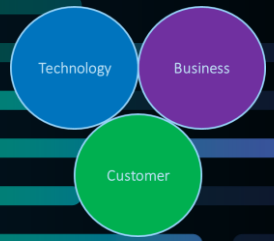


TIPS & TRICKS

Special Edition

The best of Solving Benchmarking Challenges

Carl-Olow “Kalle” Magnusson

Nordic Senior Visualization Advisor

Benchmarking is a very common task in SAS Visual Analytics

Benchmarking *= comparing something with something else*

Comparison with Average

(455 – 512) / 512 = -11 %

Percentage of Total

14 600 / 32 500 = 0,449231 (45 %)

Benchmarking is a very common task in SAS Visual Analytics

Sometimes it is straightforward, sometimes it can be challenging

I want to calculate the total by regions, based on the average for all sites within a region….

I want to have percentage of total… when I add filters, how can I keep my totals?

Commonly used benchmark functions in Visual Analytics

Feature/Function	Type	VA feature	Usage
Aggregation operators, such as SUM, AVG, DISTINCT etc.	Aggregated Operator	Expression builder	Calculate different statistics with aggregation level control
AND, IF…ELSE, NOT and OR	Boolean Operator	Expression builder	Adding logical control in expressions
AggregateTable	Aggregated Operator	Expression builder	AggregateTable allows us to aggregate data and lock it onto a specific data/aggregation level
Scoped Calculations	Feature	Expression builder	Allows us to add multiple/different expressions into our calculated item
Parameters	Data Item	Data Item	Parameters can fetch a value to be used in a calculated item, filters, ranks etc.
IsSet	Comparison Operator	Expression builder	IsSet function is used to determine if a parameter is set (true) or not (false).
NumMiss	Aggregated Operator	Expression builder	Is used to identify if missing values are present in your aggregation
New data from aggregation…	Data	Data Action	Allows us to create a new aggregated data set in VA, useful when handling advanced aggregations or handling nested aggregated data items
Periodic operators, such as Period, CumulativePeriod etc.	Aggregated Operator	Expression builder	Calculate comparing periodic and aggregated metrics over time, such as % difference compared with last year. A date column is required
AggregateCells	Aggregated Operator	Expression builder	Aggregates values of a specific set of cells, such as 7 days average. No specific date or category column is required

Commonly used functions used for benchmarking

Feature/Function	Type	VA feature	Usage
Aggregation operators, such as SUM, AVG, DISTINCT etc.	Aggregated Operator	Expression builder	Calculate different statistics with aggregation level control
AND, IF…ELSE, NOT and OR	Boolean Operator	Expression builder	Adding logical control in expressions
AggregateTable	Aggregated Operator	Expression builder	AggregateTable… …nto a specific
Scoped Calculations			…alculated item
Parameters			…m, filters, ranks etc.
IsSet			isSet function is used to determine if a parameter is set (true) or not (false).
NumMiss	Aggregated Operator	Expression builder	Is used to identify if missing values are present in your aggregation
New data from aggregation…	Data	Data Action	Allows us to create a new aggregated data set in VA, useful when handling advanced aggregations or handling nested aggregated data items
Periodic operators, such as Period, CumulativePeriod etc.	Aggregated Operator	Expression builder	Calculate comparing periodic and aggregated metrics over time, such as % difference compared with last year. A date column is required
AggregateCells	Aggregated Operator	Expression builder	Aggregates values of a specific set of cells, such as 7 days average. No specific date or category column is required

So, the challenge is to know when and how to use these functions…

Today we have 3 different use-cases/scenarios

#30 We have a dashboard with Sales % of Total, but end-users do not want Sales % of Total to be affected by filters used in the Dashboard

A report filter challenge, how to keep totals intact when using filters

#33 End users want to see Sales % of Total, while comparing it between selectable weeks and by product group

A classic sub-totals and/or by-groups challenge

#34 Creating a KPI for Average Sales by Continent based on Total Sales by country

A very good example explaining how inner and outer table works

Today we have 3 different use-cases/scenarios

Functions/operators that we will use…

#30 Parameters will be used, IsSet operator introduction and some logic re-thinking creating new data item ☺

#33 AggregateTable introduction and a simple example from the field

#34 A more advanced example using AggregateTable

Solving Benchmarking Challenges

Tips & Tricks # 30

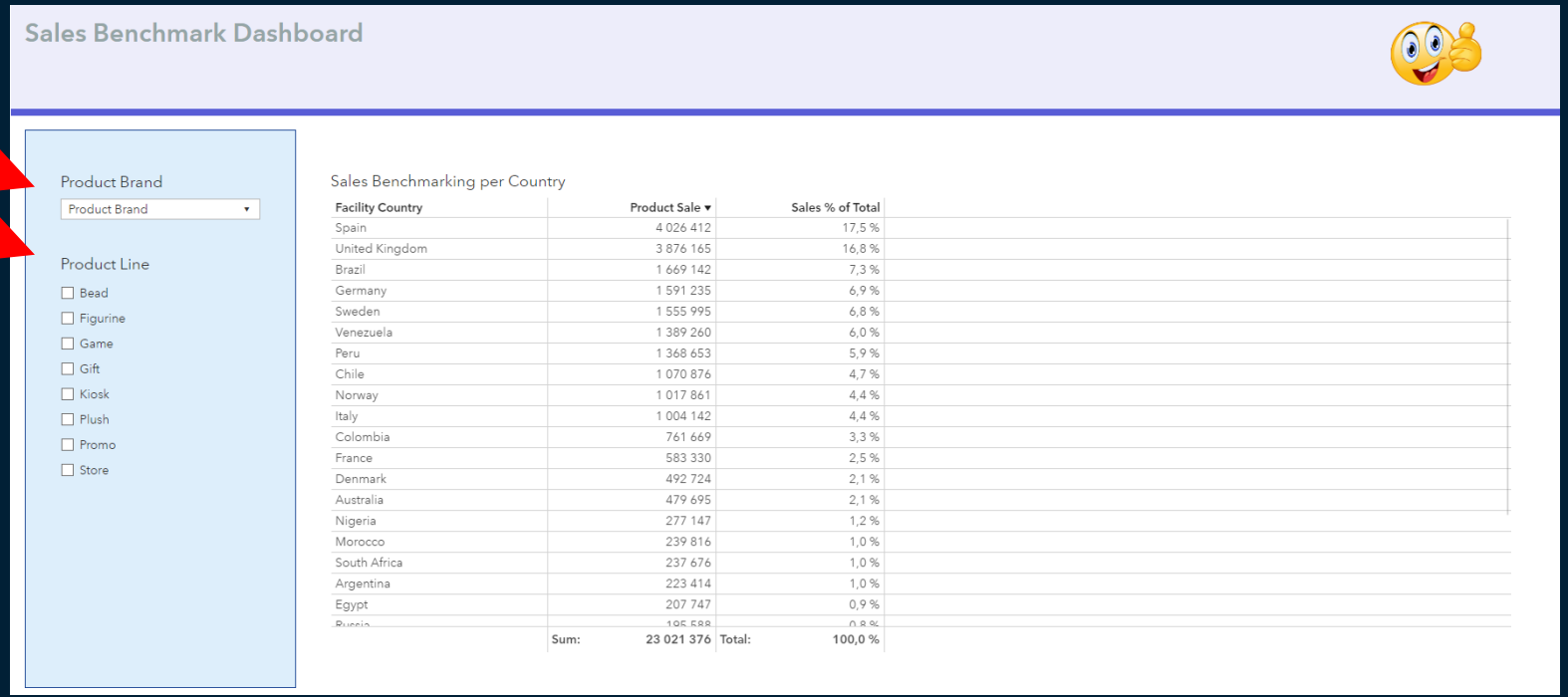
How can I keep my totals intact when using filters?

In this example Parameters will be used, IsSet operator introduction and some logic re-thinking creating new data item ☺

How can I keep my totals intact when using filters?

