Tableau:

GUI Data Visualizations

BTA 553

Spring 2024

David Gerbing

Week	Content
1	Read Data into Tableau
1	Tableau Continuity vs Categories
2	Visualize Categorical Data
3	Visualize Continuous Variables
4	Visualize Relationships
6	Visualize Processes and Forecast
7	Visualize Maps
8	Visualize Dashboards
9	Visualize Colors

Read Data into Tableau

AUTHOR
David Gerbing

PUBLISHED
Apr 11, 2024, 08:14 am

Table of contents

Data Connection

Tableau refers to reading a data file as "connecting to the data". A connection to the data file is required to read the data. However, the term *connection* alone does not adequately describe the process. Tableau does more than just connect to the data file. Once connected, it also reads the data into the current Tableau process for further analysis.

Open the Tableau app. The `Connect` tab in the left margin invites you to open a data file in a specified format, shown in Figure 1. Here, select an Excel formatted data file.

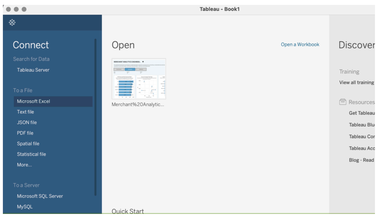


Figure 1: `Connect` to the data.

When installed, Tableau creates a `My Tableau Repository` folder in your Documents folder. You can also create folders (directories) in the repository, or anywhere else in you computer file directory. Here, I created a folder named `data` in addition to the `Datasources` folder that Tableau created, shown in Figure 2.

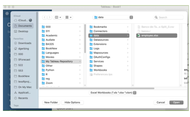


Figure 2: Locate the data file.

Click on `Open` to read the data table into Tableau. The Excel worksheet is named *d*, which is the name of the data table adopted by Tableau.

The data table is then automatically displayed in the `Data Source` window pane, which can be manually access by the tab of the same name in the lower-left corner of the Tableau window.

> **Field**
>
> Reference for a *variable* in the terminology of relational database, adopted also by Tableau .

The variables, or `fields`, are listed first. Following, the beginning rows of the data table are listed, shown in Figure 3.

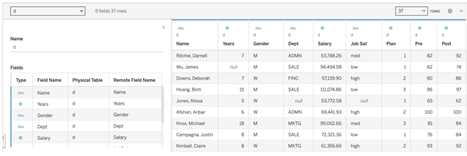


Figure 3: Data display in the `Data Source` tab.

Not all data values were present in the original Excel data file.

> **Tableau missing data code**
>
> `null` a indicates a missing data value.

The blank cells in an Excel file are replaced with `null` for the Tableau representation.

Multiple Excel Worksheets

When Tableau accesses an Excel data file with multiple worksheets, the worksheets are listed and you need to select at least one as a data source. To select, drag the worksheet name into the center of the window.

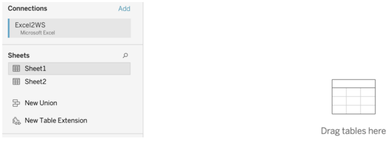


Figure 4: An Excel data source with two worksheets.

> **Relational data base option**
>
> If you select more than one worksheet, Tableau provides the option of setting up a relational database in which the tables are related to each other by a unique ID shared by both tables.

When multiple data tables are selected, Tableau lists the tables under the `Data` tab by their names, shown in Figure 5. If only one day table is present, it is not named.

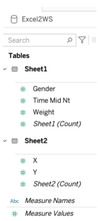


Figure 5: From multiple Excel worksheets, multiple data tables.

Next Step

The next step visualizes the data. To begin that analysis, open a new Tableau worksheet. Tableau prompts you for that step with the message in the lower-left corner of the data display window, show in Figure 6.

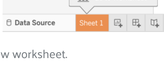


Figure 6: Prompt to open a new worksheet.

Click on `Sheet1` and move on to data visualization. Return to the `Data Source` tab at any time to view the data.

Tableau: Continuity vs Categories

AUTHOR
David Gerbing

PUBLISHED
Apr 2, 2024, 02:03 pm

This document is the Tableau interpretation of the more general, conceptual discussion regarding continuous vs categorical variables.

Data Storage Types

Tableau refers to a variable with the corresponding name from relational database, a `field`. The data table read into Tableau is called a `table`.

Continuous Variables

Tableau calls numerical variables `measures`. The term `measures` is a reasonably appropriate term given that the data values of continuous variables are measured. The more generally recognized terms for continuous variables are either *continuous* or *quantitative* but `measure` it is for Tableau.

Categorical Variables

Tableau refers to categorical variables with what must be one of the more misleading and inappropriate terms in all of data science: `dimensions`. According to everyone else, from middle school algebra students to Ph.D. data scientists who use any analysis software other than Tableau, the meaning of *dimensions* is clear. That meaning has nothing to do with defining a variable as categorical. As discussed in the reading on visual aesthetics, a dimension is an axis, which corresponds to a variable, in a space for which data values of that variable and usually one or two other variables are plotted.

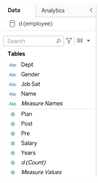


Figure 1: Initial Tableau classification of read variables into storage types: `text` and `measures`.

When reading data, Tableau classifies variables with data values that consist of alphanumeric characters as storage type `text`, equivalent to the R type `character`. Tableau indicates a `text` variable by the icon `Abc` as shown in Figure 1. Tableau indicates `measures` with the icon `#`.

The `text` variables are automatically and properly classified as categorical variables, the Tableau `dimensions`. However, Tableau initially mis-classifies integer categorical variables as `measures`. Categorical variable Plan has three unique integer values, and so is incorrectly initially classified as a `measure`. Fortunately, as shown shortly, you can manually correct this mis-classification.

Figure 1 also shows that after reading the data, Tableau automatically creates three additional variables: Measure Names, Measure Values, and d(Count).

Analyze Categorical Variables

Tableau is generally an elegant, straightforward data visualization system. However, its handling of categorical variables is, in my opinion, not ideal. For the three situations that require manual adjustment in the analysis of categorical variables, each situation requires a different response without the consistency and efficiency of R's `factor` variable.

Accept Existing Levels and Order

As shown in Figure 1, the `Data` pane on the left side of the screen lists the variables that Tableau has pre-categorized into `dimensions` and `measures`. The first step after reading the data is to correctly classify the categorical variables. Locate the integer-valued categorical variable in the `Measures` section. Then either:

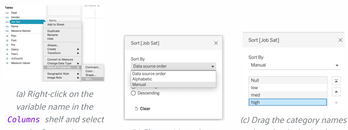
- Drag the field from the `Measures` section to the `Dimensions` section.
- Right-click on the variable name and select `Convert to Dimension` from the context menu.

Order Character String Levels

Consider a categorical variable with text data values "low", "med", and "high". With Tableau, to specify an order of the levels other than alphabetical, manually sort the levels. To sort, do the following.

1. Under the `Data` tab in the left-side of the `Worksheet` window, right-click on the categorical variable's name. Then select `Default Properties` followed by select `Sort`.
2. In the `Sort` dialog, select the `Manual` sort option.
3. Drag and arrange the values "low", "med", and "high" in the desired order.
4. Click OK to apply the sort.

Figure 2 illustrates this process of sorting the categories for the JobSat variable.



(a) Right-click on the variable name in the `Columns` shelf and select the Sort option. *(b) Choose Manual sort.* *(c) Drag the category names into the desired order.*

Figure 2: Ordering the levels of a categorical variable in Tableau.

After this sort of the JobSat levels, all subsequent visualizations will show the levels in the desired order.

Label Integer Values

In Tableau, attaching value labels to integer values, especially for categorical variables, involves creating a `calculated field` to replace these integers with the corresponding text values.

> **Calculated field**
>
> The Tableau reference to what is typically referred to in data analytics as a transformed variable, with values computed from the data values of other variables.

Create a new categorical variable or field with text values in place of the integers. In this example, attach value labels such as GoodHealth, GetWell, BestCare to integers 1, 2, and 3, respectively. In this scenario, the text descriptors are not value labels per se, they are the actual data values for the newly created variable. Unlike R's approach with the `factor` variable type, this approach wastes memory by storing all those longer text data values across all rows of data.

This transformation of data values apparently requires some SQL code, provided below, and easily adapted to other situations. After reading your data into Tableau, do the following.

1. Create a Calculated Field (Variable):
   - Navigate to the `Data` pane on the left side.
   - Right-click on the variable name Plan and select `Create` -> `Calculated Field` to open the calculated field editor.
2. Define the transformation to obtain the Calculated Field:
   - Name your calculated field, e.g., "Plan Labels", shown in Figure 3.
   - Enter the following SQL code that maps each integer to its corresponding text value, then click `OK`.

Figure 3: SQL code to define a new variable, Plan Labels, from the given variable Plan.

3. Now, there are two variables in place of a single variable: the original integer-valued categorical variable Plan and the new categorical variable Plan Labels with the corresponding descriptive text values. Use this new variable (field) in any subsequent visualizations.

Add Levels Beyond the Data

Unlike R, Tableau does not provide a method for directly indicating that a potential response in the original data table of individual data values for each person did not occur. The example here is of Gender, where, in this small Employee data table, the response of O for Other did not occur.

A workaround to display the missing level requires three steps.

1. Count the values of the existing data values for each category, such as with a bar chart.
2. Manually create the summary table of counts for the categories, such as in Excel or similar, here for existing values M and W and then add a row for O with a 0 count.
3. Do the visualization directly from the summary table read as the data.

Figure 4 shows the manually constructed summary table.

Gender	Count
M	18
W	19
O	0

Figure 4: Manually created summary table of counts.

From this summary table, construct the visualization, such as with a bar chart, as shown in the corresponding reading.

Tableau: Visualize Categorical Data

AUTHOR
David Gerbing

PUBLISHED
Apr 16, 2024, 08:15 am

This document is the Tableau implementation of the more general, conceptual discussion regarding categorical data visualizations.

Table of contents

The Worksheet

After loading your data into Tableau, open a worksheet by clicking on the highlighted orange tab at the bottom-left of the data window, shown in Figure 1.

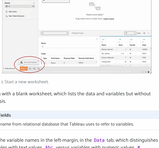


Figure 1 Start a new worksheet.

Begin with a blank worksheet, which lists the data and variables but without analysis.

> **Fields**
>
> The name from relational database that Tableau uses to refer to variables.

Find the variable names in the left-margin, in the `Data` tab, which distinguishes variables with text values, `Abc`, versus variables with numeric values, `#`.

Tableau avoids the standard notation for the *x*- and *y*-axes. It uses the term `shelves` for identifying the variables scaled on the axes, identified by `Columns`, the *x*-axis reference, and `Rows`, the *y*-axis reference Figure 2 illustrates the meaning of different aspects of the worksheet, which is embedded in a Tableau `Workbook` consisting of one or more worksheets and also dashboards and stories, concepts illustrated in later postings.

Figure 2 The new worksheet, accessing the employee data.

Regardless if a system is based on literal vision instructions or a graphic user interface (GUI), information is entered into a function to data analysis according to its parameters, sometimes called arguments. Tableau's function calls and parameter specifications are spread across the worksheet.

> **Cards**
>
> A grouped set of icons that each sets related parameter values.

One `card` often referenced is `Marks`, which defines the physical characteristics of the visualization, such as colors and labels.

Bar Chart

The following discussions show how to create the bar chart from various sources, beginning with the summary table that results from a prior data aggregation.

From Summary Table

For employment in various company departments, suppose the summary table of the counts is already available but not the raw data, the original table of data values for each individual. Maybe you located a management report that listed the number of employees in each department and wish to create the corresponding bar chart from that table. Enter the summary table directly into a worksheet app, such as Excel.

```
  Dept   n
1 ACCT   5
2 ADMN   8
3 FINC   4
4 MKTG   6
5 SALE  15
```

This summary or pivot table contains the two variables relevant to the analysis categorical variable Dept and numerical variable n There is only one row for each unique value (category) of Dept.

To create the bar chart from the summary table, specify these two variables categorical variable *x*, and numerical variable *y*, which maps to each bar's height. For a bar chart with vertical bars, identify the categorical variable Dept under tables and drag to the `Columns` shelf which represents the *x*-axis. Then identify the variable N and drag that to the `Rows` shelf. The bar chart results. Update The name of the worksheet by right clicking on the tab at the bottom left of the window and enter a new name.

The link to the video of this process follows.

Video Bar Chart from summary table [1:11]

Find the rendered bar chart in Figure 5.

Aggregate Counts

Creating the bar chart from the original data is generally the same as for the previous example of creating the bar chart from the summary table. The distinction is that we have access to all of the variables in the original data table. As before, we drag the department variable from the tables area to the `Columns` shelf if we want a vertical bar chart with the columns located on the *x*-axis. When the mouse is on the original data, Tableau creates a `count` variable, which we drag to the `Rows` shelf, which represents the *y*-axis. The vertical bar chart results.

The link to the video of this process follows.

Video Bar Chart from counts from the raw data [3:27]

Aggregate *y*

Again, generally the same process as the previous to examples, but to do data aggregation on a numerical variable we will drag the numerical variable of the shelf instead of the count variable as we did before. The key element of this process is automatic aggregation.

> **Automatic aggregation**
>
> Tableau automatically aggregates data when you drag a numerical field to the shelf or to a mark, often defaulting to `sum`.

We want the mean salary not the sum of the salaries in each department. To change from the sum to the mean, right-click on the variable name Salary in the `Rows` shelf, choose `measure`, and then `average`. The result is the bar chart for the average salary in each department.

The link to the video of this process follows.

Video Bar Chart from aggregating a numerical variable [1:24]

Export the Visualization

To export to an image or png file, in the open Worksheet, go to the top of the screen to `Worksheet` menu. Then elect `Export` and then `Image...`, shown in Figure 3.

Figure 3 Access the menu for exporting an image.

Tableau does not just produce a literal copy of the visualization. Instead, it offers several options that can be included in the exported image. In the Export Image dialog, select the image options for a png such as size and legend, then click `Save`, illustrated in Figure 4.

Figure 4 Dialogue box for exporting an image.

The result follows in Figure 5.

Figure 5 Exported image as a png file of the Employee Counts bar chart.

Pie Chart

Creating a pie chart in Tableau involves many steps. Here, we review each successive step. The categorical variable of interest is Dept. We will have the pie slices reflect the count of the employees in each department. The procedure for creating the pie chart is distinct from the procedure previously described for the bar chart, as follows.

> **No axes for the pie chart**
>
> A pie chart visualization has no axes, so instead of using the `Columns` and `Rows` shelves, create the pie chart only from setting various `Marks` parameters.

To create the pie chart, set the `Marks` parameters by dragging the labels for the field names to specified `Marks` icons.

Also, here we introduce a general Tableau property, which is a property of any data analysis system. What is unique is the way Tableau implements the assignment of parameters to the underlying visualization function.

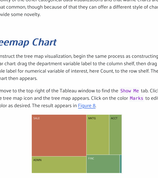
> **A variable may be assigned multiple parameter settings.**
>
> More than one parameter setting may be assigned to a given variable via the `Marks` card.

Every system allows specifying multiple parameters to a function call. To do this assignment in Tableau, duplicate the field name label for a variable by selecting a label for one mark and then holding the Command or Control key down while dragging the label to another `Mark`.

Following is the specific procedure to create the pie chart of counts for the different departments.

1. Under the `Marks` heading is a drop-down menu labeled `Automatic`. From that menu select `Pie`.
2. Drag the Dept field name toward the `Marks` card, placing the name over the `Color` mark to indicate that want colors to vary for the pie slices that represent different departments.
3. As an option, add a border between each slice. Click on the `Color` mark and select `Border`... Choose a border color such as white.
4. Select the desired `measure` for which to specify the pie slice areas, here `Count`. Drag the field name toward the `Marks` and drop on the `Detail` mark. This allows the metric of the numerical variable to be specified as a label to each of the pie slices.
5. Right-click or select the drop-down menu on the label `CNT(id)`, choose `Quick Table Calculation`, and then `Percent of Total`.
6. Drag the `CNT(id)` label to the `Angle` mark to have the angles, and so the size of the pie slices, reflect their respective percentage contribution to the overall pie.
7. Label the slices with their respective percentages by dragging the `CNT(id)` label to the `Label` mark to define another `mark` for the counts.
8. Also label the slices with their respective level names by CTRD/CTRL dragging the `Dept` label for the `Label` mark, which creates another field label for the variable beyond the one for the color parameter setting.
9. The order in which the marks appear on the visualization depends upon the order in which they are listed in the `Marks` area. Drag the field name labels into the order you wish for the display, such as listing the name of the level over the corresponding percentage.
10. Expand the view of the pie chart from the drop-down menu at the top of the Tableau window that shows `Standard`. Select from that menu the choice of `Entire View`.

