Practice Exercises for Week 3: Dynamic Data Manipulation and Presentation in Tableau

These practice exercises will direct you through some of the analyses in our analysis plan that you can run on the "dognition_data_no_aggregation" data set using the Tableau techniques you learned this week. Please refer to the reading in Week 3 > "Dognition Data Set, Data Set Description, and Analysis Plan" for a description of the data set, and to download the data set. In the text below, all variable names will be depicted in *italics* and all properties on the Marks Card will be depicted in **bold**.

Recall that the metric we are trying to increase is the number of tests customers complete. In the data set we used last week, we had one row for each Dog ID, and a column (*Total Tests Completed*) that provided the total number of tests each dog associated with a Dog ID completed.

In the data set we will use this week, there is a separate row for each test completed by each dog. That provides us with the opportunity to answer some new questions, but it also adds a layer of complexity behind the scenes of our visualizations. This is because the values in *Total Tests Completed* can only be understood at aggregation level of *Dog ID*, while other variables we are interested in have a different level of granularity (such as at the level of individual tests). Pay careful attention to this issue as you work through your analyses. If you run into trouble with your visualizations or calculations, it's probably because you are trying to combine variables with different granularities, or trying to combine variables that are aggregated at different levels.

This data set is a little larger than the data set we used last week, so if you find your computer is running slowly, you may want to try saving your data as an "extract." For a full description of what extracts are, see:
http://www.tableau.com/about/blog/2014/7/understanding-tableau-data-extracts-part1.

For a description of how to save extracts, see:
http://onlinehelp.tableau.com/current/pro/online/mac/en-us/extracting_data.html.

If you encounter any problems you cannot solve on your own while working through these exercises, post your questions to the Week 3 discussion forums on the course website ask your peers for help!

Exercise 1

Let's begin by removing entries we know are not real data. Recall from last week that Dognition's convention for indicating when an account was a "testing" account was to enter a weight that was completely infeasible for the entered breed. In particular, their testing accounts often contained Shih Tzus that were 190 pounds (Shih Tzus typically weigh between 9 and 16 pounds).

In order to remove these data points from our analyses, create a row calculation that can be used on the Filter shelf to remove all entries of Shih Tzus that weigh 190 pounds. This calculated field should use an IF/THEN/ELSE statement, with the AND operator to output one value for each row of data that indicates whether the row belongs in the "keep" category or the "exclude" category. For the rest of the exercises using the dognition_data_no_aggregation data set, use this calculated field on the Filter shelf to exclude Dognition's testing accounts from your analyses.

Now that we can exclude these data easily, let's take advantage of the new level of detail the dognition_data_no_aggregation data set provides over the aggregated data set we used last week by finding out after which games users tend to drop out. To achieve this, we need to use *Rank by DogID*.

It's worth taking some time to understand exactly what this variable provides. Recall that Dognition customers are given the tests in the same order each time (with a few exceptions that Eliot told us about in the "Meet Your Dognition Data" video. To facilitate our initial analyses, I ordered each test a dog took by its time stamp in *Created At*, and included the rank the test received in this sorted order in its own variable called *Rank by DogID*. A rank order of 1 in this field indicates the test described in that row of data was the first test that dog took, a rank order of 5 in this field indicates that the test described in that row of data was the fifth test the dog took, and so on.

I computed a similar rank for all the tests associated with a User ID. This rank, stored in *Rank by UserID,* indicates the order of tests used by each human user. When a User ID is only associated with one Dog ID, the values of *Rank by DogID* and *Rank by UserID* in a row of data will be the same. When a User ID is associated with multiple Dog IDs, the values of *Rank by DogID* and *Rank by UserID* in a row of data may differ. You will have to aggregate these variables appropriately to extract the information you want.

The *Total Tests Completed* variable we used when analyzing the dognition_data_aggregated_by_dogid data set last week was the maximum *Rank by DogID* value associated with each *Dog ID* in the dognition_data_no_aggregation data set. This week, when we need it, we will have to retrieve the maximum rank value associated with a Dog ID ourselves through the level of aggregation we ask when using *Rank by DogID*.

To determine when users are dropping out of the Dognition test progression, make a bar graph with *Rank by DogID* as a dimension on the columns shelf and *Number of Records* on the Rows shelf. Remember, the Dognition tests are organized into subcategories of cognitive abilities and personality attributes the tests are meant to assess (there are 5 subcategories in the first 20 tests comprising the Dognition Assessment). I highly recommend that you place *Subcategory Name* or *Test Name* on **Color**. After which tests do users tend to drop off? Do they drop off in the middle of the tests within a subcategory, or at the end of all the tests associated with a subcategory? What could this mean for Dognition?