Tips & Tricks # 30

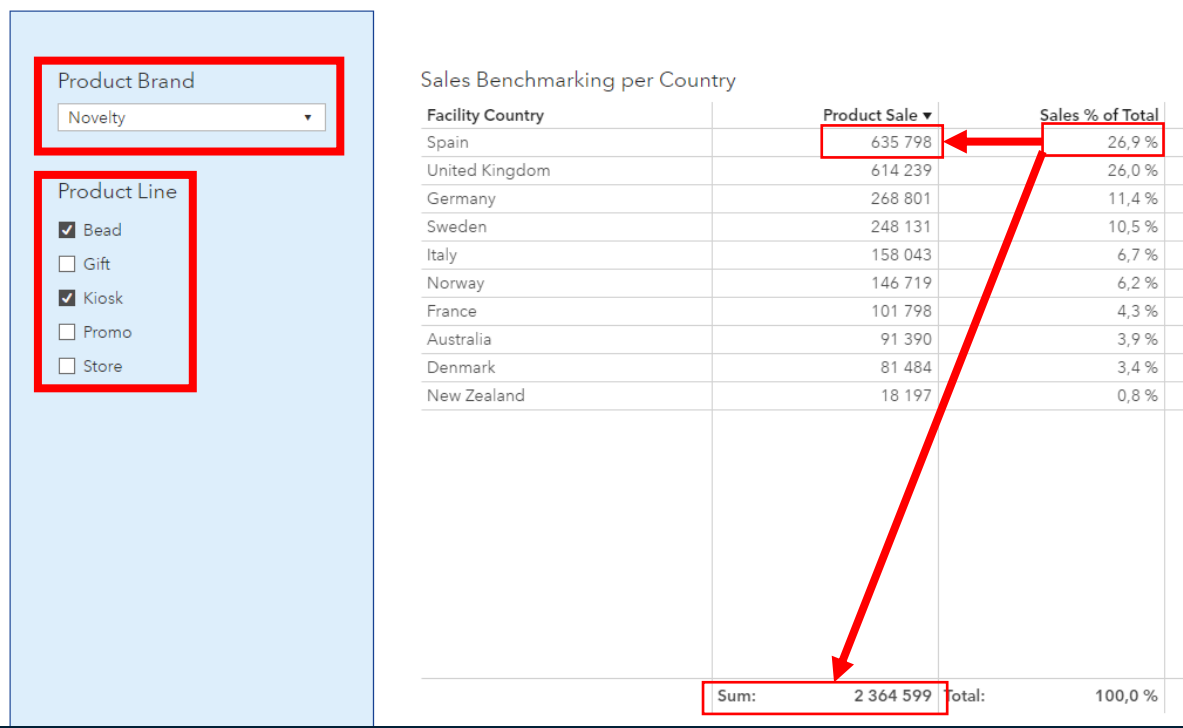
Challenge: I have a dashboard with some sale benchmarking metrics and added some filters. However, end-users DO NOT want Sales % of Total to be affected by Product Brand and Line filters.



How can I keep my totals intact when using filters?

Tips & Tricks # 30

Challenge: I have a dashboard with some sale benchmarking metrics and added some filters. However, end-users DO NOT want Sales % of Total to be affected by Product Brand and Line filters.



According to our end-users, this is not correct!

It should be 2,8 %

635 798 / 23 021 376

How can I keep my totals intact when using filters?

Tips & Tricks # 30

Challenge: I have a dashboard with some sale benchmarking metrics and added some filters. However, end-users DO NOT want Sales % of Total to be affected by Product Brand and Line filters.

Preparing data with pre-defined columns with totals is a very common solution, but it might impact flexibility…

Is there any other solution?

How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

We will keep the original data item Product Sale as our total and create a new data item that will be affected by our filters

$$\frac{\text{Product Sale (Filtered)}}{\text{Product Sale}} = \text{Sales \% of Total}$$

Product Line

- [x] Bead
- [ ] Gift
- [x] Kiosk
- [ ] Promo
- [ ] Store

Sales Benchmarking per Country

Facility Country	Product Sale ▾	Product Sale (Filtered)	Sales % of Total (Filter Control)
Spain	4 026 412	635 798	2,8 %
United Kingdom	3 876 165	614 239	2,7 %
Brazil	1 669 142	0	0,0 %
Germany	1 591 235	268 801	1,2 %
Sweden	1 555 995	248 131	1,1 %
Venezuela	1 389 260	0	0,0 %

How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

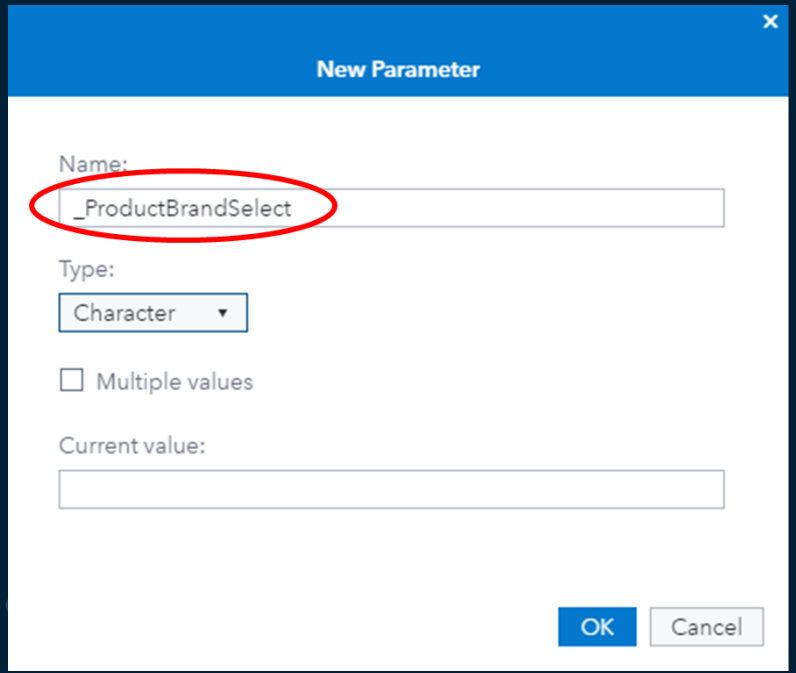
Before we start:

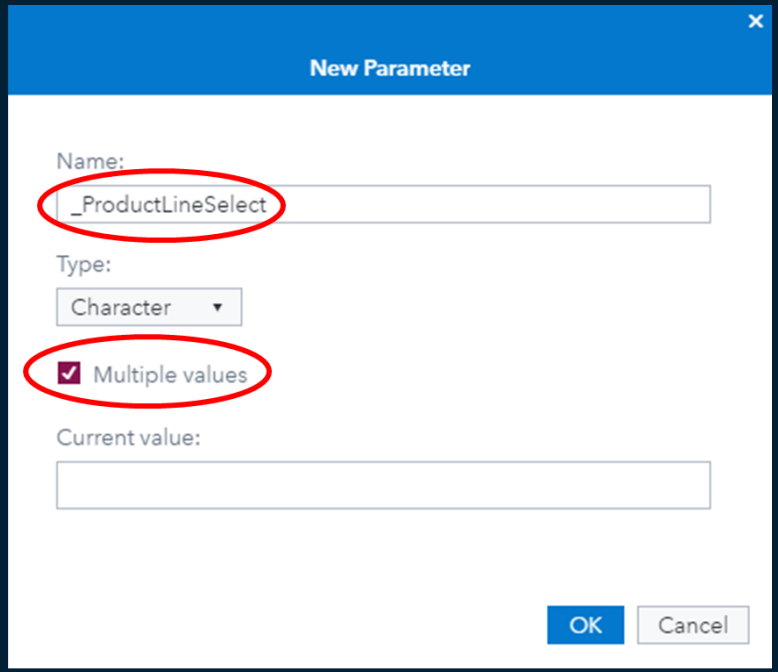
I do recommend to turn off/de-activate filters (including ranks) to avoid confusion ☺