Figure 6 shows the settings of the `Marks` parameters and the resulting Tableau pie chart.



(a) Settings of the marks parameters (b) The pie chart

Figure 6 The marks settings and finished pie chart

The link to the video of this process follows.

Video Pie Chart from of counts from the raw data [3:11]

Because Tableau treats Count as another continuous variable, any other measure could have been selected too determine the size of the pie slices.

Dot Plot

Tableau does not provide a dot (lollipop) chart. This is is not too concerning because the bar chart can be used in its place. Oh

Bubble Plot

To construct the bubble plot we will again focus on two variables Categorical variable Dept and Numerical variable Count. We want Dept to determine the number of bubbles, five, and to provide a label for each level of Dept, that is, each department. We want Count to specify the respective sizes of the bubbles. Tableau plots the bubbles in two-dimensions instead of along a single line.

To set the parameter values, drag the Dept field label over to the `Color` mark. For Count, drag to the `Size` mark. Now the size of the corresponding bubbles, that is, circles, correspond to the number of people who are employed in each department. To label the bubbles, drag the department field label over to the `Label` mark. Find the result in Figure 7.

Figure 7 shows the settings of the `Marks` parameters and the resulting Tableau bubble chart.

(a) Settings of the marks parameters. (b) The bubble chart

Figure 7 The marks settings and finished pie chart

The link to the video of this process follows.

Video Bubble plot [2:05]

Waffle Chart

Tableau does not provide a waffle chart. This is not too concerning given the availability of the other categorical data visualizations and that waffle charts are not that common, though because of that they can offer a different style of chart to provide some novelty.

Treemap Chart

To construct the tree map visualization, begin the same process as constructing the bar chart drag the department variable label to the column shelf, then drag the variable label for numerical variable of interest, here, Count, to the row shelf. The bar chart then appears.

Now move to the top right of the Tableau window to find the `Show Me` tab. Click on the tree map icon and the tree map appears. Click on the color `Marks` to edit the color as desired. The result appears in Figure 8.

Figure 8 The tree map.

The link to the video of this process follows.

Video Tree map [1:25]

Tableau: Visualize Continuous Variables

AUTHOR
David Gerbing

PUBLISHED
Apr 18, 2024, 10:12 am

This document is the Tableau implementation of the more general, conceptual discussion regarding data visualizations of continuous distributions.

Table of contents

Histogram

Tableau has no histogram function per se. There is no pre-defined histogram plot, just a bar chart. Instead, follow these steps.

1. Identify the continuous variable of interest in the list of variables on the left-side of the worksheet.
2. Create a new categorical variable version of the continuous variable, a binned version of the original variable.
   [Right-click on the variable name and select `Create` and then `Bins...`]
3. Create a bar chart of the new categorical variable.
   [Drag the `(Count)` variable to the empty shelf.]
4. Make the bars wider on the bar chart until they become adjacent to each other with no gaps or very small gaps that act like a border between adjacent bars.
   [Select all the bars, go to the `Size` mark, and drag the slider.]

The link to the video of this process follows.

Video: Histogram [4:31]

Box Plot

The Tableau default box plot is the box plot with the plotted points. There is no jitter option. Create with the following steps.

1. Drag the variable name for the continuous variable of interest to the `Rows` or `Columns` shelf depending on the desired orientation of the box plot.

Be aware of the following potential "gotcha".

> **Tableau automatic aggregation**
> Tableau often aggregates variables by default, identified such as by SUM(name). Unaggregate from the `Analysis` menu, then deselect `Aggregate Measures`.

Figure 1 show the result of the automatic aggregation after dragging the Salary variable to the `Columns` shelf. When you unaggregate from the `Analysis` menu, the `SUM` label disappears, leaving only the variable name.

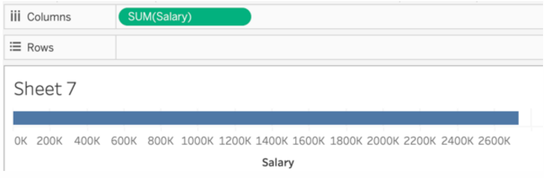


Figure 1: Automatic aggregation, which defaults to sums, as shwon by the cumulative sums of the salaries along the axis.

Tableau also remembers how you treat a variable, so once unaggregated it may be remain so with further analysis. So, depending on your recent history of working with Tableau, you may or may not need to unaggregate. For some reason, there is a way to change the aggregation by right-clicking on the variable name when in a shelf but not a way to turn off the aggregation without going to the main menu.

2. Click on the small Box Plot icon in the `Show Me` tab that displays the available plot types, shown in Figure 2.

Figure 2: Tableau box plot icon.

3. To create a Trellis plot, drag the categorical variable of interest to the opposite shelf that contains continuous variable upon which the box plot is based.

4. If desired, start the box plot at the minimum data value instead of 0.
   [Select the axis by clicking on it, right-click on the axis and select `Edit Axis`, then uncheck the box labeled `include zero`]

5. If desired, if the points overlap, add some transparency to the plotted points by selecting the `Color` mark, then slide the `Opacity` slider to achieve the desired transparency.

The link to the video of this process follows.

Video: Box Plot [3:44]

Tableau: Visualize Relationships

AUTHOR
David Gerbing

PUBLISHED
Apr 24, 2024, 08:42 am

This document is the Tableau implementation of the more general, conceptual discussion regarding data visualizations of relationships.

Table of contents

Categorical Variables

Define

Data analysis, regardless of the data analysis system, always begins with the identification and definition of categorical variables.

> **Define categorical variables**
> Before analysis begins, first define all categorical variables, what Tableau calls "dimensions".

Only categorical variables with `text` data values will be automatically classified as categorical. Properly define the categorical variables as needed: Drag all categorical variables for analysis to the list of dimensions, order the levels, and attach meaningful labels to the levels. The visualization attaches a numerical value to each group, the combination of levels of the categorical variables.

Stacked Bar Chart

For more specific guidance, assume a vertical bar chart. Switch the column and row orientation for a horizontal bark chart.

1. *Select x-axis variable*: Drag a categorical variable (dimension) to the `Columns` shelf.
2. *Select y-axis variable*: Drag a continuous variable (measure) to the `Rows` shelf, which results in a one categorical variable bar chart.
   - If the y-axis variable is the pre-defined Count variable, then the aggregation is CNT.
   - If the y-axis is another variable, then Tableau defaults to aggregating the SUM of that variable for each level of the categorical variable on the x-axis. You may want to change that aggregation to the AVG.
3. *Add the second categorical variable*: Drag another categorical variable to the `Color` mark.
4. *Add labels to the bars* [optional]: Select the numerical variable and drag to the `Label` mark.
   - Tableau will again default aggregate to Sum for the labels even if the aggregation on the bar chart y-axis is AVG.
   - Usually change that aggregation for the labels to be the same aggregation on the y-axis for consistency.

Figure 1 shows the resulting stacked bar chart with the specified `Marks` parameters.

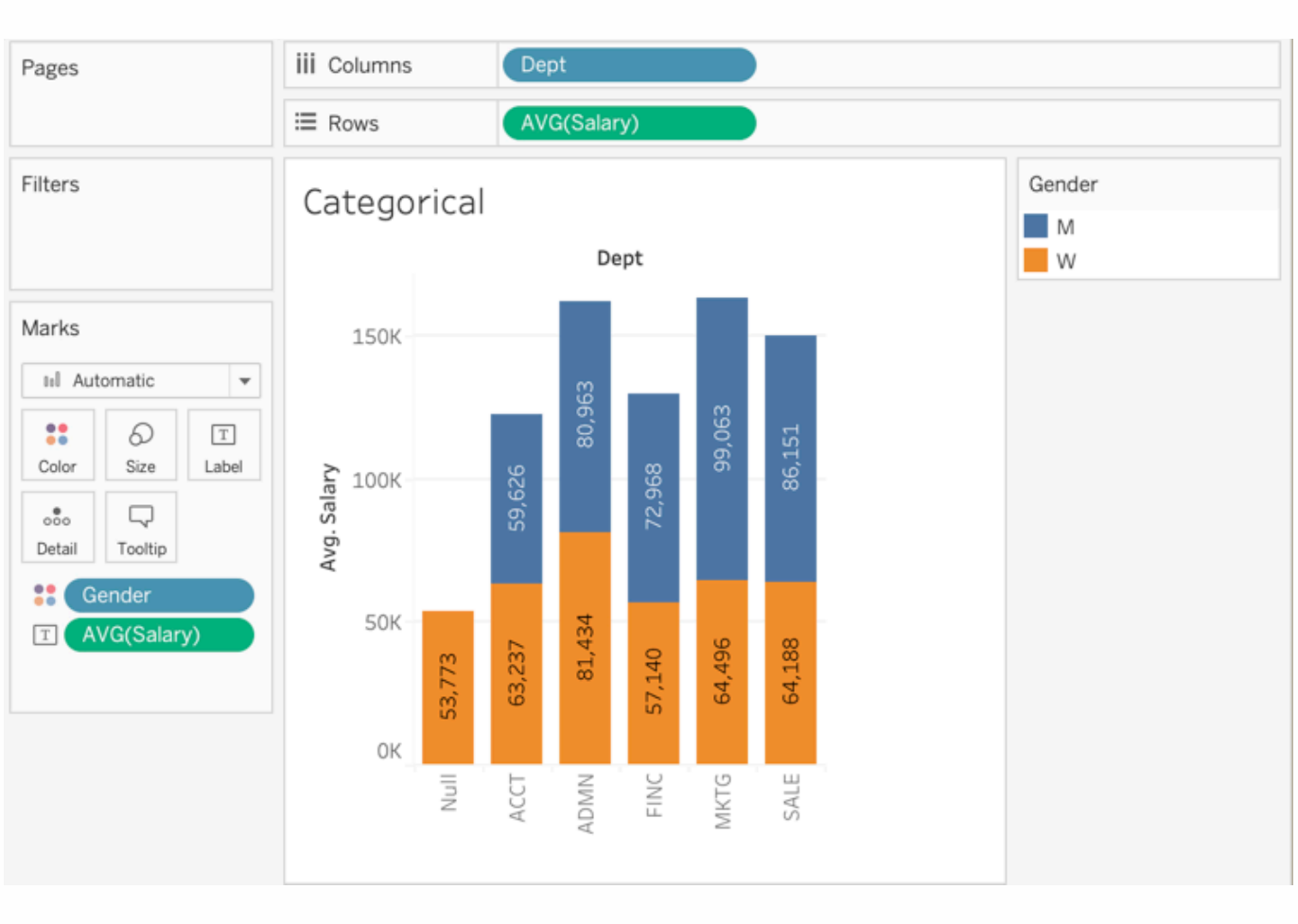


Figure 1: Stacked bar chart of average mean Salary by Dept sub-divided by Gender.

Unstacked Bar Chart

First, create the stacked bar chart.

To unstack the bars, drag the second categorical variable level over to the `Columns` shelf (or whatever shelf contains the first categorical variable).

Figure 2 shows the resulting stacked bar chart with the specified `Marks` parameters.

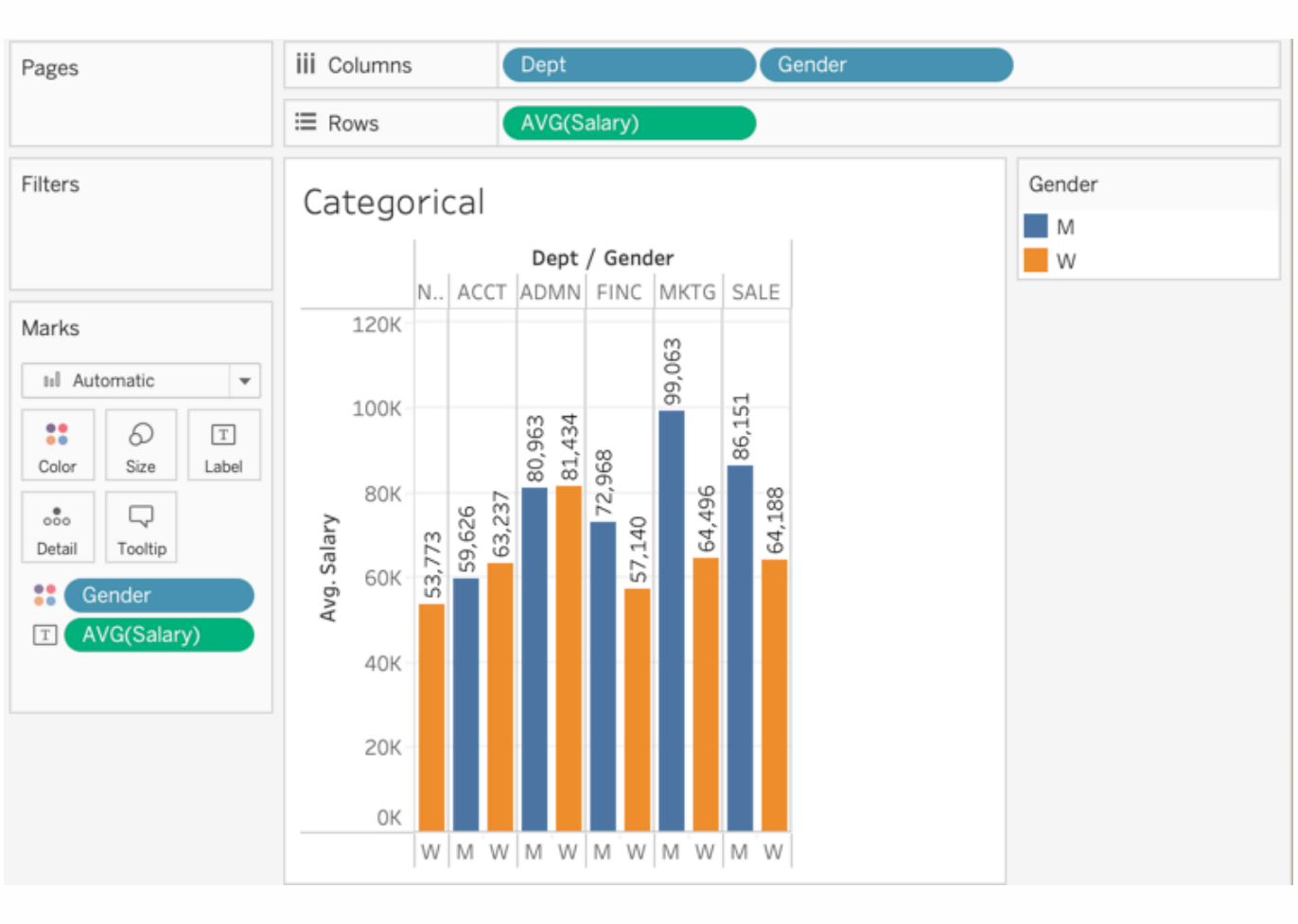


Figure 2: Unstacked bar chart of average mean Salary by Dept sub-divided by Gender.

100% Stacked Bar Chart

First create the regular stacked bar chart, except leave the aggregation of the numerical variable as SUM.

To convert to a 100% stacked bar chart:

1. Right-click the on the numerical variable name in the respective shelf
2. Select `Quick Table Calculation`
3. In the drop-down menu, select `Percent of Total`
4. Again, right-click on the numerical variable name
5. Select `Compute Using`
6. In the drop-down menu, select `Cell`

Figure 3 shows the resulting stacked bar chart with the specified `Marks` parameters.

