Filtering Join Challenge/Tutorial

Here we have a dataset preparation problem inspired by some of the problems discussed in my blog entry 'Why I Love the New Filtering Join Node'. For those who want the challenge of trying it themselves, I will start by laying out the problem with some sample datasets. I will then explain my solution to this problem in some detail. I will also attach a corresponding Angoss project of my solution for those who want to play around with it (and maybe even improve on it)!

A couple of notes:

- There are certainly multiple ways to complete the data prep task here. Many of them involve changing the order slightly, but there may be more substantial changes.

- It's also possible that there is a solution which is much more elegant than the one I found here. If you find one, I'd love to see it.

- Also, all the data here is completely made up. Don't try to build any models on the resulting datasets you get from this project, you won't find anything of value.

Ok, so here's the problem set up:

Part 1:

You are working on a project where you will eventually try to find a relationship between end sales and trade spend at a city level.

You have 2 primary datasets to work with, one with end sales data, and one with trade spend data.

- The EndSales dataset is at the store level. The dataset is simple, it contains a store ID for each store, the city and country that store is in, and the end sales value that store produced this month.
- The TradeSpend dataset is also at the store level. It is even simpler, and contains just a store ID for each store, and a trade spend value given to that store this month.

Your goal is to create a single dataset at a *city level* with the total trade spend and end sales value for each city.

However, this dataset should only include cities in scope for this project. The original datasets have data for all of the 50 largest cities in North America, but the project you are working on is supposed to include only cities in The United States, Mexico, or the two largest cities in Canada (Toronto and Montreal).

To prevent the problem from becoming harder as North American demographics change, I have included a dataset with the current list of the 50 largest cities in North America. In a real problem, you might need to obtain this data yourself. As a challenge, you can still try that, but your numbers may disagree if you use a different source than me, or if the world has changed since I created this problem.

If you like, you may begin part 1 now, but the problem is ambiguous at this point. It is ambiguous because if you are careful, you will come across challenges which don't have a single right answer.

Unfortunately, in order to explain my preferred solutions to these challenges I need to explain what the challenges are, but at the same time, I think the problem is most fun if I don't tell you what the challenges are in advance, and you come to them yourself.

So, I will explain the challenges and my preferred solutions below, but if you are feeling ambitious, I suggest trying to solve the problem without reading the challenges, and only read on when you are stuck, or when you want to see if you have the same solutions as me.

Challenge #1: Some stores in the EndSales dataset have multiple entries, and in fact belong to multiple different cities.

Solution #1: My preferred solution is to exclude all such stores. In a real-world problem, you might be able to get a clear answer about why a single store exists in multiple cities, but in this case, you can't. There is also no way to tell which city is the right one, or what information is contained in the erroneous entry. Simply excluding these stores is, in my opinion, the best compromise with no additional information.

Challenge #2: Some stores in the TradeSpend dataset have multiple entries.

Solution #2: This has nothing to do with city issues, as the TradeSpend dataset doesn't even include cities. In this case I chose to assume that the same store might get two separate injections of trade spend money, and that this is not a data quality issue. As a result, I believe the best choice here is to aggregate the dataset to include the TOTAL trade spend per store.

Part 2:

As it turns out, the solution to Challenge #2, (the challenge that some stores had multiple different entries in the TradeSpend dataset) was only partially correct. You have just been informed that there may be a few stores where the multiple entries were caused by fraud, and not by multiple injections of trade spend. Luckily, this fraud is characterized by the same store getting multiple injections of the exact same amount of trade spend. (In other words, any store with 2 trade spend entries of different amounts are both real, but any store with 2 or more identical trade spend amounts is fraudulent). You have been asked to remove all fraudulent stores from the analysis entirely, and then re-run the process.