How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

Step 2:1: Add two parameters

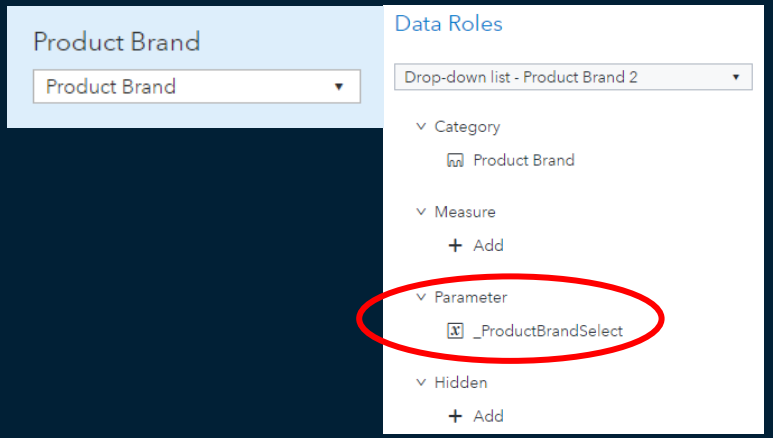


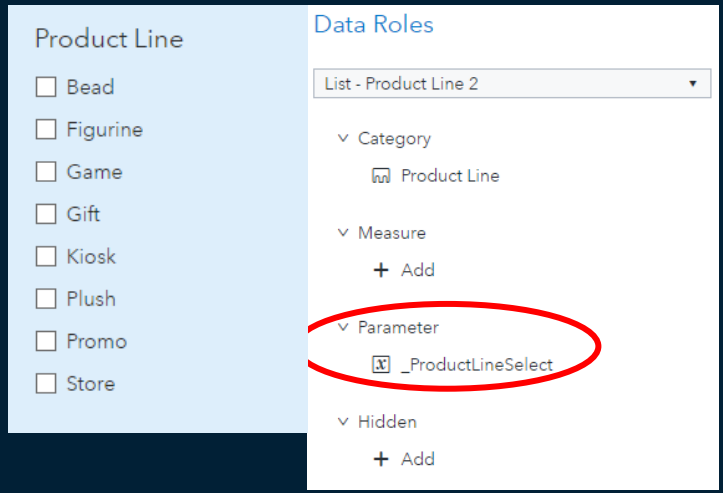


How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

Step 2:2: Assign parameters to my control objects (Product Brand & Line)





How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

Step 3: Create calculated items that is affected by our filters, either we can do a one or two step approach. This is the one step approach:

Calculated data item: Product Sale (Filtered)

IF Product Brand OR Product Line is selected RETURN Product Sales ELSE 0

IsSet function is used to check if Product brand OR Product Line is selected

```
IF AND [ ProductBrandSelect IsSet
         NOT ( ProductLineSelect IsSet ) ]
RETURN [ IF ( Product Brand = ProductBrandSelect )
         RETURN Product Sale
         ELSE 0 ]
ELSE [ IF OR [ ProductBrandSelect IsSet
               ProductLineSelect IsSet ]
       RETURN [ IF Product Line In ProductLineSelect
                RETURN Product Sale
                ELSE 0 ]
       ELSE Product Sale ]
```

How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

Step 3: Create calculated items that is affected by our filters, either we can do a one or two step approach. This is the one step approach:

Calculated data item: Product Sale (Filtered)

IF Product Brand OR Product Line is selected RETURN Product Sales ELSE 0

Product Brand

Novelty

Product Line

- [ ] Bead
- [x] Gift
- [x] Kiosk
- [x] Promo
- [ ] Store

Sales Benchmarking per Country

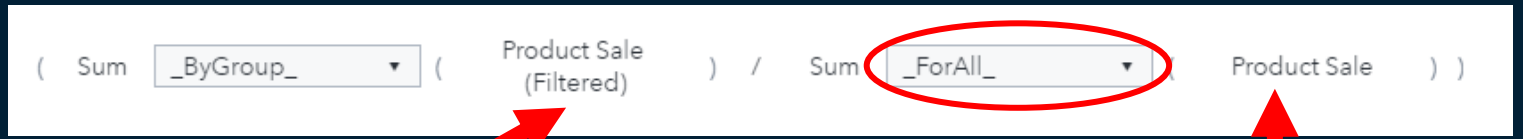
Facility Country	Product Sale ▾	Product Sale (Filtered)
Spain	4 026 412	952 865
United Kingdom	3 876 165	884 371
Brazil	1 669 142	0
Germany	1 591 235	391 335
Sweden	1 555 995	363 345
Venezuela	1 389 260	0
Peru	1 368 653	0
Chile	1 070 876	0
Norway	1 017 861	234 865
Italy	1 004 142	242 187
Colombia	761 669	0
France	583 330	140 613
Denmark	492 724	120 609
Australia	479 695	121 988

How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

Step 4: Create Sales % of Total with filter control, based on the calculated item we created in step 3

Calculated data item: Sales % of Total (Filter Control)



Product Sale (Filtered), is affected by our filters

Product Sale is our original column. As it is never affected by our Product Brand and Line filters, it will keep total sales

Make sure you set aggregation to _ForAll_

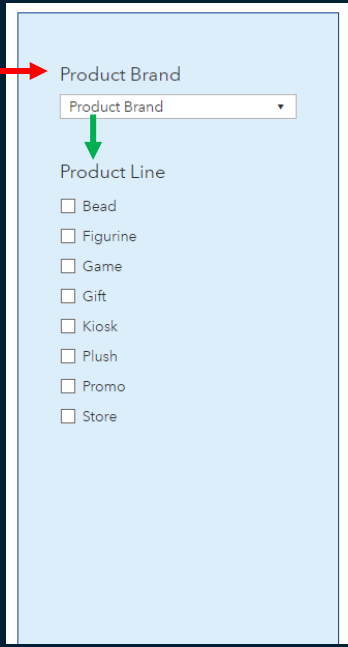
How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution

Step 5: Activate necessary filters and filter dependencies

Setup a dependency filter between Product Brand and Line control/filter objects.

Note: All filtering is controlled by parameters and the calculated item we created during step 3.



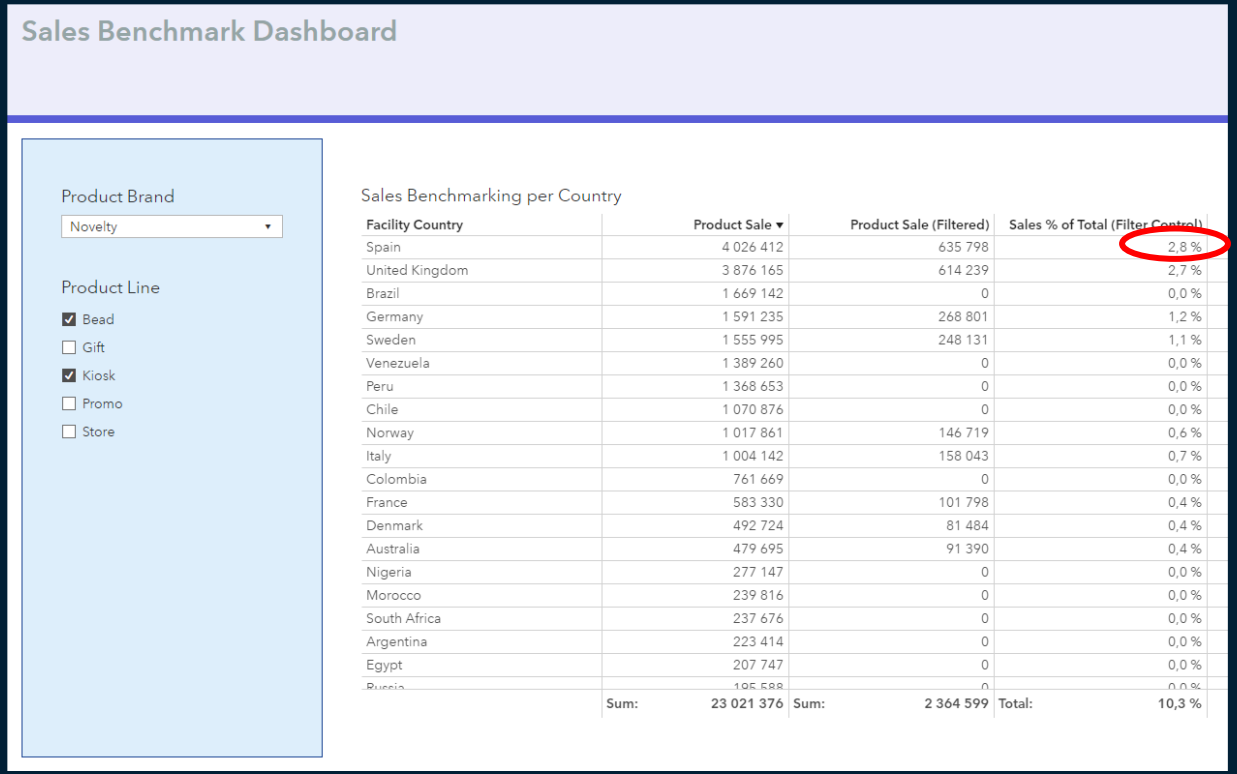
Sales Benchmarking per Country

Facility Country	Product Sale ▾	Product Sale (Filtered)	Sales % of Total (Filter Control)
Spain	4 026 412	4 026 412	17,5 %
United Kingdom	3 876 165	3 876 165	16,8 %
Brazil	1 669 142	1 669 142	7,3 %
Germany	1 591 235	1 591 235	6,9 %
Sweden	1 555 995	1 555 995	6,8 %
Venezuela	1 389 260	1 389 260	6,0 %
Peru	1 368 653	1 368 653	5,9 %
Chile	1 070 876	1 070 876	4,7 %
Norway	1 017 861	1 017 861	4,4 %
Italy	1 004 142	1 004 142	4,4 %
Colombia	761 669	761 669	3,3 %
France	583 330	583 330	2,5 %
Denmark	492 724	492 724	2,1 %
Australia	479 695	479 695	2,1 %
Nigeria	277 147	277 147	1,2 %
Morocco	239 816	239 816	1,0 %
South Africa	237 676	237 676	1,0 %
Argentina	223 414	223 414	1,0 %
Egypt	207 747	207 747	0,9 %
Russia	195 588	195 588	0,8 %
	Sum: 23 021 376	Sum: 23 021 376	Total: 100,0 %

DO NOT use “Automatic actions on all objects”!!! If we do, our regain of total control will be lost ☺

How can I keep my totals intact when using filters?