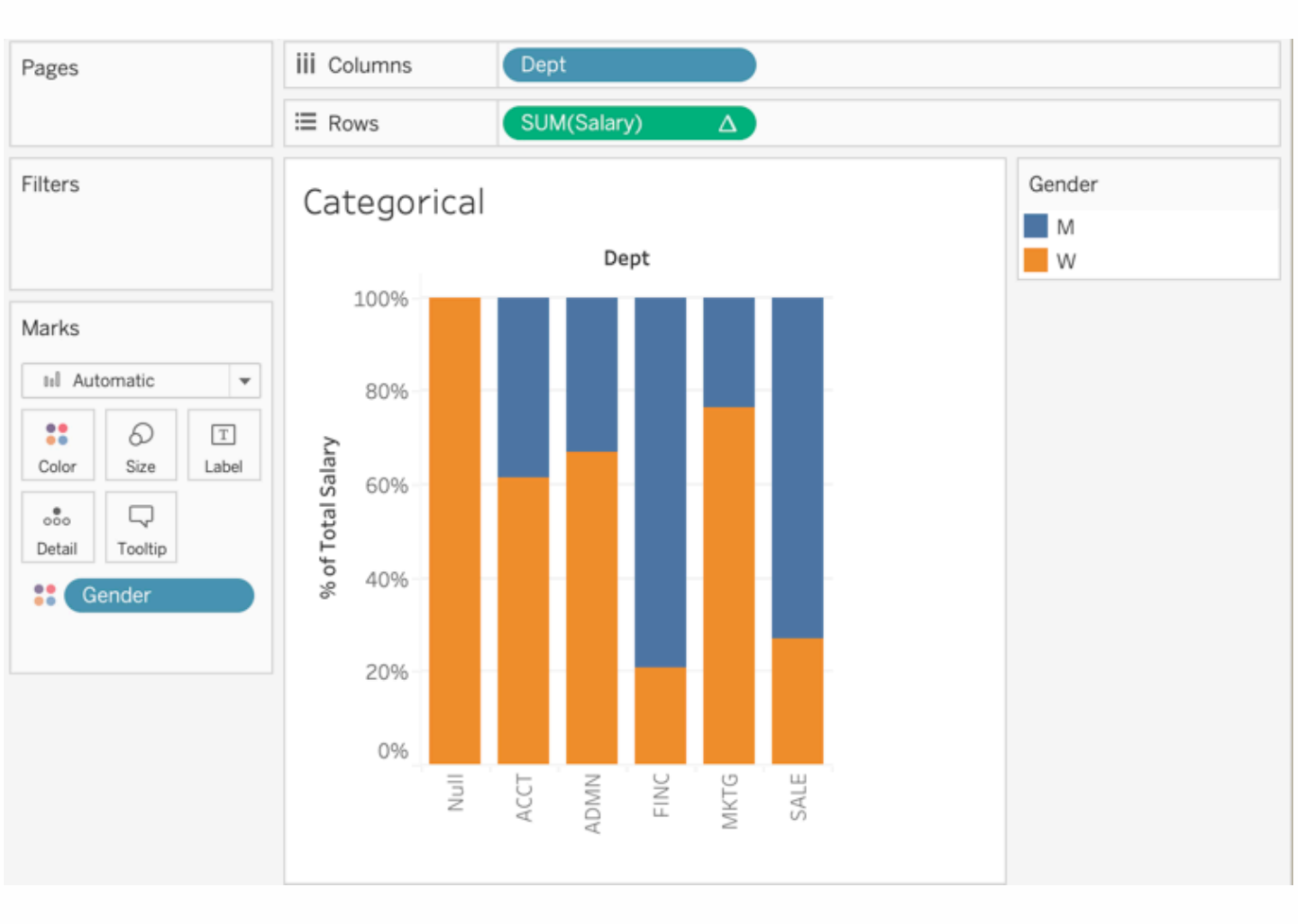


Figure 3: 100% stacked bar chart of Salary by Dept sub-divided by Gender.

Treemap

One way to proceed is to first create the regular stacked bar chart.

Then, select the Treemap icon on the `Show Me` panel.

The boxes are shaded according to the value of the numerical variable for each box.

Figure 4 shows the resulting scatterplot with the specified `Marks` parameters. In this example, aggregate on the count of the number of occurrences in each group.

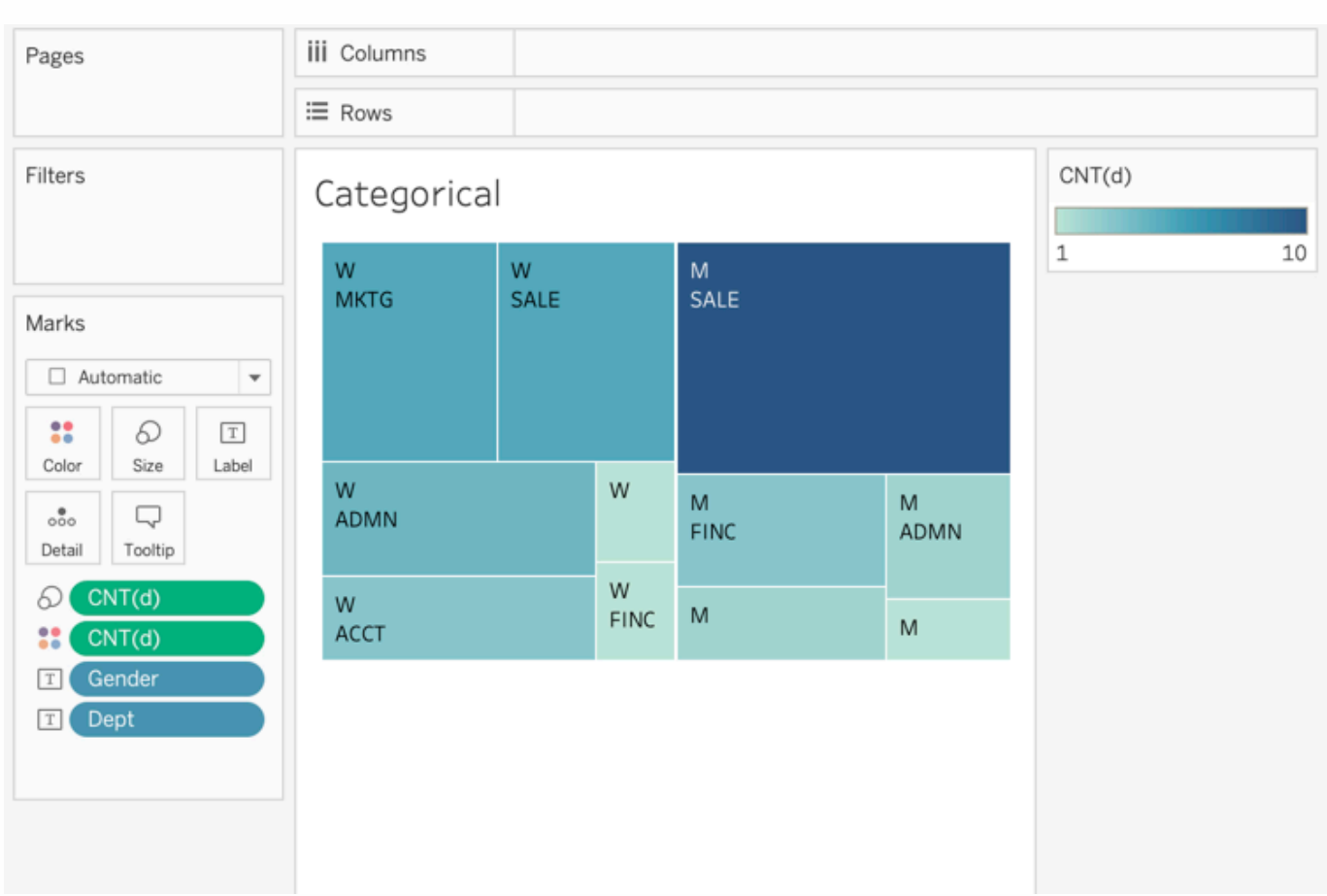


Figure 4: Treemap of counts for two categorical variables, Dept and Gender, with the size of the rectangles determined by mean Salary.

Bubble Plot

One way to proceed is to first create the regular stacked bar chart.

Then select the Bubble plot icon on the `Show Me` panel. Usually, change the aggregation of the numerical variable to AVG.

Figure 5 shows the resulting scatterplot with the specified `Marks` parameters.

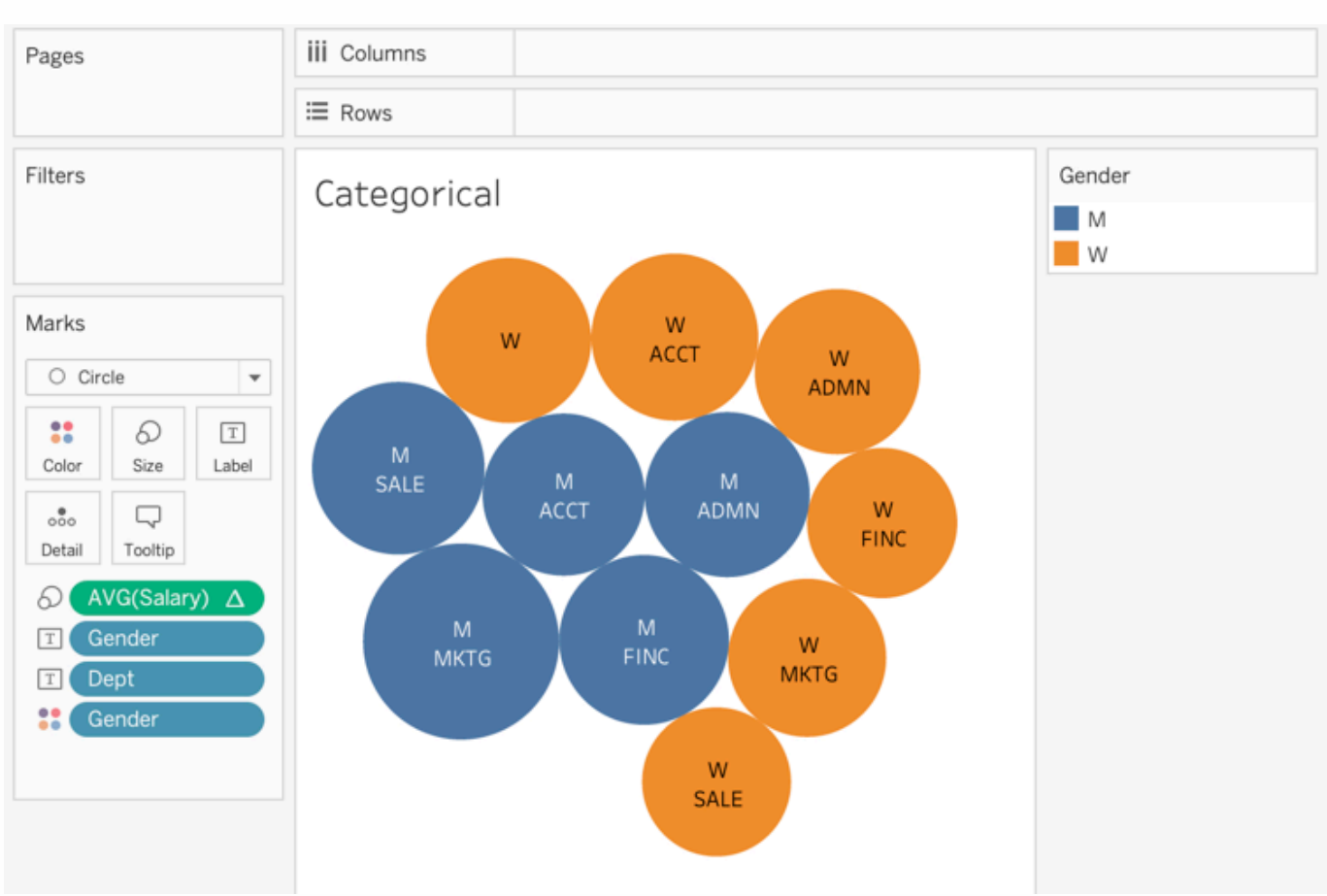


Figure 5: Bubble scatterplot of two categorical variables, Dept and Gender, with the size of bubbles determined by mean Salary.

The link to the video of examples of these processes follows.

Video: Two categorical variables. [4:25]

Continuous Variables

Two-Variable Scatterplot

1. *Select x-axis variable*: Drag one variable, measure, to the `Columns` shelf.
2. *Select y-axis variable*: Drag the other variable, measure, to the `Rows` shelf.
3. *Disaggregate*: On the Main Menu at the top of the screen, select `Analysis`, then select `Aggregate Measures`, which will uncheck the menu option, turning off aggregation.
4. *Fit line*: Select the `Analytics` tab, next to the `Data` tab at the top of the list of variables. Under `Model`, select the mis-named `Trend line` option (misnamed because "trend" as generally understood applies to an orientation over time). Then, if a linear trend line is desired, choose the displayed `Linear` option.

Figure 6 shows the resulting scatterplot with the specified `Marks` parameters.

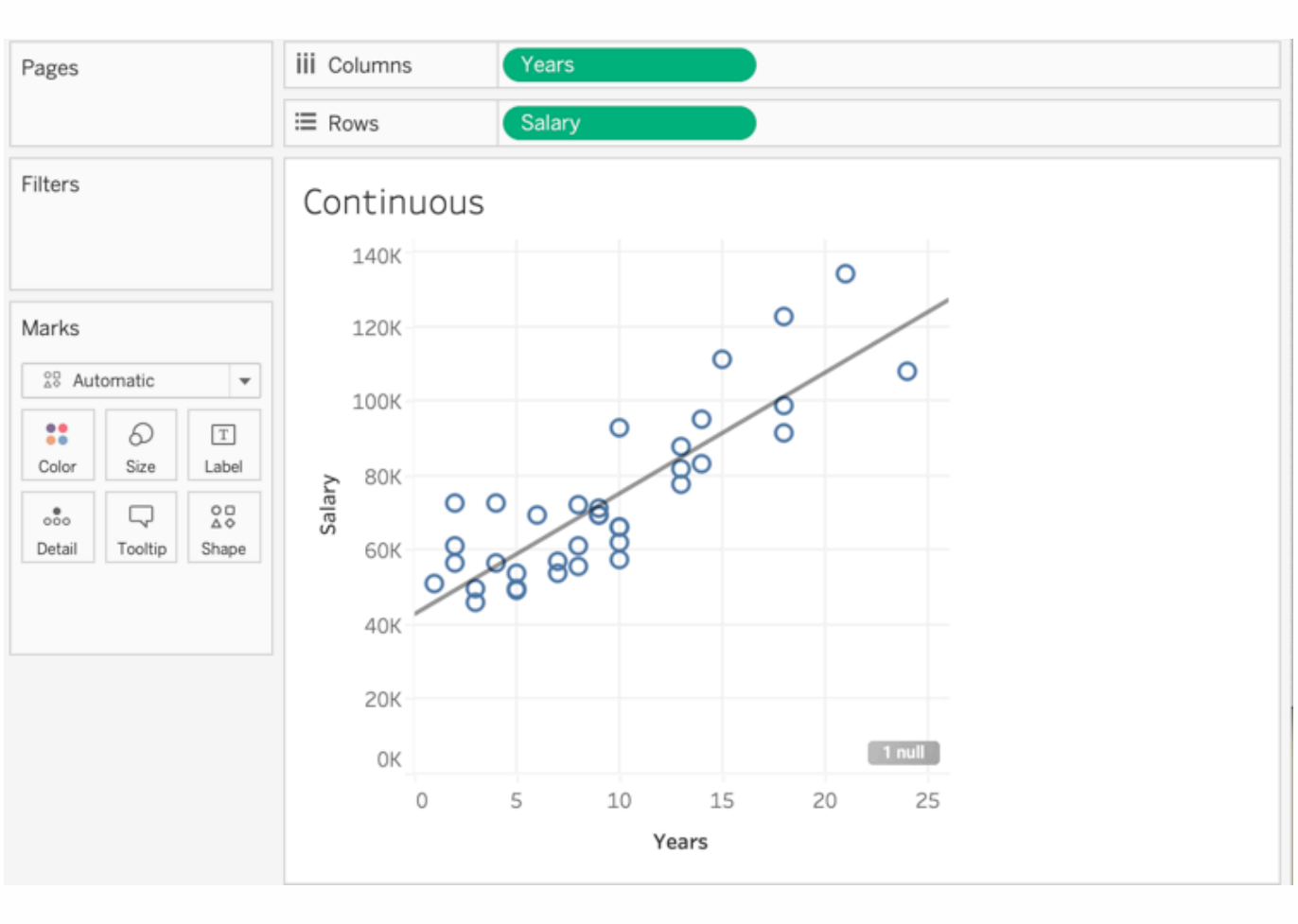


Figure 6: Scatterplot of Years and Salary with the best-fit regression line.

