Cluster Analysis – Using Flag Arrays

Release Date

2017 Q3

Summary

Flag array dimensions can now be used in Cluster analysis as long as they have 32 or less flags.

A full explanation of how the Cluster tool works can be found in the Modelling training manual that can be downloaded from the D&B website.

The Cluster Analysis requires the ability to calculate the distance or 'proximity' between two points. Also the iteration when using the KMeans technique requires the ability to take frequency counts across the dimensions at cluster centres and from that compute a new 'centre of gravity'. For each dimension coordinate a value component for this mid-point needs to be derived.

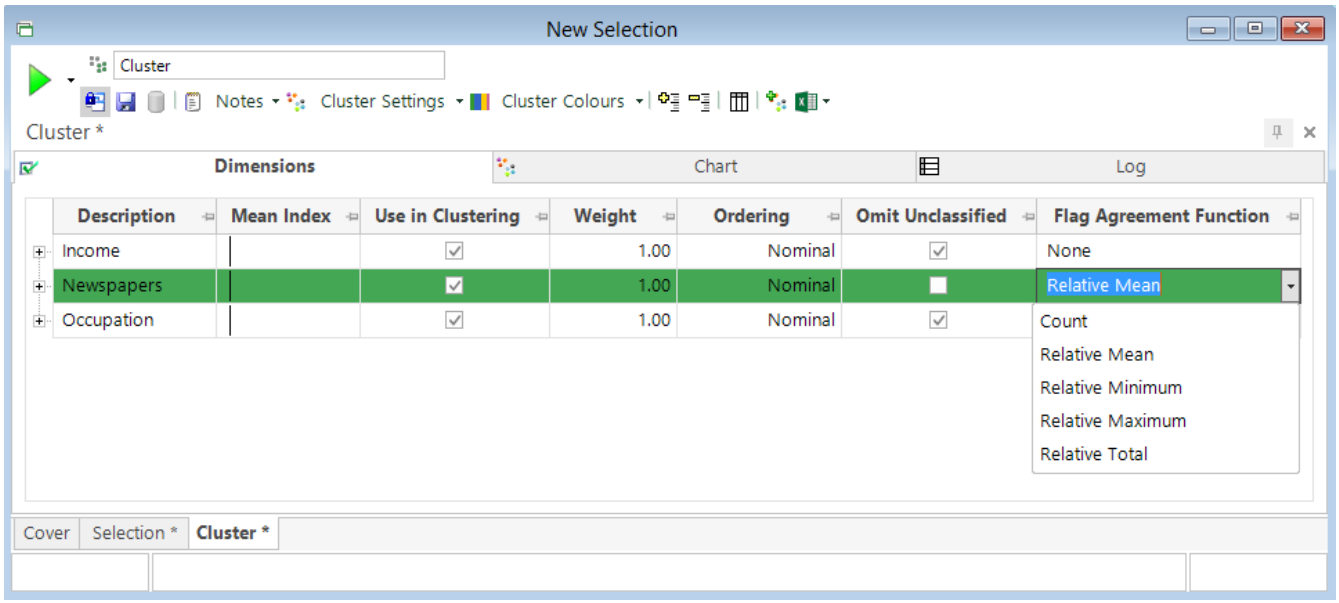
The added complication when using a flag array is the possible multiple yes flags that have to be considered when trying to calculate the required values as described above. This software release now supports the use of flag arrays with 32 or less flags e.g. the Newspaper variable in the Holidays system.

- On the **Dimensions** tab of a **Cluster** window right click on a column heading and select **Column Chooser**
- Tick the box next to **Flag Agreement Function** and click **OK**

A new column has now been added to the display. Only the Flag Array variables will have options in this column all others will show as 'None'.

- Click on the drop down arrow to show the list of possible **Flag Agreement Functions**

The Relative Mean function is used by default.



The Flag Agreement Functions available are as calculated below using the following two records as an example:

Record 1: Newspapers = Times, Guardian, Sun, FT

Record 2: Newspapers = Telegraph, Guardian, FT

Each record is represented as a bitmap in FastStats and analysed as such.

Record 1: Newspapers = Times, Guardian, Sun, FT - 1010001001

Record 2: Newspapers = Telegraph, Guardian, FT - 0001001001

But how similar are these two records in terms of newspaper readership? A proximity measure calculates how close with 1.0, meaning that the records are the same (or at least treated as equivalent) and 0.0 meaning that they are "completely different". Notice that the sense of proximity (same / different / close / distant) is going to depend on your interpretation of the data. Even knowing the source of the data you will still need to consider the context - in the above example you might say that these records are similar or not depending on your analysis aims.

The proximity is calculated based on the relative agreement between the two sets of flags. This is calculated from the number of Yes flags that are in agreement (e.g. FT, Guardian) compared to the number of Yes flags overall. There are various ways of doing this controlled by the Flag Agreement Function dimension property:

Count

e.g. min number Yes / max number Yes Proximity = 3/4 in this example

Relative Mean

e.g. matched Yes / mean number Yes Proximity = 2/3.5 in this example

Relative Minimum

e.g. matched Yes / minimum number Yes Proximity = 2/3 in this example

Relative Maximum

e.g. matched Yes / maximum number Yes Proximity = 2/4 in this example

Relative Total

e.g. matched Yes / total number Yes Proximity = 2/7 in this example

There is a further special case where no flags are set on either record e.g.

Record 1: Newspapers = none -> 0000000000
Record 2: Newspapers = none -> 0000000000

But how similar are these two records in terms of newspaper readership? We could say zero - why assume there is any similarity when we probably just don't have any information? Or we could say 1 - these are two people who have the same readership - they don't read any newspapers. The value 0 causes problems in the K-means adaptation - if a centre is established at 0,0,0,0,0, …then every point with the same coordinates is maximally distant from itself. The value 1 seems far too high, intuitively this negative match should be lower than a positive match. The value settled upon is 1/N where N is the number of flags. This is the same for all flag agreement functions.

Consideration should be given to what data a variable's flags represent and how the similarity should be interpreted when selecting a function to use. It may be advisable to try a number of examples before making a final decision.