Tips & Tricks # 30 - Solution



Facility Country	Product Sale ▾	Product Sale (Filtered)	Sales % of Total (Filter Control)
Spain	4 026 412	635 798	2,8 %
United Kingdom	3 876 165	614 239	2,7 %
Brazil	1 669 142	0	0,0 %
Germany	1 591 235	268 801	1,2 %
Sweden	1 555 995	248 131	1,1 %
Venezuela	1 389 260	0	0,0 %
Peru	1 368 653	0	0,0 %
Chile	1 070 876	0	0,0 %
Norway	1 017 861	146 719	0,6 %
Italy	1 004 142	158 043	0,7 %
Colombia	761 669	0	0,0 %
France	583 330	101 798	0,4 %
Denmark	492 724	81 484	0,4 %
Australia	479 695	91 390	0,4 %
Nigeria	277 147	0	0,0 %
Morocco	239 816	0	0,0 %
South Africa	237 676	0	0,0 %
Argentina	223 414	0	0,0 %
Egypt	207 747	0	0,0 %
Russia	195 588	0	0,0 %
	Sum: 23 021 376	Sum: 2 364 599	Total: 10,3 %

An AggregateTable Introduction

#33 End users want to see Sales % of Total, while comparing it between selectable weeks and by product group

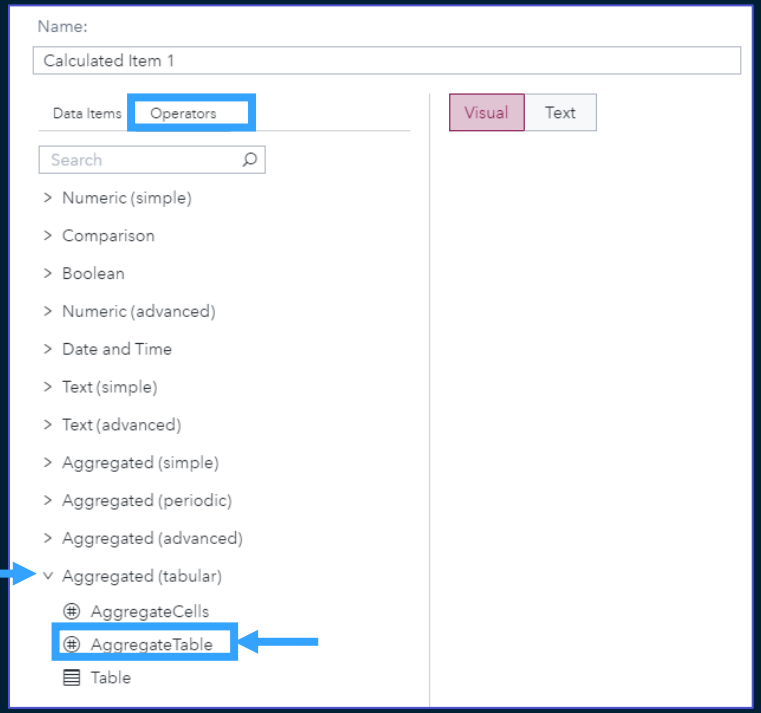
A classic sub-totals and/or by-groups challenge

#34 Creating a KPI for Average Sales by Continent based on Total Sales by country

A very good example explaining how inner and outer table works

AggregateTable - Introduction

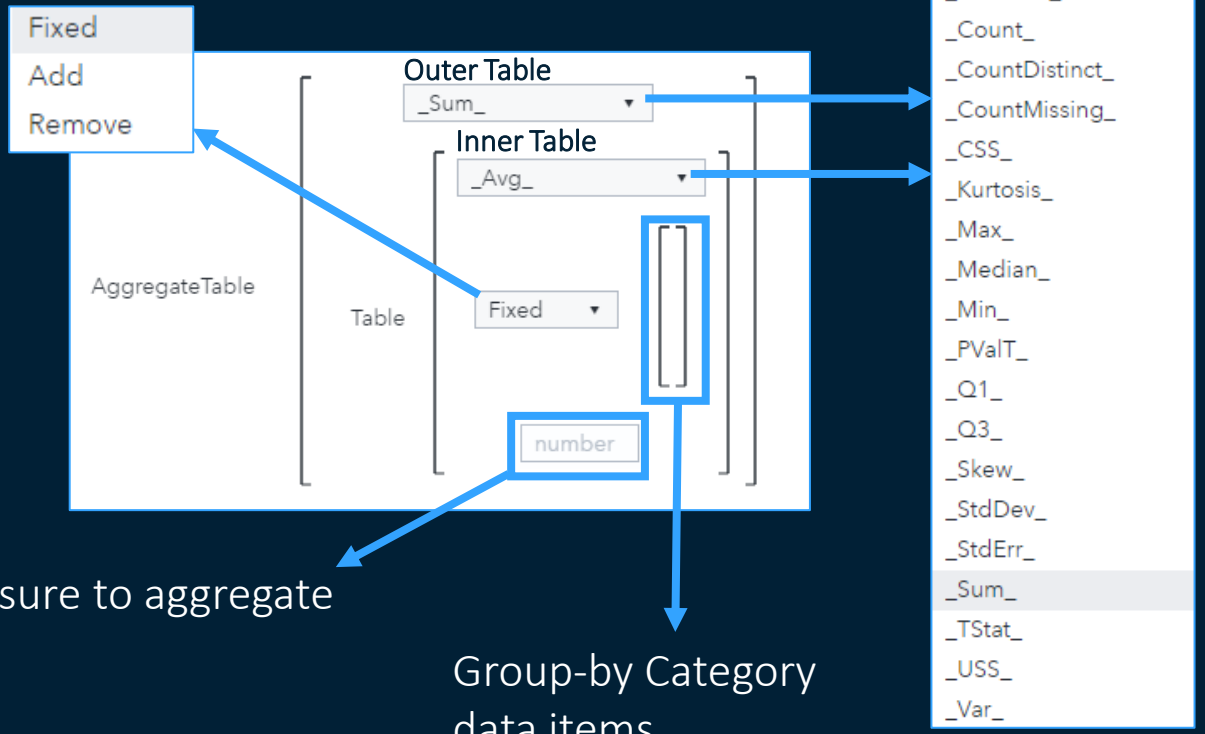
Data Item Expression Builder



AggregateTable - Introduction

Specify aggregation

Group-by Category options



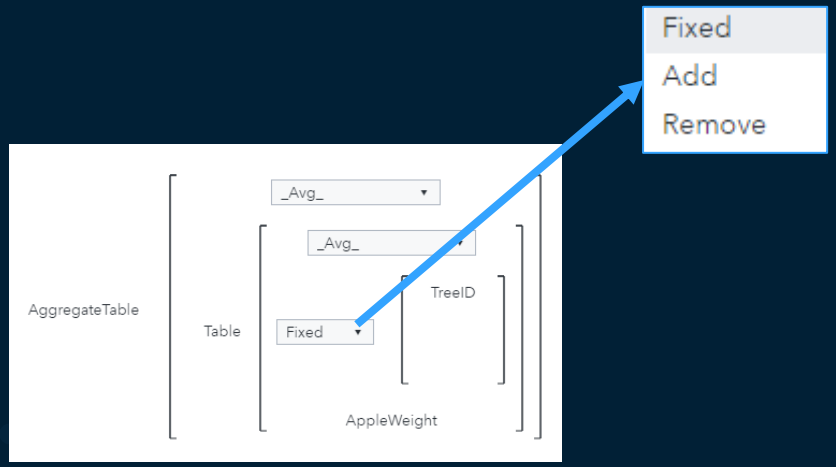
Measure to aggregate

Group-by Category data items

AggregateTable - Introduction

Group-by Category options

Important: Keep in mind, data items that you are using in your visualization will impact the outcome when AggregateTable is used. This is the reason why you can set different group-by category options.