Exercise 2

In order to better target advertisements or reminder emails, it would be useful to know when customers tend to play the Dognition games. To assess this, we can take advantage of the information provided in *Created At* in the dognition_data_no_aggregation data set. *Created At* has a time stamp of every test recorded by a customer (this information was lost when we used the version of the data set that aggregated over Dog ID last week). If we place *Created At* on the Columns shelf and *Number of Records* on the Rows shelf, we can use the full extent of Tableau's date hierarchy to see when tests are taken most often. During which days of the week are customers most likely to play the games (hint: use the "weekday" level on the date hierarchy)? During which hours of the day are customers most likely to play the games (hint: use the "hour" level on the date hierarchy)?

Look carefully at your visualizations that display which hours of the day customers are most likely to play games. Does anything look suspicious to you? It should! The data currently make it look like a lot of games are played in the middle of the night, but that doesn't make sense. If you look at the raw data behind your graphs, you'll start to get a feeling of what might be going on with this field of data.

It should become apparent that the time stamps are provided in "Coordinated Universal Time" (UTC) and do not take time zones or daylight savings into account. Unfortunately, it is not very easy to adjust UTC time stamps for changes in time zones in Tableau (or most programs, for that matter). The best way to accomplish such a correction would be to find or assemble a separate data set that would provide a UTC correction for every entry in one of our location-related variables, and then blend that secondary data set with dognition_data_no_aggregation (a process we learned about in Lesson 3 this week).

These data sets are not easy to find, but for the purposes of this course, we have assembled a data set that provides UTC corrections for a collection of United States zip codes. We will use a new method, called joining, to combine this data with the dognition_data_no_aggregation data set. The results of a joining will be easier to work with than the results of data blending in these specific circumstances.

To join the zip code correction data with the dognition_data_no_aggregation data set, download a separate file called dognition_data_no_aggregation_with_time_zone_correction from the course website. This excel file will have the original dognition_data_no_aggregation data in one worksheet called "master table," and the zip code correction data in a separate worksheet called "time zone correction." Connect to this file with Tableau.

On the initial data connection screen, drag both worksheets to the box where it says "drag sheets here" (it might take a while for the data to fully load). Make sure "master table" is on the left and "time zone correction" is on the right in the workspace. You should see that the worksheet names are connected with a line and two circles. Click on the circles and change the default from "inner" to "left join." Choosing a left join will make sure you retain all the data from the master table, and will incorporate data from the time zone correction

worksheet whenever there is a zip code that matches between the two files (You will learn more about joins in the next course of this specialization, Managing Big Data with MySQL).

Then go to a Tableau worksheet. You will see that *Diff from UTC*, which is the variable we essentially imported through the left join we just implemented, is in its own "time_zone_correction" heading in the variables pane. You can now use this variable as any other variable.

We now have what we need to correct *Created At* for differences in time zone. Make a new row calculation that will adjust each value of *Created At* by the correction provided in *Diff from UTC*. Dates have their own unique functions for adding and subtracting values; I suggest you use the DATEADD function. Type the following formula into your calculated field:

DATEADD('hour',[Diff from UTC],[Created at])

Make sure you specify "hour" in this formula so that the calculation knows to adjust the date at the level of hour rather than minute, second, or year, etc. Put your new calculated variable on the columns shelf, and number of records on the rows shelf. Keeping your new variable as a dimension, adjust it so that it is displaying data at the level of "hour" in the date hierarchy. Right-click (control-click) on the "Null" subheading in the graph and choose "Hide."

Next, make sure your calculation is outputting what you expect it to by right-clicking (control-clicking) on one of the data points to look at the underlying data. You should be able to see the original version of *Created At*, the new corrected version of *Created At* (which will have whatever title you gave it when making the calculated field), and *Diff from UTC* for each row. Does it appear that the calculation was implemented correctly? If not, double-check your calculated field.

When you are confident your calculation is correct, you are ready to interpret your new, time-zone corrected results. To aid in this, bring another copy of your (Corrected) *Created At* variable to the filter shelf. Choose the options that allow you to filter by "Weekdays." Show the quick filter. What hours of the day do customers tend to play games (although, remember, these results only reflect US countries where you have zip code data)? Do those hours change at different times of the week? What implications could this have for when you should send reminders or advertisements to customers or potential customers?

Exercise 3

We heard Eliot describe in the "Meet Your Dognition Data" video the story of how Dognition tried giving the Dognition tests to some customers in a different order than they typically do now. The hope was that doing so would increase the number of tests users completed overall (it didn't, according to Eliot).