Stratification

Trellis plot: Drag the categorical variable (dimension) label over to one of the shelves.

Figure 7 shows the resulting Trellis (facet) scatterplot with the specified `Marks` parameters.

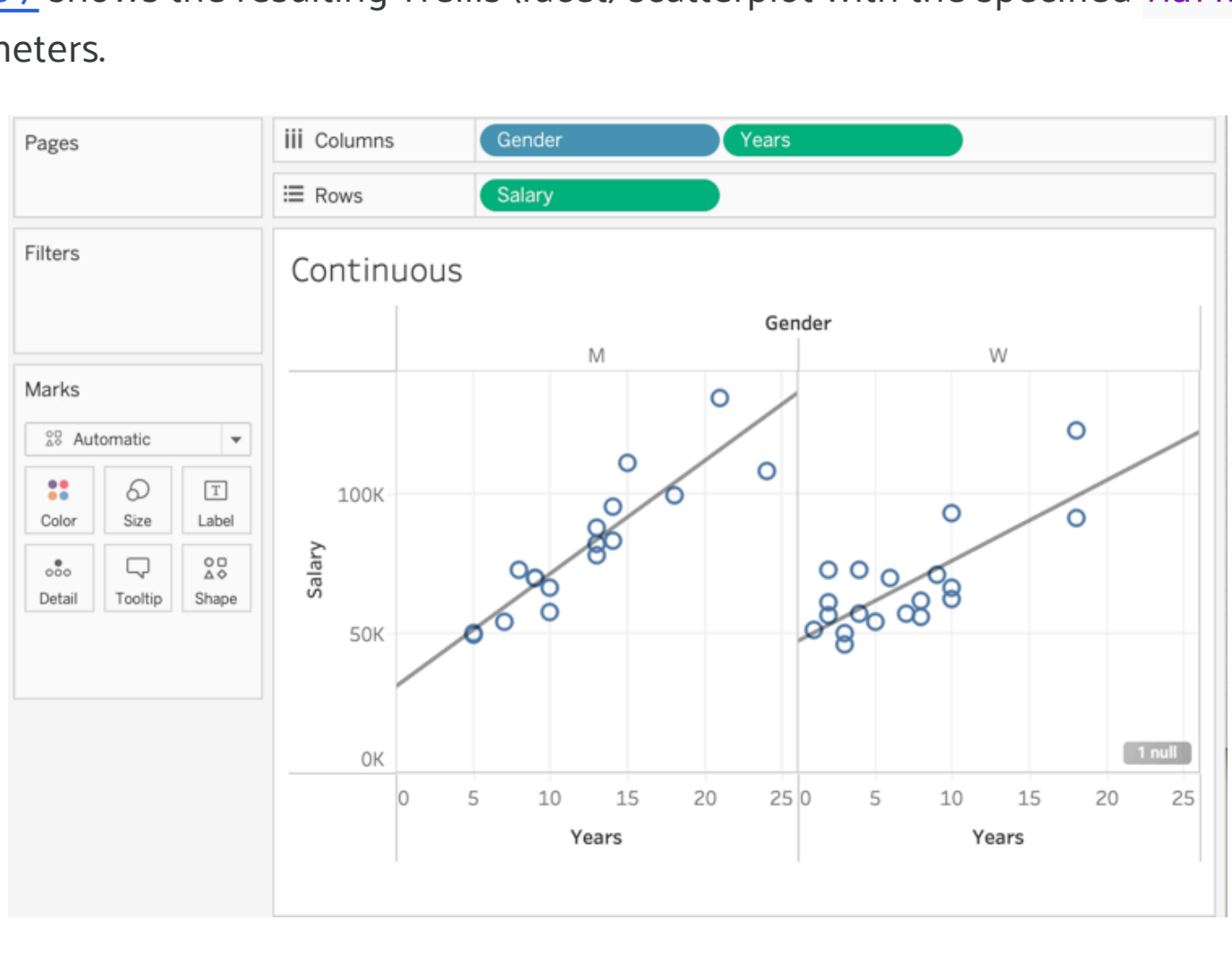


Figure 7: Scatterplot of Years and Salary with the best-fit regression line.

Same panel plot: Drag the categorical variable (dimension) label over to `Color` mark.

Figure 8 shows the resulting scatterplot with the specified `Marks` parameters.

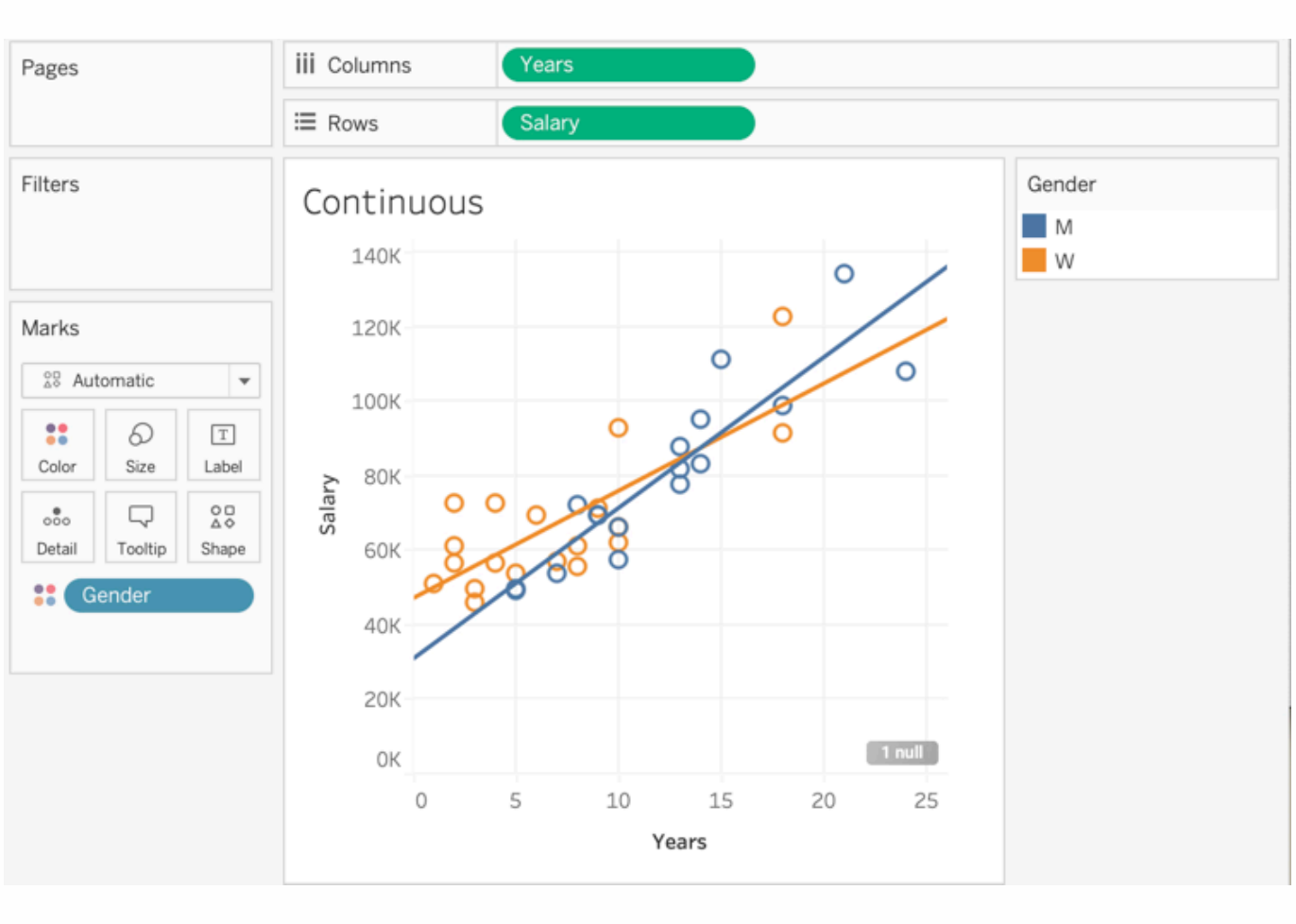


Figure 8: Scatterplot of Years and Salary with the best-fit regression line.

Bubble Scatterplot

Drag the continuous variable (measure) over to the `Size` mark. Perhaps adjust the sizes, though apparently not a way to better differentiate among the different sizes other than make them all larger or smaller at the same scale.

Figure 9 shows the resulting scatterplot with the specified `Marks` parameters.

Figure 9: Bubble scatterplot for cars of MidPrice with HP (horsepower) with bubble size city MPG.

The link to the video of examples of these processes follows.

Video: Continuous variables. [3:31]

Tableau: Visualize Processes and Forecast

AUTHOR
David Gerbing

PUBLISHED
Jun 7, 2024, 10:57 am

Table of contents

This document is the Tableau implementation of the more general, conceptual discussion regarding data visualizations of data values collected over time.

The link to the video that illustrates the run chart follows.

Visualize a Run Chart

The run chart is the value of the variable over time plotted against the Index variable, the sequence of integers from 1, 2, 3, ..., to the number of data values to plot. If not present in the data, the Index must be computed.

Data: https://web.pdx.edu/~gerbing/data/Sales07.xlsx

1. Begin with the data values to plot on the y-axis, Sales, sorted in the proper order from the first collected data value to the last.
2. Create the Index variable for the x-axis. Right-click in the open area under the list of variables in the left-side of the window.
   - Select: Create -> Calculated Field...
   - Name the new variable Index
   - Access the INDEX function by typing its name, INDEX()

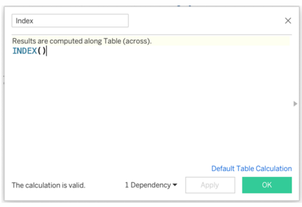


Figure 1: Create the Index variable.

3. The created Index variable appears in the variable list in the left-side. Drag to the `Columns` shelf.
4. Drag the variable to plot, Sales, to the `Rows` shelf.
5. Under `Marks`, select the `Line` chart.

The link to the video that illustrates the run chart follows.

Video: Plot a Run Chart [1:47]

Visualize a Time Series

The time series chart is the value of the variable over time plotted against the date/time when each data value was created. Both the date and the value to plot are included in the data.

Data: https://web.pdx.edu/~gerbing/data/StockPriceWide.xlsx

1. Begin with the wide form of a data table for the stock price of three companies: Apple, IBM, Intel. In the white form, each company is it's own variable, three columns of stock prices. Month is another variable.
2. Apple Time Series.
   - Drag the Apple variable to the `Rows` shelf.
   - Drag the date variable, Month, to the `Columns` shelf. It will automatically aggregate by Year.
   - To use all the available space in the Tableau window for your plot, go to the `Standard` drop-down menu at the top of the window and select `Fit Width`.
3. Area chart: For the drop-down mneu under `Marks`, select `Area`.
4. Trellis multiple plots: To visualize all three times series stock prices on different panels, drag the IBM variable and the Intel variable over to the `Rows` shelf next to the SUM(Apple) variable.
5. Same panel multiple plots:
   - Drag the date variable, Month, over to the `Columns` shelf.
   - Tableau has created a new variable, Measure Values, found in the measures section of the variable (field) list. Drag that variable over to the `Rows` shelf.
   - Remove the CNT(StockPriceWide) variable from under the `Measure Values` card below the `Marks` card.
   - To use all the available space, go to the `Standard` drop-down menu at the top of the window and select `Fit Width`.
   - To color each time series line differently, drag the Measure Names categorical variable over to the `Color` mark in the `Marks` card.
   - To create the stacked area chart version, for the drop-down menu on the `Marks` card, select `Area`.

The link to the video that illustrates the time series chart follows.

Video: Plot a Time Series Chart [5:15]

Visualize a Forecast

Tableau estimates an exponential smoothing model of the time series data and then is able to forecast future values from this model. The model can be estimated by default or additive or multiplicative components specified for any combination of the error, trend, and seasonality.

The data are the same as for the run chart but with an added column, the Month, indicated as the first day of each month.

Data: https://web.pdx.edu/~gerbing/data/Sales07date.xlsx

1. Time series plot of the data.
   - Drag the date variable, Month, to the `Columns` shelf.
   - Drag the variable to plot, Sales, to the `Rows` shelf.
   - The result is that the date variable, Month, is automatically aggregated by Year, which hides the quarterly seasonality. To plot by Quarter, click on the icon of the `+` sign inside of a square at the beginning of YEAR(Month) on the `Columns` shelf, which then disaggregates to QUARTER.
2. Exponential smoothing forecast from the data.
   - Click on the `Analytics` tab next to the `Data` tab in the top-left corner.
   - Under `Model`, select `Forecast` and drag to the plot area, dropping the icon on the `Forecast` icon that appears. The forecast for the next quarter appears in a lighter blue color the for the data, complete with the 95% prediction interval.
   - To customize the underlying exponential smoothing model on which the forecast is based, right-click near the forecasted line segments, select `Forecast` then `Forecast Options...`.
   - To View the fit indices of the underlying exponential smoothing model on which the forecast is based, right-click near the forecasted line segments, select `Forecast` then `Describe Forecast...`.

The link to the video that illustrates the exponential smoothing model and subsequent forecast follows.

Video: Forecast [5:24]

Tableau: Visualize Maps

AUTHOR
David Gerbing

PUBLISHED
May 14, 2024, 10:02 pm

This document is the Tableau implementation of the more general, conceptual discussion regarding maps.

Table of contents

Symbol Map: Largest Italian Cities

We use the simplemaps.com free data base of almost 48,000 cities.

1. Download and prepare the data file, *worldcities.xlsx*. Delete the *city* column with its non-ASCII characters.
2. Read the data into Tableau.
3. Tableau does not recognize the abbreviation Lng for Longitude, so select and choose `Geographic Role`, then `Longitude`.
4. Use the Longitude and Latitude provided by the data file, not that generated by Tableau as it cannot probably classify thousands of the provided cities. Drag *Lat* to the `Rows` shelf and *Lng* to the `Columns` shelf.
5. Drag the *country* variable to the `Filters` card and select `Italy`.
6. Go to the main menu select `Analysis` and then click on `Aggregate Measures`, which will disaggregate.
7. Drag the *population* variable over to the `Filters` card and indicate a minimum value of 250,000.
8. Create a bubble chart by dragging the *population* variable over to the `Size` mark
9. Options:
   - change the color of the plotted bubbles: Click on the `Color` mark
   - enhance the tooltips when hovering the mouse over a city: Drag the `city` over the `Tooltip` mark
   - change the background map: Map menu, choose Background Maps, then Satellite
10. *Add labels to the bars* [optional]: Select the numerical variable and drag to the `Label` mark.
    - Tableau will again default aggregate to Sum for the labels even if the aggregation on the bar chart y-axis is AVG.
    - Usually change that aggregation for the labels to be the same aggregation on the y-axis for consistency.

The link to the video of examples of these processes follows.

Video: Larger Italian Cities. [6:44]

Choropleth Map: Gini Index USA States

If you wish to follow along with the video, find the data file at:

web.pdx.edu/~gerbing/0Viz/Maps/data/Gini2021byState.xlsx

1. Read the Excel data file containing the State names and the Gini coefficients into Tableau.
2. Assign the State variable, *ID*, as a geographic object. Right-click the *ID* variable in the data pane, then choose `Geographic Role` -> `State/Province`.
3. Create the map.
   - drag the *ID* field to the `Rows` shelf
   - drag the *Gini* variable to the `Color` mark
   - go to the `Show Me` palette and click on the filled map icon: Second row. middle column
4. Hold the Shift key and drag the map to focus on the main body of the USA
5. Option: Choose a different color palette from the `Color` mark

---

The link to the video of examples of these processes follows.