Fixed: Most common used option, we simply want the inner table to be fixed by specified category data item(s), in this case "lock" it by TreeID and TreeID must be part of our visualization.

Add: This options allow us to add specified category data item(s) to our inner table without the need to visualize it/them.

Remove: This options allow us to remove specified category data item(s) to our inner table when we want to visualize it/them.

Bonus Use-Case

Group-by Category options

- Fixed
- Add
- Remove

#35 Engineers want to review average milage by car model and production year, based on service data

A great example where the “add” option is needed

ProductionPeriod	ProductionYear	ChassiID	CarModel	ServiceDate	CarReadoutKM
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2020-07-12	4560
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2020-11-17	7650
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2021-11-30	10805
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2022-10-25	15870
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2021-09-01	11900
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2021-10-10	12345
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2022-06-30	18700
202008	2020	A10044	Audi A3 Sportsback 45 TFSI e	2021-11-13	9055
202008	2020	A10044	Audi A3 Sportsback 45 TFSI e	2022-06-12	15890
202008	2020	A10049	Audi A3 Sportsback 45 TFSI e	2022-01-16	15600
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2021-12-12	4570
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2022-03-25	7650
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2022-10-10	8700
202103	2021	A10129	Audi A3 Sportsback 45 TFSI e	2022-01-13	9800
202104	2021	A10141	Audi A3 Sportsback 45 TFSI e	2021-06-12	1560
202104	2021	A10141	Audi A3 Sportsback 45 TFSI e	2022-08-09	6790
202104	2021	A10145	Audi A3 Sportsback 45 TFSI e	2022-09-12	13450
202106	2021	A10147	Audi A3 Sportsback 45 TFSI e	2022-09-18	14560
202110	2021	A10161	Audi A3 Sportsback 45 TFSI e	2022-02-03	5675
202110	2021	A10161	Audi A3 Sportsback 45 TFSI e	2022-10-19	11900
202111	2021	A10189	Audi A3 Sportsback 45 TFSI e	2022-11-10	8045
202112	2021	A10212	Audi A3 Sportsback 45 TFSI e	2022-03-15	15605
202112	2021	A10212	Audi A3 Sportsback 45 TFSI e	2022-11-09	21890

CarModel	ProductionYear ▲	Average Readout Km
Audi A3 Sportsback 45 TFSI e	2020	14 952
Audi A3 Sportsback 45 TFSI e	2021	12 348

The Apple Pile Example

AggregateTable - Introduction

The Apple Pile Example

Apple ID	TreeID	AppleWeight
A1	Tree 1	155
A2	Tree 1	145
A3	Tree 1	156
A4	Tree 1	143
A5	Tree 1	156
A6	Tree 1	149
A7	Tree 1	159
A8	Tree 2	164
A9	Tree 2	161
A10	Tree 2	159
A11	Tree 2	170
A12	Tree 2	177
A13	Tree 3	171
A14	Tree 3	167
A15	Tree 3	164
A16	Tree 3	157
A17	Tree 3	159
A18	Tree 3	155
A19	Tree 3	167
A20	Tree 4	188
A21	Tree 4	178
A22	Tree 4	191
A23	Tree 4	178
A24	Tree 4	176
A25	Tree 4	181

AggregateTable - Introduction

The Apple Pile Example

Apple ID	TreeID	AppleWeight
A1	Tree 1	155
A2	Tree 1	145
A3	Tree 1	156
A4	Tree 1	143
A5	Tree 1	156
A6	Tree 1	149
A7	Tree 1	159
A8	Tree 2	164
A9	Tree 2	161
A10	Tree 2	159
A11	Tree 2	170
A12	Tree 2	177
A13	Tree 3	171
A14	Tree 3	167
A15	Tree 3	164
A16	Tree 3	157
A17	Tree 3	159
A18	Tree 3	155
A19	Tree 3	167
A20	Tree 4	188
A21	Tree 4	178
A22	Tree 4	191
A23	Tree 4	178
A24	Tree 4	176
A25	Tree 4	181

Apple benchmark Use-case:

- Simple Apple weight comparison
- For a fairer comparison, apple weight is compared to average apple weight for each tree

AppleWeight – AVG(AppleWeight) by TreeID

AggregateTable - Introduction

The Apple Pile Example

AppleWeight – AVG(AppleWeight) by TreeID

Apple ID	TreeID	AppleWeight	Average Apple Weight by TreeID
A1	Tree 1	155	152
A2	Tree 1	145	152
A3	Tree 1	156	152
A4	Tree 1	143	152
A5	Tree 1	156	152
A6	Tree 1	149	152
A7	Tree 1	159	152
A10	Tree 2	159	166
A11	Tree 2	170	166
A12	Tree 2	177	166
A8	Tree 2	164	166
A9	Tree 2	161	166
A13	Tree 3	171	163
A14	Tree 3	167	163
A15	Tree 3	164	163
A16	Tree 3	157	163
A17	Tree 3	159	163
A18	Tree 3	155	163
A19	Tree 3	167	163
A20	Tree 4	188	182
A21	Tree 4	178	182
A22	Tree 4	191	182
A23	Tree 4	178	182
A24	Tree 4	176	182
A25	Tree 4	181	182

To be able to calculate the comparison by Tree, we need to calculate the average apple weight by TreeID…

Do I need to create another column before loading data to VA?

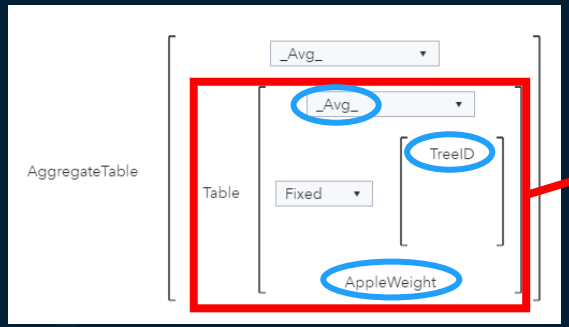
Or can we do this directly in VA?

AggregateTable - Introduction

The Apple Pile Example

Yes, we can achieve this by using the AggregateTable operator/function

Average Apple Weight by TreeID



Inner table aggregation

Avg. AppleWeight
Tree 1

Avg. AppleWeight
Tree 2

Avg. AppleWeight
Tree 3

Avg. AppleWeight
Tree 4

Apple ID	TreeID	AppleWeight	Average Apple Weight by TreeID
A1	Tree 1	155	152
A2	Tree 1	145	152
A3	Tree 1	156	152
A4	Tree 1	143	152
A5	Tree 1	156	152
A6	Tree 1	149	152
A7	Tree 1	159	152
A10	Tree 2	159	166
A11	Tree 2	170	166
A12	Tree 2	177	166
A8	Tree 2	164	166
A9	Tree 2	161	166
A13	Tree 3	171	163
A14	Tree 3	167	163
A15	Tree 3	164	163
A16	Tree 3	157	163
A17	Tree 3	159	163
A18	Tree 3	155	163
A19	Tree 3	167	163
A20	Tree 4	188	182
A21	Tree 4	178	182
A22	Tree 4	191	182
A23	Tree 4	178	182
A24	Tree 4	176	182
A25	Tree 4	181	182

