**1-7– Gathering Numbers**

The transform tool set of Alteryx summarizes and rearranges data to prepare data for reporting and analysis. One of the ways it summarizes data is by applying an aggregation method to fields which accumulates the data from record level to the grand total level. One of the transform tools that maximizes this functionality is the weighted average tool.

**Weighted Average Tool is used to calculate the weighted average of an incoming data field. A weighted average is similar to a common average, but instead of all records contributing equally to the average, the idea of weight means some records contribute more than others. This takes into account the relative importance or frequency of some factors in a data set. In calculating a weighted average, each number in the data set is multiplied by a predetermined weight before the final calculation is made.**

The input we are going to use is this supermarket sales dataset which contains sales transaction within the year 2019. We are going to find the weighted average by each branch and product line using Quantity as the weight and Gross Income as value. Place a weighted average tool from the transform palette and connect it to the output anchor of the select tool. The first configuration is where we need to set the numeric value field which contains the data to return the weighted average. Let’s set this to “gross income”. Next, set the numeric weight field. As mentioned earlier, the weight column is needed to be present in the dataset to show the relative importance of each record. If only 1 weight is assigned for the whole dataset, its returned value will be same to that of the common average. We will map the weight to “Quantity”. Next is the Output field name. By default, it is set to “WeightedAverage”. We will just keep the default as is. Finally, this setting is optional, but you can tick the checkbox for the fields you want to group on. The weighted average will be calculated for each group. Not specifying a grouping field will result to a single weighted average value for the whole dataset. We will tick “Branch” and “Product Line” to group the calculation according to these fields. Once configurations are all done, run the workflow. The output contains 18 records grouped by branch and product line. The weighted average field is also appended. Food and Beverage of Branch C has the greatest average value of 23.038 while Health and Beauty of Branch A has the least with 15.167.

**Running Total is the cumulative sum of a value as it grows over time. It is quite common in everyday use. When buying groceries, cash registers show you a running total of the goods that has been scanned. Once all products are taken accounted for, it also shows you the total amount that you need to pay for. Another example would be game scoreboards. Every time you get a point, it is added to the total score you’ve accumulated. This method is also possible in alteryx using the Running Total Tool in alteryx.**

The **Running Total Tool** is used to calculate a cumulative sum on a numeric field per record in a data stream.

Let’s create a running total on our supermarket sales dataset using the gross income field. From the transform tool tab, drag the Running Total tool into the canvas, place it below the weighted average tool and connect it to the output of the select tool. The first configuration of this tool is optional. It lets you set a field to group by. Meaning, the running total will reset once it reaches a different group. For this example, we are going to count the gross income by branch. Tick the branch checkbox to set it as a grouping field. Next, we need to select the numeric fields that we want to create a running total. You choose 1 or more field and its output will have the prefix “RunTot\_” to show that it is the running total of the specific field. Check “gross income” then run the workflow. At the end of our output records is the RunTot\_gross income which was generated by the tool. As we can observe, the first record is equal to the first gross income which is 23.288, but has increased to 43.94 on the 2nd row since the 2nd gross income of 20.652 was added to it. This cumulative sum proceeds to the rest of the dataset and only resets when we reach record 341 which was under a different branch, thus a different grouping.

**Count Records Tool returns a simple count of the number of records that pass through the tool. You can use this tool when you want to report on the resulting record count of a process. It even returns a count of 0 which is something that the Summarize tool doesn't do.**

Place a Count Records tool from the Transform palette to the end of the workflow. Connect it to the output anchor of the running total tool. The Count Records tool does not require any configurations, so you simply need to run the workflow to show its output. The tool outputs a single row under the column name “Count”. It shows that our running total data stream has 1000 records in it. The summarize tool also has its own count function, but it will not give you any output if there were no records on the data stream. Unlike the summarize tool, the count records tool will still give generate the output value of Count = 0 if your workflow had no records which can be essential in avoiding errors and in monitoring your scheduled workflows.

**Arrange Tool lets you manually transpose and rearrange your data fields for presentation purposes. Data is transformed so that each record is turned into multiple records, and columns can be created by using field description data.**

We will use the arrange tool to arrange this text input that contains the sequence and location of a bus route in London. Each record has the Sequence Number, Stop Name, Latitude 1 and Longitude 1 for the coordinates of the route’s origin. And Latitude 2 and Longitude 2 for the coordinates of the destination. What we would like to have as output is to have a single column for the latitude and a single column for longitude, with an additional column that dictates whether the coordinate was an origin or a destination of the bus stop.

First, drag an arrange tool into the canvas and connect it to the text input. The first configuration of the tool which are the key fields works similar to that of the transpose tool. Meaning, every field that you select will be preserved on the output. If you merge multiple fields and create additional rows on the arrange tool, the Key fields will be duplicated for each new row. We wanted to retain the Sequence and Stop Name on our output so we are going to check these 2 fields. The next configuration is where we can create and manipulate the output fields. You must first create an output column before you can create an additional row. The order you select fields in the output fields window in each field is the order they will be transposed in for each row. Add a column by clicking the column dropdown and selecting “Add”. This opens a new window where you can type a column name, set its values, and set up a description column. The first column we are going to create is for the latitude, Type in the column name “Latitude” then check “Latitude1” and “Latitude2” for its fields to set it as the values. The Description Column will be the new field that will contain the tag for our “Origin” and “Destination” route. Name the description header as “Route”. This also gives identification to the new columns that we’ve created. We will leave the other description options to the default so we can modify the tags later on. Hit OK to close the window and reflect the new column we created into the configuration box. Next, we need to add another new column for the longitude. Click column drop down again and select “Add”. Type the Header name “Longitude” and check “longitude1” and “longitude2” for its values. Since we already added a description column using the latitude column that we created, the Route field is already mapped here. Hit OK to finish this column. Now that we have the 2 columns we need, we can now add the tags for our Route or description field. For the first line which has latitude1 and longitude1, type in “Origin” under the route column. Once done, type “Destination” on the 2nd line. This completes the configuration for our arrange tool. Run the workflow to check the output.

As we can see from the output, we have retained the key fields; sequence and stop name, and we have additional 3 new columns, Route, Latitude and Longitude. Before the arrange tool, we have a unique sequence and stop name for each row but since we moved the origin coordinates and destination coordinates to a single column, the Key fields were duplicated.