Video: Choropleth map of the United States. [3:39]

City Map: Portland with Custom Geocodes

If you wish to follow along with the video, find the data file at:

web.pdx.edu/~gerbing/0Viz/Maps/data/PDXgeocodes.xlsx

1. Prepare the Excel data file with the location name, address, and any additional information.
2. Read the data file into R.
3. Check to make sure that the longitude and latitude variables are recognized as geographical information, geocodes.
4. Drag the latitude variable to the `Rows` shelf and the longitudinal variable to the `Columns` shelf.
5. Go to the main menu select `Analysis` and then click on `Aggregate Measures`, which will disaggregate the plotted values.
6. Go to the main menu, select `Maps`, then `Background Maps`, then `Streets`.
7. To create a bubble plot, drag a numerical variable such as *Profit* over to the `Size` mark

---

The link to the video of examples of these processes follows.

Video: Map of Portland Locations. [3:43]

Tableau: Visualize Dashboards

AUTHOR
David Gerbing

PUBLISHED
Jun 7, 2024, 11:18 am

The Concept

A dashboard simultaneously displays multiple, previously constructed items, usually mostly data visualizations, on the same layout. Presumably, the reason for displaying the multiple visualizations simultaneously is that they all share a common theme. For example, consider sales analysis. We can observe sales by geographical region, by product, and over time. All three corresponding data visualizations can be placed on a dashboard along with some overall sales values and profitability.

A Sales dashboard is presented here, using the Tableau superstore data, which lists sales by customers over a range of products from 2021 through (at least on my version) 2024. The previously constructed Tableau items that will compose the dashboard each have their own worksheet. So, first we construct a new item we have not previously encountered, a header worksheet that displays numbers: overall profitability and sales and the number of customers. Then, we repeat types of visualizations we have already accomplished, a choropleth map of sales by state, a bar chart of sales by product, and the time series chart of sales over months.

When these four worksheets are complete, we are ready for the dashboard.

Data: Tableau Superstore

The Worksheets

The link to the video of constructing these four worksheets follows.

Video: Four Worksheets, [7:47]

Sales Statistics

Here, we pursue a new form of a worksheet, a data summary without a visualization.

Construct

1. Drag Sales to the Sheet
2. Drag Profit on top of Sales and release when you see `Show Me`
3. Orient the table horizontally:
   Drag `Measure Names` to the columns shelf
4. List Sales before Profit:
   Drag SUM(Sales) on the `Measure` values card ahead Profit
5. Compute and display the number of unique customers:
   Customer Name -> Create -> Calculated Field
   Name: # Customers
   Formula: `COUNTD([Customer Name])`
   Drag # Customers to the Measure Values card

Format

Any formatting is optional, and, of course, many more possibilities exist than what is shown.

1. Center headings:
   Click on `Text` mark
   `Alignment`: Choose Center
2. Increase headings font size:
   Right-click on a heading
   Choose Font the left margin, 14 pt
3. Increase text font size:
   Right-click on the `Text` mark
   Select
   Choose 18 point font size
4. Remove horizontal lines:
   Select the `Border` option
   Under `Row Divider` choose `None`

Sales by State Map

Create a choropleth sales map. The data cover the states of the United States and the provinces of Canada.

Construct

1. Show the hierarchy locations to identify our geographic regions by which to show Sales:
   Click on the arrow at points to the location under Location
2. Double click on State/Province
3. Drag Sales to the `Color` mark

Format

1. Remove other countries from the map:
   Right-click on the map
   Choose `Background Layers`
   Drag the `Washout` slider to 100 percent
2. Get rid of the header borders:
   Right-click on the map
   Choose `Format`
   In the left margin, choose `Borders`
   Choose `Row Divider`
   Set `Pane` to `none`

Sales by Month Time Series

Construct

1. Set the time unit:
   Drag Order Date to the Column shelf
   Right-click Month
   Month is aggregated by Year, can leave as is or choose Month/Year to see each individual month
2. Drag Sales to the Rows shelf

Format

1. Abbreviate month labels:
   Right click on any month
   Select `Format Months` in the left margin
   Select`Dates`, then `Abbreviation`
2. Remoce Order Date header:
   Right-click on Order Date
   Select Hide Field Labels for Columns

Sales by Product

Construct

1. Click on the arrow for the product hierarchy to reveal the sub-category
2. Drag subcategory to the `Rows` shelf
3. Drag Sales to the `Columns` shelf

Format

1. Sort descending from the toolbar
2. Remove header:
   Right-click on the sub-category label for the columns
   Select `Hide Field Labels for Rows`
3. Remove the Sales horizontal axis label:
   Right-click
   Select `Format` and remove the label Sales

The Dashboard

The link to the video of constructing the dashboard follows.

Video: Four Worksheets, [6:08]

Construct

1. Begin by clicking on the New Dashboard icon at the bottom of the Tableau window.
2. If needed, adjust the size of the dashboard in the left margin under `Size`:
   Set to a size in which the entire dashboard is visible, here, `Width`: 1050
   `Height`: 625

To place the individual worksheets on the dashboard, use the Tableau concept of horizontal and vertical containers. The container provides control for aligning the individual worksheets horizontally and vertically on the dashboard. We group related items together into their shared container so that their sizes can be adjusted separately from other items outside the container. Moreover, containers can be nested within each other.

> **Horizontal container**
> Arrange the items placed within it side-by-side in a row. Can adjust width.

> **Vertical container**
> Stack the items within it on top of each other in a column. Can adjust height.

1. Vertical container to cover entire dashboard:
   Drag under `Objects`, the `Vertical Container` to the top of the worksheet
   Click the `Layout` tab
   `Position`: x=0, y=0 `Size`: w=1050, h=625
2. Add dashboard title:
   Drag the Text Object to the worksheet, call Sales Analysis
   Increase font size
3. Add the map to the extreme right side:
   Drag Sales by State to the top right corner
   Edit width by the down pointing arrow at top right of map
4. Create a new vertical container:
   Drag a `Vertical Container` to the left of the Title in that container
   Drag the Title by the handle into that container
5. Drag Stats underneath the Title
6. Drag a `Horizontal Container` object under the Stats to allow Time Series and Bar Chart to be next to each other:
   Drag the Time Series to the left of the container Drag the Bar Chart to the right of the container

Format

1. Sizes:
   Set Title height to 85
   Set Stats to fit entire view and height
   Set width of bar chart to 350
2. Borders and padding:
   Select `Dashboard` on the main menu, choose `Format Dashboard Shading`:
   Light Gray Remove the `Dashboard Format` menu
3. Each item, Title and the four worksheets:
   Select the item
   Right-click `Format`
   Set `Outer Padding` and `Inner Padding` to 10
   Set `Background` to White `Border` to Light Gray

Filters

In addition to the simultaneous display, another advantage of Tableau dashboards is that all items are linked in terms of filtering. Filter one item by choosing a specific value and all the other items on the dashboard update accordingly. For example, to see monthly sales and sales by product for California, select that state in the map display.

Filter:
Click on any item
Select the `Filter` icon
Select a specific value, such as a Month, Product, or State/Province depending on the item

Tableau: Visualize Colors

AUTHOR
David Gerbing

PUBLISHED
May 29, 2024, 03:31 pm

Tableau Default Palettes

The link to the video of implementing the Tableau instructions for this section follows.

Video: Tableau Colors. [2:32]

Create the Bar Chart

We begin with territory already covered, creating the standard bar chart. Once again, we will return to the employee data set and create the bar chart for the average salary across departments of the company. Open the employee data set Excel file, drag the department variable Dept over to the `Columns` shelf and the created Count variable over to the `Rows` shelf. Our bar chart emerges in Figure 1 with the default darkish blue bar colors.

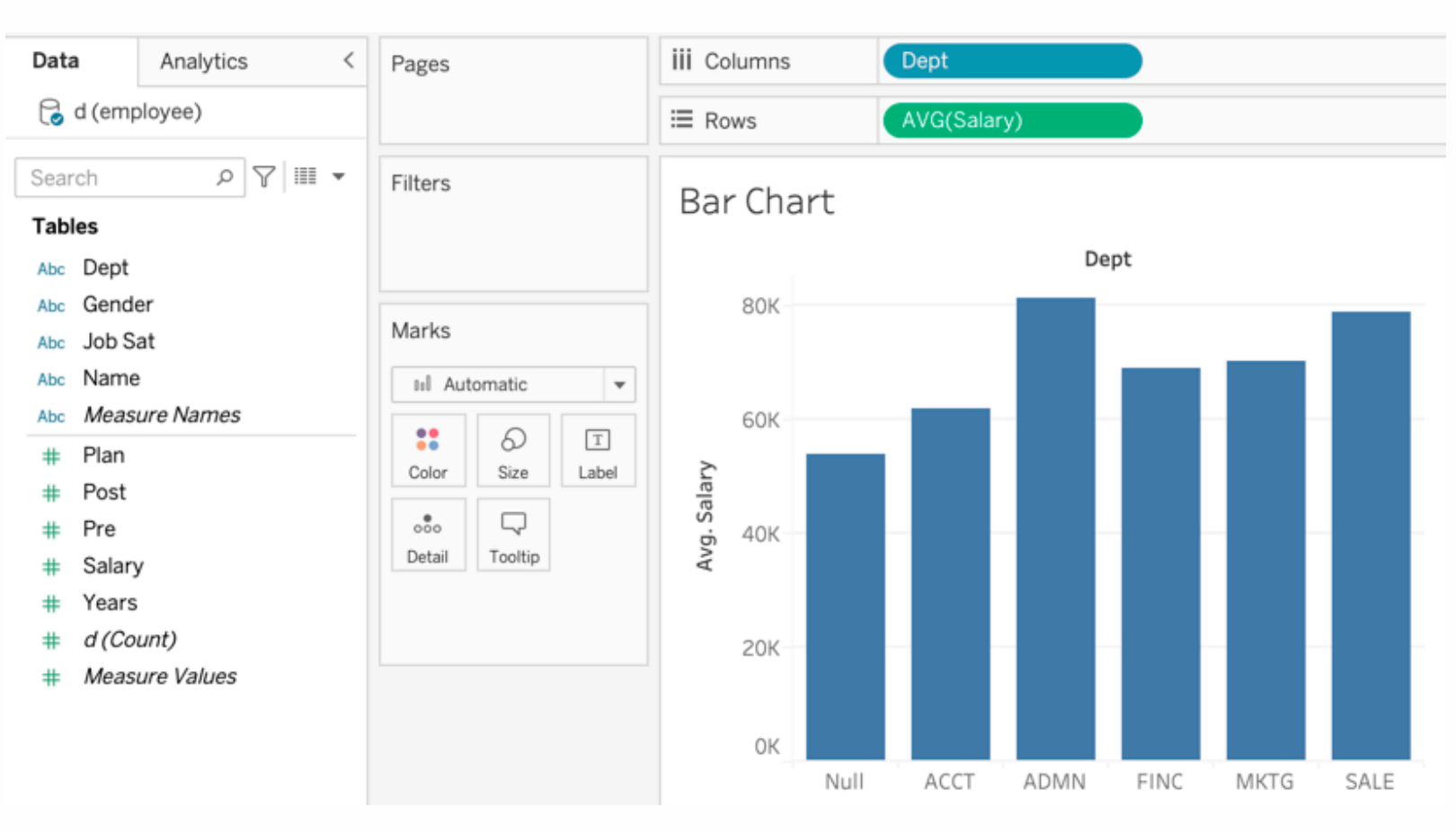


Figure 1: Bar chart of Employee variable Dept with default bar color.

There is one missing value for department in the data set, which Tableau renders as `Null`. As an option, remove the `Null` value. Drag the Dept name to to the `Filters` card, which then opens the dialogue box in Figure 2 that lets us retain or delete levels of the categorical variable as we wish.

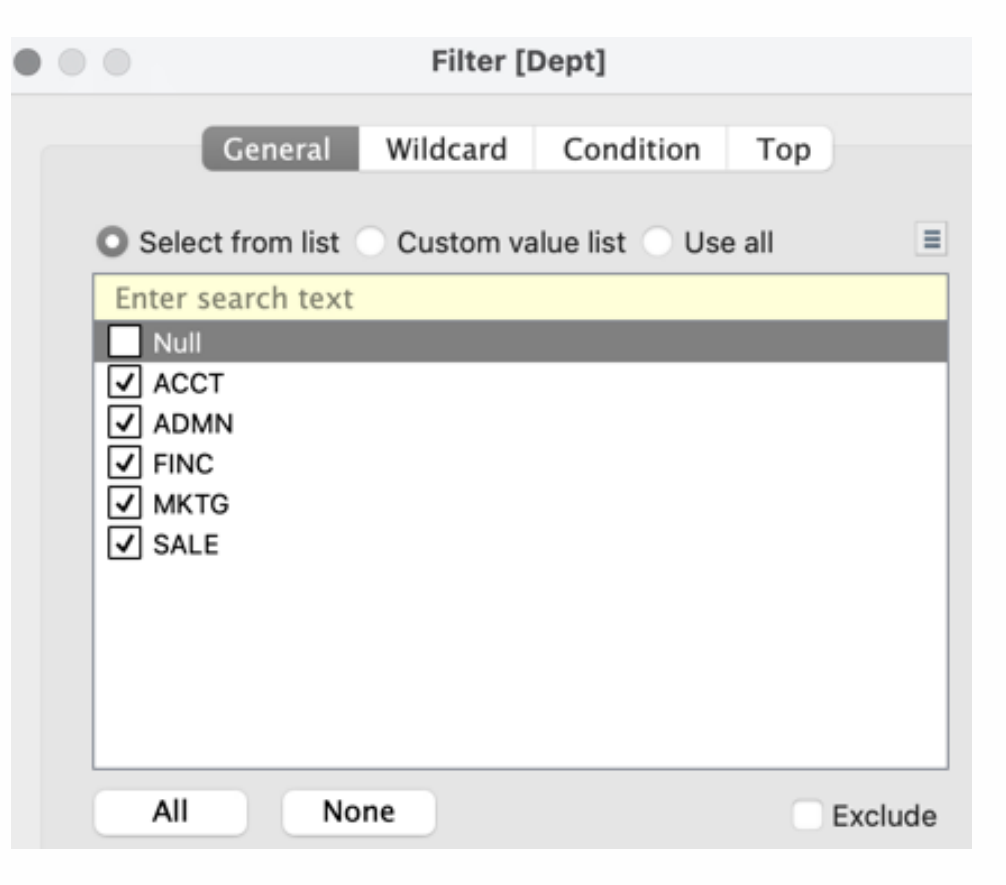


Figure 2: As an option, filter out the NULL value for missing data.

With the `Null` value removed, the five remaining non-missing levels of Department appear in the bar chart, which displays the average salary across the five departments, Figure 3.