During discussions off-camera, Eliot shared some information about other experiments Dognition has implemented as well. In particular, Dognition periodically tries offering a "Free Start" promotion to customers that gives the customers the first four tests for free. The hypothesis (or hope) would be that once potential customers get a chance to experience the product first-hand, they will be more likely to buy a subscription. In this exercise, we are going to assess whether the "Free Start" promotions worked.

Free Start User indicates whether the user began their Dognition experience with a free start. Currently *Free Start User* has three possible values: 1, 0, and NULL. The Dognition team confirmed that the 0 and Null entries should be considered the same group, and neither group had free starts. To implement the suggested grouping, right-click (control-click) on *Free Start User* to make a grouped variable that has just two options: "Free Start" or "No Free Start." Next, use either a visualization or text tables through the Marks Card to get an idea of how many data points you have for both categories. Also examine when Free Start users were enrolled, and how many non-Free Start users were enrolled at the same time. This will give you an idea of the kind of control groups to which you might have access when drawing conclusions.

Next, put the *Free Start User(Group)* on the Columns shelf, followed by *Rank by DogID* as a dimension. Put Dog ID on the rows shelf, aggregated by Count (distinct). You should see two graphs that depict the number of dogs who have completed 1- 45 tests, one for those who began with a free start, and one for those who didn't. These visualizations make it look like perhaps Free Start users do not finish as many tests as non-Free Start users, but it's hard to know for sure because there are so many fewer Free Start users to start with. A more direct way to assess the success of Free Start users would be to compute a table calculation that would tell you what percentage of users who start the Dognition Assessment make it to each test along the way. Fortunately, this is one of the default table calculations Tableau offers.

Without removing any of the pills you currently have on the Columns or Rows shelves, drag another instance of Dog ID to the Rows shelf. Click on the drop down and choose quick table calculation option, followed by "percent of total." You should now have one set of bar charts that show you the raw numbers of dogs who completed a certain number of tests, and a second set of charts that show you what percentage of a total those raw numbers represent.

The goal in making the table calculation was to determine what percentage of those who start their Dognition experience with a free start finish each number of tests. Do the charts you see depict that? Probably not, because the calculated field is likely making calculations with <u>all</u> the data in the work space together at the same time, rather than doing the calculation separately for different partitions within the work space. You have to tell Tableau what partitions the table calculation should take into account. To do this, click on the table calculation to edit it. Can you figure out how to set the advanced options so that the table calculation computes percentages separately for Free Start users vs non-Free Start users? Based on these results, do you think the Free Start tests seem to be working as intended?

Exercise 4

In this exercise, we are going to use table calculations to determine how we could use Tableau to dynamically recreate the ranks stored in the *Rank by DogID* and *Rank by UserID* variables. These variables, remember, chronologically rank each test according to its associated time stamp in *Created At*, and restart the ranking either for every dog or for every customer.

To begin, make a group of Dog IDs that only has a few Dog IDs included (randomly choose 4 or 5 DogIDs). Use your new grouped variable and the filter shelf to ensure that you only analyze the data from the few dogs you chose in your workspace. This will make troubleshooting your calculations much faster.

Next, place *Dog ID* on the rows shelf, followed by *Test Name* on the right-hand side. Then put *Created At* on **Text,** and adjust the variable aggregation so that you get a level of detail that is appropriate for the purpose of the exercise. You should see a table with *Dog ID* in the left-most column, *Test Name* in the column to the right of that, and the time the test was created in the right-most column.

Our goal is going to be to create a column between the *Test Name* and *Created At* columns that uses a calculation (NOT *Rank by DogID* – that would be cheating!) to indicate the ranked order each test was completed, sorted from 1 to the last test taken. To achieve this, make a calculation that starts with RANK, and then drag the *Created At* pill from your workspace into the parentheses Tableau automatically inserts in your calculated field. Notice that Tableau will likely insert a date function with the level of detail you specified on the *Created At* pill in your table. You will need to choose whether the rank should be ascending or descending.

As you troubleshoot your calculation, think carefully about whether the variables in your calculation need to be aggregated, and if so, how. Also think about what level of detail of the time stamps provided in *Created At* would be most useful.

Once you have a valid calculation, drag it to **Details**. Then right-click (control-click) to edit the calculation, and navigate to the "Compute Using" screen. Chose the options you think are most appropriate for your goals. Once selected, convert the pill to a dimension and place it in the appropriate place on the Rows shelf. Assess whether your calculation succeeded by comparing your rank to the rank provided in *Rank by DogID*.

Can you make a similar calculation that ranks the tests completed by each human user, mimicking the information provided in *Rank by UserID*? Can you figure out how to include both ranks in the same table?

Exercise 5

Choose one of the analyses from your analysis plan for this data set that we didn't address, and design a visualization (or set of visualizations) that addresses it!