * SELLIN – means CPG sales to its customer, * SELLOUT – means CPG’s customer sales to their clients, * POS – means sales in the point of sales, can be relevant for both Retailer and CPG | POS |
| **value\_src** | Name of the target variable from the source table. It should be quantity of sales.  Feasible values: INVOICE\_QTY, SALES\_QTY, SHIPMENT\_QTY, ORDER\_QTY. | SALES\_QTY |
| **act\_flag** | Activity flag, whether this target variable is needed to be forecasted. Feasible values: 0 or 1 | 1 |
| **vf\_product\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 8, which means PRODUCT\_ID | 7 |
| **vf\_location\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 6, which means LOCATION\_ID | 1 |
| **vf\_customer\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 6, which means CUSTOMER\_ID | 5 |
| **vf\_distr\_channel\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 3, which means DISTR\_CHANNEL\_ID | 1 |
| **vf\_time\_lvl** | Accumulation level for ML ABT by time hierarchy, default value is WEEK.2, which means weeks began from Monday | MONTH |
| **ml\_product\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 8, which means PRODUCT\_ID | 7 |
| **ml\_location\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 6, which means LOCATION\_ID | 5 |
| **ml\_customer\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 6, which means CUSTOMER\_ID | 4 |
| **ml\_distr\_channel\_lvl** | Aggregation level for ML ABT by product hierarchy, default value is 3, which means DISTR\_CHANNEL\_ID | 1 |
| **ml\_time\_lvl** | Accumulation level for ML ABT by time hierarchy, default value is WEEK.2, which means weeks began from Monday | WEEK.2 |
| **out\_time\_lvl** | Accumulation level regarding time hierarchy regarding output table | DAY or WEEK.2 |

## INITIAL\_GLOBAL parameters

All parameters are listed in initial\_global file.

|  |  |
| --- | --- |
| **INITIAL\_GLOBAL parameters init** | |
| Column Name | Description |
| **IB\_HIST\_END\_DT** | Last known date (i.e. sales and stock information is known) |

## Other Dependencies

# Algorithm Definition

## Disaggregation

**Inputs:** AGG\_HYB\_FCST = AGG\_HYB\_FCST\_<fcst\_type>\_<target var postfix>

**Transformation algorithm:**

1. Check whether period\_dt and period\_end\_dt in AGG\_HYB\_FCST correspond to out\_time\_lvl:

IF table is empty:

SELECT DISTINCT PERIOD\_DT, PERIOD\_END\_DT FROM AGG\_HYB\_FCST

WHERE intnx(out\_time\_lvl, PERIOD\_DT, 0) <> intnx(out\_time\_lvl, PERIOD\_END\_DT, 0)

THEN **FINAL\_GRANULARITY\_DELIVERED** = True

ELSE **FINAL\_GRANULARITY\_DELIVERED** = False.

1. If **FINAL\_GRANULARITY\_DELIVERED** == False then transform original table
   1. Provide all periods [OUT\_PERIOD\_DT := MAX(PERIOD\_DT, intnx(out\_time\_lvl, OUT\_PERIOD\_DT, 0,’b’)), OUT\_PERIOD\_END\_DT:=MIN(PERIOD\_END\_DT, intnx(out\_time\_lvl, OUT\_PERIOD\_END\_DT, 0,’e’))] that split period\_dt and period\_end\_dt to more granular time stamps, let’s denote them

OUT\_PERIOD\_DTand OUT\_PERIOD\_END\_DT.

Note: If input time granularity is Month and out\_time\_lvl is like ‘wee’ then output of the Disaccumulation step will be in split-week granularity.

1. Calculate forecast share and volume of VF\_FORECAST\_VALUE, ML\_FORECAST\_VALUE, HYBRID\_FORECASTproportionally to number of days in interval [PERIOD\_DT, PERIOD\_END\_DT]:
   1. For each row

VF\_FORECAST\_VALUE = VF\_FORECAST\_VALUE \* ,

ML\_FORECAST\_VALUE = ML\_FORECAST\_VALUE \*

HYBRID\_FORECAST\_VALUE = HYBRID \_FORECAST\_VALUE \*

Assumption: no need to consider more complex logic when splitting forecast volumes to the interval [PERIOD\_DT, PERIOD\_END\_DT]