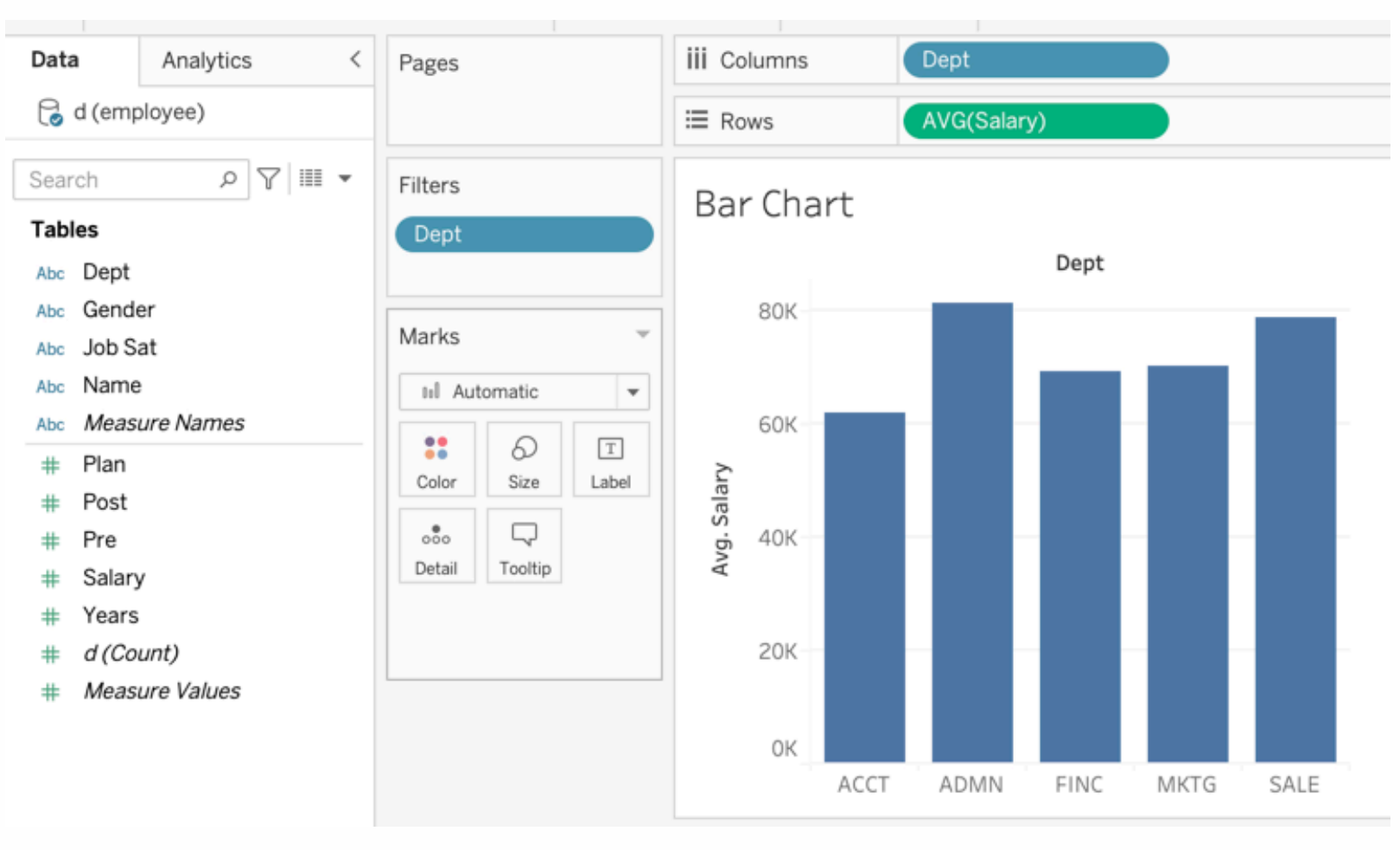


Figure 3: Bar chart of Dept with the Null value removed.

Default Quantitative Palette

There is no consistency of opinion as to whether the bars in a categorical variable bar chart should be the same color or vary in color. An argument for retaining the same color across bars is that different bar colors do not distract the viewer in different ways for the different bars. An argument for different bar colors is that the bar graph is more attractive and visually appealing, perhaps to longer hold the viewers attention.

Display the bars of the bar chart in the default tableau qualitative ballet is straightforward. Simply drag the Dept variable name over to the `Color` mark. The result is shown in Figure 4.

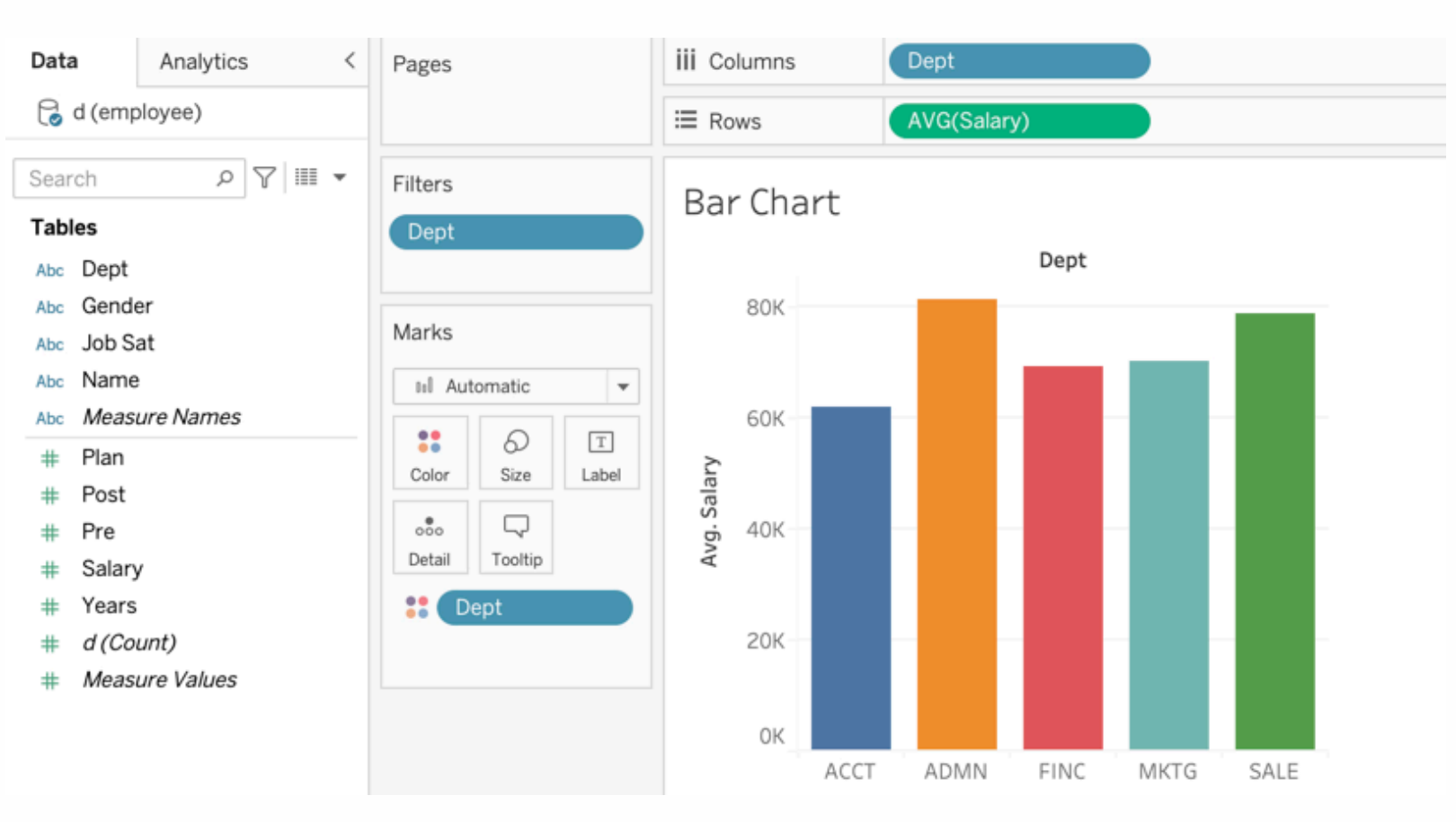


Figure 4: Default Tableau quantitative palette.

Additional Palettes

Tableau has several dozen built-in and available qualitative palettes. To access these palettes, double-click on the `Color` mark, which results in the `Edit Colors` prompt shown in Figure 5.

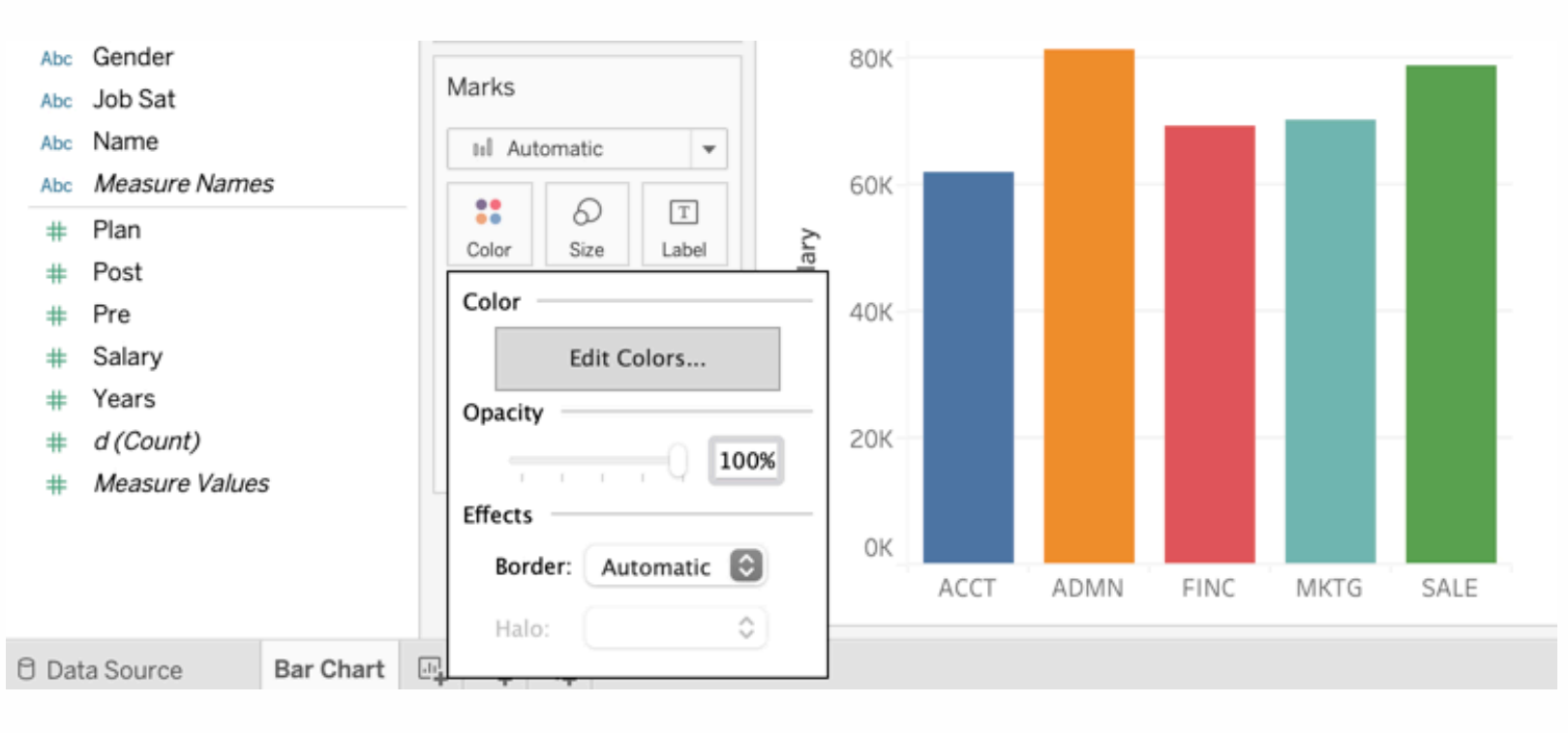


Figure 5: Edit Colors dialogue box from clicking on the Colors mark.

Clicking on `Edit Colors` leads to the access of the built-in palettes. The dropped down menu under `Select Color Palette:` displays the available palettes. The default palette is referenced by the name `Automatic`. This palette can accommodate the10 different colors shown in Figure 6. Also displayed are the levels of the categorical variable with their assigned colors,

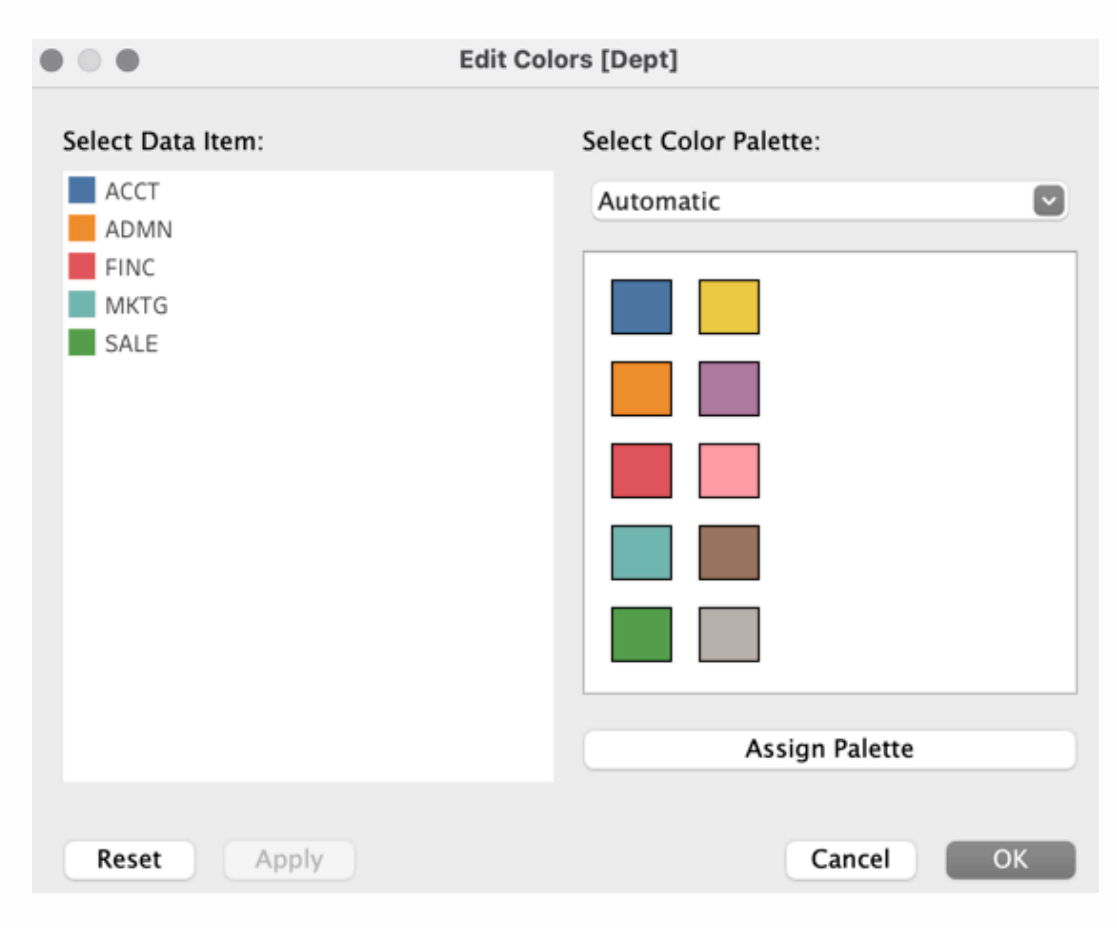


Figure 6: Access the built-in palettes.

If going to choose another palette, after selecting that palette from the drop-down, click `Assign Palette` and then click `Apply`.

Highlight a Bar

Another way to display a visualization, such as a bar chart, is to highlight one specific level of the categorical variable. Usually, the non-highlighted bars are displayed in a neutral color, such as a light-medium gray. In this example, select the `Gray Warm`` palette. Select one of the non-highlighted levels, here of Dept, then click on the desired color in the palette, shown in Figure 7.