AggregateTable - Introduction

The Apple Pile Example

Apple Weight Diff by Tree Average

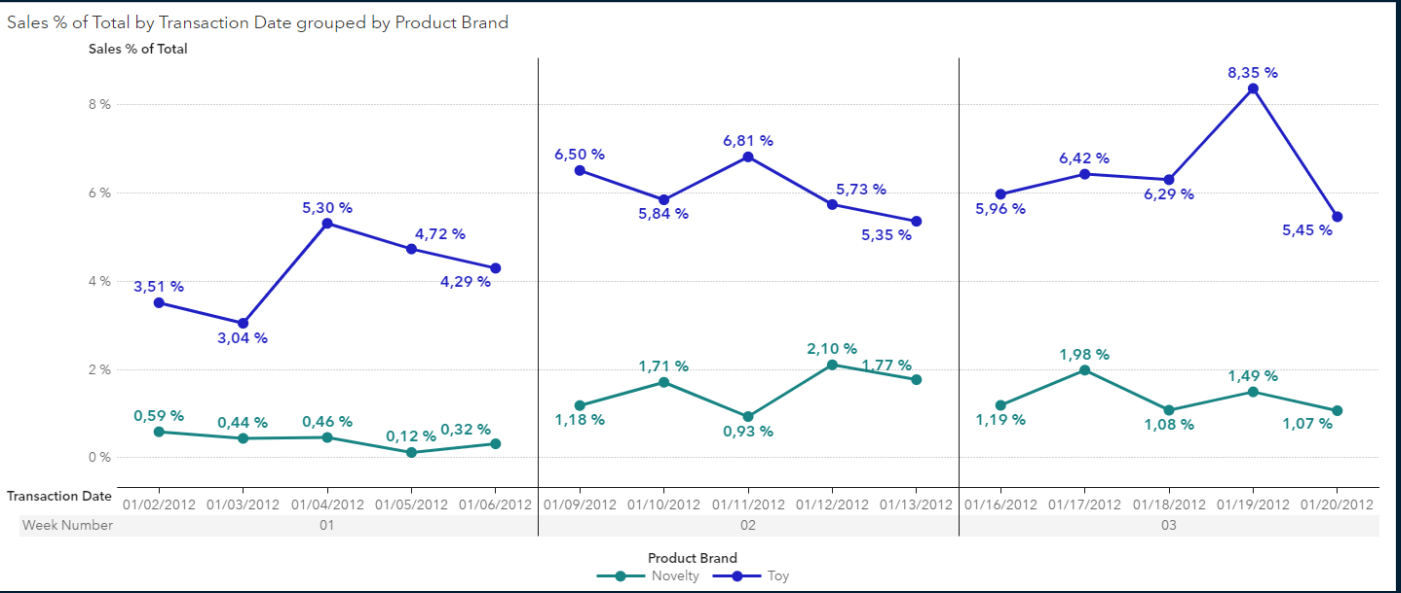
Apple ID	TreeID	AppleWeight	Average Apple Weight by TreeID	Apple Weight Diff by Tree Average
A1	Tree 1	155	152	3
A2	Tree 1	145	152	−7
A3	Tree 1	156	152	4
A4	Tree 1	143	152	−9
A5	Tree 1	156	152	4
A6	Tree 1	149	152	−3
A7	Tree 1	159	152	7
A10	Tree 2	159	166	−7
A11	Tree 2	170	166	4

AggregateTable – Use-case #33

Locking totals on specific levels in your data

#33 End users want to see sales % of total, while comparing it between selectable weeks and by Product Brand

A classic sub-totals and/or by-groups challenge

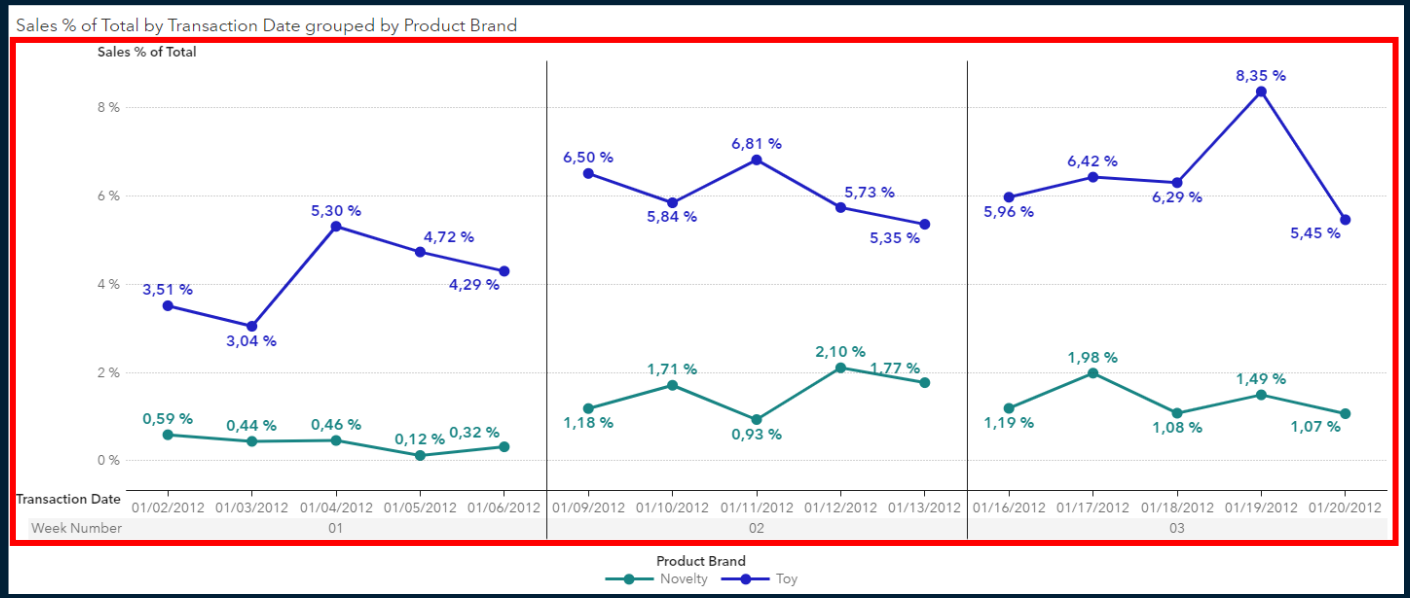


#33 End users want to see sales % of total, while comparing it between selectable weeks and by Product Brand

A classic sub-totals and/or by-groups challenge

Sales % of Total

(Sum _ByGroup_ (Product Sale) / Sum _ForAll_ (Product Sale))



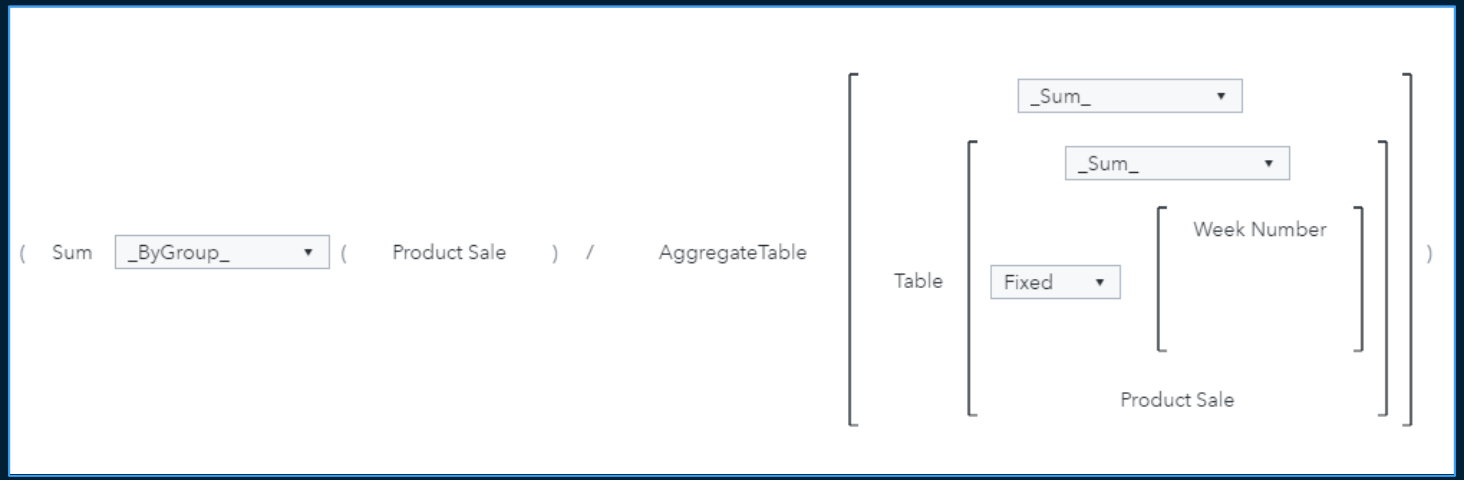
Challenge:
End users want to see sales % of total by week, not all weeks displayed...

