

EXAMPLE - PREV Function

This example describes how you can use the PREV function to analyze data that is available in a window in rows before the current one.

Functions:

Item	Description
PREV Function	Extracts the value from a column that is a specified number of rows before the current value.
IF Function	The IF function allows you to build if/then/else conditional logic within your transforms.

The following dataset contains orders for multiple customers over a period of a few days, listed in no particular order. You want to assess how order size has changed for each customer over time and to provide offers to your customers based on changes in order volume.

Source:

Date	CustId	OrderId	OrderValue
1/4/16	C001	Ord002	500
1/11/16	C003	Ord005	200
1/20/16	C002	Ord007	300
1/21/16	C003	Ord008	400
1/4/16	C001	Ord001	100
1/7/16	C002	Ord003	600
1/8/16	C003	Ord004	700
1/21/16	C002	Ord009	200
1/15/16	C001	Ord006	900

Transformation:

When the data is loaded into the Transformer page, you can use the PREV function to gather the order values for the previous two orders into a new column. The trick is to order the window transform by the date and group it by customer:

Transformation Name	Window
Parameter: Formulas	PREV(OrderValue, 1)
Parameter: Group by	CustId
Parameter: Order by	Date

Transformation Name	Window
Parameter: Formulas	PREV(OrderValue, 2)
Parameter: Group by	CustId
Parameter: Order by	Date

Transformation Name	Rename columns
----------------------------	----------------

Parameter: Option	Manual rename
Parameter: Column	window
Parameter: New column name	'OrderValue_1'

Transformation Name	Rename columns
Parameter: Option	Manual rename
Parameter: Column	window1
Parameter: New column name	'OrderValue_2'

You should now have the following columns in your dataset: Date, CustId, OrderId, OrderValue, OrderValue_1, OrderValue_2.

The two new columns represent the previous order and the order before that, respectively. Now, each row contains the current order (OrderValue) as well as the previous orders. Now, you want to take the following customer actions:

- If the current order is more than 20% greater than the sum of the two previous orders, send a rebate.
- If the current order is less than 90% of the sum of the two previous orders, send a coupon.
- Otherwise, send a holiday card.

To address the first one, you might add the following, which uses the IF function to test the value of the current order compared to the previous ones:

Transformation Name	New formula
Parameter: Formula type	Single row formula
Parameter: Formula	IF(OrderValue >= (1.2 * (OrderValue_1 + OrderValue_2)), 'send rebate', 'no action')
Parameter: New column name	'CustomerAction'

You can now see which customers are due a rebate. Now, edit the above and replace it with the following, which addresses the second condition. If neither condition is valid, then the result is send holiday card.

Transformation Name	New formula
Parameter: Formula type	Single row formula
Parameter: Formula	IF(OrderValue >= (1.2 * (OrderValue_1 + OrderValue_2)), 'send rebate', IF(OrderValue <= (1.2 * (OrderValue_1 + OrderValue_2)), 'send coupon', 'send holiday card'))
Parameter: New column name	'CustomerAction'

Results:

After you delete the OrderValue_1 and OrderValue_2 columns, your dataset should look like the following. Since the transformations with PREV functions grouped by CustId, the order of records has changed.

Date	CustId	OrderId	OrderValue	CustomerAction
1/4/16	C001	Ord001	100	send rebate
1/7/16	C001	Ord002	500	send rebate
1/15/16	C001	Ord006	900	send rebate
1/8/16	C003	Ord004	700	send rebate
1/11/16	C003	Ord005	200	send rebate
1/21/16	C003	Ord008	400	send coupon
1/7/16	C002	Ord003	600	send rebate
1/20/16	C002	Ord007	300	send rebate
1/21/16	C002	Ord009	200	send coupon