Data Summary How To
Matthew Weinstein
10/24/03

The data summary (Report->Data summary->Data summary…) can provide complex counts of your data. It works on the same principle as the program dataBoiler and many other databases; it creates summary records to provide counts or sums of where your data changes value (as it is scanned from top to bottom). To produce a data summary you first must sort your data so that the program can summarize. The sort order determines the summary; as the following examples demonstrate. After sorting your data you need to specify the fields that the program will summarize and the field that contains the data you want to count or sum. What TA does is work through the visible data and when one of your group fields change value it creates a summary record saving the count/sum up to that point. It then zero's the count or sum of itself and any groups lower than it in the group list. Finally it presents the saved sums in a record that keeps the different sum/count values in different columns, along with a total count/sum

For each group, and for the field that you are counting/summing you need to specify the field (column), the type of data in that field (alphanumeric, code, integer, real, or date: it inherits the date format from the results window), and for code data how many levels to compare.

Code data is a new data type for both sorting and grouping in TA. You need to specify a code level for any time you sort or group code data (usually the _code field of your results window). A 0 (zero) code level means that the codes must be absolutely identical (not counting coder information) to be considered the same group. If the code level is 1 (one) then a, a>b, and a>c>z are all equal; the program will not save the count until it finds something different (i.e., they are all in the same group). A code level of 2 means that a>b and a>b>c and a>b>z are all the same, but they are not the same group as a or a>c. The importance of this should become clear with the examples below.

When determining what to count, the user has two options that are very useful. The first, count duplicates in the summary column when unchecked only will count one instance of a series of the same value in the column being counted. An example where this may be useful is in determining how many documents use each code. This example is given below.

The other switch controls whether blanks count in doing a count. In other words does the program count every record or only records of the field being counted/summed where there is some data. An example that might require you to uncheck this is one where you wanted to see how many comments were attached to each code.

Examples:

1. Code count

An alternative to the code count report (it's easy, and it will set us up for the next example). Here we simply want to know how many passages have been coded for each code.

First sort _code field alphanumerically. Click on the column header for _code and pick the Result->Sort up->alpha menu option

Pick Result->Data summary->Data summary…

Create one group by picking _code, pick Code as its type, and 0 as its code level.. To add this as a group click the + button.

It doesn't matter what you count; you might just pick _code again and make it an alpha type field. Make sure that you indicate that you want a "Count of records"

Leave both options checked

Hit OK and you should have an alphabetic count of your codes.

2. Code family count

In example 1 families were counted separately: a and a>b were two different codes to be counted. Here we're going to find a count of each code family: a, a>b, a>c>x will all be counted as one item: "a".

To start with set the code level of your result window to 1: pick Result->Code level. Then sort the _code column by Code. Click on the column header for _code and pick the Result->Sort up->alpha menu option

Pick Result->Data summary->Data summary…

Create one group of field _code, type Code, level 1 (typically you will group in exactly the same order you sort)

Again, it doesn't really matter what we're counting, keep both options checked and click ok

3. Count of codes in each document

Example 3 breaks down the data a little more. It will show you what the count of codes are in each document. Like the other examples you will sort and then group the same fields in the same order (if you sort _code and then _doc you'll group _code and then _doc)

Here we're going to sort up the _doc column as an alpha first. Click on the _ doc column and pick the Result->Sort up->alpha menu option.

Then pick the _code option (click on the column title) and pick Result->Sort up->alpha (same as Result->Sort up->code with code level = 0) within menu option the "alpha within" means that the first sort will be preserved.

Now go to your Result->Data summary->Data summary… menu option and create the same groups you sorted on. First pick from the "Add group" _doc of field type "alpha" and click the + button, then from the same part of the dialogue pick _code of field type "code" and pick 0 as the level. Click the + button to add this second group.

Again we're just counting all records so it really doesn't matter what you count. Just pick _data of type alpha; make sure the two checks are both on and press OK. Voila

4. Count of code families in each document

It should be pretty obvious that this combines examples 2 and 3; but just to review.

Here we're going to sort up the _doc column as an alpha first. Click on the _ doc column and pick the Result->Sort up->alpha menu option.

Set the code level to "1" by picking Result->Code level… menu option and typing in 1 (one). Then pick the _code option (click on the column title) and pick Result->Sort up->code within menu option the "code within" means that the first sort will be preserved.

Now go to your Result->Data summary->Data summary… menu option and create the same groups you sorted on. First pick from the "Add group" _doc of field type "alpha" and click the + button, then from the same part of the dialogue pick _code of field type "code" and pick 1 as the level. Click the + button to add this second group.

Again we're just counting all records so it really doesn't matter what you count. Just pick _data of type alpha; make sure the two checks are both on and press OK. Voila

5. Count of documents in which each code appears.

In this example we find out how many of your source documents (not which) each code appears. To do this we'll need to use some options that we haven't used before.

First select and sort the codes by alpha (use the examples above to figure out how: hint select the column by clicking on it's title and then use something on the Result->sort up menu).

Second sort the _doc column as "alpha within" so that you don't lose your sorting of _code.

Now go to the Data summary… dialogue (Result->Data summary->Data summary…) and create the _code group by picking _code of type alpha (or _code with level 0) and press the + button.

Fourth, pick the count of records for _doc that is type alpha and then uncheck "Count duplicates in summary column" because you want each document name to be counted only one time.

Press OK and you should have a nice chart of how many documents each code appeared in.

REMEMBER: THE SORTING IS IMPORTANT SINCE THE DATA SUMMARY ROUTINE JUST STARTS AT THE TOP OF THE VISIBLE RECORDS AND SIMPLY MONITORS (WHILE COUNTING AND SUMMING) WHEN THE VALUES OF YOUR "GROUP" VARIABLES CHANGE.

Final note: you can save your grouping and count field information to be modified or to be used in other results by filling in a name, checking the project wide box (if you want these criteria to be available to other result windows), and pressing the + button by the pop up at the top of the window. As always in TA + means add a report/step/set – means delete a report/step/set and - - means delete all reports/steps/sets.