Output: As a result of this step, a table of the following structure is constructed, *T1.*

|  |  |
| --- | --- |
| Column Name | Description |
| **product\_lvl\_id<m>** | ID of product group for which specific ML model will be trained, m= **out\_product\_lvl** |
| **location\_lvl\_id<n>** | ID of location group for which specific ML model will be trained, n= **out\_location\_lvl** |
| **customer\_lvl\_id<l>** | ID of customer group for which specific ML model will be trained, l= **out\_customer\_lvl** |
| **distr\_channel\_lvl\_id<k>** | ID of distr\_channel group for which specific ML model will be trained, k= **out\_distr\_channel\_lvl** |
| **PERIOD\_DT** | Date of the first sales of forecasting timestamp |
| **PERIOD\_END\_DT** | Date of the last sales of forecasting timestamp |
| **SEGMENT\_NAME** | Name of segment that was linked to a pair product/location within VF Project (can be missing) |
| **VF\_FORECAST\_VALUE** | VF forecast value |
| **DEMAND\_TYPE** | ‘promo’ or ‘regular’ – type ML model is used to forecast |
| **ASSORTMENT\_TYPE** | new, or old |
| **ML\_FORECAST\_VALUE** | forecasted value |

## Providing PRODUCT Life Cycle information

**Inputs:** T1

**Transformation algorithm:**

1. Keep the Same Table

Output: As a result of this step, a table of the following structure is constructed, *T2.*

|  |  |
| --- | --- |
| Column Name | Description |
| **PRODUCT\_ID** | Product ID (the lowest level of the product hierarchy) |
| **LOCATION\_ID** | Location ID |
| **PERIOD\_DT** | Date of sales (calendar day) |
| **SEGMENT\_NAME** | Name of segment that was linked to a pair product/location within VF Project (can be missing) |
| **VF\_FORECAST\_VALUE** | VF forecast value |
| **DEMAND\_TYPE** | ‘promo’ or ‘regular’ – type ML model is used to forecast |
| **ASSORTMENT\_TYPE** | new, or old |
| **ML\_FORECAST\_VALUE** | forecasted value |

## Providing LOCATION Life Cycle information

**Inputs:** T2

**Transformation algorithm:**

1. Keep the Same Table

Output: As a result of this step, a table of the following structure is constructed, *T3.*

|  |  |
| --- | --- |
| Column Name | Description |
| **PRODUCT\_ID** | Product ID (the lowest level of the product hierarchy) |
| **LOCATION\_ID** | Location ID |
| **PERIOD\_DT** | Date of sales (calendar day) |
| **SEGMENT\_NAME** | Name of segment that was linked to a pair product/location within VF Project (can be missing) |
| **VF\_FORECAST\_VALUE** | VF forecast value |
| **DEMAND\_TYPE** | ‘promo’ or ‘regular’ – type ML model is used to forecast |
| **ASSORTMENT\_TYPE** | new, or old |
| **ML\_FORECAST\_VALUE** | forecasted value |

## Providing CUSTOMER Life Cycle information

**Inputs:** T3

**Transformation algorithm:**

1. Keep the Same Table

Output: As a result of this step, a table of the following structure is constructed, ACC\_AGG\_HYBRID\_FORECAST (detailed description see in section 3.5)*.*

## Output from the Algorithm

1. Table with all types of forecast

|  |  |
| --- | --- |
| **ACC\_AGG\_HYBRID\_FORECAST** | |
| Column Name | Description |
| **product\_lvl\_id<m>** | ID of product group for which specific ML model will be trained, m= **out\_product\_lvl** |
| **location\_lvl\_id<n>** | ID of location group for which specific ML model will be trained, n= **out\_location\_lvl** |
| **customer\_lvl\_id<l>** | ID of customer group for which specific ML model will be trained, l= **out\_customer\_lvl** |
| **distr\_channel\_lvl\_id<k>** | ID of distr\_channel group for which specific ML model will be trained, k= **out\_distr\_channel\_lvl** |
| **PERIOD\_DT** | Date of the first sales of forecasting timestamp at out\_time\_lvl granularity |
| **PERIOD\_END\_DT** | Date of the last sales of forecasting timestamp at out\_time\_lvl granularity |
| **SEGMENT\_NAME** | Name of segment that was linked to a pair product/location within VF Project (can be missing) |
| **VF\_FORECAST\_VALUE** | VF forecast value |
| **DEMAND\_TYPE** | ‘promo’ or ‘regular’ – type ML model is used to forecast |
| **ASSORTMENT\_TYPE** | new, or old |
| **ML\_FORECAST\_VALUE** | forecasted value |
| **product\_lvl\_id<m>** | ID of product group for which specific ML model will be trained, m= **out\_product\_lvl** |
| **location\_lvl\_id<n>** | ID of location group for which specific ML model will be trained, n= **out\_location\_lvl** |