#33 End users want to see sales % of total, while comparing it between selectable weeks and by Product Brand

A classic sub-totals and/or by-groups challenge

Solution

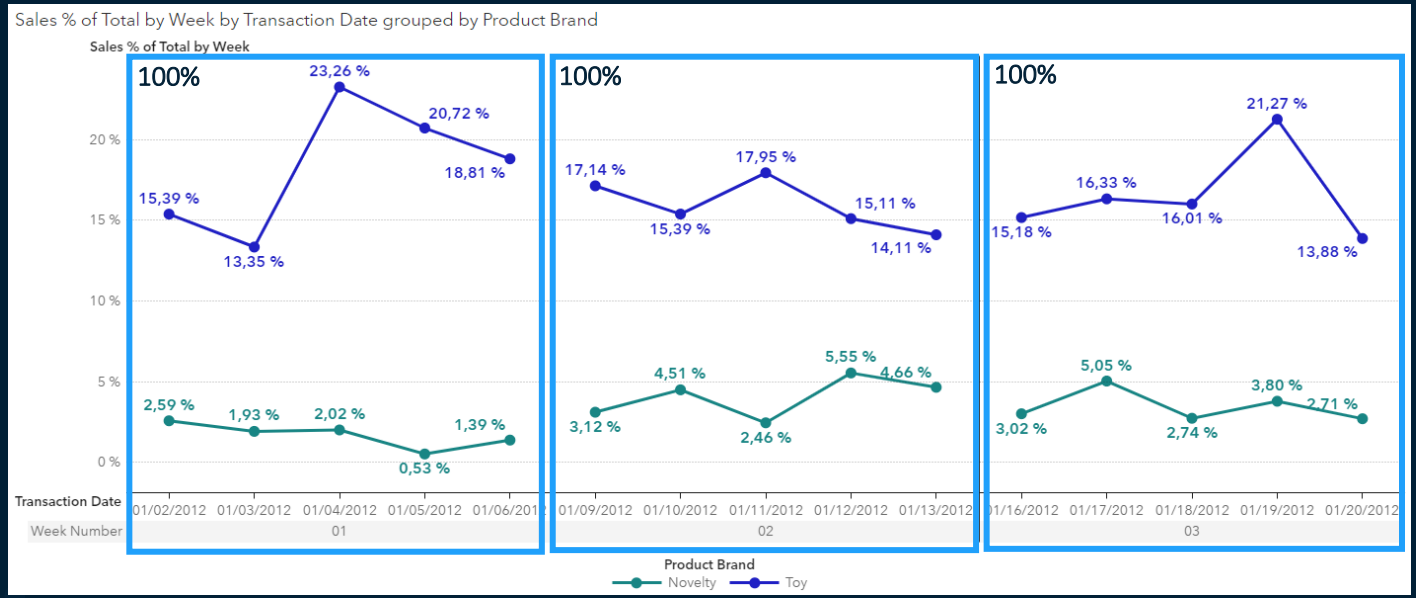
Sales % of Total by Week



#33 End users want to see sales % of total, while comparing it between selectable weeks and by Product Brand

A classic sub-totals and/or by-groups challenge

Solution



% of totals are now calculated by week (100% per week)

AggregateTable - Use-case #34

Outer vs. Inner Table

#34 Creating a KPI for Average Sales by Continent based on Total Sales by country

A very good example explaining how inner and outer table works

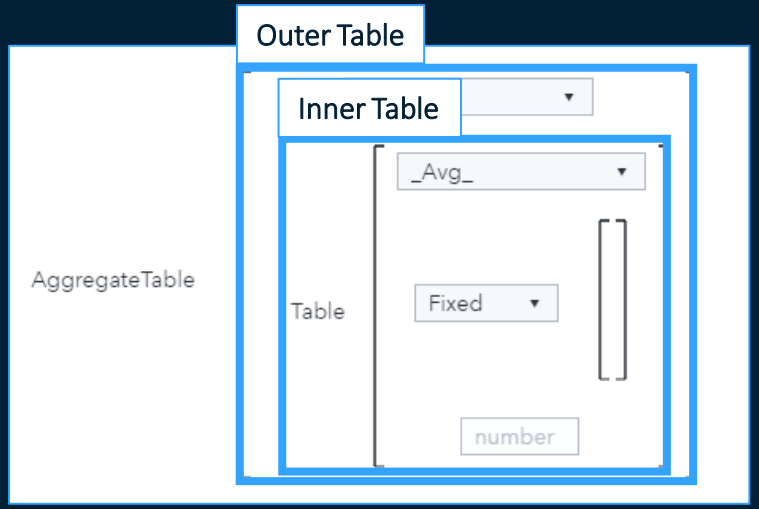
Facility Continent ▲	Average Sales based on total sale per Country
Africa	240 596
Asia	91 727
Europe	1 768 483
Oceania	301 284
South America	1 080 503

AggregateTable - Introduction

Outer vs. Inner Table

Aggregate data using different aggregation types in one calculation ☺

Outer Table is our final aggregated data we want to use in our visualization.



Inner Table is what we want to pre-aggregate.

Remember: Average Apple weight by TreeID

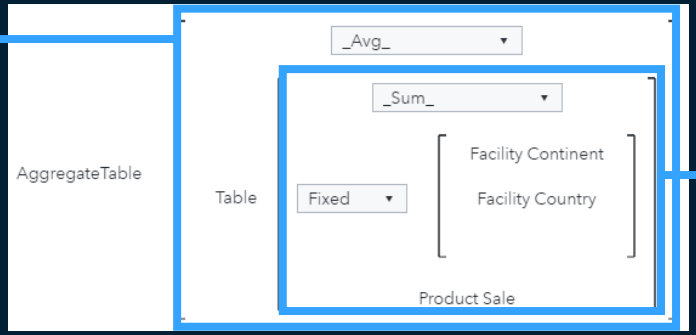
#34 Creating a KPI for Average Sales by Continent based on Total Sales by country

A very good example explaining how inner and outer table works

Solution

Outer Table

Facility Continent ▲	Average Sales based on total sale per Country
Africa	240 596
Asia	91 727
Europe	1 768 483
Oceania	301 284
South America	1 080 503



Inner Table

Facility Country	Total Sales
Egypt	207 747
Nigeria	277 147
Morocco	239 816
South Africa	237 676
South Korea	63 147
Russia	195 588
Israel	65 226
Saudi Arabia	61 040
Singapore	63 078
China	140 487
Indonesia	59 034
Japan	121 247
India	56 696
Spain	4 026 412
Denmark	492 724

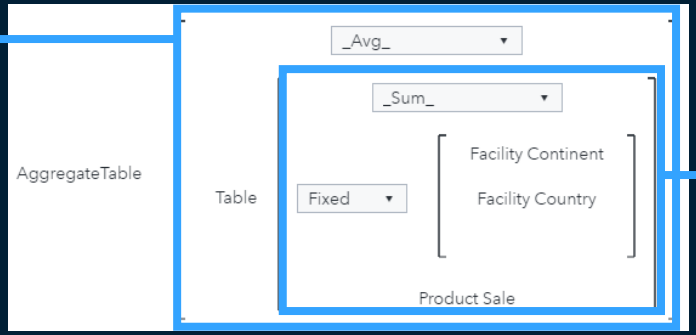
#34 Creating a KPI for Average Sales by Continent based on Total Sales by country

A very good example explaining how inner and outer table works

Solution

Outer Table

Facility Continent ▲	Average Sales based on total sale per Country
Africa	240 596
Asia	91 727
Europe	1 768 483
Oceania	301 284
South America	1 080 503



Inner Table

Facility Country	Total Sales
Egypt	207 747
Nigeria	277 147
Morocco	239 816
South Africa	237 676
South Korea	63 147
Russia	195 588
Israel	65 226
Saudi Arabia	61 040
Singapore	63 078
China	140 487
Indonesia	59 034
Japan	121 247
India	56 696
Spain	4 026 412
Denmark	492 724

Why do we need Facility Continent in our inner table?