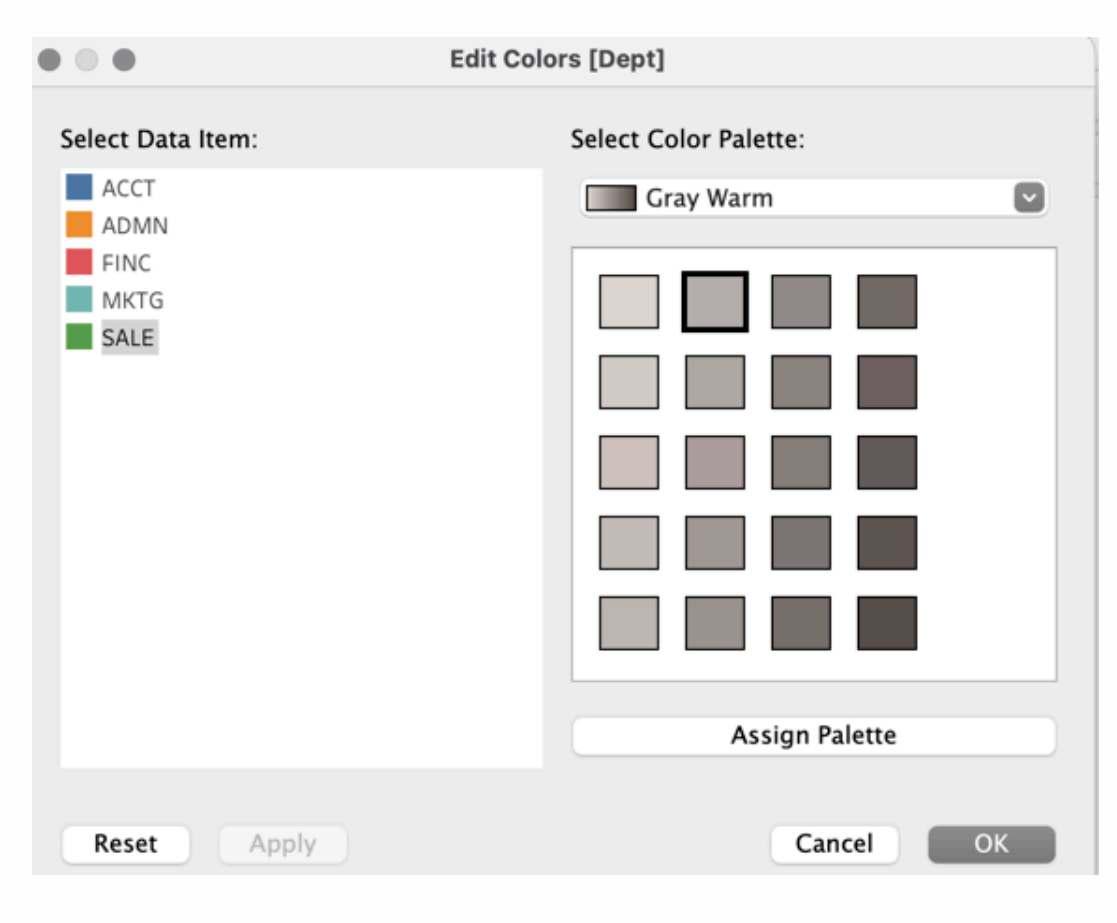


Figure 7: Assign a light gray color to the Sales bar.

Repeat that procedure for all non-highlighted bars, resulting in Figure 8.

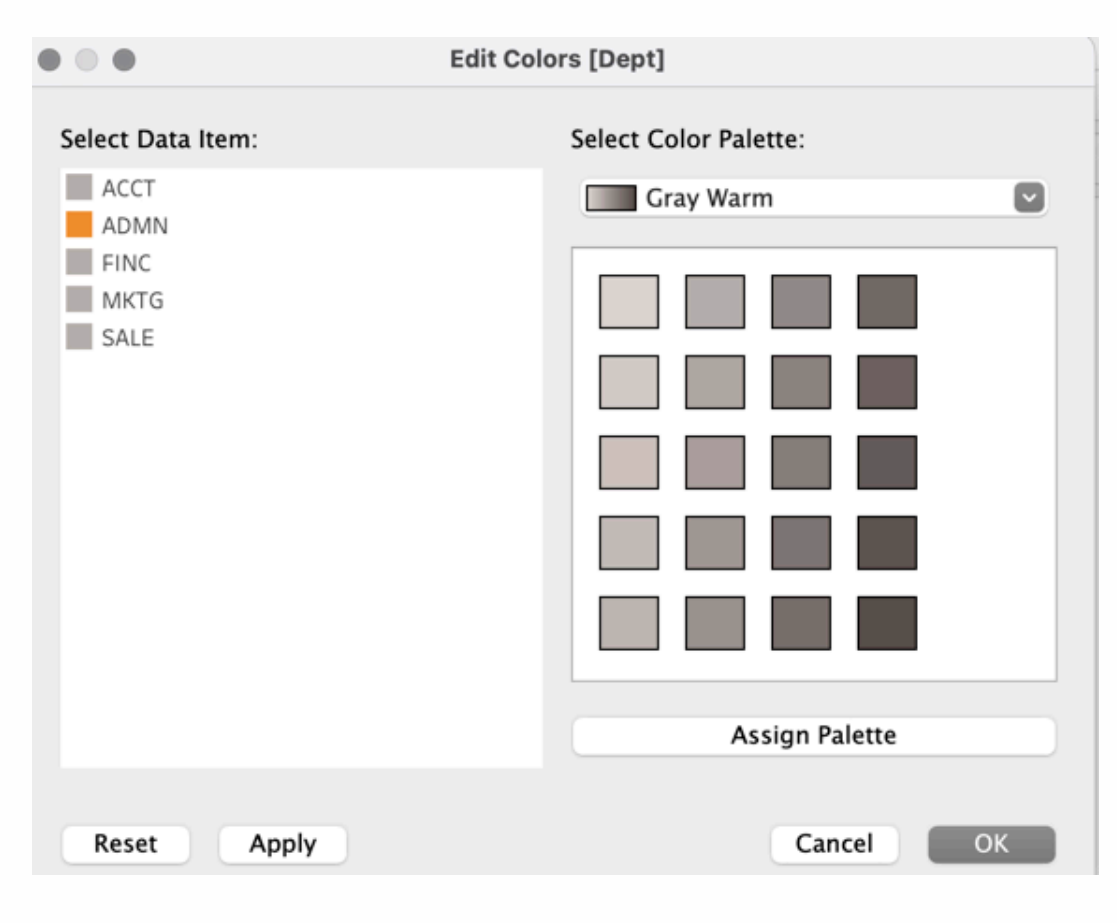


Figure 8: All bars assigned the same color except the bar to be highlighted, ADMN.

Click on `Assign Palette` and then `Apply` to obtain the bar chart shown in Figure 9.

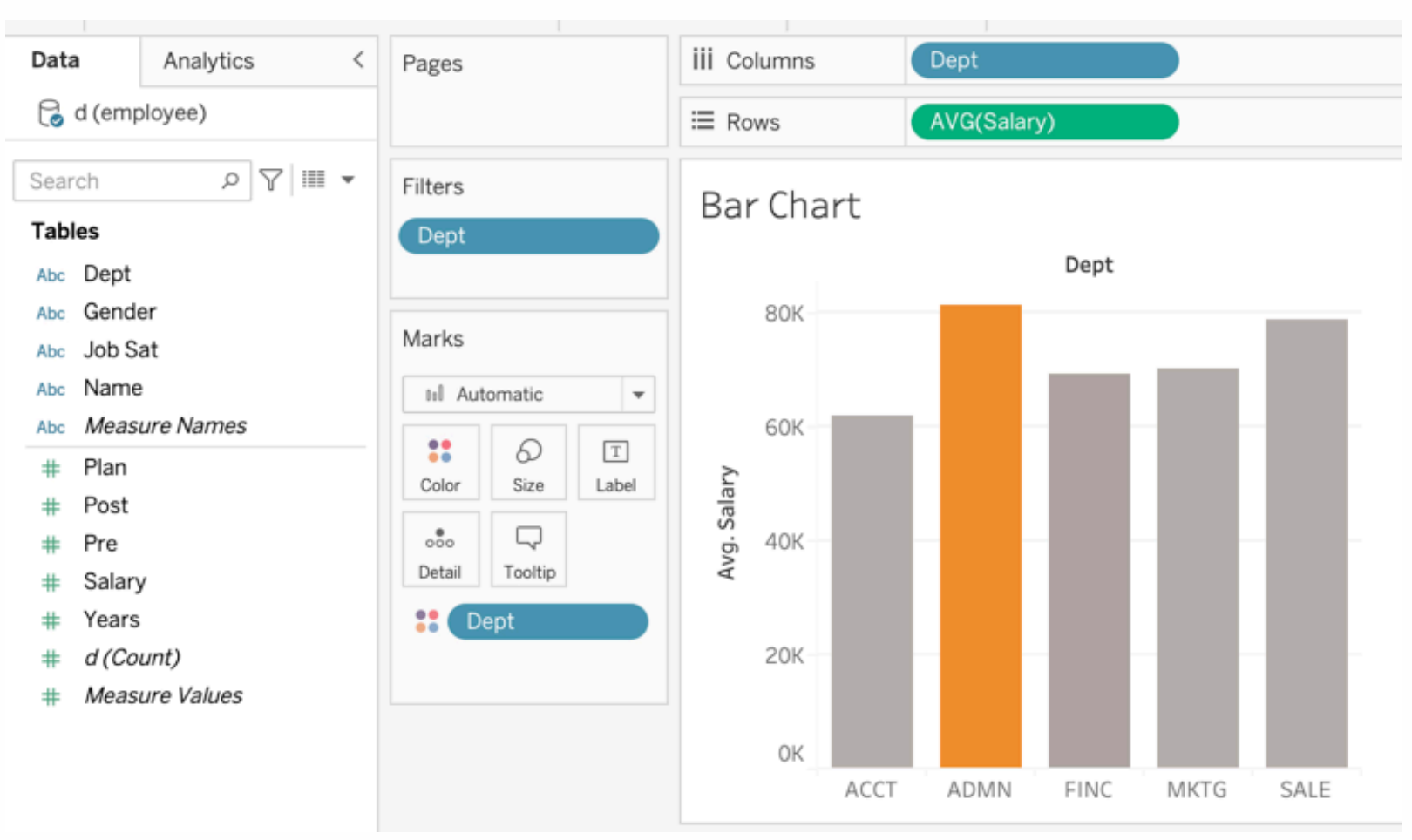


Figure 9: The bar chart with the ADMN bar highlighted.

Of course, this procedure could be repeated for each level of the categorical variable, with each corresponding bar highlighted a different color that does not conform to any built-in palettes.

Add a Custom Palette

The link to the video of implementing the Tableau instructions for adding a custom palette follows.

Video: Tableau Custom Palette. [5:03]

Create the Palette

In this example, we add the `lessR` qualitative palette to Tableau. First, we need the palette obtained from the `lessR` function `getColors()`.

```
pal <- getColors()
pal
```

```
[1] "#4398D0" "#B28B2A" "#5FA140" "#D57388" "#9A84D6" "#00A898"
    "#C97E5B" "#909711" "#00A3BA" "#D26FAF" "#00A76F" "#BD76CB"
```

How do we enter this palette into Tableau? Within your `Documents` folder is the `My Tableau Repository` folder. Inside that folder, find the `Preferences.tps` file. That file is in `XML` format, a generalization of HTML that allows for custom tags. We will need to add some HTML-like instructions in a text editor such as RStudio. If not modified before, opening that file in RStudio shows the following in Figure 10.

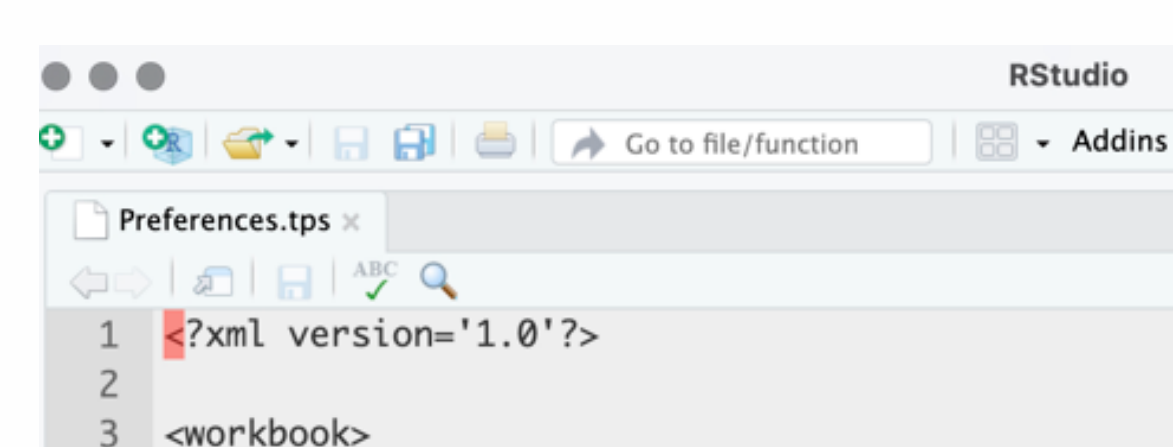


Figure 10: The unmodified Preferences.tps file.

The first task is to insert the begin `<preferences>` and close `</preferences>` tags, resulting in the following layout, shown in Figure 11.

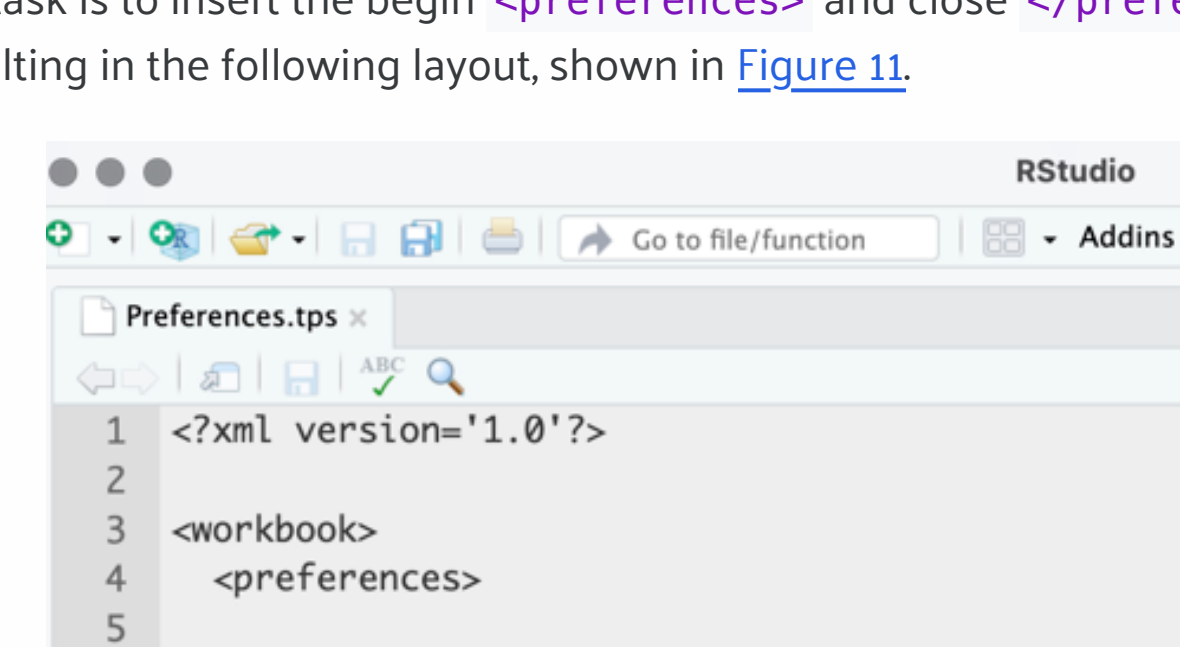


Figure 11: Add the preferences opening and closing tags.

Between the preferences tags create a new `color-palette` tags as follows: followed by the closing tag shown in Figure 12.

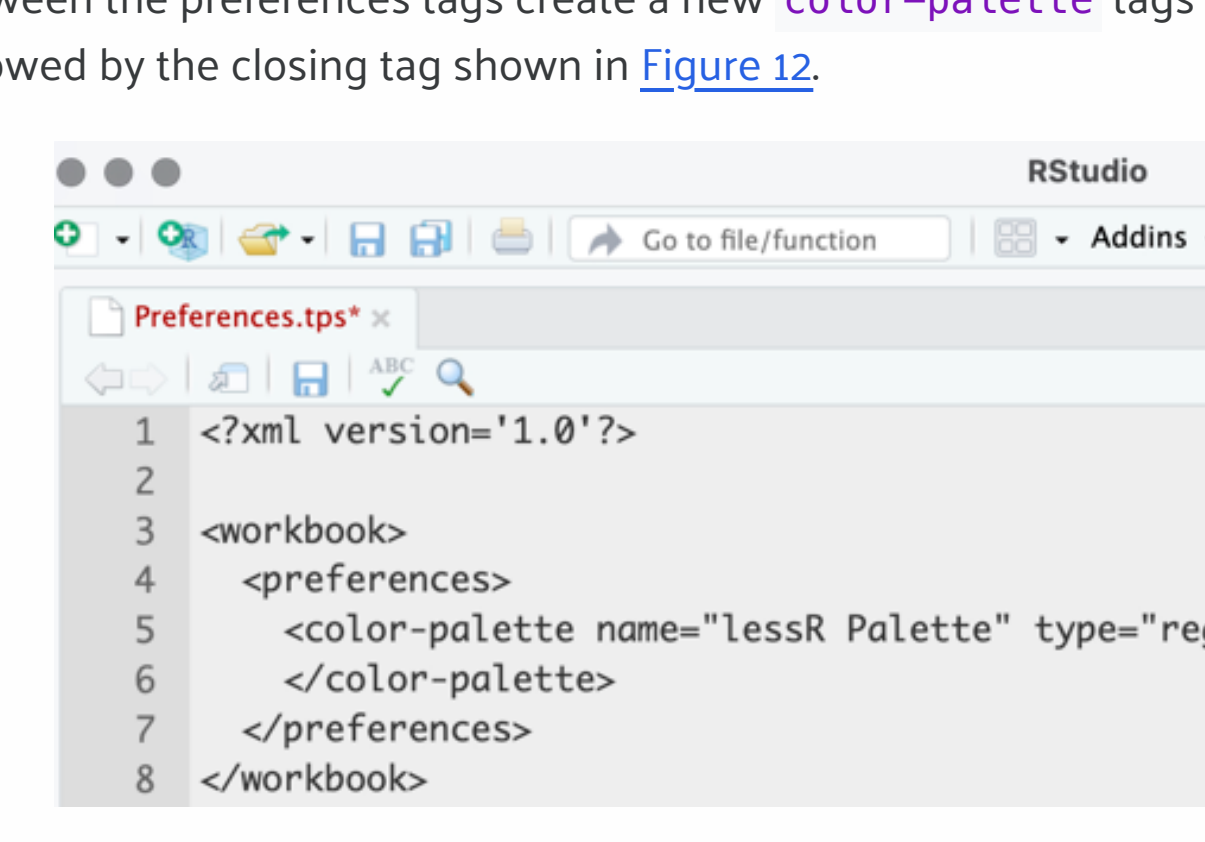


Figure 12: Add the opening and closing color-palette tags.

Whatever you specify for the `name` becomes the palette name in the Tableau system. Also, the specified value of `type` is `"regular"`, which denotes the creation of a categorical color palette.

Between the `color-palette` tags, now define the color palette. For *each* color in your palette, create an open and closed `color` tag, with its defining hexidecimal code inbetween. For example, a fully formed color palette looks like this, here with the first six colors from the `lessR` default quantitative palette as in Figure 13.

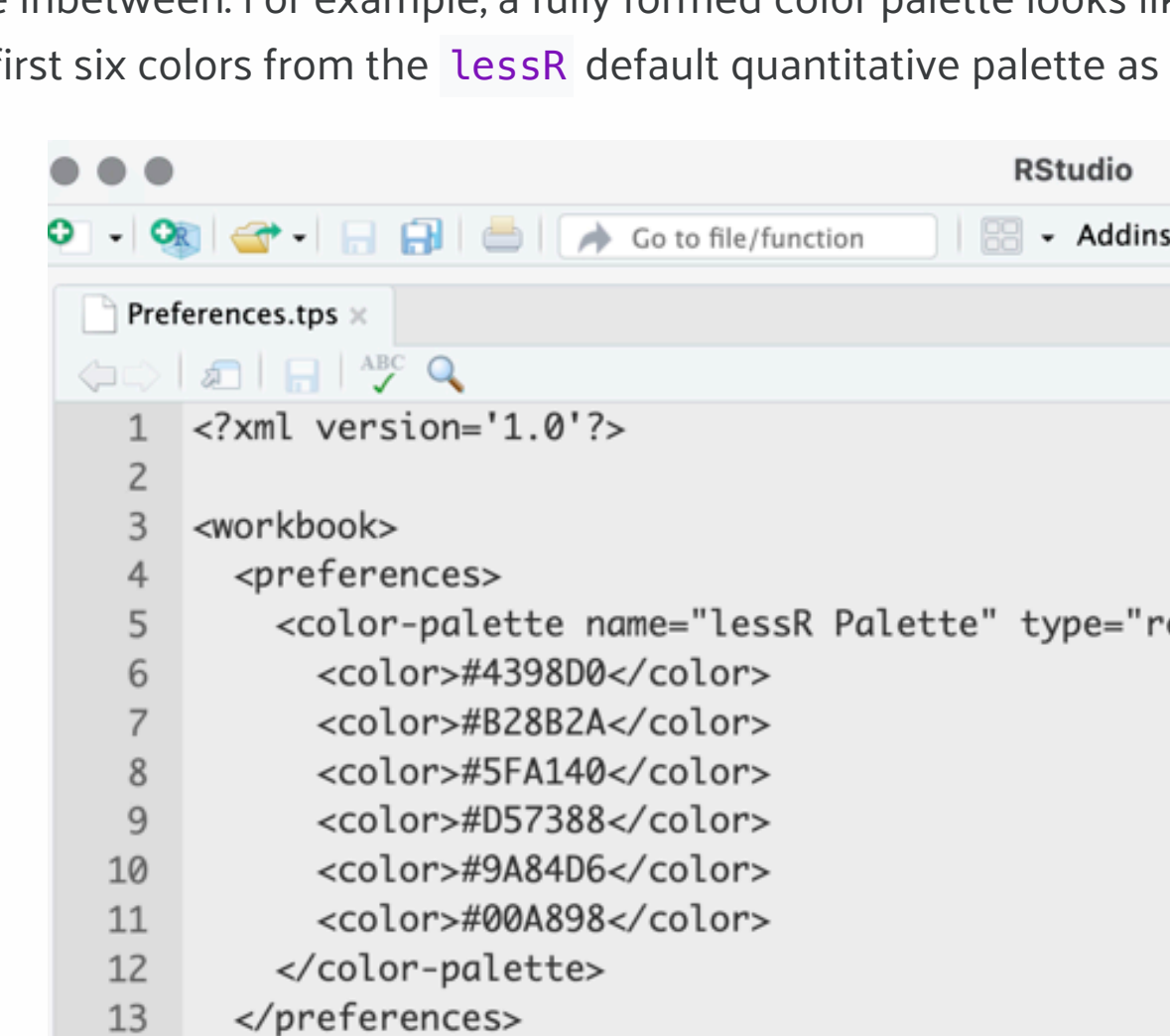


Figure 13: Add the opening and closing color tags for each color in the new palette.

When your color palette is complete, save the `Preferences.tps` file and restart Tableau.

Access the Palette

When back in Tableau, you will be able to locate your custom palette at the bottom of the drop-down list of palettes. As before, to access the palettes, click on the `Colors` mark and then the `Select Color Palette:` drop-down menu as in Figure 14.

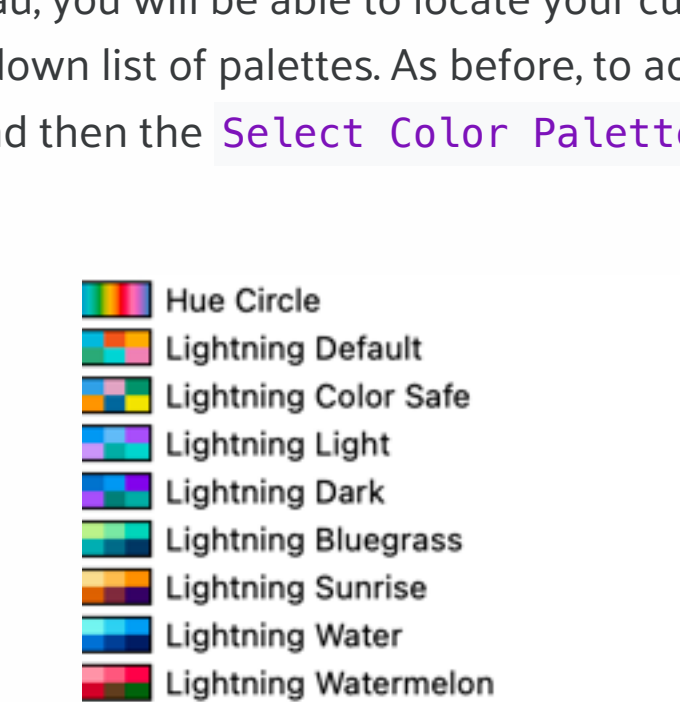


Figure 14: Access the added palette from the Edit Colors dialogue by clicking on the Colors mark.

Select at the bottom of the palette list the defined custom palette, here named `lessR palette`, as shown in Figure 15.

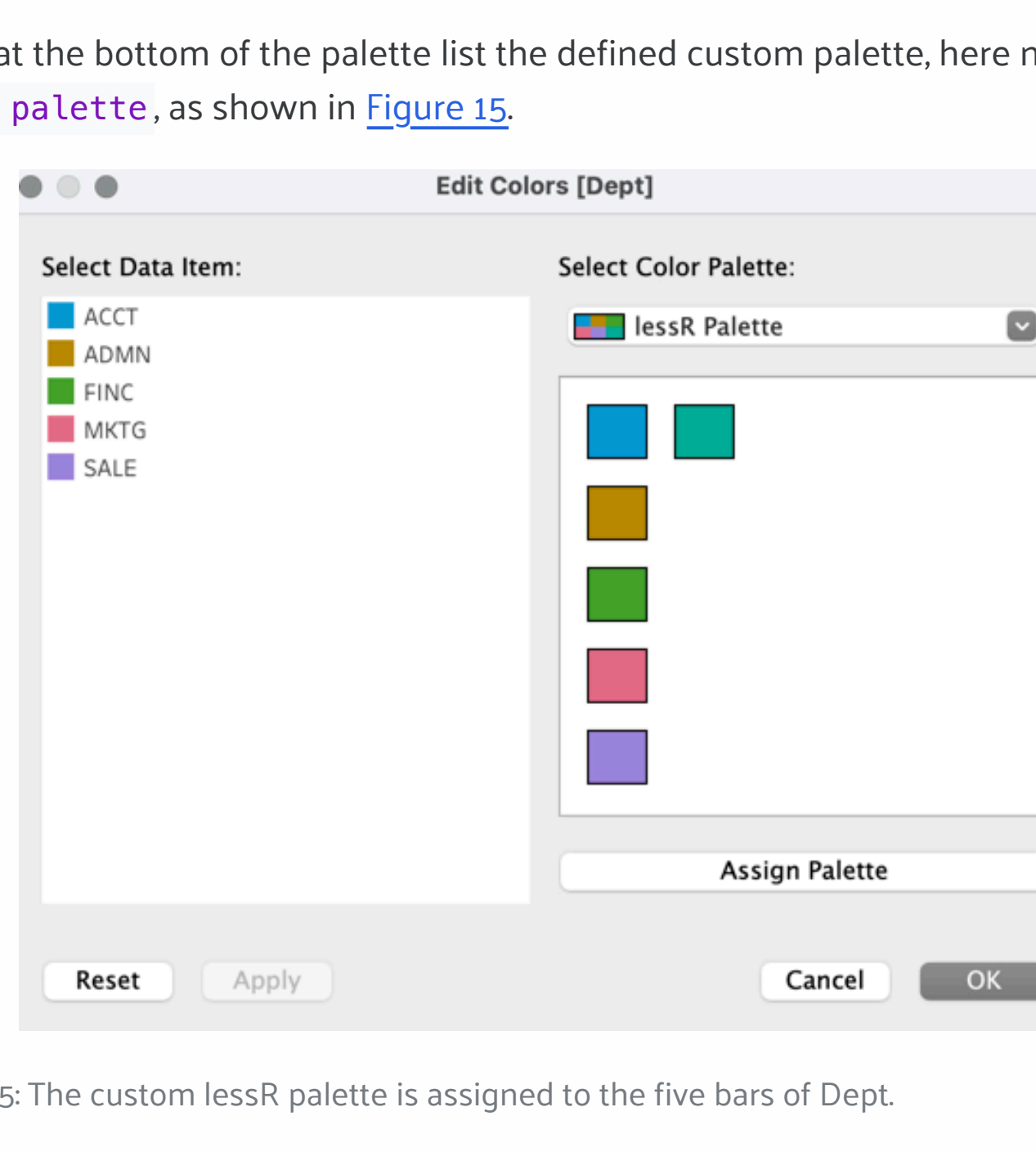


Figure 15: The custom lessR palette is assigned to the five bars of Dept.

Click on `Assign Palette` and then `Apply` to display the Dept bar chart in native `lessR` colors shown in Figure 16.

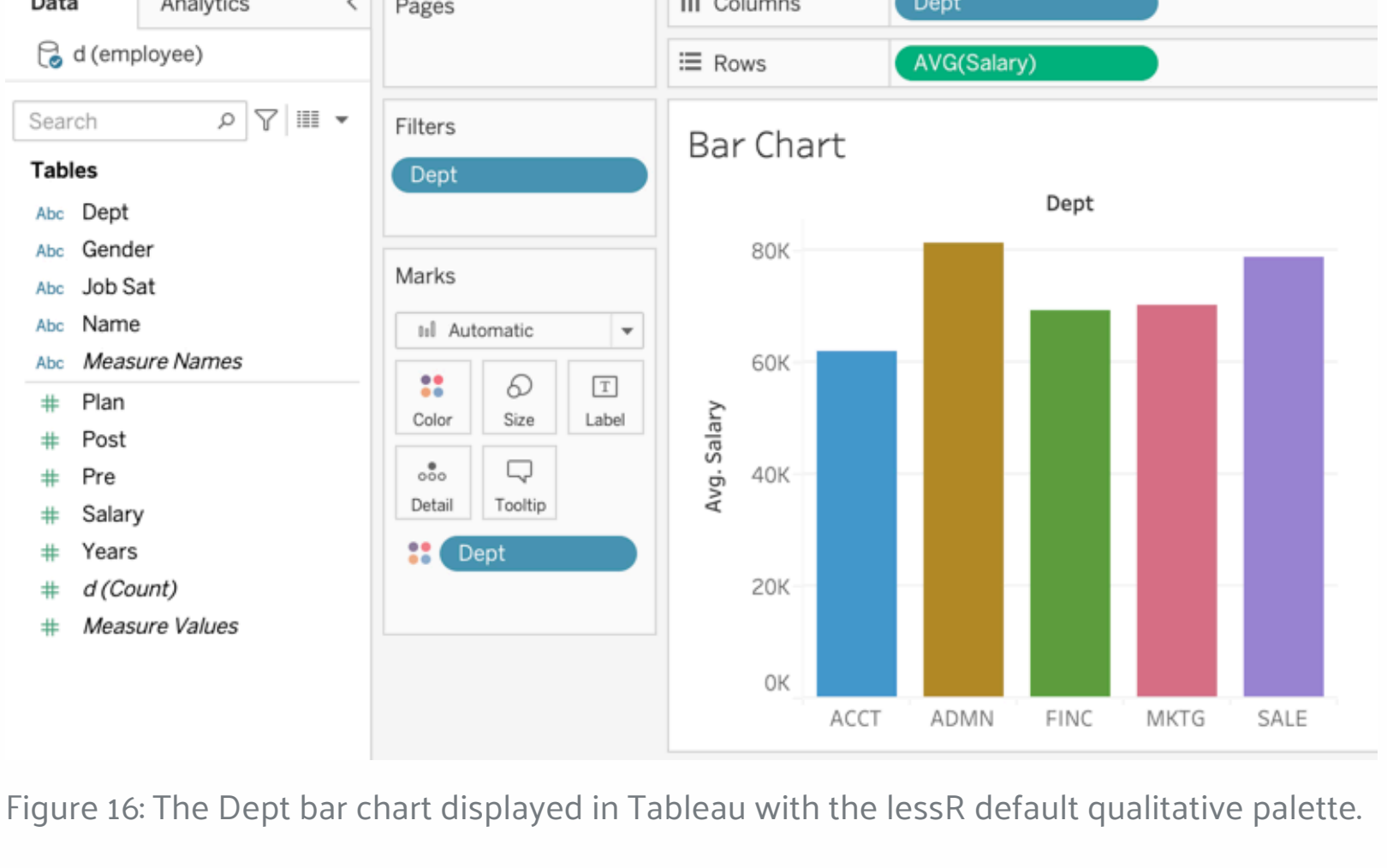


Figure 16: The Dept bar chart displayed in Tableau with the lessR default qualitative palette.