Answer: Because we are using Facility Continent in our final visualization (outer table)

Summary: Tips using AggregateTable

- It is a very powerful function, training is key – use my use-cases to get started
- Could be an option to avoid creating unnecessary pre-calculated columns in our data
- Remember WYSIWYG ☺ - What you visualize might impact the result
  - Outer table is what you want to visualize
  - Use group-by options Fixed, Add or Remove
  - Hidden role can be useful as well

Fantastic article by Renato Luppi

SAS Visual Analytics Advanced Calculations (part 2 of 4): AggregateTable

https://communities.sas.com/t5/SAS-Communities-Library/SAS-Visual-Analytics-Advanced-Calculations-part-2-of-4/ta-p/538541

Q&A

Bonus Use-Case

Group-by Category options

- Fixed
- Add
- Remove

#35 Engineers want to review average milage by car model and production year, based on service data

A great example where the “add” option is needed

ProductionPeriod	ProductionYear	ChassiID	CarModel	ServiceDate	CarReadoutKM
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2020-07-12	4560
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2020-11-17	7650
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2021-11-30	10805
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2022-10-25	15870
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2021-09-01	11900
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2021-10-10	12345
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2022-06-30	18700
202008	2020	A10044	Audi A3 Sportsback 45 TFSI e	2021-11-13	9055
202008	2020	A10044	Audi A3 Sportsback 45 TFSI e	2022-06-12	15890
202008	2020	A10049	Audi A3 Sportsback 45 TFSI e	2022-01-16	15600
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2021-12-12	4570
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2022-03-25	7650
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2022-10-10	8700
202103	2021	A10129	Audi A3 Sportsback 45 TFSI e	2022-01-13	9800
202104	2021	A10141	Audi A3 Sportsback 45 TFSI e	2021-06-12	1560
202104	2021	A10141	Audi A3 Sportsback 45 TFSI e	2022-08-09	6790
202104	2021	A10145	Audi A3 Sportsback 45 TFSI e	2022-09-12	13450
202106	2021	A10147	Audi A3 Sportsback 45 TFSI e	2022-09-18	14560
202110	2021	A10161	Audi A3 Sportsback 45 TFSI e	2022-02-03	5675
202110	2021	A10161	Audi A3 Sportsback 45 TFSI e	2022-10-19	11900
202111	2021	A10189	Audi A3 Sportsback 45 TFSI e	2022-11-10	8045
202112	2021	A10212	Audi A3 Sportsback 45 TFSI e	2022-03-15	15605
202112	2021	A10212	Audi A3 Sportsback 45 TFSI e	2022-11-09	21890

CarModel	ProductionYear ▲	Average Readout Km
Audi A3 Sportsback 45 TFSI e	2020	14 952
Audi A3 Sportsback 45 TFSI e	2021	12 348

#35 Engineers want to review average milage by car model and production year, based on service data

A great example where the “add” option is needed

Using option “Fixed” will generate an average on max Km readout for all ChassiID’s, but we want average by Production Year…

ProductionPeriod	ProductionYear	ChassiID	CarModel	ServiceDate	CarReadoutKM		MaxReadout
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2020-07-12	4560		
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2020-11-17	7650		
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2021-11-30	10805		
202003	2020	A10001	Audi A3 Sportsback 45 TFSI e	2022-10-25	15870		15870
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2021-09-01	11900		
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2021-10-10	12345		
202007	2020	A10023	Audi A3 Sportsback 45 TFSI e	2022-06-30	18700		18700
202008	2020	A10044	Audi A3 Sportsback 45 TFSI e	2021-11-13	9055		
202008	2020	A10044	Audi A3 Sportsback 45 TFSI e	2022-06-12	15890		15890
202008	2020	A10049	Audi A3 Sportsback 45 TFSI e	2022-01-16	15600		15600
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2021-12-12	4570		
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2022-03-25	7650		
202012	2020	A10102	Audi A3 Sportsback 45 TFSI e	2022-10-10	8700		8700
202103	2021	A10129	Audi A3 Sportsback 45 TFSI e	2022-01-13	9800		9800
202104	2021	A10141	Audi A3 Sportsback 45 TFSI e	2021-06-12	1560		
202104	2021	A10141	Audi A3 Sportsback 45 TFSI e	2022-08-09	6790		6790
202104	2021	A10145	Audi A3 Sportsback 45 TFSI e	2022-09-12	13450		13450
202106	2021	A10147	Audi A3 Sportsback 45 TFSI e	2022-09-18	14560		14560
202110	2021	A10161	Audi A3 Sportsback 45 TFSI e	2022-02-03	5675		
202110	2021	A10161	Audi A3 Sportsback 45 TFSI e	2022-10-19	11900		11900
202111	2021	A10189	Audi A3 Sportsback 45 TFSI e	2022-11-10	8045		8045
202112	2021	A10212	Audi A3 Sportsback 45 TFSI e	2022-03-15	15605		
202112	2021	A10212	Audi A3 Sportsback 45 TFSI e	2022-11-09	21890		21890
					AvgeragMaxReadout All Cars:		13433

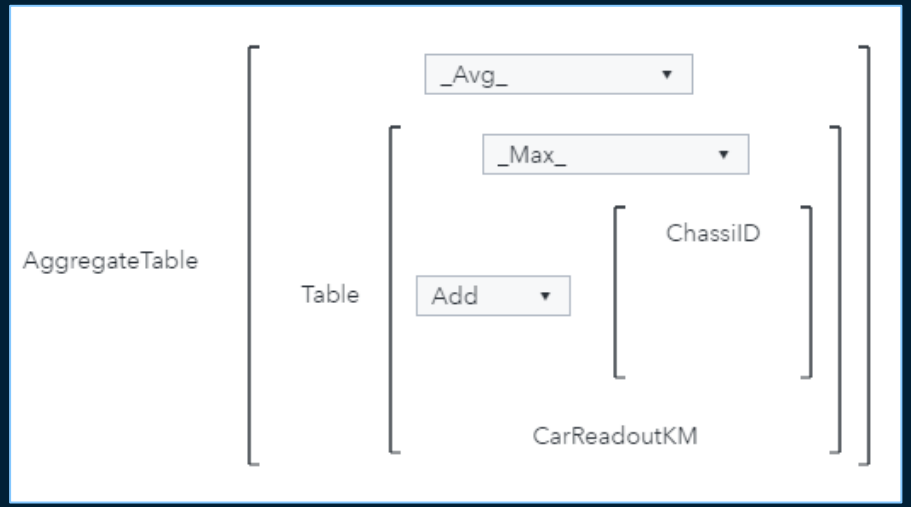
This is wrong!!!

CarModel	ProductionYear ▲	Average Readout Km
Audi A3 Sportsback 45 TFSI e	2020	13 433
Audi A3 Sportsback 45 TFSI e	2021	13 433

#35 Engineers want to review average milage by car model and production year, based on service data

A great example where the “add” option is needed

Solution



Think like this:

Option “Add” adds ChassiID into our inner table calculation, without the need to visualize it ☺

Or, adding a new non-visual dimension into our calculation ☺

#35 Engineers want to review average milage by car model and production year, based on service data

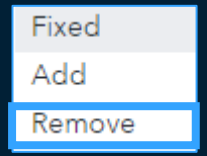
A great example where the “add” option is needed

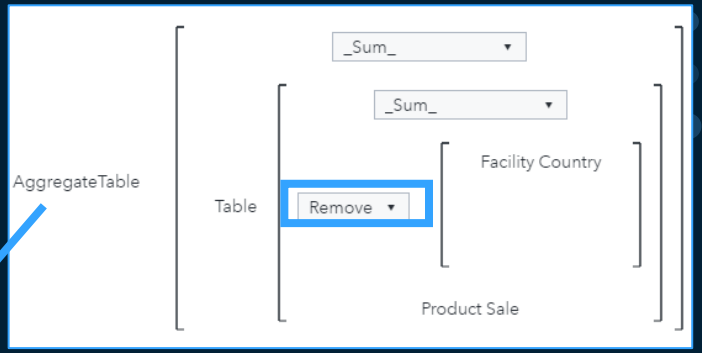
Solution

CarModel	ProductionYear ▲	Average Readout Km
Audi A3 Sportsback 45 TFSI e	2020	14 952
Audi A3 Sportsback 45 TFSI e	2021	12 348

AggregateTable – Example using Remove

Group-by Category options





Option Remove means that we are removing Facility Country from our inner table calculation

Facility Continent	Facility Country	Product Brand	Product Sale	AggregateTable: Product Sale by Country (Remove)
Europe	Denmark	Novelty	202 747	5 590 681
		Toy	289 977	8 557 184
	France	Novelty	245 680	5 590 681
		Toy	337 650	8 557 184
	Germany	Novelty	640 820	5 590 681
		Toy	950 415	8 557 184
	Italy	Novelty	411 230	5 590 681
		Toy	592 913	8 557 184
	Norway	Novelty	390 460	5 590 681
		Toy	627 400	8 557 184
	Spain	Novelty	1 602 235	5 590 681
		Toy	2 424 177	8 557 184
	Sweden	Novelty	619 261	5 590 681
		Toy	936 735	8 557 184
	United Kingdom	Novelty	1 478 247	5 590 681
		Toy	2 397 918	8 557 184

Benchmark use-case, calculating Product Sale % of total by Product Brand only, but we still want to use Facility Continent and Country in our crosstab: