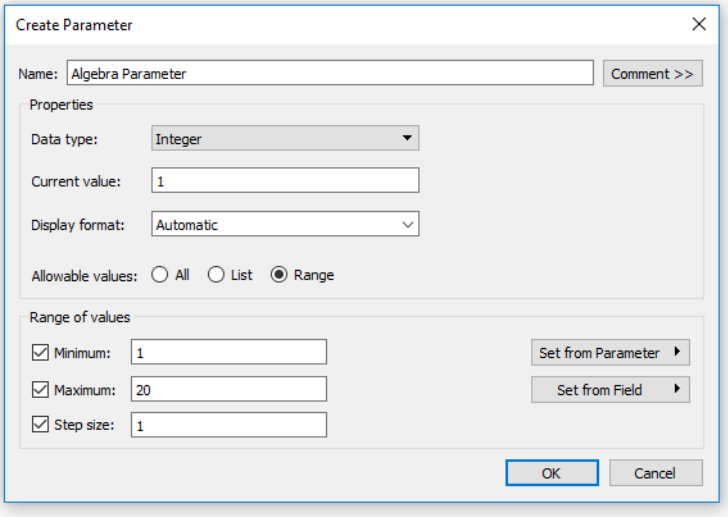
well as a step size, or multiple, for the parameter. After setting up the parameter to have a range of 1 to 20 with a step size of 1, my parameter looks like this and is ready to be saved by clicking the OK button:

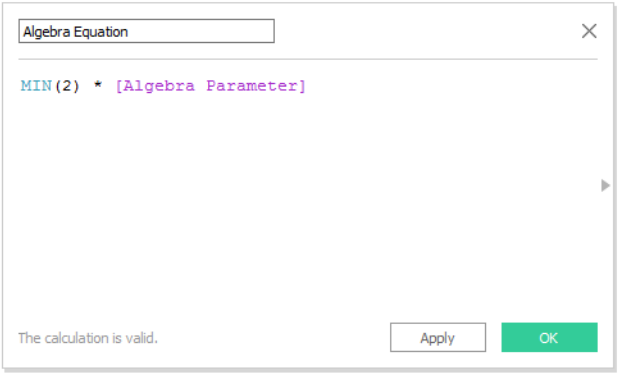


In order for a parameter to be useful, you must provide Tableau instructions for how to use it. This is accomplished through calculated fields. Being that the equation for our use case is 2 multiplied by *X*, or our algebra parameter, your first instinct may be to create a calculated field that says:

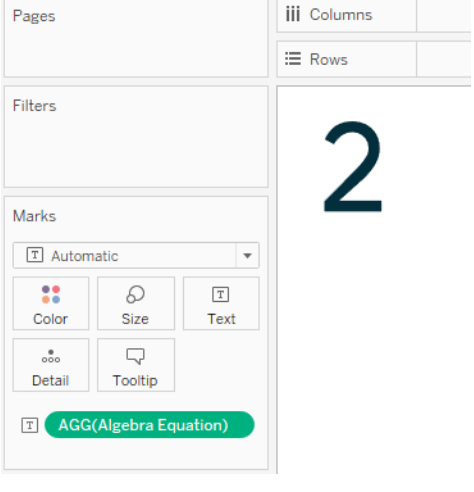
```
2*[Algebra Parameter]
```

Being that the current value of the parameter is 1, we would expect the answer to be 2*1 = 2. However, this calculation will be computed for every row in the data, which would not provide the answer we are looking for. Should we add this calculated field to the view, we would get an answer of 19,988, which is 2 multiplied by 1, multiplied by the number of records in the dataset (9,994 for the Sample – Superstore dataset).

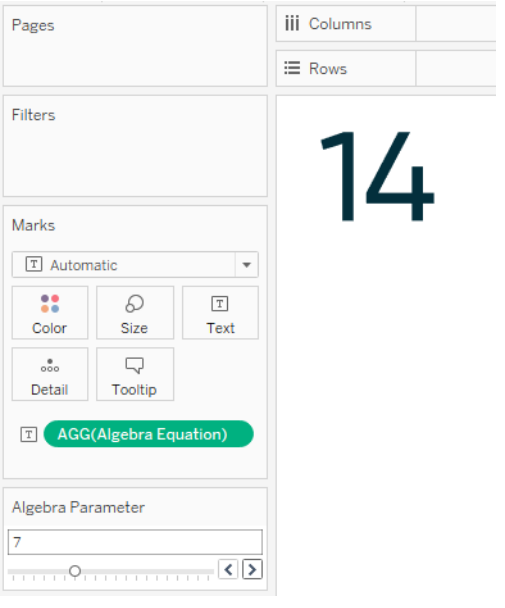
To alleviate this, add an aggregation of MIN to the 2 so the Algebra Parameter is guaranteed to be multiplied by 2:



Now if I add my newly created Algebra Equation calculated field to the Text Marks Card to display the answer on a view, I see the answer I am looking for, MIN(2) multiplied by 1 equals 2:



Finally, to provide the power of selecting the value of *X* in our example to the end user, you must provide access to the parameter. To do this, right-click the newly created Algebra Parameter and choose "Show parameter control." A new selector will appear in the upper-right corner of the view that allows the user to choose from the allowable values that were set up when the parameter was created. Notice that the current value is 1, the maximum value is 20, and the numbers can be selected in multiples of 1, just like we set up. Here's what my final view looks like when I've shown the parameter control, moved the parameter control from its default location to below the Marks Cards, and changed the selection from 1 to 7:



In this example, the end user has chosen the number 7, which powered the underlying calculation of 2*7, to produce the answer of 14. Imagine all of the power that can be put into the hands of your end users through the use of parameters!

For just a few other possibilities, see the following chapters:

- Chapter 69, "How to Compare Two Date Ranges on One Axis"
- Chapter 66, "How to Change Date Aggregation Using Parameters"
- Chapter 48, "How to Make a What-If Analysis Using Parameters"
- Chapter 53, "How to Create and Compare"
- Chapter 64, "Allow Users to Choose Measures and Dimensions"

CHAPTER 15

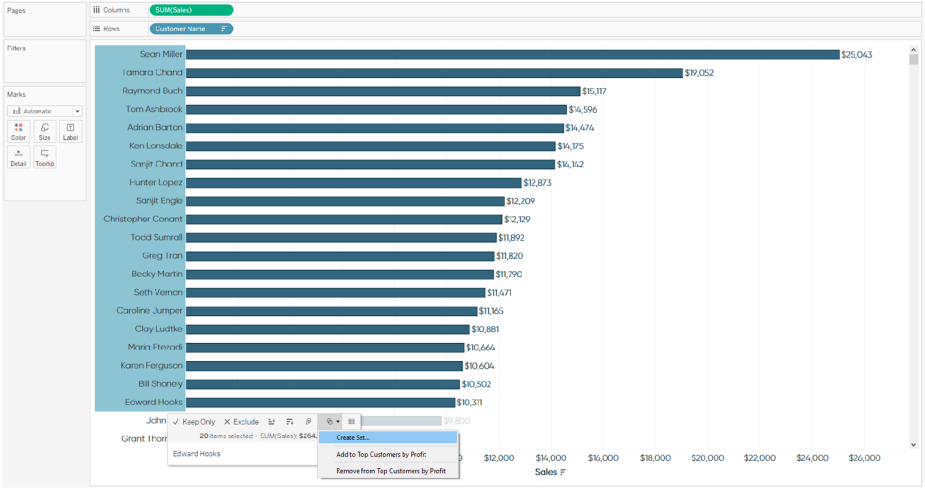
An Introduction to Sets

Tableau sets allow you to isolate specific segments of a dimension, which can then be used in several different ways to find insights in your data. This chapter provides instructions on how to build sets as well as five different ways they can be used to enhance your analyses.

Sets can be thought of as custom segments, but unlike dimension fields, they are always binary. In other words, you are either in the set or not. Other than that one restriction, sets can be created for just about anything. You can pick individual dimension members to place in a set, have sets be based on quantitative thresholds, created with the top or bottom performing dimension members, and more.

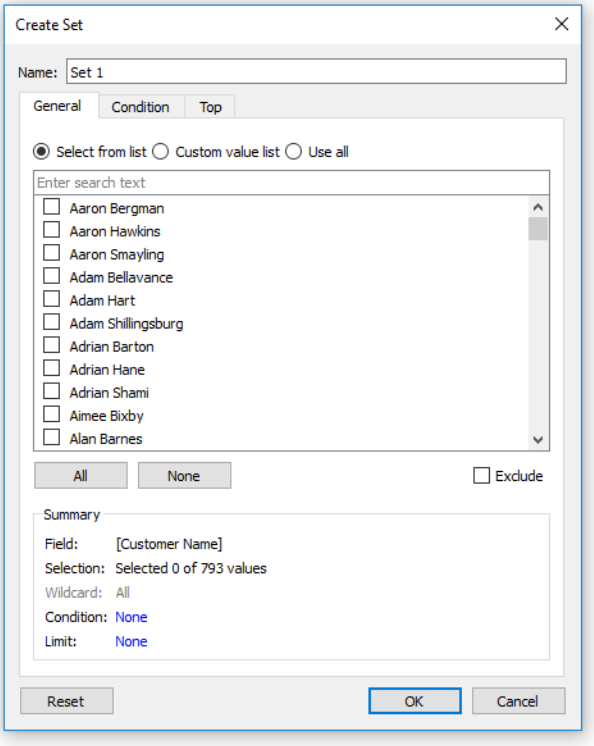
How to Create a Set in Tableau

Similar to creating filters in Tableau, sets can be created in a couple of different ways. The first and most straightforward method is to simply select the dimension members on a view, then hover over one of the dimension members, click the Venn diagram icon that appears, and choose Create Set. Here's what that would look like if I wanted to create a set from the top 20 customers by sales in the Sample – Superstore dataset:



After clicking Create Set and giving the set a name, the set will appear in its own area on the Data pane. This set that tells us whether each customer name is in or out of the top 20 customers by sales is now available to use in our analyses. Note that sets created with this method are static, so the top 20 will not dynamically change should a new customer enter the top 20.

You can also create a set by right-clicking the dimension the set will be created from on the Dimensions area of the Data pane, hovering over Create, and clicking Set. After following the preceding steps on the Customer Name dimension, the following dialog box appears:

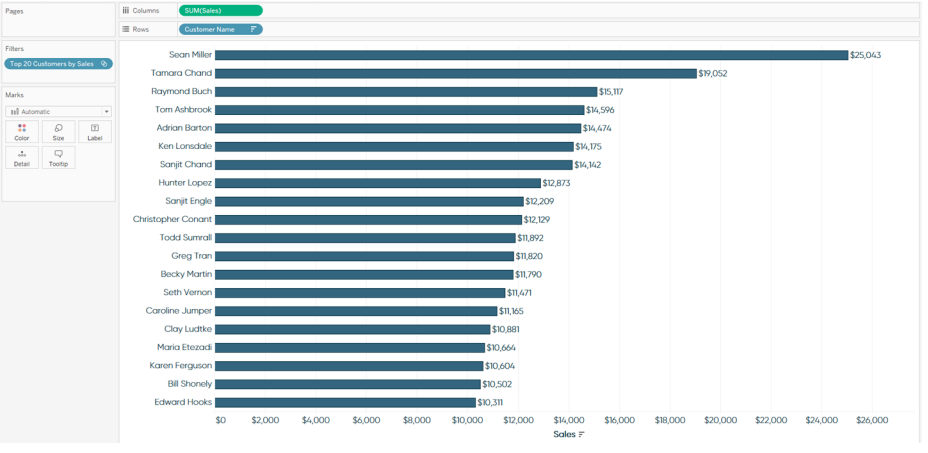


The first tab provides the ability to manually pick and choose the dimension members to be included in the set as we have done with the first method. The remaining two tabs can be used to base the set on a condition or Top N, respectively. Computed sets like this will dynamically change when the dataset is updated. Whichever method you choose to create your set, a new field will be created to use in the following examples.

Five Ways to Use Tableau Sets

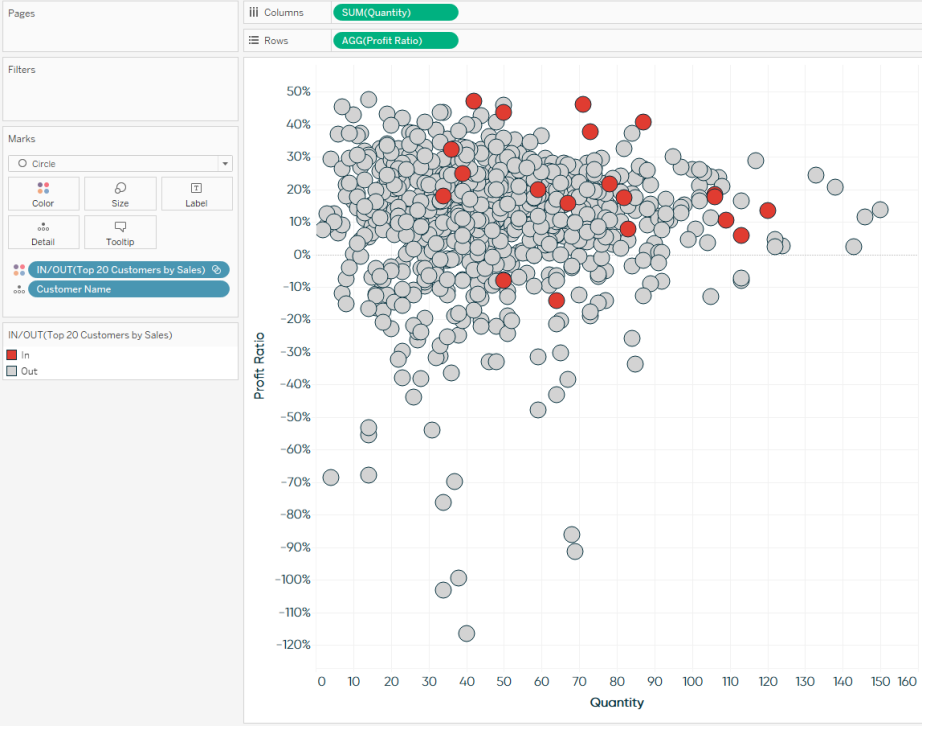
As a filter

Sets can be used as a filter by right-clicking a set from the Sets area of the Data pane and choosing Show Filter. Sets are binary, so when you add them as a filter, you will only have the option to choose whether the marks on the view are in or out of the set. Here's how my bar chart of sales by customer name in descending order looks after showing the filter for Top 20 Customers by Sales and keeping only the names that are in the set:



To encode marks

You can encode marks on a view by a set by dragging the set from the Sets area of the Data pane to the Color Marks Card. Here's an example where I've done this to make the top 20 customers by sales stand out on a scatter plot:



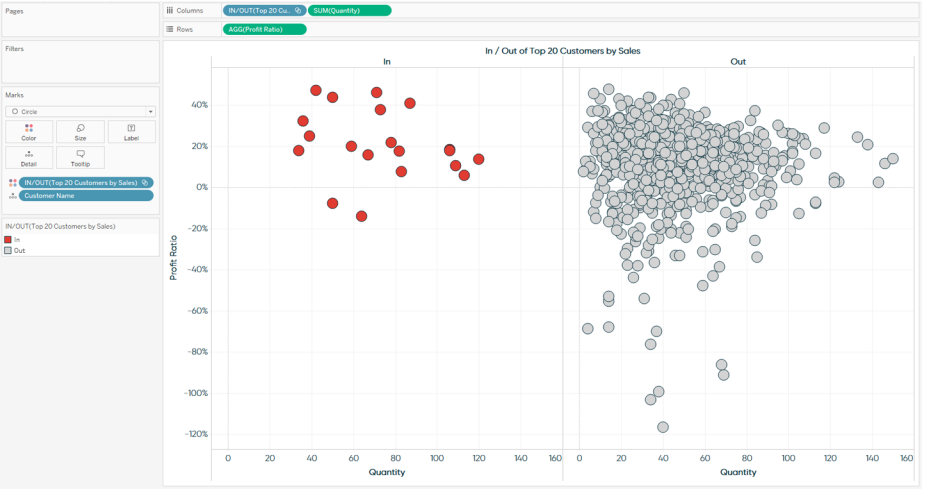
In calculated fields

Sets can be used in calculated fields just like dimensions or measures. This way you can treat dimension members differently based on whether or not they are in a set. Here is a simple example being used to rename the sets:

This formula is saying that if a customer name is in the set, name the set "Top 20 Customers"; otherwise call it "Other." Note that this example is used to show you that sets can be used within calculated fields, but this same renaming could have been accomplished by editing the aliases of the set.

As dimension fields

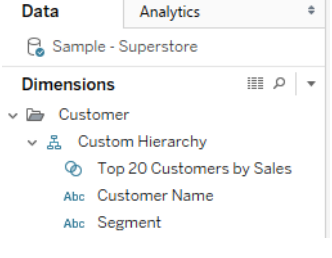
Sets can be used just like dimensions in that if you add a set field to the Columns Shelf or Rows Shelf, the view will be "sliced and diced" by that field. Here is how my scatter plot from earlier looks if I create two separate columns for in and out of the set by dragging the set from the Sets area of the Data pane to the Columns Shelf:



Within a custom hierarchy

Sets can be used as part of a custom hierarchy in Tableau, which allows you and your end users to easily drill down and back up across different dimensions. For example, you may want to create a hierarchy with customer names that starts with the Top 20 Customers by Sales set, then drills down to individual customer names, then to their segment.

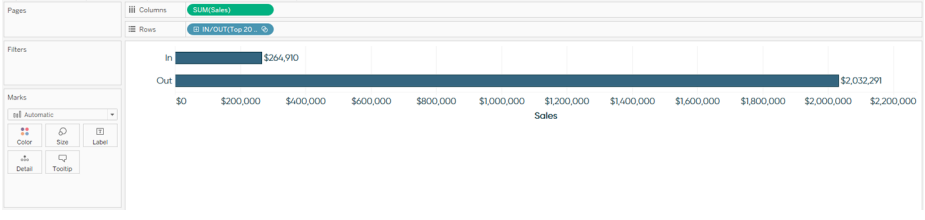
Create a custom hierarchy by selecting the Customer Name and Segment dimensions, right-click, hover over Hierarchy, and choose Create Hierarchy. Once the custom hierarchy is created, you can drag the Top 20 Customers by Sales set into the hierarchy on the Dimensions area of the Data pane and rearrange them to the desired order of the drilldown. At this point, I see this hierarchy in the Dimensions area of the Data pane:



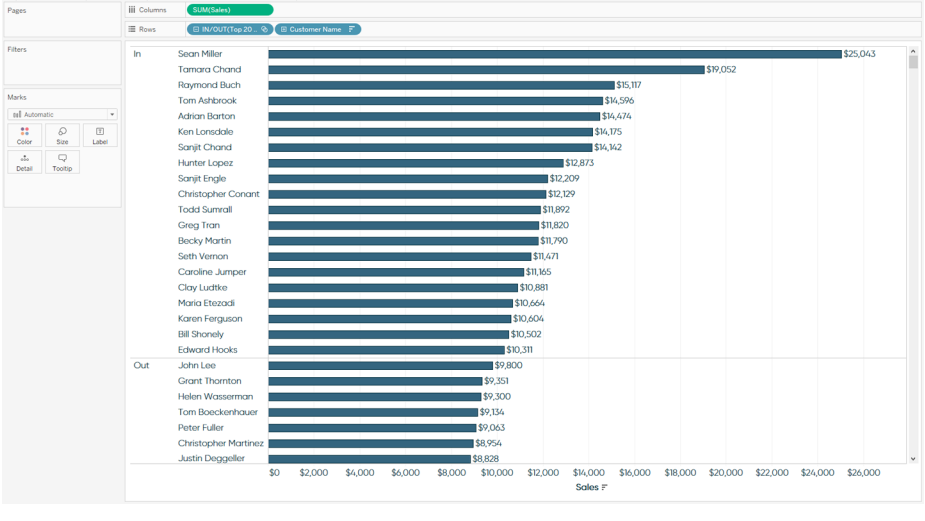
Now that the hierarchy is in place, if I replace the Customer Name dimension with the Top 20 Customers by Sales set dimension in the preceding bar chart example, I will be able to drill down from the top 20 set, to customer name, to

segment. This is accomplished by clicking the "+" symbol on the field(s) on the Rows Shelf.

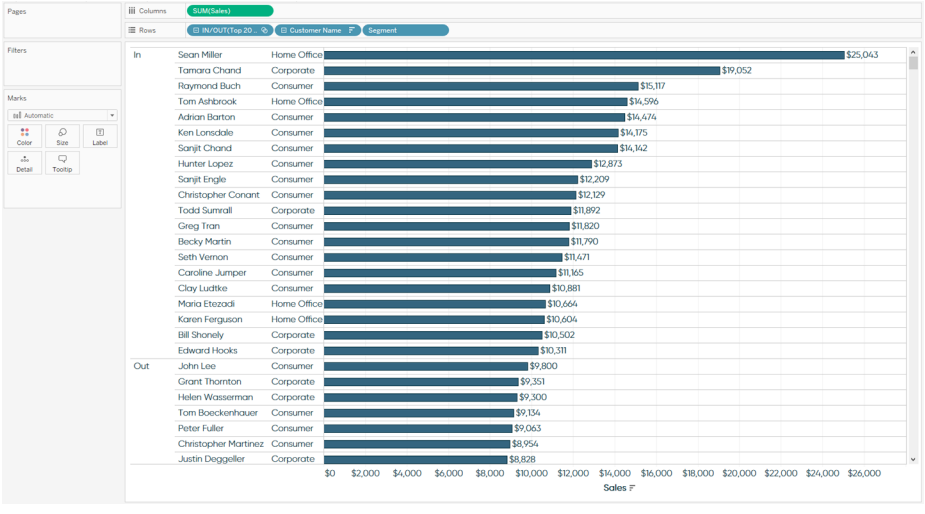
Here's how the view looks when the bars are aggregated at the highest level of the hierarchy. This view shows sales generated from customers in the set versus sales generated from customers out of the set:



Here's how the same view looks after clicking the + symbol on the blue pill on the Rows Shelf. We now see sales by whether or not the customer is in our set, but also the names of individual customers:



Lastly, because we built a third level of the hierarchy in for the Segment dimension, we can drill down one more time by clicking the + symbol on the Customer Name pill. This view first breaks sales down by the set that we created, then Customer Name, then Segment:


Pages
Columns
SUM(Sales)
Rows
IN/OUT(Top 20 ...
Customer Name
Segment
Filters
Marks
Automatic
Color
Size
Label
Detail
Tooltip
In
Sean Miller Home Office $25,043
Tamara Chand Corporate $19,052
Raymond Buch Consumer $15,117
Tom Ashbrook Home Office $14,596
Adrian Barton Consumer $14,474
Ken Lonsdale Consumer $14,175
Sanjit Chand Consumer $14,142
Hunter Lopez Consumer $12,873
Sanjit Engle Consumer $12,209
Christopher Conant Consumer $12,129
Todd Sumrall Corporate $11,892
Greg Tran Consumer $11,820
Becky Martin Consumer $11,790
Seth Vernon Consumer $11,471
Caroline Jumper Consumer $11,165
Clay Ludtke Consumer $10,881
Maria Etezadi Home Office $10,664
Karen Ferguson Home Office $10,604
Bill Shonely Corporate $10,502
Edward Hooks Corporate $10,311
Out
John Lee Consumer $9,800
Grant Thornton Corporate $9,351
Helen Wasserman Corporate $9,300
Tom Boeckenhauer Consumer $9,134
Peter Fuller Consumer $9,063
Christopher Martinez Consumer $8,954
Justin Deggeller Corporate $8,828
$0 $2,000 $4,000 $6,000 $8,000 $10,000 $12,000 $14,000 $16,000 $18,000 $20,000 $22,000 $24,000 $26,000
Sales

CHAPTER 16

An Introduction to Level of Detail Expressions

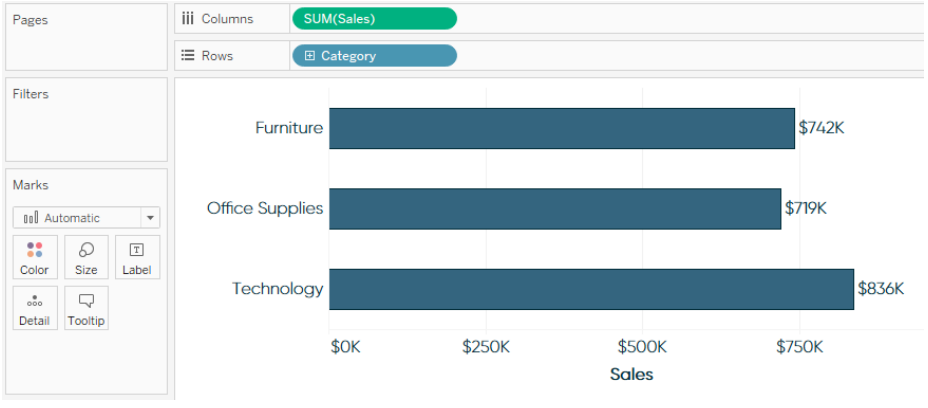
Tableau level of detail expressions (LODs) allow you to change the most granular place where an analysis takes place. An analysis such as AVG(Sales) by State includes the dimension that you are using to break down the measure by as well as the aggregation of the measure. Prior to Tableau version 9.0, and without some clever technical hacking such as duplicating a data source or leveraging table calculations, you were stuck using the same level of detail for an entire view. Now that you can alter the level of detail for specific measures, you can compare and contrast numbers at different granularities within the same chart.

With level of detail expressions in Tableau, you can now compute things such AVG(Sales) by State minus AVG(Sales) for the entire dataset to see how the sales per state compare to the overall average.

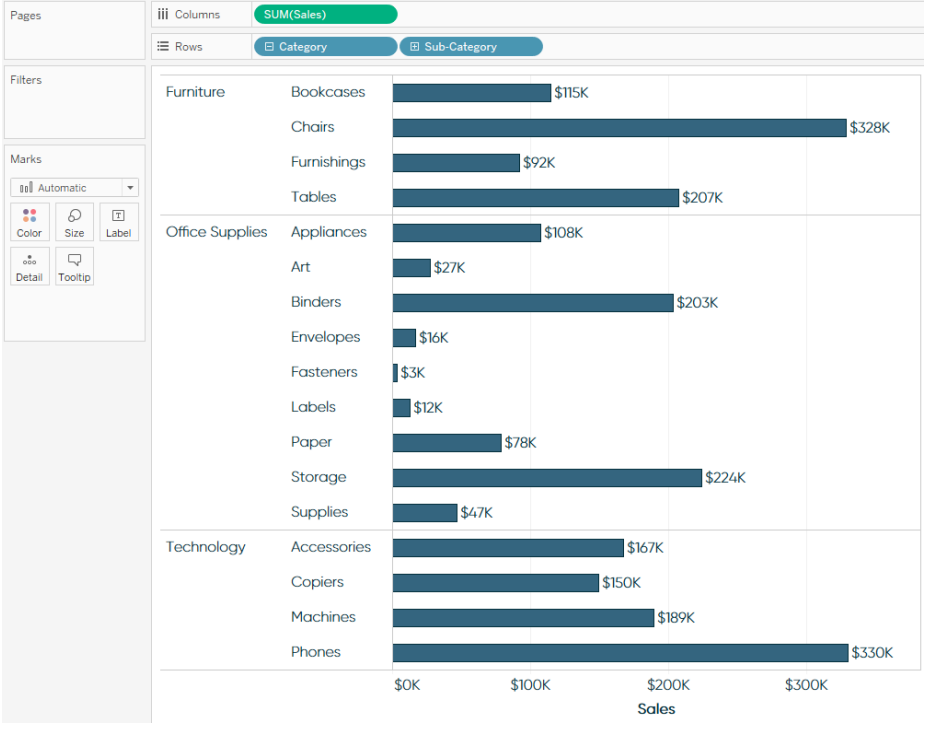
This unlocks a great deal of flexibility in your analyses because you can explicitly define the level of detail for different numbers in your view. Here are just 15 of limitless ways to take advantage of Tableau LOD expressions. I will also share a couple of resources that show you how I've used level of detail in my own Tableau workbooks at the end of this chapter, but this chapter mainly serves as an introduction to what level of detail is and the syntax needed to control it.

An Introduction to Tableau Level of Detail Expressions

In Chapter 10, we introduced the idea of level of detail and how you can make a visualization more granular by adding dimensions to the Detail Marks Card. Here is another example to help illustrate what a Tableau visualization's level of detail is:



In this simple Sales by Category bar chart from the Sample – Superstore dataset, the most granular level of detail in the analysis is Category. Since we are only slicing and dicing the Sales measure by the Category dimension, there are only three marks; one for each category. We can make this analysis more granular by adding an additional dimension to the view. Here's what the bar chart looks like if I add the Sub-Category dimension to the Rows Shelf:



We are now slicing and dicing the Sales Measure by both the Category and Sub-Category dimensions. We have changed the level of detail for this view and made our analysis more granular.

But what if we wanted to look at it both ways? Perhaps we want to show the sales per sub-category in one column, the sales per category in a second column, and even divide the two by each other to determine how much each sub-category is contributing to its respective category.

That's where Tableau's level of detail expressions come in. level of detail expressions have their very own syntax in Tableau, which looks like this:

- An open curly bracket "{"
- Followed by one of three level of detail expressions: FIXED, INCLUDE, or EXCLUDE
- Followed by zero or more dimensions you want to be fixed, included, or excluded from the computation
- Followed by a colon ":"
- Followed by an aggregated measure
- Followed by a closing curly bracket "}"

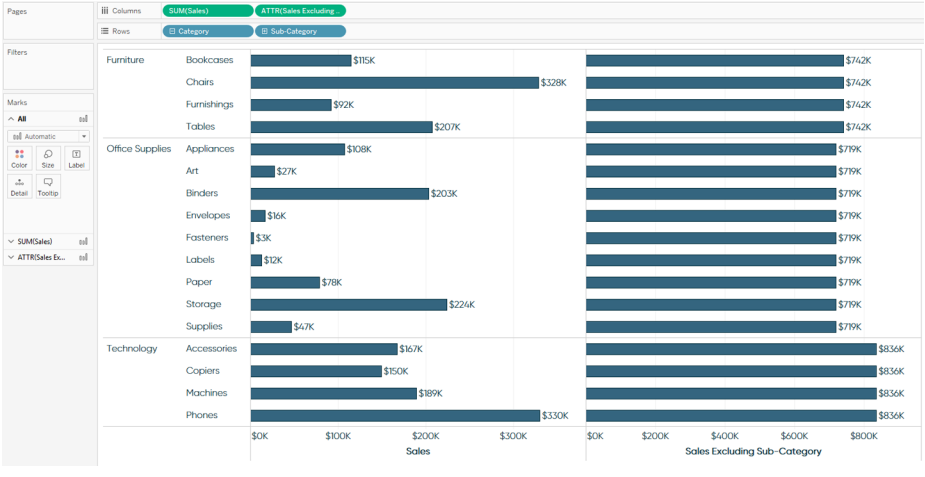
The three level of detail expressions—FIXED, INCLUDE, and EXCLUDE—are fairly intuitive names for what they're going to do (i.e., fix the measure at a certain level of detail, include dimensions that are not on the view, or exclude dimensions that are on the view, respectively). As with many things in Tableau, there are multiple ways to get to the same answer, and there is not always a "best" way to go about a certain solution.

Your level of detail expressions may take some experimenting to get the result you want, but I have found the EXCLUDE expression to align best with the way I think about aggregations. The reason this works well for me is because I can make a view as granular as I'd like as I normally would, then "back out" to a comparison metric by excluding certain levels of granularity.

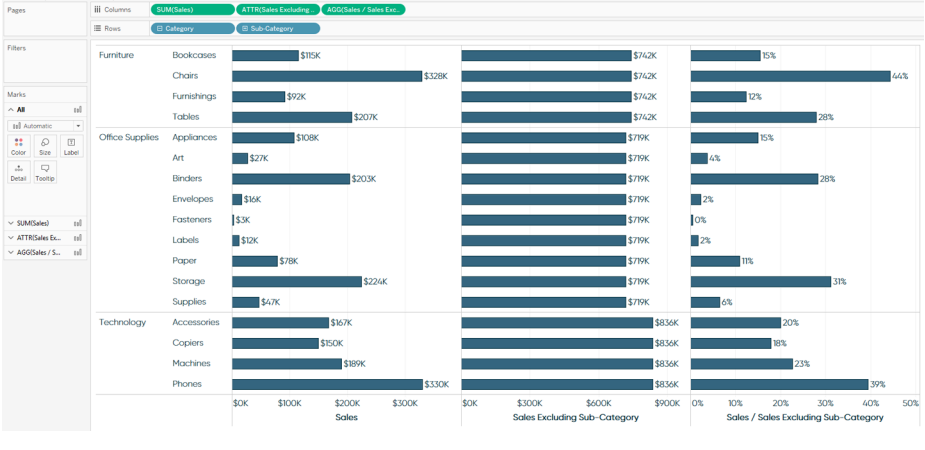
Take another quick look back at the last bar chart. I've made the analysis as granular as I'd like, slicing and dicing the Sales measure by the Category and Sub-Category dimensions. If I wanted to compare each row by the Sales at the Category level only, so excluding the Sub-Category dimension, I would create a calculated field with this formula:

```
{EXCLUDE [Sub-Category]: SUM([Sales])}
```

Now when I add my new calculation to the Columns Shelf, the left side shows my original analysis, while the right side "backs out" the level of detail to the Category level, excluding the Sub-Category dimension:



We're now displaying two different levels of detail on one view! This new measure can now be used for further analysis. If I want to calculate the contribution of each sub-category to its respective category, I can create another calculated field that divides SUM(Sales) by the newly created SUM(Sales Excluding Sub-Category):



This third column could have been computed using a Tableau table calculation, as covered in Chapter 13, that took the percent of total per pane, but using level of detail expressions provides several benefits including:

- You can "show your math"; in the preceding example, we would not have been able to show the first and second column together without LOD expressions. You could have also used a different table calculation called WINDOW_SUM, but with LOD expressions.
- You no longer need to worry about setting and maintaining the direction and scope of a table calculation; set the level of detail once in the LOD calculation and don't worry about it again.
- You can definitely get creative with table calculations in Tableau, but LOD calculations are even more flexible, allowing you to specify the precise level of detail and aggregation for a measure.

We have barely scratched the surface on arguably the most powerful feature to ship with Tableau version 9.0, but this example should have provided a good foundation about what level of detail is and how it can be changed to benefit your analysis. You will have to experiment with your own use cases and LOD calculations, but in addition to the 15 uses of Tableau LOD expressions linked to at the beginning of this chapter, here are a couple more from *Practical Tableau*:

- Chapter 59, Using Level of Detail Expressions to Create Benchmarks
- Chapter 70, How to Compare Unequal Date Ranges on One Axis

CHAPTER 17

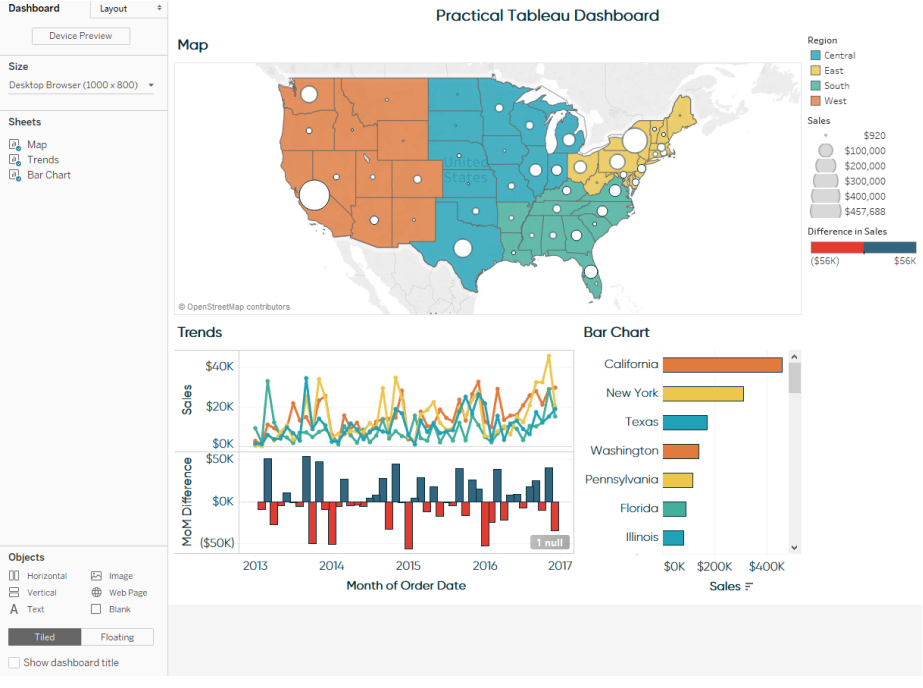
An Introduction to Dashboards and Distribution

Individual worksheets in Tableau can lead to powerful insights that help your business, but many times, it makes sense to combine the worksheets into a single dashboard. By combining varying visualizations into a dashboard, you and your audience are able to analyze different aspects of the data in context of each other. This is a much more intuitive experience than viewing the visualizations individually.

In addition to this one obvious benefit, Tableau comes with several technical features that allow you to control the user experience and even the ability to have the individual components of the dashboard interact with each other. This chapter offers an introduction to dashboards in Tableau and several ways to distribute the dashboard after it has been created.

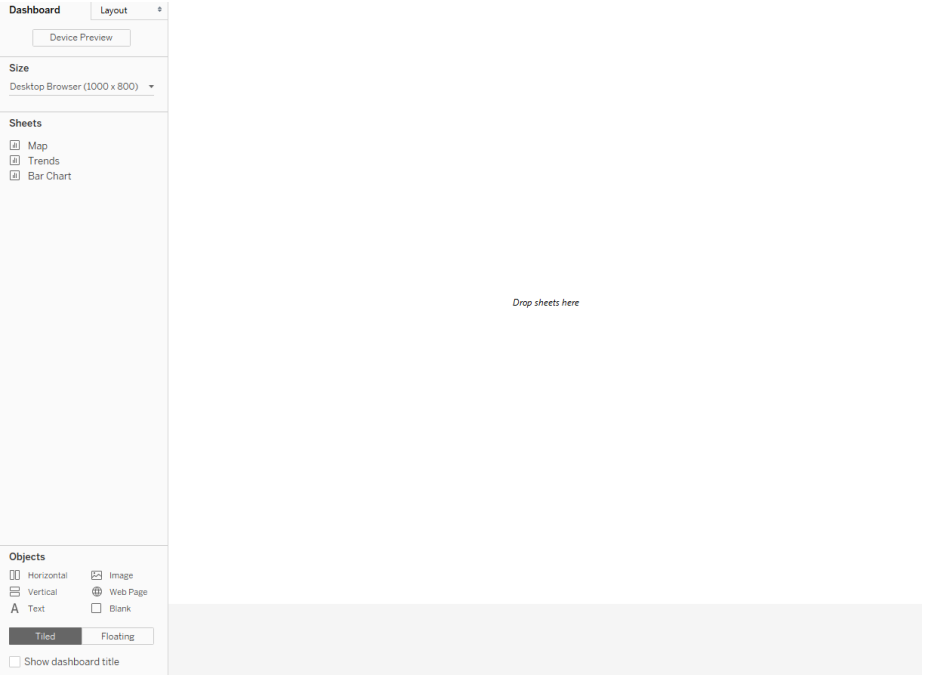
An Introduction to Dashboards in Tableau

For this introduction, we will re-create this dashboard in Tableau:



To create a new dashboard in Tableau, either click Dashboard in the top navigation and then New Dashboard or click the New Dashboard icon at the bottom of any worksheet. The New Dashboard icon is the second icon immediately following the existing worksheets in the workbook.

Upon creating a new dashboard, you will be provided a blank slate that looks like this:



Here's a quick overview of all the different dashboard options in the left navigation:

Dashboard and Layout tabs
: By default, you will be working on the Dashboard tab, which allows you to set most aspects of the dashboard. The Layout tab allows you to set the dimensions and location of individual dashboard components. All sizes on the Layout tab are in pixels.

Device Preview button
: The Device Preview button allows you see what the dashboard will look like on different devices and you can even save different versions of the dashboard so that it looks different depending on what device it is displayed on. For more, see Chapter 60.

Size
: This is where you can set the height and width of the dashboard in pixels. There are several preset size options or you can set an exact size of your choosing. If you choose the Automatic option, the dashboard will change to fill all available space on the screen it is being displayed on and resize the individual components of the dashboard accordingly. While this option sounds good on the surface, beware that it is not truly "responsive," and the display can be somewhat unpredictable.

Sheets
: These are the individual worksheets in your workbook that can be added to the dashboard. Note that to create the example pictured, there are three sheets in the workbook: Map, Trends, and Bar Chart. It helps to give the worksheets good names so you can easily find them, but you can also get a thumbnail preview of the worksheet by hovering over the name in the left navigation.

Objects

Horizontal
: Adds a horizontal layout container that additional objects can be added to.

Vertical
: Adds a vertical layout container that additional objects can be added to.

Text
: Opens a mini word processor where you can add and format any text you wish.

Image
: Adds an image from your computer to the dashboard.

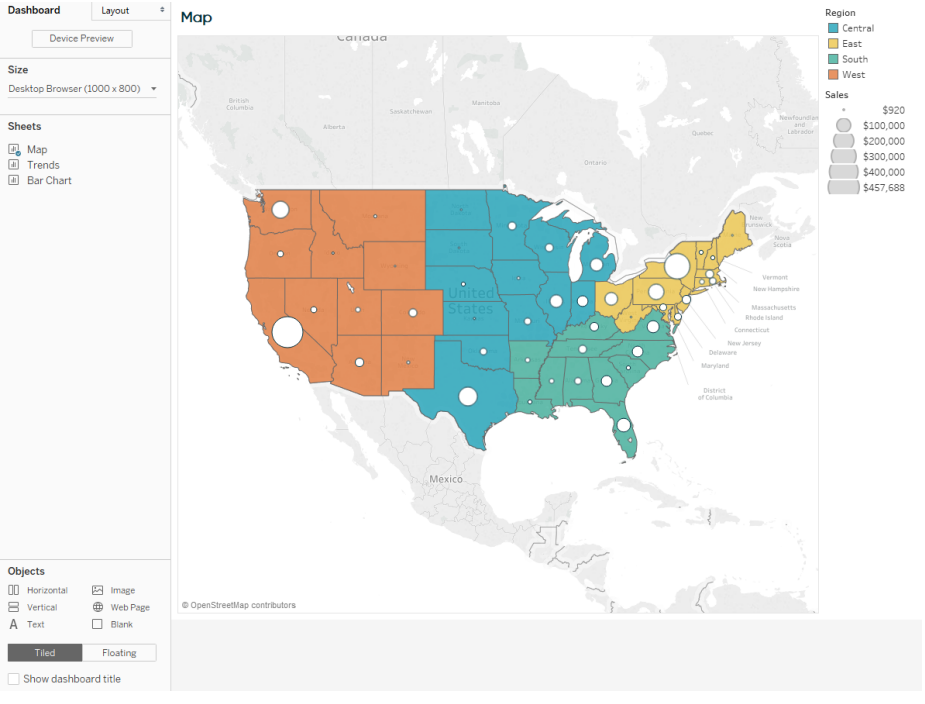
Web Page
: Embeds a web page in the dashboard (requires an internet connection to display the web page).

Blank
: Adds blank space to the dashboard, which can be helpful when dashboard elements are too close to each other and in a tiled layout.

Tiled or Floating
: When dashboard elements are tiled, they fill all available space in their respective tiles. When dashboard elements are floating, you control their exact size and location on the dashboard. Each of these layouts comes with their own pros and cons and the choice is largely dependent on your specific use case. Most Tableau users prefer the automatic resizing that comes with a tiled layout; I personally prefer the predictability and precision that comes with floating elements.

  The dashboard pictured is tiled in the Desktop Browser size, which is 1000 pixels wide by 800 pixels tall. To create the dashboard, I first ensured the size was 1000 by 800 by clicking Size in the left navigation, choosing Fixed Size, then choosing the preset Desktop Browser option.

  Tiled is the default layout option, so I am ready to go ahead and add individual sheets to my view. I will add the map first, and when there is nothing else on the dashboard, it will fill all the available space on the view:



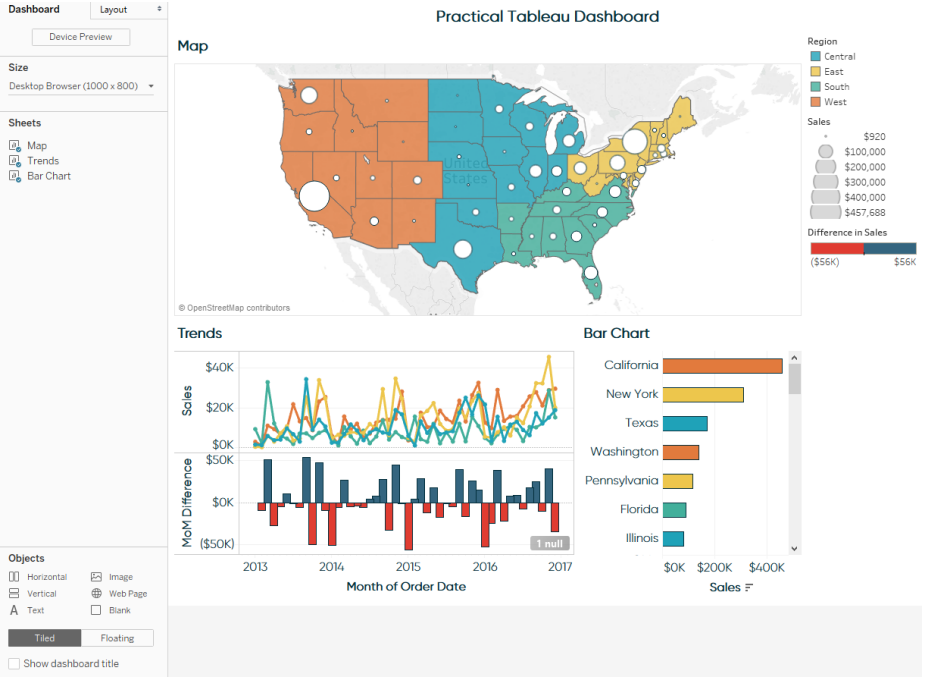
To add a worksheet to a dashboard, left-click and drag the worksheet name from the left navigation to the dashboard. When you drag new objects onto the view, you will see gray shading that represents where the object will be placed. You can always move an object after it has been placed by hovering over it, and dragging and dropping the gray icon that appears at the top-middle of the object.

Note that when I added a worksheet that has legends, the legends were added for me automatically. The color legend can be used as a highlighter by simply clicking a color in the legend. To learn how to create the map pictured, see Chapter 32.

I will then add the Trends worksheet in the space below the map by dragging it toward the bottom of the dashboard. To learn how to make a line graph, see Chapter 9. To learn how to calculate the month-over-month difference pictured, see Chapter 13.

I will then add the Bar Chart worksheet by dragging it near the lower-right corner of the dashboard. To learn how to make a bar chart in Tableau, see Chapter 8.

Lastly, I added a title to the top by dragging a text object to the top of the dashboard, and ended up with this view:



This is a more than serviceable dashboard that we were able to make in a matter of minutes in Tableau. You can add even more value by building in some user interactivity. Perhaps you want the trend lines and bar chart to update when a region is clicked on the map. To learn how to do this, see Chapter 56. Another option would be to add filters that allow the end user to filter the three worksheets. For more on filters, see Chapter 11.

Distributing Tableau Dashboards

After you've created a dashboard in Tableau, there are several ways the dashboard can be distributed.

Packaged Workbooks

If you have created a dashboard using Tableau Desktop, you can package the workbook for offline distribution. To package a workbook, navigate to File in the top navigation and click Export Packaged Workbook. This will package the data with the instructions for how to visualize the data. Anybody with Tableau Desktop or Tableau Reader can open the file and interact with the visualizations you have created. Note that packaged workbooks do not automatically update so the data within the workbook will only be as recent as the last update.

Tableau Public

Any dashboard built in Tableau Desktop or Tableau Public can be published to the web for public consumption. This is not a realistic option for sensitive business data, but if you are able to make your data public, this is the perfect solution for distributing your dashboard to the widest audience possible. To publish a dashboard from Tableau Desktop to Tableau Public, navigate to Server, hover over Tableau Public, and choose Save to Tableau Public As.

Tableau Server/Tableau Online

You can also distribute your workbooks privately on-premise or in the cloud by using Tableau Server or Tableau Online, respectively. Tableau Server requires incremental licenses for you and your end users, but is the most scalable Tableau solution for distributing your business-related workbooks. Publishing to Tableau Server is very similar to publishing to Tableau Public, but you will be presented with several additional options, including where to publish the workbook within Tableau Server, who has permission to view and interact with the workbook, and if/when you want the data in the workbook to update. To publish a workbook to Tableau Server, navigate to Server in the top navigation, and choose Publish Workbook. Tableau Online works similarly, but it is hosted by Tableau in the cloud.

PART II
Chart Types

CHAPTER 18

A Spreadsheet Is Not a Data Visualization

Like the movie *Groundhog Day* (but without its Certified Fresh 96% score on Rotten Tomatoes) there is one conversation I find myself having over and over again in my career as a data visualization consultant. It typically goes something like this:

Excel fan: "Thanks for this—looks great. Would it be possible to take your world-class data visualization and turn it into an unreadable wall of numbers?"

(OK, the "world-class" part is an exaggeration—no one has ever said that.)

Ryan (me): "But Excel fan, look how much easier it is to gain valuable business insights more accurately and efficiently when the numbers are visualized! Isn't that really cool?"

Excel fan: "Steven, my stakeholders are more comfortable seeing the exact numbers."

Ryan (me): "It's Ryan, actually. Let me just show…"

Excel fan: "Brian, please provide a text table of this information. Also, if there's room along the bottom, please add at least one pie chart."

At least this is how I sometimes feel the conversation goes when I'm having these discussions. I admit it can be discouraging, but mostly because I feel I have let my stakeholders down by my own failure to educate them on a better way of looking at data. It can also be encouraging, because it reminds me there is still a long way to go before best practice techniques are widely adopted and that I have a long career in data visualization ahead of me. (That means I have quite a bit of time to earn money that I can use to purchase barbecue.)

Scott Klein of ProPublica gave an inspiring talk on the history of data journalism at the 2016 Tapestry Conference, sharing several fascinating examples of early data visualization. What I found most interesting was that William Playfair, inventor of the bar

chart and line graph, was not celebrated in his time. In fact, most people did not understand the value in visualizing quantitative information and were critical of not using tables.

In other words, data visualization practitioners have been fighting this same fight since, quite literally, *the beginning of data visualization time*.

A Spreadsheet Is Not a Data Visualization

Spreadsheets are extremely valuable in business. They can be used in accounting, to store information, or to do calculations, just to name a few purposes.

However, a spreadsheet or text table view of data is not a data visualization. In my opinion, putting one or more of these text tables in one place does not make for a dashboard. To me, dashboards and data visualizations help users interpret data by leveraging the power of visualization. Further, once there is more than just *one* number in a report, I firmly believe that a text table is the worst possible choice for analyzing the data.

Consider this example of just two numbers: 23.4K versus 3.42M. Even when the numbers are in different units, you can likely tell that the first number is smaller than the second number, but it is challenging to consider the scale of the difference. Take a look at the same two numbers visualized as a bar chart:

The values can still be included as labels on the bars if the end users need those in order to sleep at night, but now they have a much better insight into what is happening with the numbers. Not only is the bar chart easier to interpret than the raw numbers, it is more memorable, and thus, effective.

I am not saying that raw numbers provide no value, but I am saying that visualizing raw numbers always provides additional value. If this is clear even when comparing two numbers, think about how much more value a visualization adds when we're looking at a tableful of numeric information.

Are Text Tables or Crosstabs Ever the Best Choice?

While my immediate reaction to tables in data visualization is *"Don't!"* we should nevertheless consider the times when it is appropriate to use crosstab views in Tableau:

Crosstabs can be used to export raw data
: Sometimes there are legitimate reasons to get to the raw data. If end users are more comfortable using a tool like Excel and want to add their own calculations, for example, it would be nice to provide the data in an Excel-ready format. Fortunately, any view in Tableau can easily be duplicated as a crosstab. The crosstab view can then be opened in Excel. Here are the steps:

  - From any view in Tableau, hover over Worksheet in the top navigation and select Duplicate as Crosstab.
  - Once the crosstab view is generated, hover over Worksheet again, hover over Export, and choose Crosstab to Excel.

Crosstabs can be used to create "Callout Numbers"
: In Chapter 95, I encourage data designers to use "callout numbers" to clearly communicate the most important points of the story. These numbers are standalone, which makes them descriptive in nature (typically with no prescriptive value), but they can help engage the viewers so they can immediately see what KPIs are prioritized by the designer. From here, hopefully the end users will continue exploring your dashboard so they can help answer why a callout number is the value that it is.

  Callout numbers are created by making a crosstab view, filtering it down to one number, and formatting the number so that is oversized (think a minimum of 36-point font). These numbers are then typically placed along the top or left side of a dashboard to provide a natural starting point of an analysis.

Crosstabs can be used to create dashboard filters or navigation
: Another clever application of text table views in Tableau is to use the view as a dashboard filter or navigation. This approach has an added advantage of making your data process more efficiently.

  For example, if you wanted to create a dashboard filter for Region out of a crosstab, you would first make a crosstab view that displayed the four regions: North,

East, South, and West. You would then place this view on a dashboard and add a dashboard action that filters the rest of the dashboard when a Region selection is made.

Not only does this type of dashboard action process more quickly than a filter in Tableau, but it can be used for other purposes. Perhaps you want to add a dashboard action to the table view that takes end users to interior pages of your workbook or even conducts an online search.

For more dashboard action ideas, see Chapter 56.

CHAPTER 19

How to Make a Highlight Table

In Chapter 87, I demonstrate an exercise for "Smoothing the Excel Transition" by introducing the value of data visualization. In the example, we simply encode the values in a large text table of data by color—the higher the sales values, the darker the green. The result is called a highlight table.

Highlight tables are one of the simplest chart types to create, but are also among the most powerful. When compared to a crosstab (a.k.a. text table) view, this basic data visualization helps reduce the time to insight and improve the accuracy of insights. Highlight tables are definitely worth making a part of your chart type toolbelt, right alongside bar charts and line graphs. They are best used for:

- Quickly identifying highs and lows or other points of interest in your data
- As a means of enhancing a crosstab
- As a tactic for helping smooth the Excel transition

Highlight tables consist of one or more dimensions and exactly one measure (the color). Let's take a look at how they're made.

How to Make a Highlight Table in Tableau

Before we begin, let's see how the final product will look:

	January	February	March	April	May	June	July	August	September	October	November	December
Accessories	$5,478	$5,369	$8,735	$7,984	$9,613	$8,858	$17,177	$12,376	$24,900	$12,927	$25,957	$28,007
Appliances	$3,176	$4,933	$6,734	$6,042	$7,526	$7,479	$3,540	$13,345	$10,193	$9,152	$18,970	$16,443
Art	$966	$1,006	$1,413	$2,407	$2,231	$2,182	$2,102	$1,690	$3,660	$1,911	$3,966	$3,584
Binders	$12,412	$4,237	$13,889	$13,357	$9,160	$13,294	$8,557	$20,430	$37,342	$18,079	$20,858	$31,798
Bookcases	$5,352	$1,650	$7,352	$4,720	$6,290	$7,881	$9,856	$5,622	$22,849	$8,771	$23,561	$10,977
Chairs	$11,285	$7,583	$21,016	$18,855	$25,703	$21,714	$23,016	$18,340	$51,577	$21,905	$49,636	$57,819
Copiers	$3,960		$25,590	$3,880	$18,400	$900	$9,780	$5,730	$10,320	$37,020	$15,150	$18,800
Envelopes	$750	$669	$1,657	$945	$1,096	$514	$1,200	$701	$2,177	$1,403	$2,907	$2,458
Fasteners	$88	$159	$150	$258	$109	$116	$182	$243	$406	$326	$550	$438
Furnishings	$3,980	$2,316	$5,159	$7,404	$6,996	$5,956	$7,336	$4,307	$11,805	$5,447	$16,783	$14,218
Labels	$207	$300	$940	$430	$863	$1,207	$1,692	$876	$1,496	$1,248	$1,850	$1,376
Machines	$7,215	$8,990	$35,052	$18,190	$11,268	$12,183	$4,065	$6,262	$26,386	$10,613	$33,807	$15,210
Paper	$2,287	$2,813	$6,286	$3,964	$6,184	$6,751	$4,180	$6,523	$10,690	$4,965	$12,563	$11,274
Phones	$13,772	$8,984	$28,163	$17,890	$24,182	$26,493	$23,128	$28,413	$37,775	$26,472	$56,185	$38,551
Storage	$9,374	$6,125	$14,793	$15,806	$14,670	$18,606	$12,491	$17,719	$29,511	$15,822	$37,945	$30,983
Supplies	$4,403	$289	$10,637	$6,216	$1,154	$1,267	$8,816	$866	$6,436	$816	$1,372	$4,402
Tables	$10,952	$4,218	$16,913	$9,913	$9,288	$16,405	$9,299	$17,752	$19,626	$20,223	$33,182	$39,193

You can create a highlight table with any combination of one or more dimensions and one measure. You can also create a highlight table using Show Me, but I prefer to learn by creating chart types manually so I understand how Tableau is generating the data visualization:

1. Make a crosstab of data.

   First, make a basic text table. In my example, I'm looking at the measure of Sales by the dimensions of Sub-Category and Month of Order Date (Discrete). At this point, my view looks like this:

	January	February	March	April	May	June	July	August	September	October	November	December
Accessories	$5,478	$5,369	$8,735	$7,984	$9,613	$8,858	$17,177	$12,376	$24,900	$12,927	$25,957	$28,007
Appliances	$3,176	$4,933	$6,734	$6,042	$7,526	$7,479	$3,540	$13,345	$10,193	$9,152	$18,970	$16,443
Art	$966	$1,006	$1,413	$2,407	$2,231	$2,182	$2,102	$1,690	$3,660	$1,911	$3,966	$3,584
Binders	$12,412	$4,237	$13,889	$13,357	$9,160	$13,294	$8,557	$20,430	$37,342	$18,079	$20,858	$31,798
Bookcases	$5,352	$1,650	$7,352	$4,720	$6,290	$7,881	$9,856	$5,622	$22,849	$8,771	$23,561	$10,977
Chairs	$11,285	$7,583	$21,016	$18,855	$25,703	$21,714	$23,016	$18,340	$51,577	$21,905	$49,636	$57,819
Copiers	$3,960		$25,590	$3,880	$18,400	$900	$9,780	$5,730	$10,320	$37,020	$15,150	$18,800
Envelopes	$750	$669	$1,657	$945	$1,096	$514	$1,200	$701	$2,177	$1,403	$2,907	$2,458
Fasteners	$88	$159	$150	$258	$109	$116	$182	$243	$406	$326	$550	$438
Furnishings	$3,980	$2,316	$5,159	$7,404	$6,996	$5,956	$7,336	$4,307	$11,805	$5,447	$16,783	$14,218
Labels	$207	$300	$940	$430	$863	$1,207	$1,692	$876	$1,496	$1,248	$1,850	$1,376
Machines	$7,215	$8,990	$35,052	$18,190	$11,268	$12,183	$4,065	$6,262	$26,386	$10,613	$33,807	$15,210
Paper	$2,287	$2,813	$6,286	$3,964	$6,184	$6,751	$4,180	$6,523	$10,690	$4,965	$12,563	$11,274
Phones	$13,772	$8,984	$28,163	$17,890	$24,182	$26,493	$23,128	$28,413	$37,775	$26,472	$56,185	$38,551
Storage	$9,374	$6,125	$14,793	$15,806	$14,670	$18,606	$12,491	$17,719	$29,511	$15,822	$37,945	$30,983
Supplies	$4,403	$289	$10,637	$6,216	$1,154	$1,267	$8,816	$866	$6,436	$816	$1,372	$4,402
Tables	$10,952	$4,218	$16,913	$9,913	$9,288	$16,405	$9,299	$17,752	$19,626	$20,223	$33,182	$39,193

2. Color the numbers by sales.

Second, color the displayed sales values in the text table by dragging the Sales measure from the list of measures to the Color Marks Card. Note that the Sales measure is now on the Marks Shelf twice: once for text and once for color. At this point, my view looks like this:

	January	February	March	April	May	June	July	August	September	October	November	December
Accessories	$5,478	$5,369	$8,735	$7,984	$9,613	$8,858	$17,177	$12,376	$24,900	$12,927	$25,957	$28,007
Appliances	$3,176	$4,933	$6,734	$6,042	$7,526	$7,479	$3,540	$13,345	$10,193	$9,152	$18,970	$16,443
Art	$966	$1,006	$1,413	$2,407	$2,231	$2,182	$2,102	$1,690	$3,660	$1,911	$3,966	$3,584
Binders	$12,412	$4,237	$13,889	$13,357	$9,160	$13,294	$8,557	$20,430	$37,342	$18,079	$20,858	$31,798
Bookcases	$5,352	$1,650	$7,352	$4,720	$6,290	$7,881	$9,856	$5,622	$22,849	$8,771	$23,561	$10,977
Chairs	$11,285	$7,583	$21,016	$18,855	$25,703	$21,714	$23,016	$18,340	$51,577	$21,905	$49,636	$57,819
Copiers	$3,960		$25,590	$3,880	$18,400	$900	$9,780	$5,730	$10,320	$37,020	$15,150	$18,800
Envelopes	$750	$669	$1,657	$945	$1,096	$514	$1,200	$701	$2,177	$1,403	$2,907	$2,458
Fasteners	$88	$159	$150	$258	$109	$116	$182	$243	$406	$326	$550	$438
Furnishings	$3,980	$2,316	$5,159	$7,404	$6,996	$5,956	$7,336	$4,307	$11,805	$5,447	$16,783	$14,218
Labels	$207	$300	$940	$430	$863	$1,207	$1,692	$876	$1,496	$1,248	$1,850	$1,376
Machines	$7,215	$8,990	$35,052	$18,190	$11,268	$12,183	$4,065	$6,262	$26,386	$10,613	$33,807	$15,210
Paper	$2,287	$2,813	$6,286	$3,964	$6,184	$6,751	$4,180	$6,523	$10,690	$4,965	$12,563	$11,274
Phones	$13,772	$8,984	$28,163	$17,890	$24,182	$26,493	$23,128	$28,413	$37,775	$26,472	$56,185	$38,551
Storage	$9,374	$6,125	$14,793	$15,806	$14,670	$18,606	$12,491	$17,719	$29,511	$15,822	$37,945	$30,983
Supplies	$4,403	$289	$10,637	$6,216	$1,154	$1,267	$8,816	$866	$6,436	$816	$1,372	$4,402
Tables	$10,952	$4,218	$16,913	$9,913	$9,288	$16,405	$9,299	$17,752	$19,626	$20,223	$33,182	$39,193

3. Create a highlight table by changing the mark type.

While more helpful than a raw table of numbers, the view we have created so far can still be improved. While the numbers themselves are encoded by color, which helps us recognize highs and lows in the data more quickly than we could with a crosstab, there is a lot of whitespace around each number.

The final step in converting this view to a highlight table is to simply change the mark type from Automatic (which at this point is Text) to Square. When you convert the mark type from Text to Square in a crosstab view, the "Square" mark type fills each cell, creating a highlight table. When building highlight tables, I also like to add a white border around each cell; this can be accomplished by clicking the Color Marks Card and changing the border.

My final highlight table looks like this:

	January	February	March	April	May	June	July	August	September	October	November	December
Accessories	$5,478	$5,369	$8,735	$7,984	$9,613	$8,858	$17,177	$12,376	$24,900	$12,927	$25,957	$28,007
Appliances	$3,176	$4,933	$6,734	$6,042	$7,526	$7,479	$3,540	$13,345	$10,193	$9,152	$18,970	$16,443
Art	$966	$1,006	$1,413	$2,407	$2,231	$2,182	$2,102	$1,690	$3,660	$1,911	$3,966	$3,584
Binders	$12,412	$4,237	$13,889	$13,357	$9,160	$13,294	$8,557	$20,430	$37,342	$18,079	$20,858	$31,798
Bookcases	$5,352	$1,650	$7,352	$4,720	$6,290	$7,881	$9,856	$5,622	$22,849	$8,771	$23,561	$10,977
Chairs	$11,285	$7,583	$21,016	$18,855	$25,703	$21,714	$23,016	$18,340	$51,577	$21,905	$49,636	$57,819
Copiers	$3,960		$25,590	$3,880	$18,400	$900	$9,780	$5,730	$10,320	$37,020	$15,150	$18,800
Envelopes	$750	$669	$1,657	$945	$1,096	$514	$1,200	$701	$2,177	$1,403	$2,907	$2,458
Fasteners	$88	$159	$150	$258	$109	$116	$182	$243	$406	$326	$550	$438
Furnishings	$3,980	$2,316	$5,159	$7,404	$6,996	$5,956	$7,336	$4,307	$11,805	$5,447	$16,783	$14,218
Labels	$207	$300	$940	$430	$863	$1,207	$1,692	$876	$1,496	$1,248	$1,850	$1,376
Machines	$7,215	$8,990	$35,052	$18,190	$11,268	$12,183	$4,065	$6,262	$26,386	$10,613	$33,807	$15,210
Paper	$2,287	$2,813	$6,286	$3,964	$6,184	$6,751	$4,180	$6,523	$10,690	$4,965	$12,563	$11,274
Phones	$13,772	$8,984	$28,163	$17,890	$24,182	$26,493	$23,128	$28,413	$37,775	$26,472	$56,185	$38,551
Storage	$9,374	$6,125	$14,793	$15,806	$14,670	$18,606	$12,491	$17,719	$29,511	$15,822	$37,945	$30,983
Supplies	$4,403	$289	$10,637	$6,216	$1,154	$1,267	$8,816	$866	$6,436	$816	$1,372	$4,402
Tables	$10,952	$4,218	$16,913	$9,913	$9,288	$16,405	$9,299	$17,752	$19,626	$20,223	$33,182	$39,193

CHAPTER 20

How to Make a Heat Map

Heat maps are a visualization where marks on a chart are represented as colors. As the marks "heat up" due their higher values or density of records, a more intense color is displayed. These colors can be displayed in a matrix/crosstab, which creates a highlight table, but can also be displayed on a geographical map or even a customized image—such as a web page used to show where users are clicking.

Density plots are coming in Tableau version 11, but in the meantime, heat maps are defined somewhat differently in Tableau. This chapter shares how to create a Tableau heat map. If you are interested in creating a traditional heat map using a custom image, see Chapter 35.

To first get more specific about how Tableau defines heat map, let's take a look at the requirements to draw a heat map under Tableau's Show Me options:

"For heat maps try 1 or more dimensions and 1 or 2 measures"

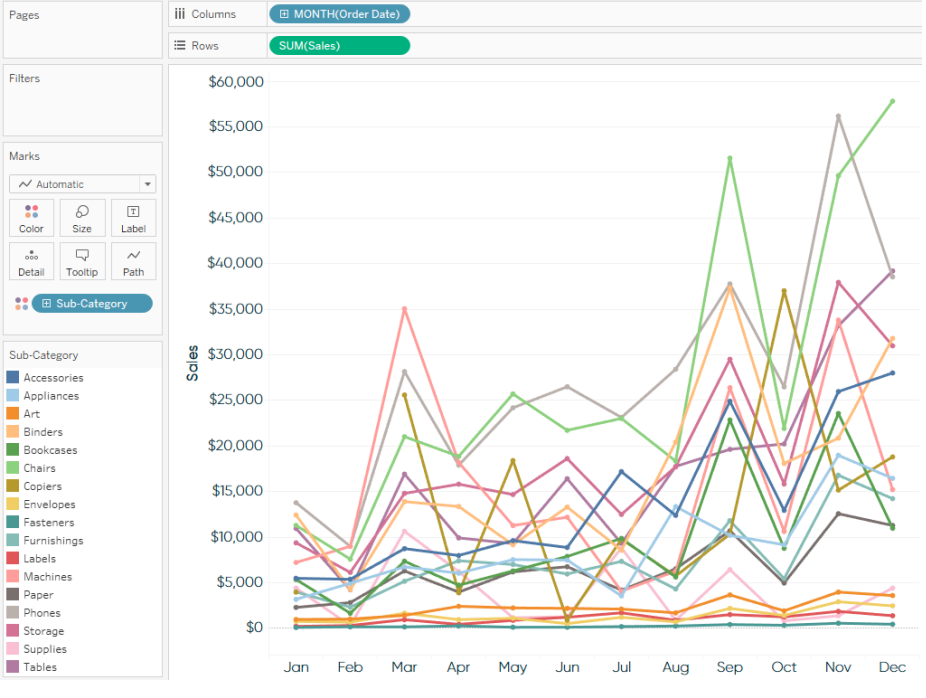
This is very close to the requirements for drawing a highlight table with Show Me:

"For highlight tables try 1 or more dimensions and 1 measure"

The key distinction between the two chart types is that with a heat map, you are able to encode the marks by one additional measure. With a highlight table, your only option is to color the marks by one measure. Since you can only color marks by one thing at a time, your encoding is limited to exactly one measure. With a heat map in Tableau, you can color the marks by one measure, but also size the marks by a second measure. Depending on your analysis, this additional encoding can add value to your visualization.

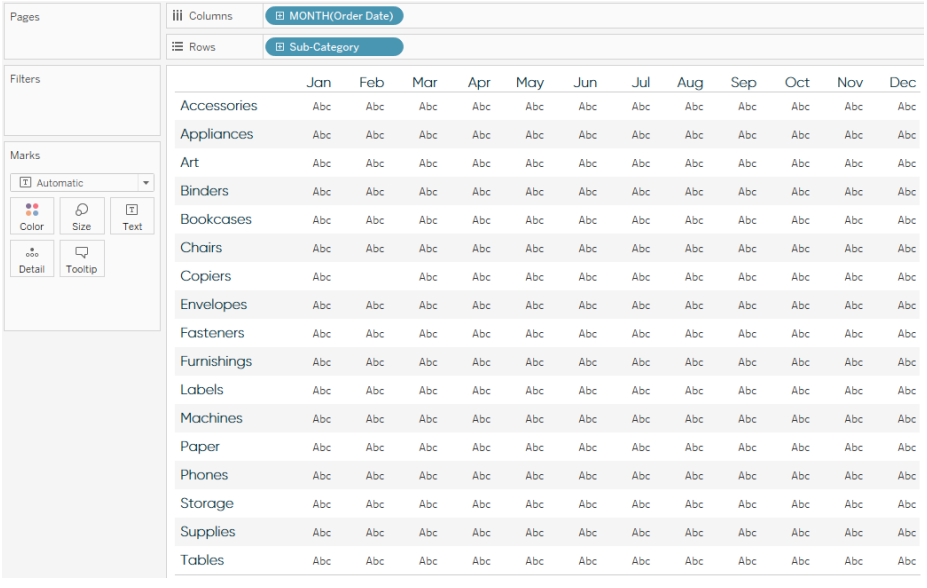
How to Make a Heat Map in Tableau

Let's say we've been tasked with evaluating the product sub-category sales in the Sample – Superstore dataset by Month of Order Date to see if we can identify any seasonal trends in the data. The element of time (Month of Order Date) may give you the instinct to go with a line graph for this analysis, which would look like this:

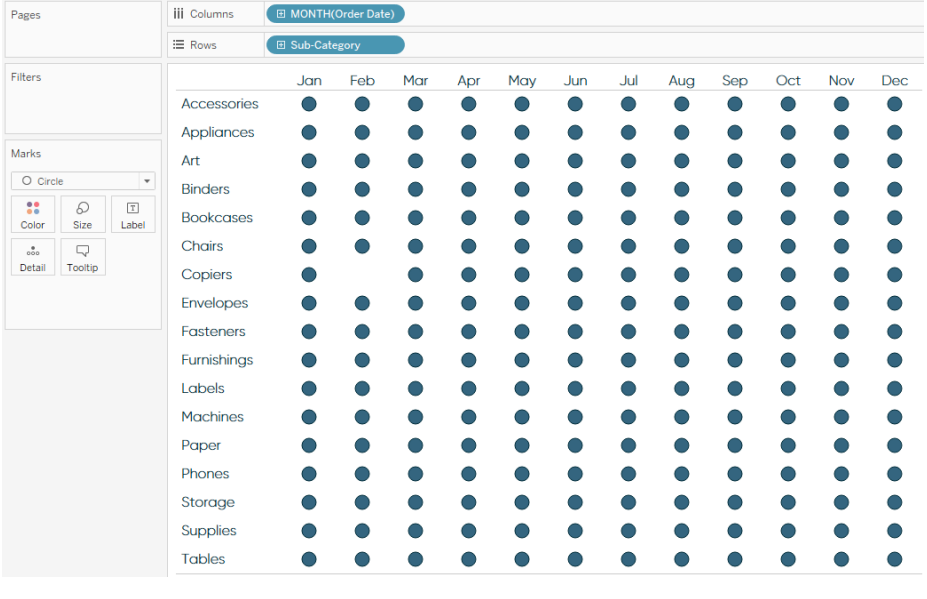


As you can see, this graph is a bit of a mess. The 17 lines are causing a lot of overlap and several of the sub-categories at the bottom are on a much smaller scale than the rest, making it challenging to gain insights. In this case, a heat map may be a better option.

To create a heat map in Tableau, start by laying out the rows and columns that will serve as the grid for the visualization. I would like the months in this analysis to be listed along the top of the view. Since the months will create columns, I know that I should put the Month of Order Date dimension on the Columns Shelf. Conversely, I would like each sub-category to have its own row, so I will place that dimension on the Rows Shelf:



By default, the mark type is set to Text. I personally prefer my Tableau heat maps to be created with circles, so I will change the mark type to Circle to lay the foundation for the view. The Shape or Square mark types are also good choices:



Now that I have a mark at each intersection of Sub-Category and Month of Order Date, I can encode them by two measures; one that will determine the size of the marks and one that will determine the color intensity of the marks. This encoding is produced by placing the measures we want to encode the marks by onto the Size Marks Card and Color Marks Card, respectively. For my first analysis, I will size and color the circles by the same measure—Sales:

When the Sales measure—which always has a positive value—is used for color, we get a sequential color palette by default. If you were to place a measure such as Profit—which has both negative and positive values—on the Color Marks Card, you would see a diverging color palette by default with 0 at the center (i.e., one color for negative values; a different color for positive values).

This visualization uses the exact same fields as the line graph from before, but it is now much easier to compare sub-categories within a specific month (reading the chart vertically) or compare the seasonality across each sub-category (reading the chart horizontally). The "double-encoding," where the size and color are both based on the same field, are meant to help the insights "pop." However, you have the option to use one measure for the size, and a different measure for the color. For example, here is what the heat map looks like if I size the circles by the Quantity measure instead of Sales:

With this analysis, large and light circles would mean that a sub-category sold a relatively large quantity but made a relatively low amount of revenue: low sales per item. Conversely, small and dark circles would mean the sub-category sold a relatively small quantity, but generated a relatively high revenue: high sales per item.

Beware that this type of mixed encoding can be confusing for end users unless you explicitly state what the size and color represents. Despite some of their limitations, Tableau heat maps provide a viable alternative to a line graph or highlight table if you need to compare dimension members with varying scales across multiple measures.

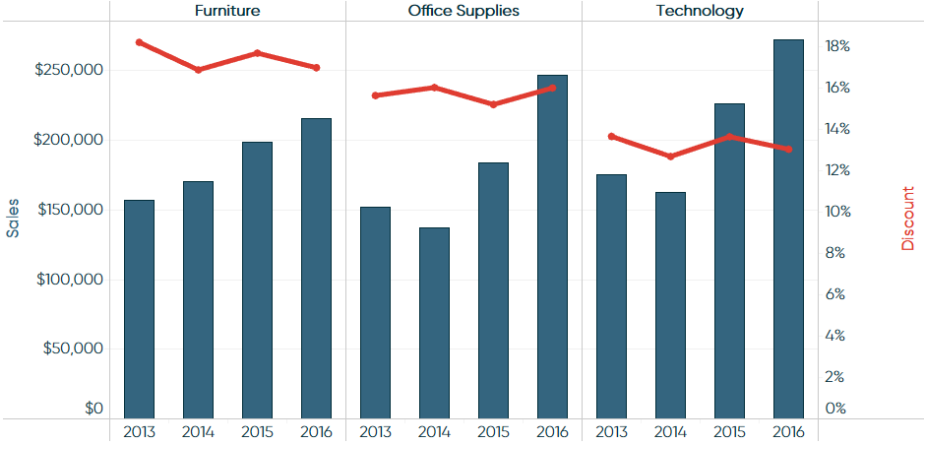
CHAPTER 21

How to Make a Dual-Axis Combination Chart

Dual-axis combination charts, or Combo Charts, are an effective chart type for showing related information while saving real estate by combining views. This chart type is created with one shared axis, such as an x-axis for date, and two separate axes, such as y-axes for two different measures. This chapter will show you how to make a dual-axis combo chart in Tableau that looks at Sales and Discount by Year and Product Category.

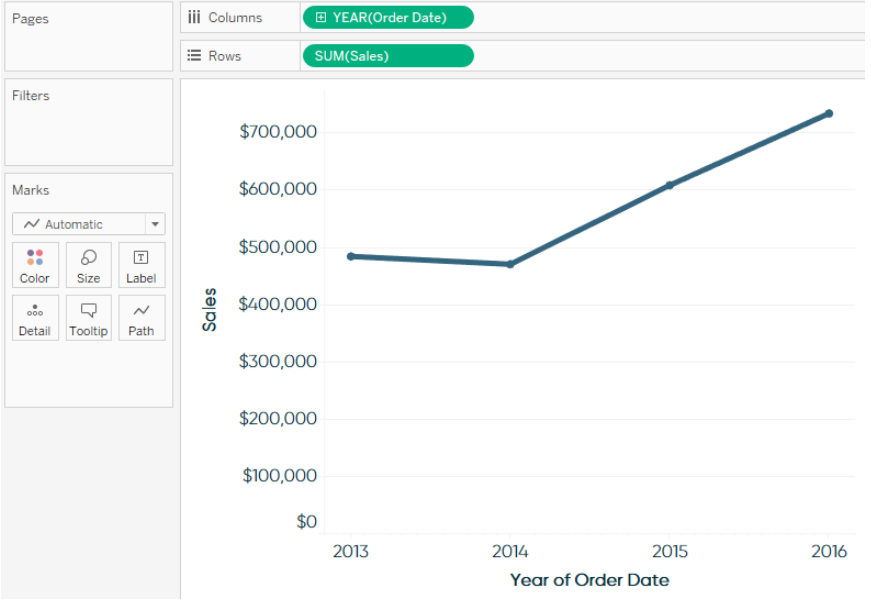
How to Make a Dual-Axis Combo Chart in Tableau

Before we begin, let's take a look at how our final product will look:



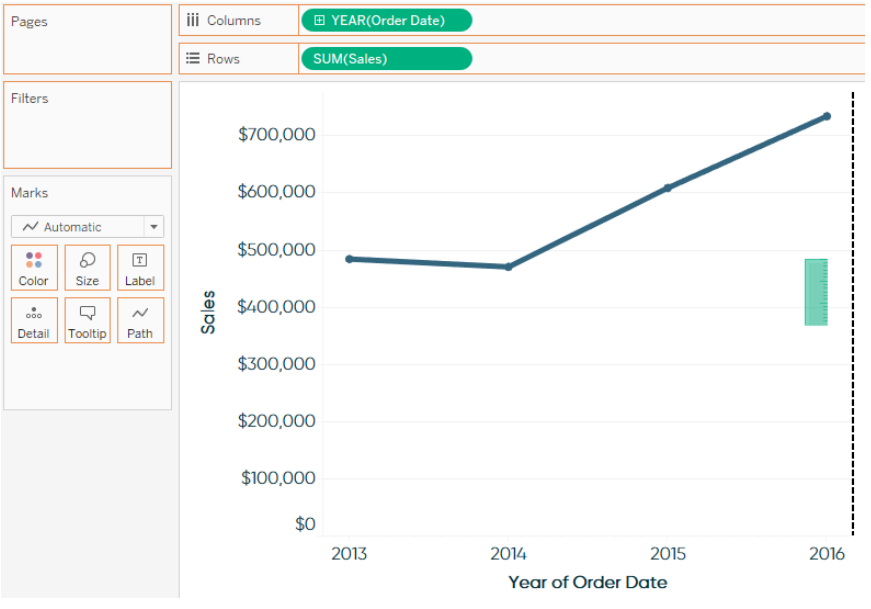
1. Make a graph of one of the measures.

   The first step is to make a graph for one of your measures. It doesn't matter too much which of your two measures you begin with, but the first measure you select will always form the lefthand y-axis or bottom x-axis (depending on the chart's orientation). Let's start by making a line graph for sales by year. To create the first graph, drag the Order Date field to Columns Shelf with an aggregation of Year Continuous. Then drag the Sales field to the Rows Shelf. At this point, the view looks like this:

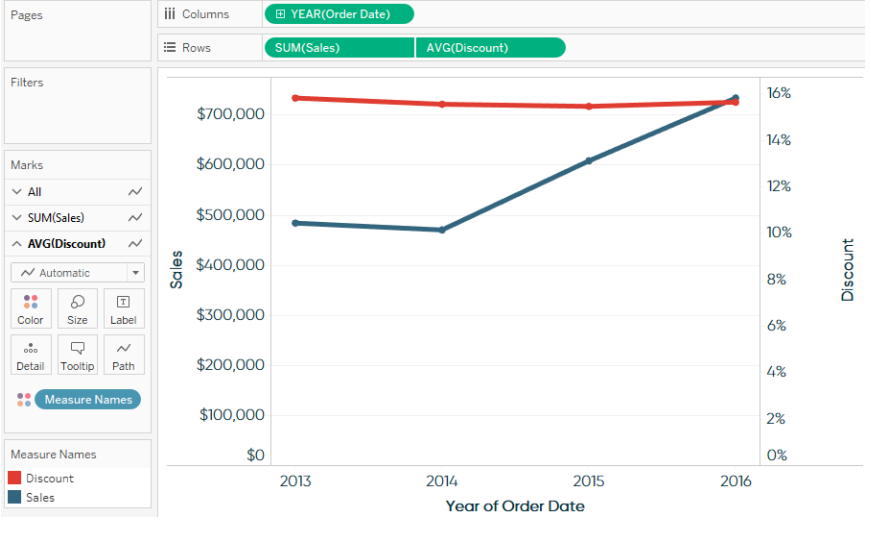


2. Drag the second measure onto the opposite axis.

   In order to create a dual-axis chart, you will drag the second measure onto the opposite axis from your first measure. In my case, Sales is on the left y-axis. I want to add the Discount measure to the view, so I will drag it to where the right y-axis will appear. As I get close to the opposite axis, Tableau is giving me a hint of where I can drop the measure as indicated by a dashed line:



Once I drop the field on the opposite axis, Tableau generates a dual-axis chart. Note that this is a dual-axis chart at this point, but not a dual-axis *combination* chart. The name combination chart comes from using a combination of mark types, and so far, we only have one mark type (line):



3. Create a dual-axis combination chart by changing one of the mark types.

When the second measure was dropped onto the view, not only did the field appear on the Rows Shelf, but two new Marks Shelves were generated. The first new Marks Shelf, "All," affects all of the marks on the view, and the second new Marks Shelf is for the marks on the right y-axis. The marks for each measure can now be edited independently of each other, allowing you to display a *combination* of mark types on your view.

Let's change the mark type of the Sales field to Bar and leave the mark type for Discount as Line. This is accomplished by clicking the Marks Shelf for SUM(Sales) and changing the mark type from Automatic to Bar. At this point, the view looks like this:

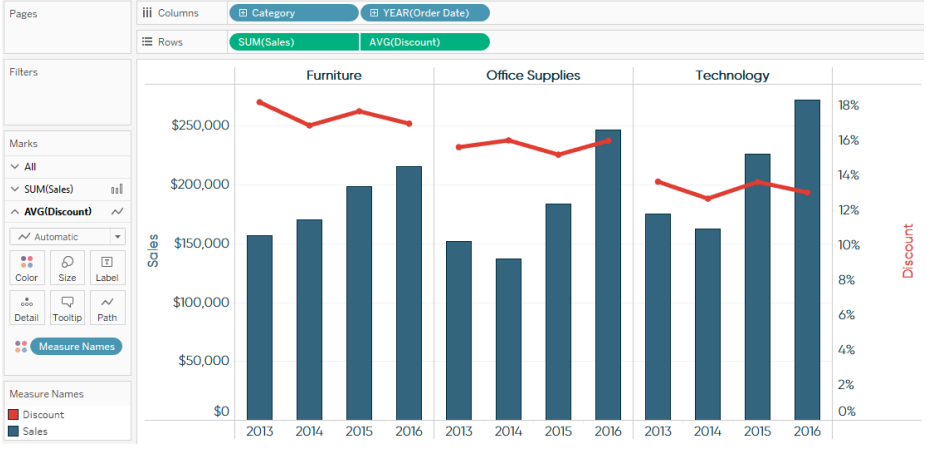
Some Additional Thoughts

- I decided to make the first graph, Sales by *continuous* Year, because it was meant to be a continuous (i.e., chronological) trend over time. Many people ask why the bars are so skinny (prior to version 10) and so fat (with version 10) when Year is continuous. You can get a better look by changing the Year field from continuous to discrete by right-clicking the field on the Columns Shelf and choosing Discrete. The catch with discrete date dimensions is that the dates can now be sorted out of chronological order. If you decide to make this change for formatting purposes, be careful not to sort the view in an unintended way.
- A mark type of Line should only be used when connecting dots between continuous dates. This is because a line implies that there is a chronological relationship

between points. This works well for our example, where discount is being connected by years in chronological order. However, dual-axis combination charts do not necessarily need to use a date as the shared axis, so choose your mark types wisely.

- You can create small multiples, or comparison views, by dragging another dimension onto the Rows Shelf or Columns Shelf.
- If your axes should be on the same scale, you can synchronize the axes by right-clicking either axis and choosing Synchronize Axis.

Here is my final view after changing Year to Discrete, adding the Category dimension to the Columns Shelf, and cleaning up the formatting:



CHAPTER 22

How to Make a Scatter Plot

After the bar chart and line graph, I find the scatter plot to be one of the most effective visualization options for analyzing data. A scatter plot displays data points at their respective intersections of two measures, and displays every data point on the same view. The marks can then be further encoded by up to three additional measures and/or dimensions by leveraging the Color and/or Size and/or Shape Marks Cards.

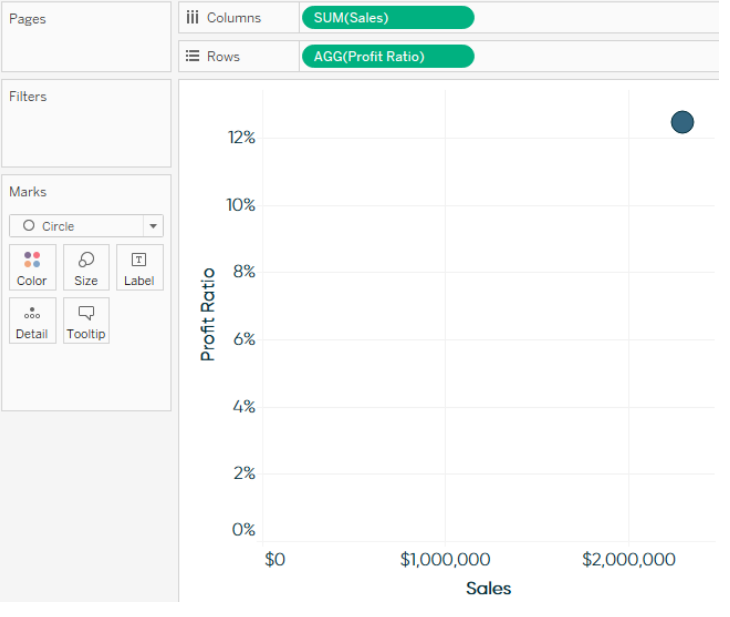
While scatter plots allow you to use several combinations of dimensions and measures, each new encoding increases the cognitive load on your end users. In other words, it makes it harder for them to process the view.

This ability to slice and dice data points in several ways within a condensed space provides an effective means for identifying patterns. Not only that, scatter plots provide a natural way to segment the marks into four quadrants by simply adding a reference line to each of the two axes. You can even take this a step further in Tableau by creating sets for each of the four segments to use for deeper analysis later. This chapter shares how to create a scatter plot in Tableau and use the results to create segments.

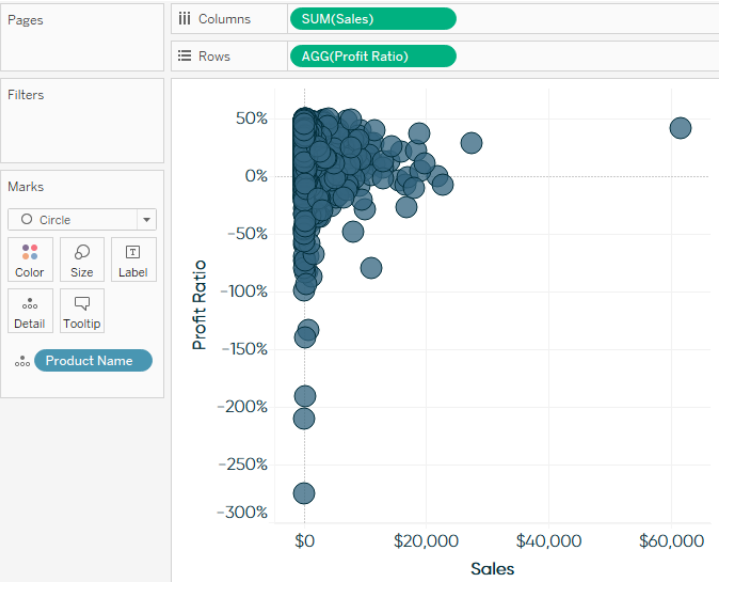
How to Make a Scatter Plot in Tableau

For this walkthrough, we'll be evaluating all of our products across the Sales and Profit Ratio measures. When you build a scatter plot, one measure will form the y-axis and one measure will form the x-axis. The marks on the view will then be plotted at the intersection of the values on the two axes. It is typically best to put the most dependent metric on the y-axis and the explanatory metric on the x-axis.

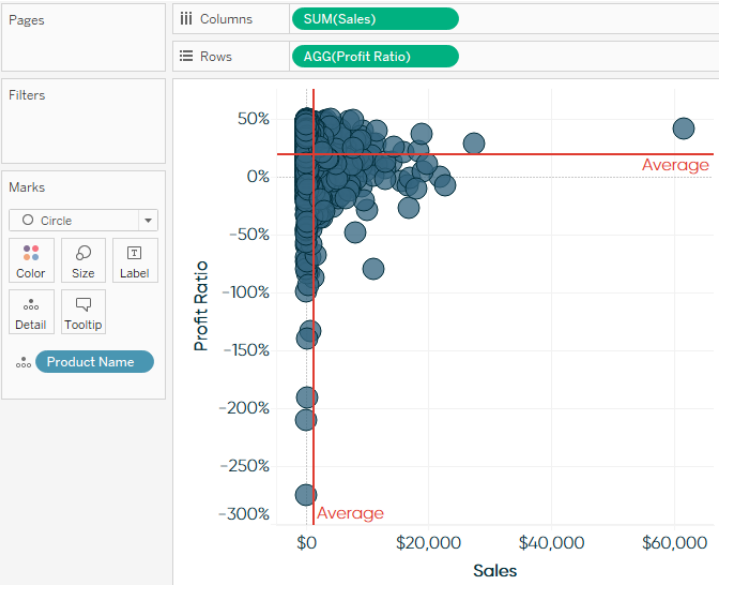
To create a scatter plot, drag and drop the Profit Ratio measure to the Rows Shelf and the Sales measure to the Columns Shelf. Scatter plot is the default chart type in Tableau when two measures are used, so you could have got to this same point by just double-clicking Profit Ratio, then double-clicking Sales to add them to the view. At this point, your view should look similar to this:



So far, we have just one point that represents the intersection of Profit Ratio and Sales for all of the records in our dataset. This is because we have yet to specify a level of detail for our analysis. For more on this topic, you can review Chapter 10. Our analysis is going to look at products, so change the level of detail by dragging the Product Name dimension to the Detail Marks Card. The view has now been changed to this:



What's powerful here is that we are looking at all 1,850 of our products at once, which helps us evaluate them quickly in context of each other. You can see outliers, unprofitable products, and segments are beginning to emerge (i.e., high sales/high profit ratio, high sales/low profit ratio, etc.). One way to make the segments more apparent is to add reference lines to each axis. Here's what the view looks like when I add a reference line for the average of each axis by right-clicking each axis and choosing "Add reference line":



These reference lines create four quadrants on the view that can be used to segment the data:

- **Top-left quadrant:** High Profit Ratio & Low Sales
- **Top-right quadrant:** High Profit Ratio & High Sales
- **Bottom-left quadrant:** Low Profit Ratio & Low Sales
- **Bottom-right quadrant:** Low Profit Ratio & High Sales

This provides some areas to focus on. For example, we don't want to have high sales of products that are causing us to lose money. Also, are there opportunities to increase the sales of our most profitable items? This example created segments using the averages of each measure, but you can easily change the reference lines to the thresholds of your choice for segmentation.

Lastly, these segments or products of interest can be made more permanent by placing them into sets. In this example, our two highest-selling products are also above average in profitability. Let's put these two products into a "positive-outlier" set by selecting them on the view (either through multi-select or dragging a box around them), right-clicking, and choosing "Create set." Once in a set, these positive outliers can be highlighted in different visualizations, which can lead to valuable insights for the business.

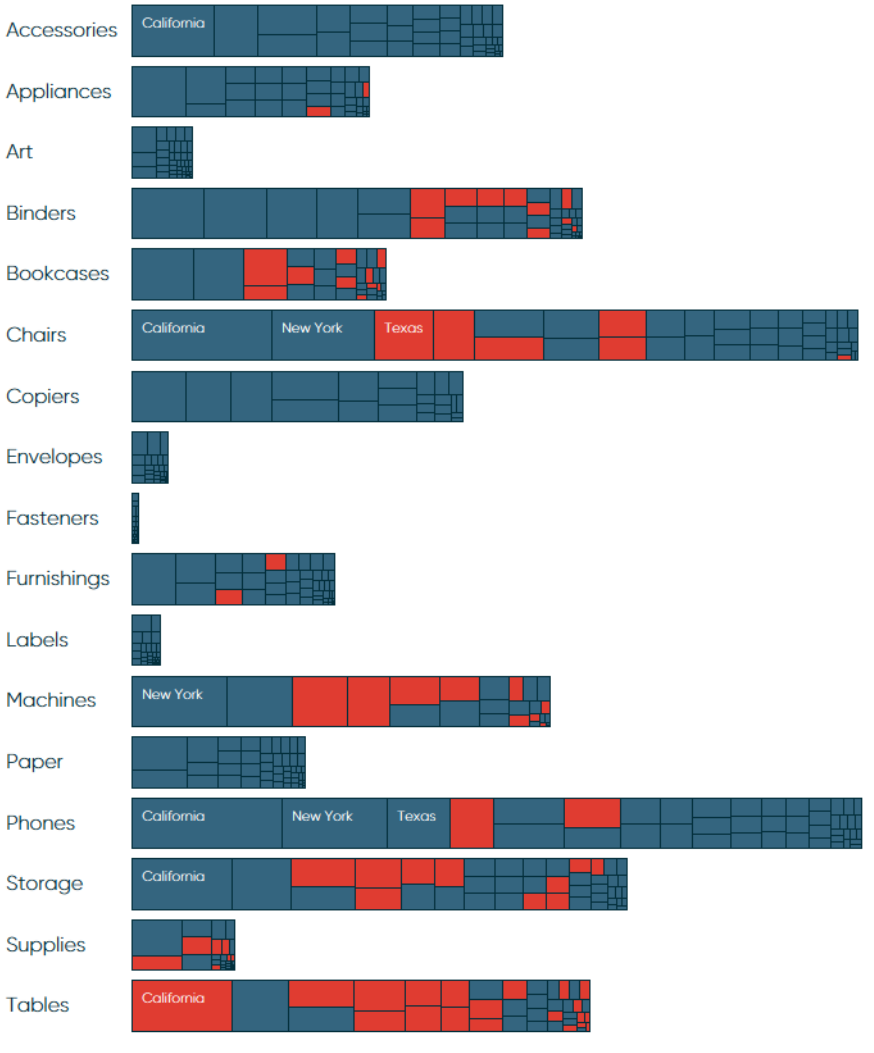
CHAPTER 23

How to Make a Tree Map

A tree map is a visualization that nests rectangles in hierarchies so you can compare different dimension combinations across one or two measures (one for size; one for color) and quickly interpret their respective contributions to the whole. When used poorly, tree maps are not much more than an alternative pie chart. When used well, they provide at least two big benefits:

- Depending on the analysis, some portions of the tree map will be composed of large rectangles where additional context can be added as labels. This is beneficial when the visualization will not be interactive and you still want the written information represented.
- In addition to the scatter plot, tree maps are one of the only visualization types that allow you to reasonably communicate and consume hundreds of marks on a single view. This makes it easier to spot patterns and relationships that you would not otherwise be able to see.

For this tutorial, we will be making the following set of tree maps:



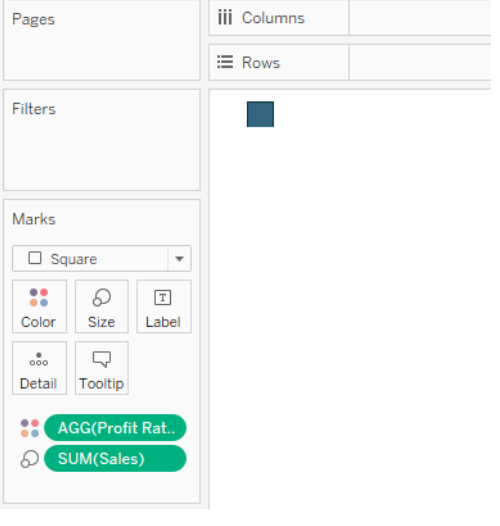
This analysis shows not only the sales amount by sub-category, but the sales contribution of each US state per sub-category and whether or not those states were profitable within each sub-category.

By changing the level of detail to make our analysis more granular and encoding the marks by a second measure of Profit Ratio provides more context to the view and helps us avoid the dreaded question, "So what?"

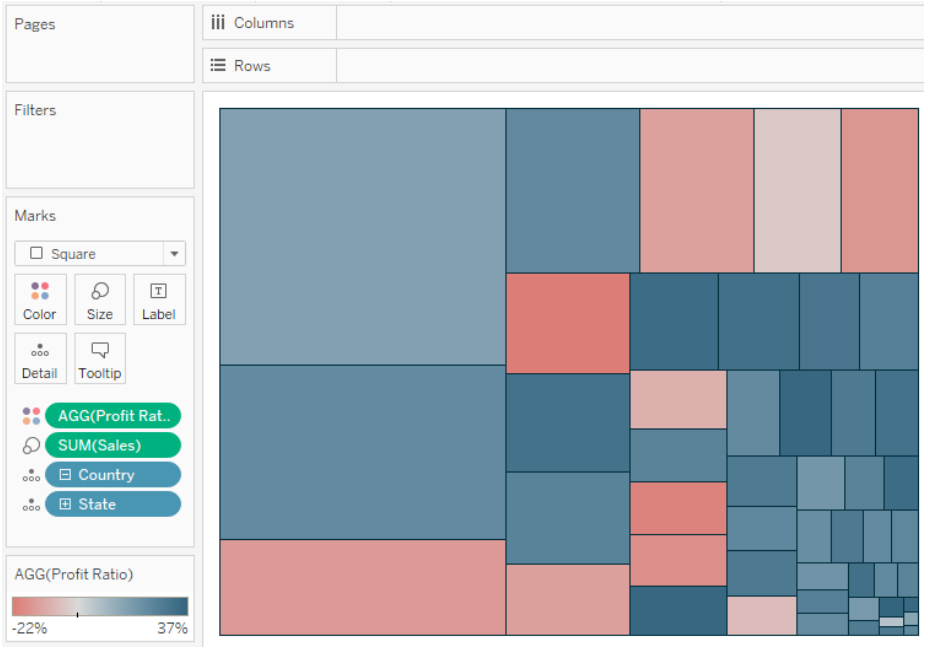
How to Make a Tree Map in Tableau

Tree maps are one of the out-of-the-box Show Me options, but as with most charts in Tableau, I find building them from scratch helps me not only understand how they work, but also helps me get to my desired output faster.

To make a tree map in Tableau, begin by changing the mark type for a view from Automatic to Square. Then drag the primary measure that you want to evaluate to the Size Marks Card and the secondary measure to the Color Marks Card. Just as it sounds, the primary measure will control the size of the squares on the view, and the secondary measure will control the color of the squares. In our case, I've dragged Sales to Size and Profit Ratio to Color:



At this point, we've simply laid the foundation of the tree map. There is no detail, so we just see one square colored by the overall profit ratio in the Sample – Superstore dataset. To create a square for each state, drag State to the Detail Marks Card:



At this point, we have a tree map and a solid analysis. You can see there is quite a bit of real estate to add information to the Label Marks Card and have it displayed on the view (the first benefit mentioned in the introduction).

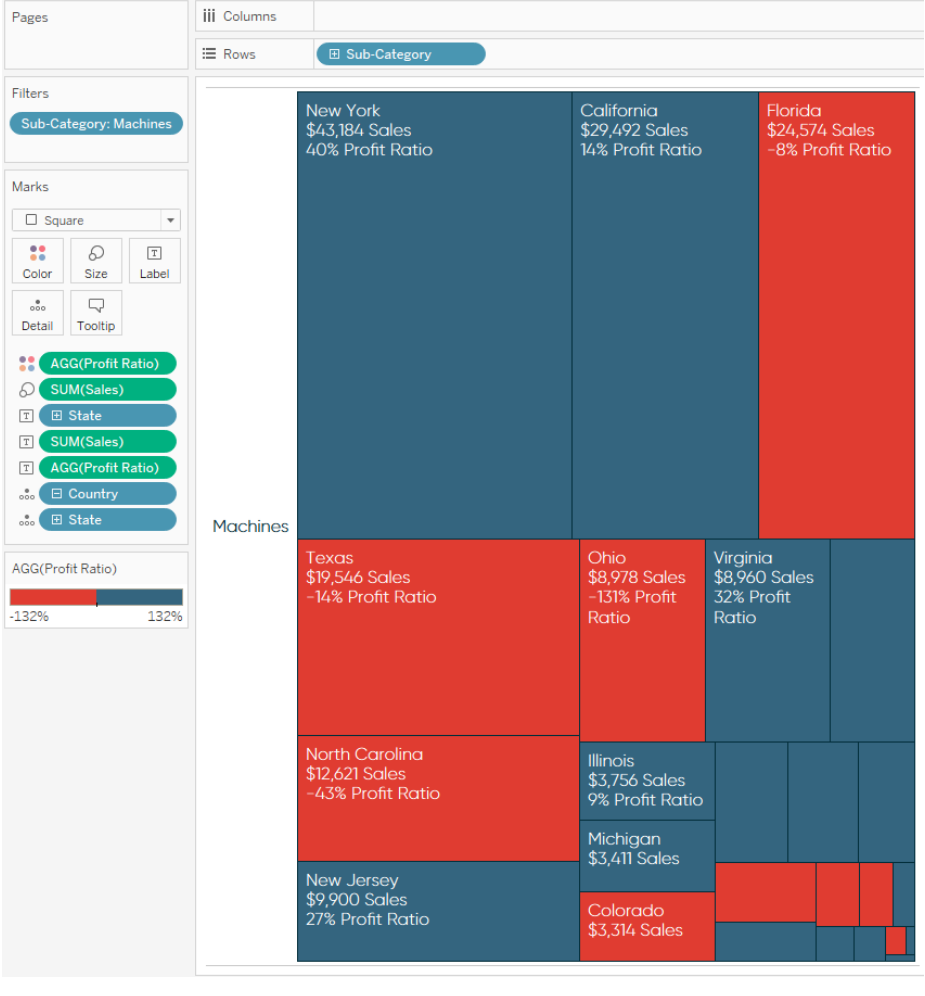
If I wanted to see this same analysis done at the Sub-Category level as just pictured, I would drag the Sub-Category dimension to the Rows Shelf. This will create a row with the sales and profit ratio by state tree map for each sub-category:

The final steps required to match the example would be to drag the State dimension to the Label Marks Card and double-click the color legend to choose the colors and steps. I changed the steps to 2 and changed the colors so that any state with a negative profit ratio per sub-category would be colored red and any state with a positive profit ratio per sub-category would be colored navy:

This tree map provides several insights, including:

- The overall sales are strongest in the Chairs and Phones sub-categories. This is the main insight we would see when looking at a simple sales by sub-category bar chart.
- The Tables sub-category is largely composed of unprofitable states, and two out of the top three highest selling states are unprofitable.
- The overall best-selling state and sub-category combination is Phones in California, which was profitable.

I also see that California is the highest-selling state in all sub-categories except for Machines and Binders. Note that I was not able to see that insight in the screenshot alone, but by hovering over marks in the interactive version of the workbook. You can also simply provide a filter that looks at one sub-category at a time and add even more context to the mark labels:



These are just a few examples of the insights gleaned from this visualization, and a couple of the limitless applications of tree maps.

CHAPTER 24

How to Make Sparklines

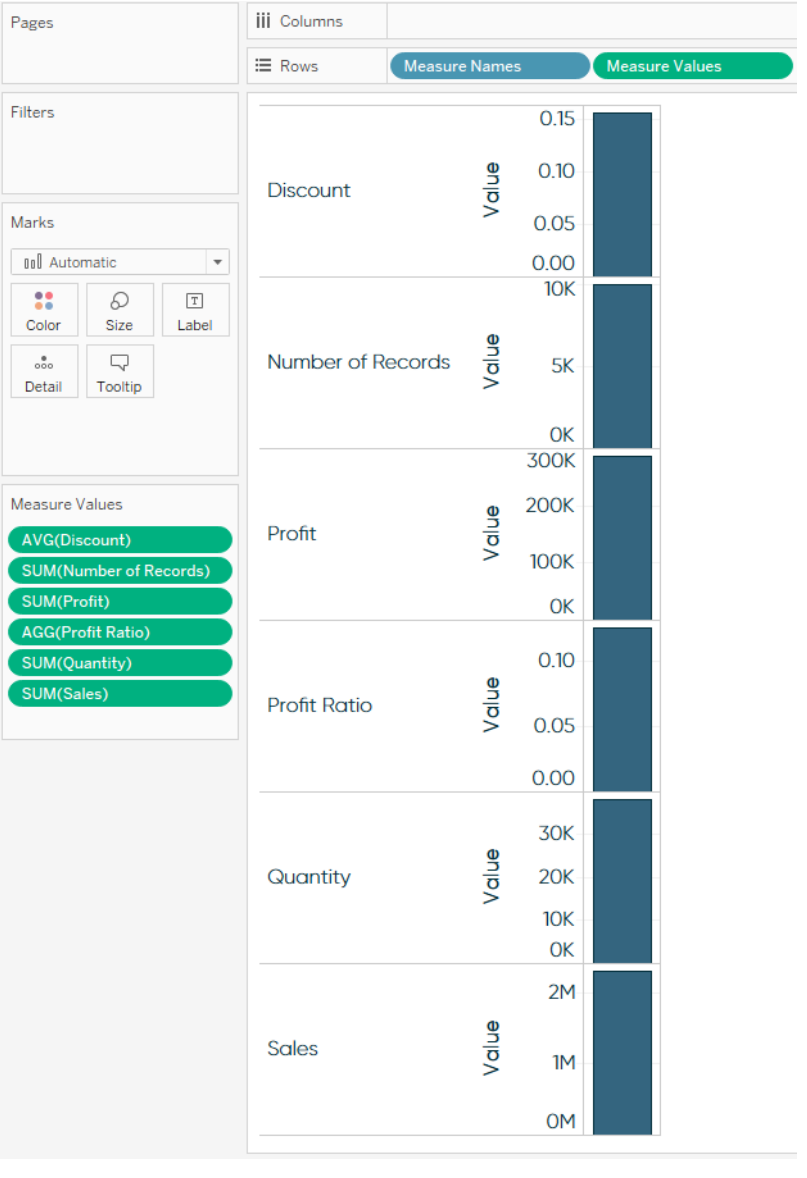
Sparklines are condensed graphs or charts that can be used in-line with text or grouped to show trends across several different measures. The term sparkline was introduced by Edward Tufte, a data visualization pioneer, and proponent of the chart type. Sparklines are typically so small that the chart itself usually does not contain familiar context that you would find in a full-size chart, such as axes. Despite this limitation, I find sparklines to be one of the most effective corporate chart types for quickly communicating trends across KPIs. After all, you can always provide context in the surrounding text, and if you are using Tableau, context can be added through other approaches such as tooltips (the information that appears when you hover over a data point).

It is very common for sparklines to be a foundational piece of the corporate dashboards I create. I think they are a great place to guide an end user to start, so I usually place sparklines in a prominent area of my dashboards, such as down the left side. For more on dashboard layout and design, see tip five in Chapter 90.

How to Make Sparklines in Tableau

Sparklines are easy, and (dare I say?) fun, to create in Tableau. To get started, you will leverage two special fields in your data, *Measure Names* and *Measure Values*. These fields are automatically generated in your data by Tableau so they will be available to use even though they do not exist in your underlying data:

1. PlaceMeasure Names, then Measure Values on the Rows Shelf:

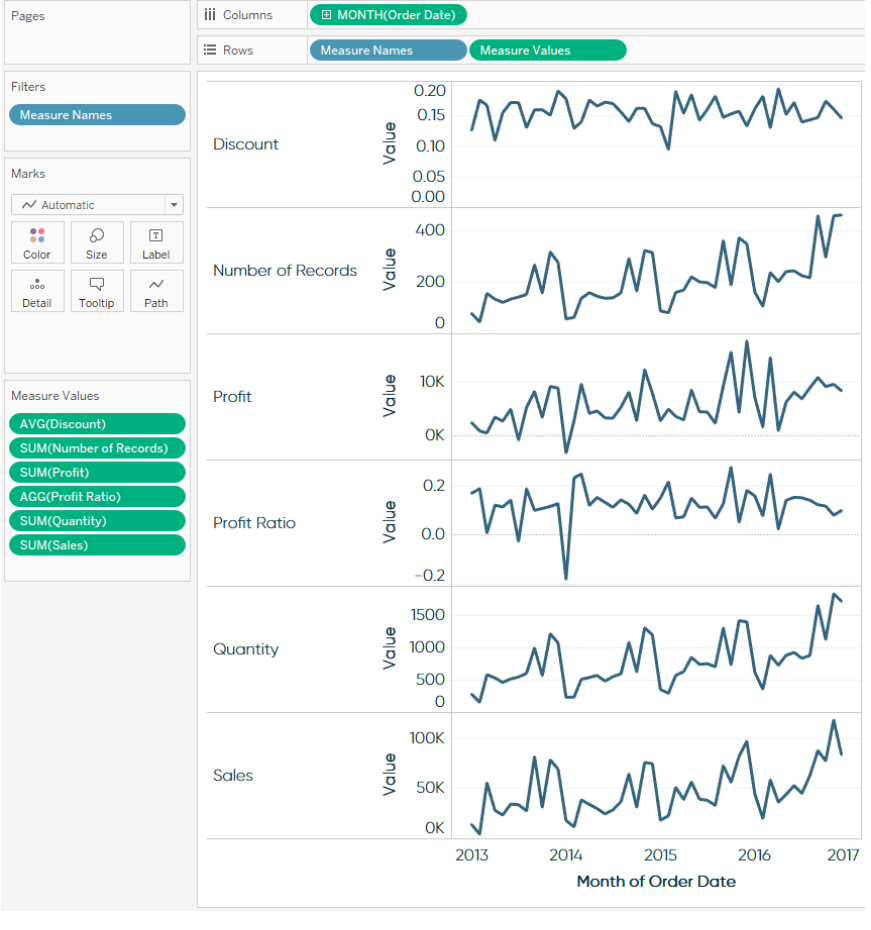


Notice that *every* measure name appears in your chart, whether it is relevant to your analysis or not—more on filtering out specific measure names later. Since you placed Measure Values on your view as well, each measure name has an accompanying value, shown by default as a bar in each measure's default aggregation (i.e., SUM, AVG, etc.). I have also changed the view's fit to Fit Height so that I can see all of the measure names and values without scrolling.

This is a good start, but now we need an element of time to trend the measure values.

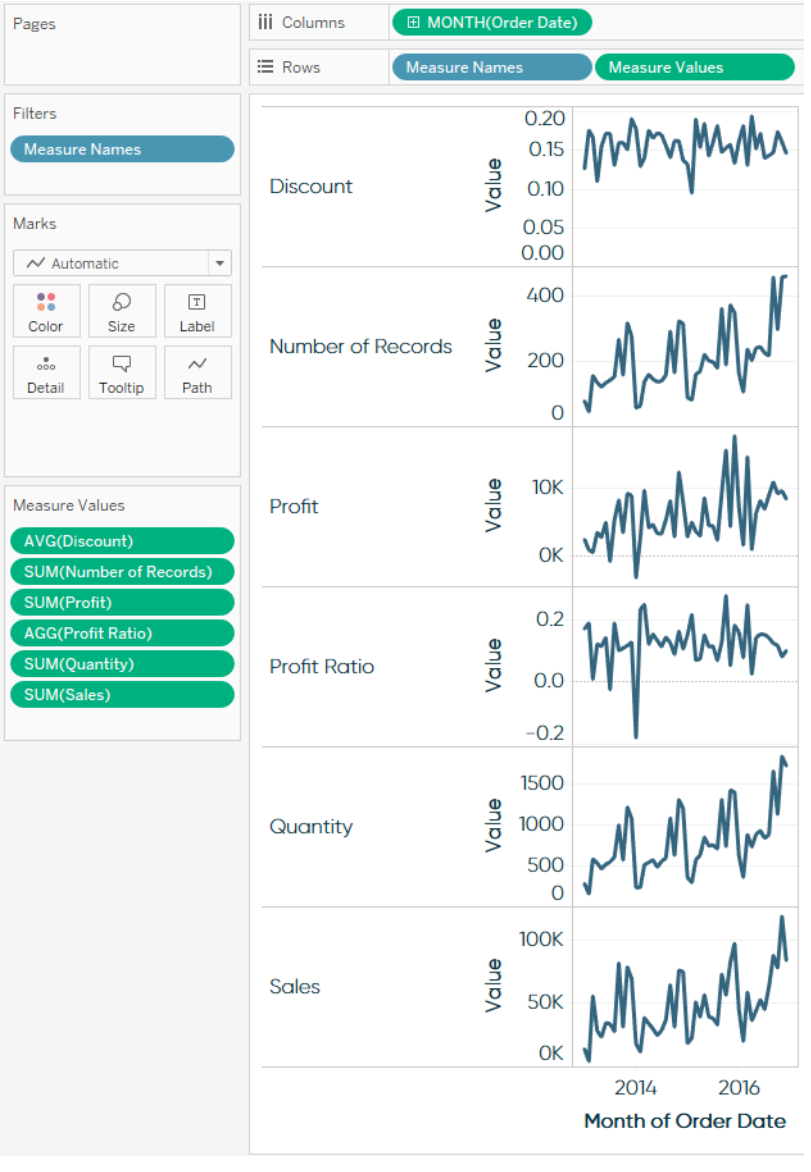
2. Place a date field on the Columns Shelf.

   By right-clicking and dragging my Order Date field onto the Columns Shelf, I was given an extra option to select the date part (i.e., Year, Month, Week). To get the view to look as it does here, I chose the MONTH option preceded by a green icon. The green indicates that the date will be continuous:



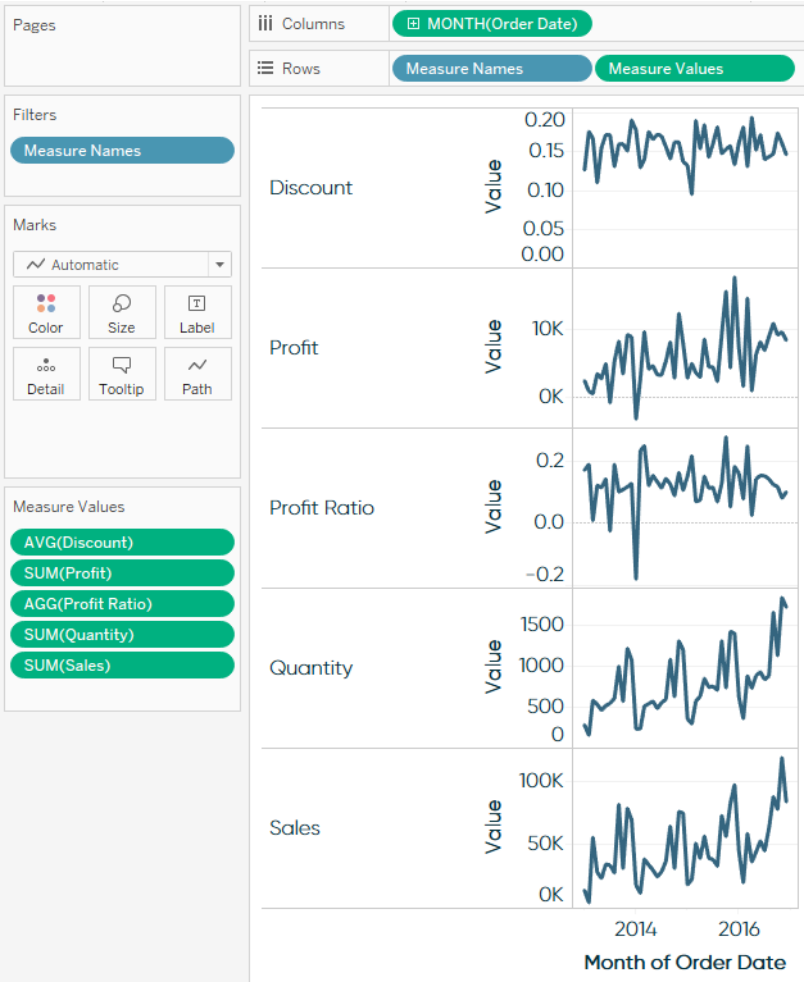
We have now essentially made a series of line graphs, but they are not very "sparky," making it difficult to quickly glean insights. This is an easy fix in Tableau by clicking and dragging the right side of the graph to the left to reduce the width of the view.

3. Reduce the width of the sparklines view to make the trends pop:



You can see at this point that the sparklines are coming together, but as previously mentioned, we have an irrelevant measure name that is not adding much to our analysis.

4. Remove irrelevant measures from your view:



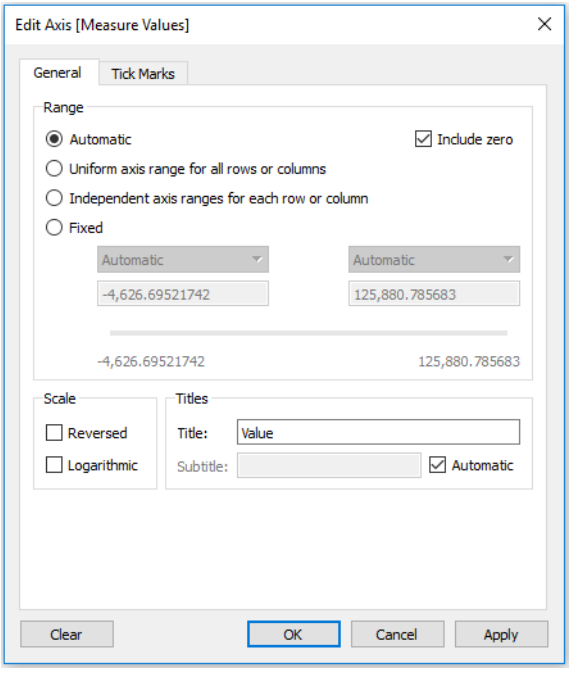
For the purposes of this analysis, I removed the Number of Records measure by dragging its green "pill" from the Measure Values Shelf. Number of Records is a generated field and not needed for this analysis.

We're getting even closer now, but notice that Discount is not providing much insight because this measure has very little fluctuation. For this reason, I typically remove zero from my axes in sparklines. There is much debate around whether it is ever appropriate to exclude zero from your axes because it is easy to mislead your audience when an axis starts anywhere but zero. In the case of sparklines, and measures with little to no volatility, I recommend either excluding zero on the axes, or removing these types of measures completely from your view. Remember: if you cannot gain any insight from these measures in your spark-

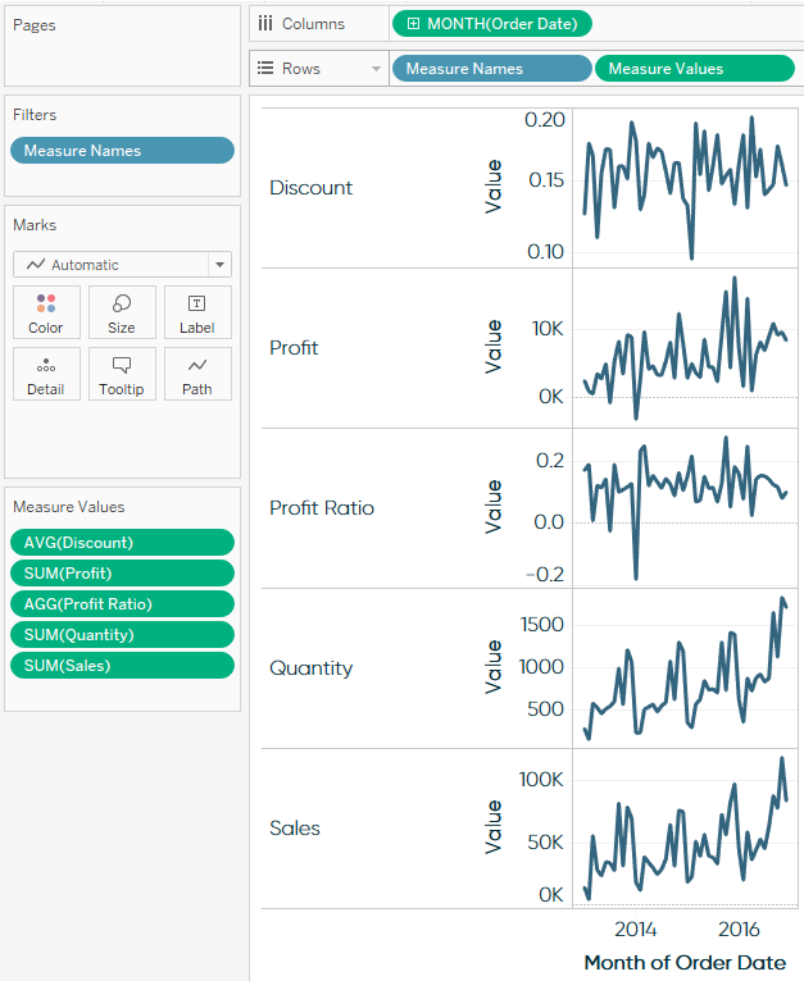
lines, they are not adding any value. There may be times when you expect to see little to no volatility, in which case control charts (covered in Chapter 43), may be a better choice.

5. Exclude zero from your axes or remove measures that have little to no fluctuation.

   If you're comfortable excluding zero from your axes, right-click any of the axes in your sparklines and choose Edit Axis. You will see a box specifically created to give you the option to include or exclude zero in your axes. By default, the box to "Include zero" is checked. To exclude zero, uncheck this box:

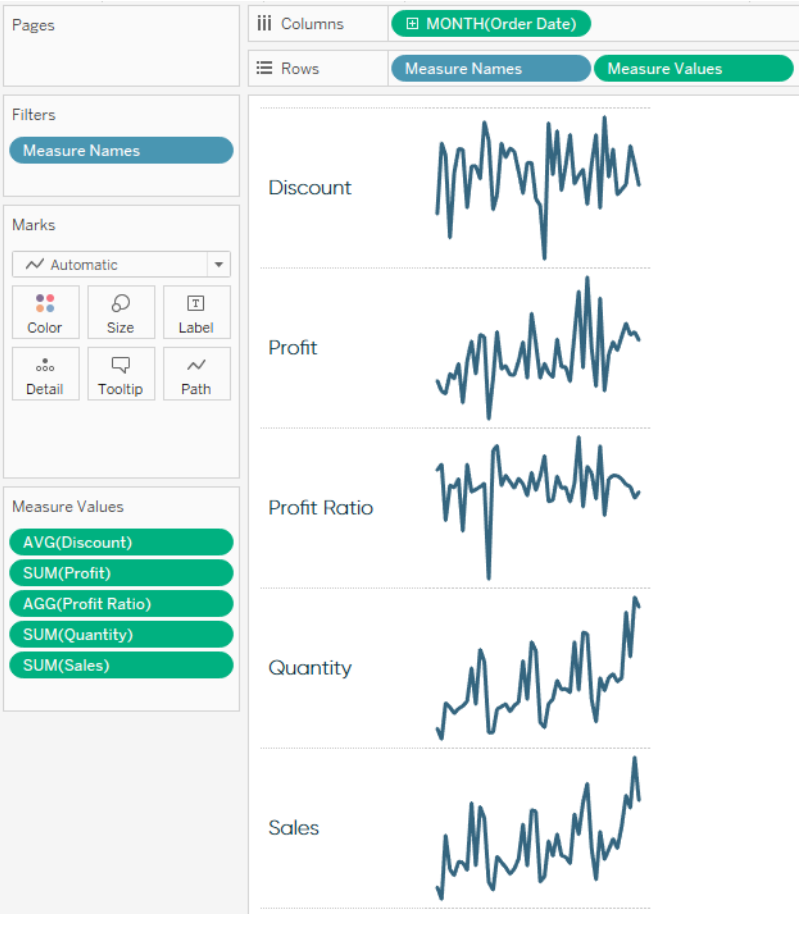


   You can now see the fluctuation in Discount:



From here, all that's left is to format the sparklines to your preference. Remember, sparklines are not quite like regular charts or graphs in that they are meant to provide quick trends at a glance. They don't usually contain typical context, such as the axis values.

6. Hide axes and format your view:



Formatting changes I made to finalize this view include:

- Hid the y-axis by right-clicking the axis and deselecting Show Header.
- Hid the x-axis. Some prefer to keep the axis that shows the date range. I personally exclude this from my sparklines, but if you need to show it, I recommend only showing the start and end points.
- Removed the gridlines.
- Removed the column separators.
- Softened the row separators by choosing a dotted line instead of a solid line.

Final Considerations

The sparklines pictured in this chapter can definitely stand on their own and provide a great deal of insight and a starting point for any dashboard. That being said, Tableau allows you to customize from here in many different ways. For example, you may choose to show labels for the minimum and maximum values for each sparkline. This approach provides an alternative to showing the y-axis because it shows the end user how low and high the axis goes. You can also customize the tooltips to provide additional context such as the exact values and dates that the end user is viewing.

CHAPTER 25

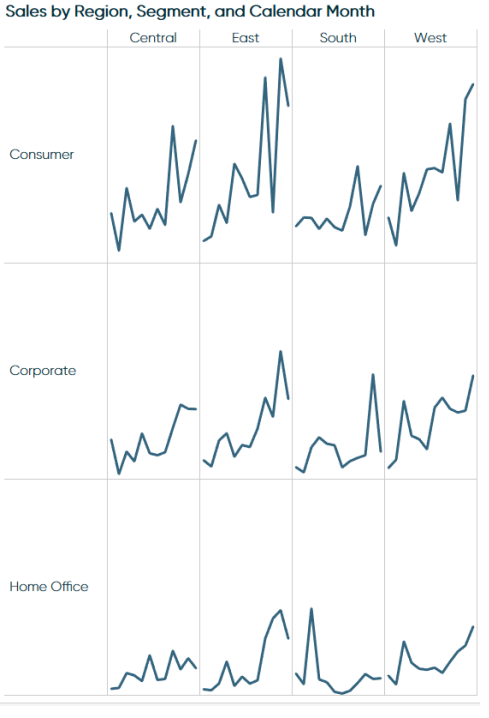
How to Make Small Multiples

Small multiples are a group of charts or graphs that share the same axes and scales, which allows the user to compare trends across dimensions in a single view. They have been praised for their ability to provide a great deal of context, reducing the need for end users to ask the dreaded, "So what?" The term "small multiples" was popularized by Edward Tufte, who puts it best in his book, *Envisioning Information*:

"At the heart of quantitative reasoning is a single question: *Compared to what?*"

While smaller series of small multiples can work well on an executive summary dashboard, I typically like to use them as a second layer in an analysis. My executive level view, or "first layer," may provide higher-level information about a particular measure, such as the overall sales trend and progress toward goals. This is a "descriptive" view of the data answering the question, "What is happening with sales?" From here, I may provide an option to view sales across different dimensions and sub-categories as a series of small multiples in a second layer of the dashboard (often located away from the first view). While small multiples is still a "descriptive" view, it helps answer the question, "Compared to what?"

In this chapter, we will be re-creating the following small multiples view:

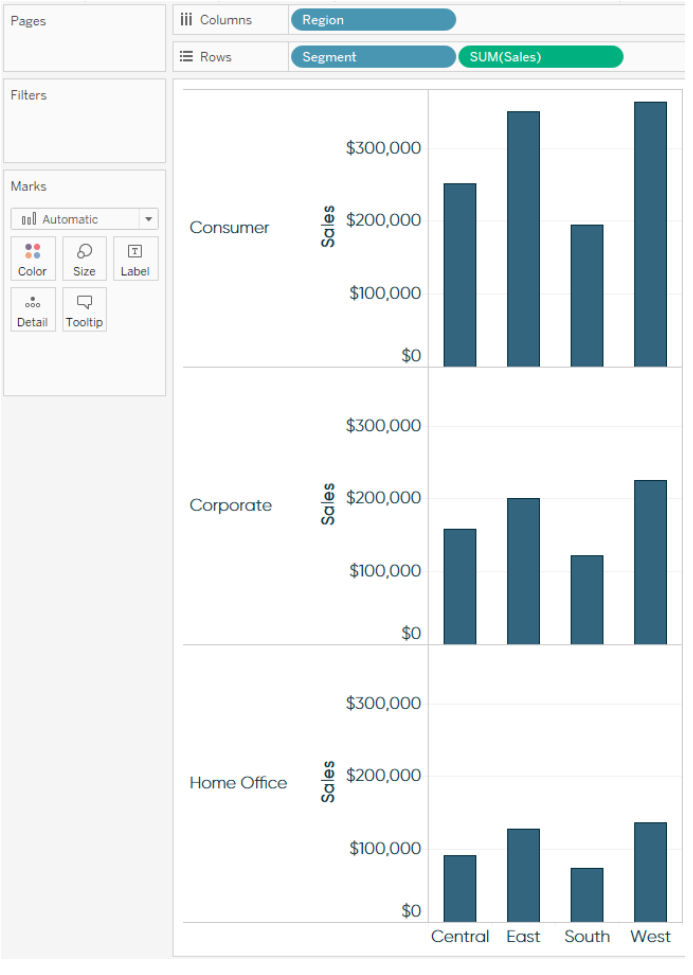


How to Make Small Multiples in Tableau

Small multiples can come in several different forms, but for the purposes of this tutorial, I am going to show you how to compare a single measure across two different dimensions. To get started, select two dimensions and a measure. I have chosen the dimensions of Segment and Region and the measure of Sales for the view to follow:

1. Place one dimension on the Columns Shelf and the other dimension and your measure on the Rows Shelf.

   By default, Tableau has created a bar chart for you. By placing Segment on the Rows Shelf, performance for each segment is read left to right, in rows. By placing Region on the Columns Shelf, each region is represented vertically, in columns:



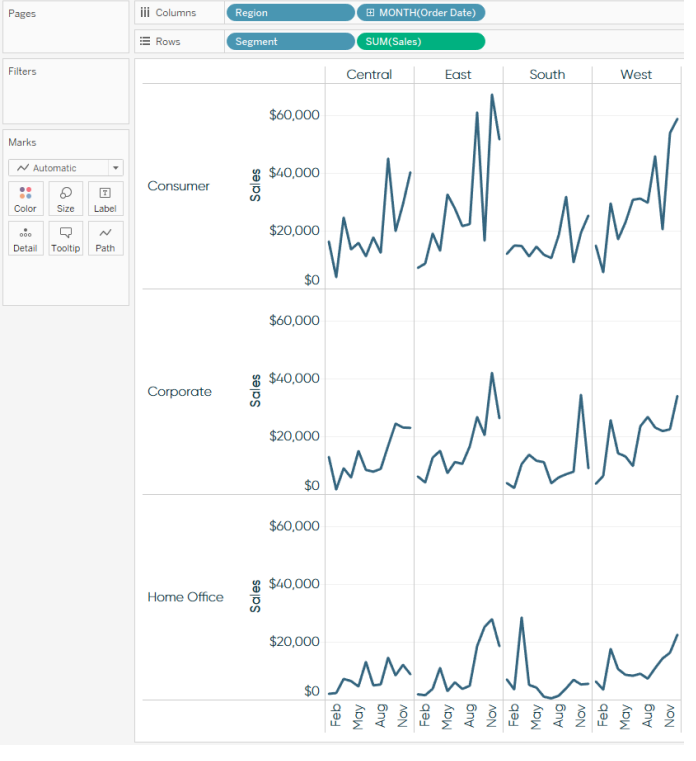
Small multiples may be a series of bar charts, but we still need to add an element of time to trend the Sales measure.

2. Place a date field on the Columns Shelf.

   Let's pretend that we would like to evaluate the seasonality of sales to answer a question such as, "Is there a certain month that I can expect a spike in sales?" For this type of analysis, we will use a discrete date field with a monthly aggregation, which will always show us the performance per distinct calendar month. By using discrete months, I know that I will always have twelve data points per small multiple, one for each calendar month (January, February, March, etc.). The Superstore sample dataset contains four years of data, so if we used a continuous date

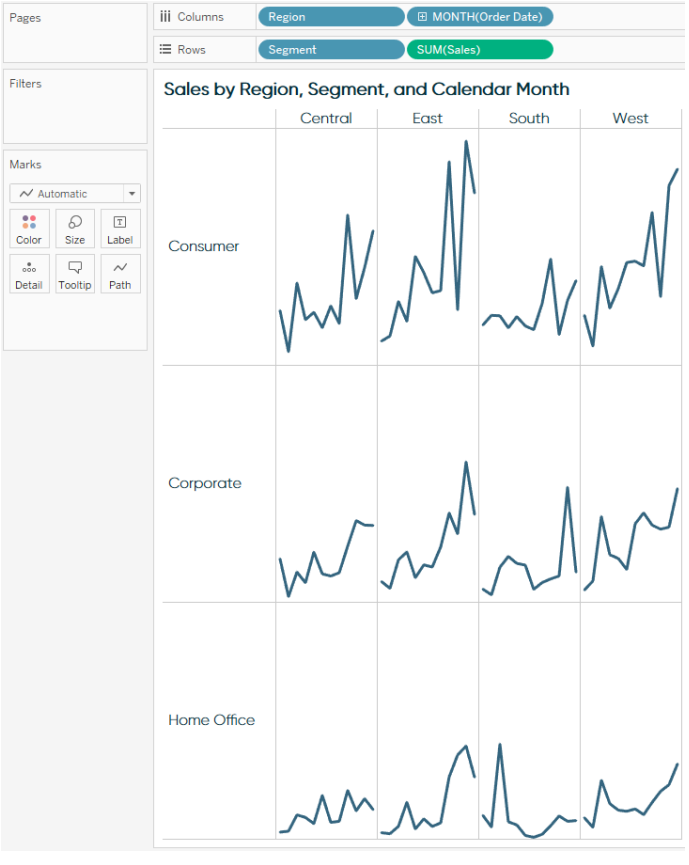
field with a monthly aggregation, we would have up to 48 data points per small multiple (4 years × 12 months = 48 points).

By right-clicking and dragging the Order Date field onto the Columns Shelf, I was given an extra option to select the date part (i.e., Year, Month, Week). To get the view to look as it does here, I chose the MONTH option that was colored blue. The blue indicates that the date will be discrete:



3. Format the small multiples to your preference.

   From here, all that is left to do is format the view to your liking:



Formatting changes I made to finalize this view include:

- Hid the field labels for columns by right-clicking Region/Order Date.
- Hid the field labels for rows by right-clicking Segment.
- Hid both the x- and y-axes by right-clicking them and deselecting Show Header. This is a personal preference of mine. You may choose to keep the axes in the view for additional context. To keep the view as clean as possible, I opted to provide this context in the form of a title, and would typically provide additional information in the tooltips that appear when an end user hovers over different data points.
- Increased the font size in the headers.

Final Considerations

In this tutorial, we have laid a strong foundation for using small multiples, but Tableau makes it easy to build on this view by customizing it in many different ways. For example, instead of using lines, you can change the mark type to bars, which are particularly useful for measures that can be positive or negative, or areas—which provide a beautiful design alternative. I have also seen small multiple views that display year-over-year data nicely and Ben Jones of Tableau has shown how to create small multiple *maps* in Tableau. So get creative, and build small multiples that help provide the answer to "Compared to what?"

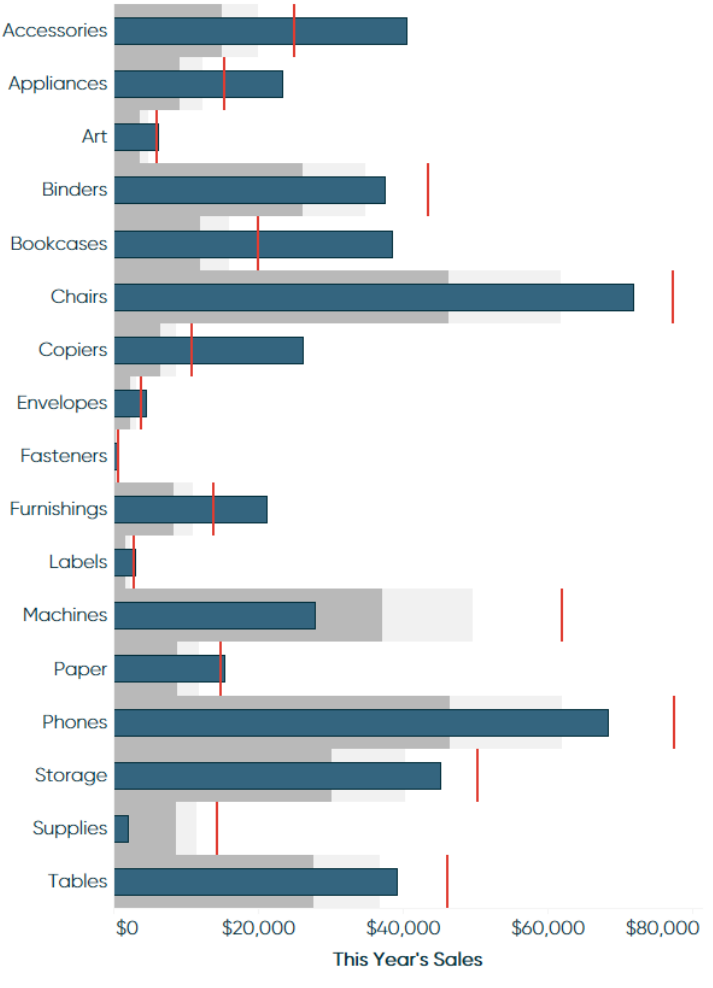
CHAPTER 26

How to Make Bullet Graphs

Bullet graphs are a variation of the bar chart invented by Stephen Few. Bullet graphs are one of my go-to chart types and are often featured prominently throughout my dashboards because I find that when I use them, I hear the question "So what?" less often. As powerful as bar charts are at communicating data, when viewed in a vacuum—or without additional context—their comparisons are limited to only the dimension members that are displayed in the chart. Bullet graphs improve on the bar chart by providing additional points of comparison. For example, in addition to showing a bar for widget sales, a bullet graph would also include a point of comparison that shows either last year's sales or a target sales amount. Further, bullet graphs can include shading to illustrate how close your sales number is to last year's number or your target.

While I find this to be one of the most powerful visualization types, it took me some time to wrap my head around how to make them when I was getting started with Tableau. Bullet graphs are a "Show Me" option in Tableau, but I found that this option was not always providing the results that I expected. Once I figured out that bullet graphs are simply a combination of bars and reference lines, it was much easier for me to create them. The good news is that if you can create a bar chart in Tableau, you can create bullet graphs!

In this chapter, we will re-create the following chart:



How to Make Bullet Graphs in Tableau

Bullet graphs are only an appropriate chart type to use if you have a point of comparison, such as last year's performance or annual performance goals. The Sample – Superstore dataset does not contain goal information, so my point of comparison will be last year's performance.

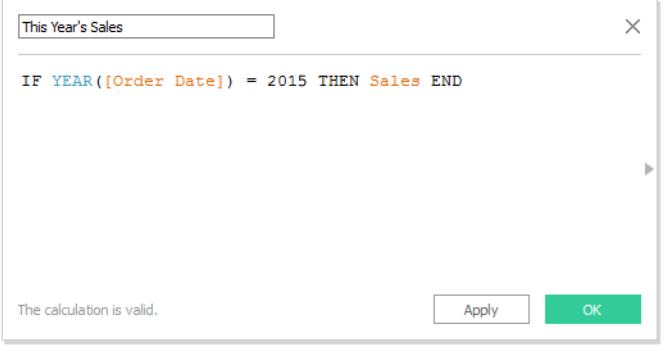
1. Break out this year's performance and last year's performance.

   For this bullet graph, I am going to look at sales by sub-category. In order to create a bullet graph, I will need to break out this year's sales and last year's sales.

The Superstore dataset currently runs through 2017, but for the purpose of illustration, I am going to use 2015 as "this year" and 2014 as "last year."

The isolation of sales for these two years is achieved through calculated fields. To create a field that contains only 2015 sales, right-click the Sales measure and select Create Calculated Field. This approach provides a small shortcut because sales is already part of your formula when the Calculated Field dialog box opens. Once the dialog box is open, enter a formula like this to isolate the current year's sales:

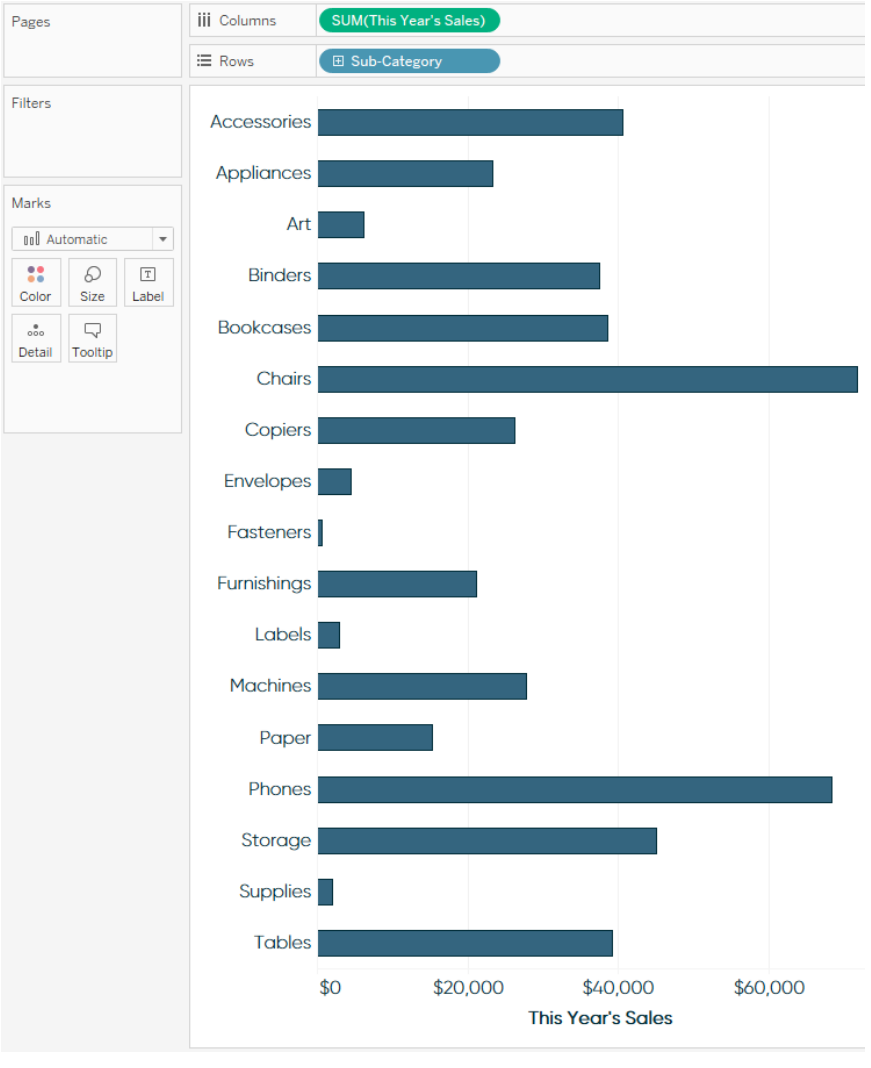
```
IF YEAR([Order Date]) = 2015 THEN [Sales] END
```



Repeat this step to isolate last year's sales (in this case, 2014).

2. Create a bar chart that will serve as the foundation for your bullet graph.

   Create a bar chart as you normally would by placing the current year's sales on the Columns Shelf and a dimension on the Rows Shelf. I am looking at sales by sub-category, so I have placed the Sub-Category dimension on the Rows Shelf. I have also fit the height for more visibility:

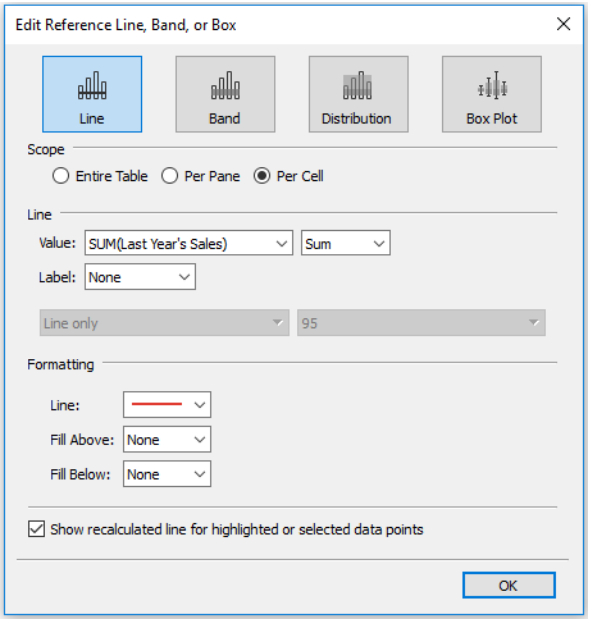


3. Add a reference line for' last year's sales.

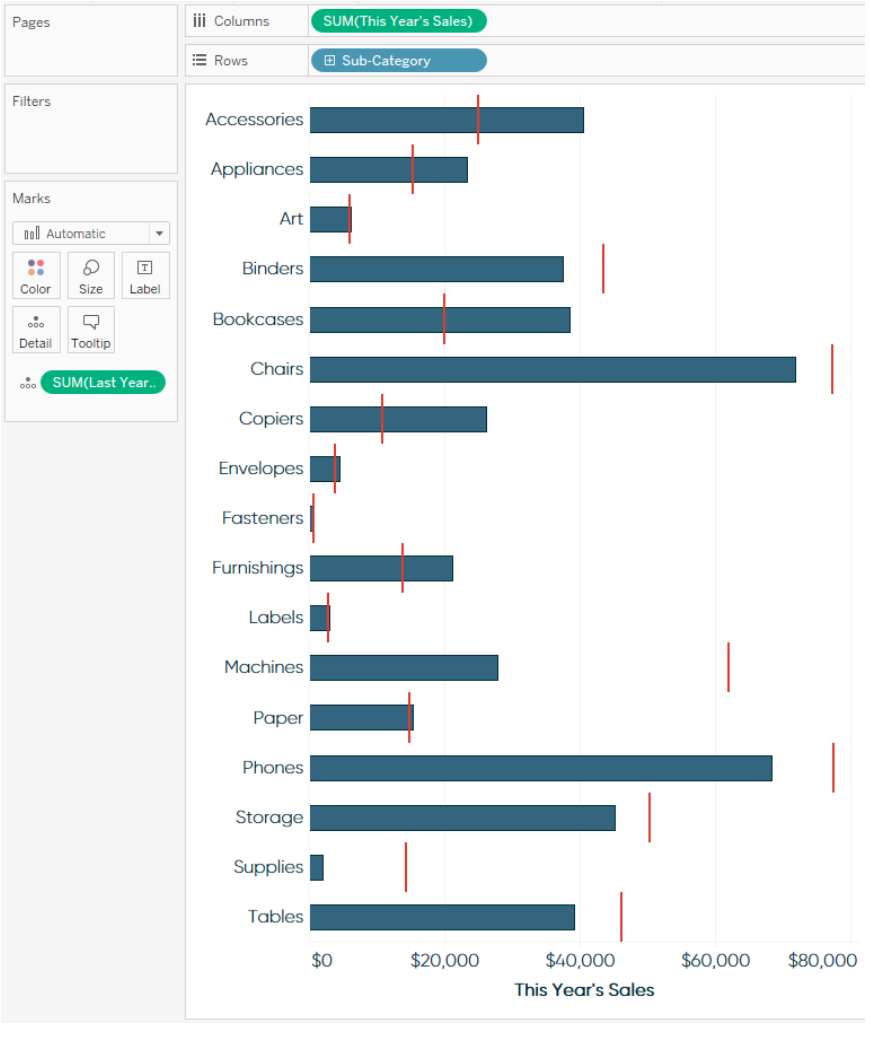
   In order to use last year's (2014) sales as a reference line, the isolated calculation that you created in Step 1 needs to be a part of your view. Even though it is not yet visible, you can make Last Year's Sales part of your view by dropping the field on your Detail Marks Card. Notice that when you place Last Year's Sales on the Detail Marks Card, your view does not change, but now that data is available to use as a reference line.

   To create a reference line, right-click the x-axis and select Add Reference Line. Change Value to SUM(Last Year's Sales) and Label to None. The most important

change is to toggle the Scope radio button from Per Pane to Per Cell. This will give you a reference line for each distinct sub-category. You may also choose to make the line a bolder color and heavier weight to make it stand out on your view. After the appropriate selections, your reference line dialog box should look something like this:



At this point, your view should look like this:



4. Add a reference distribution for last year's sales.'

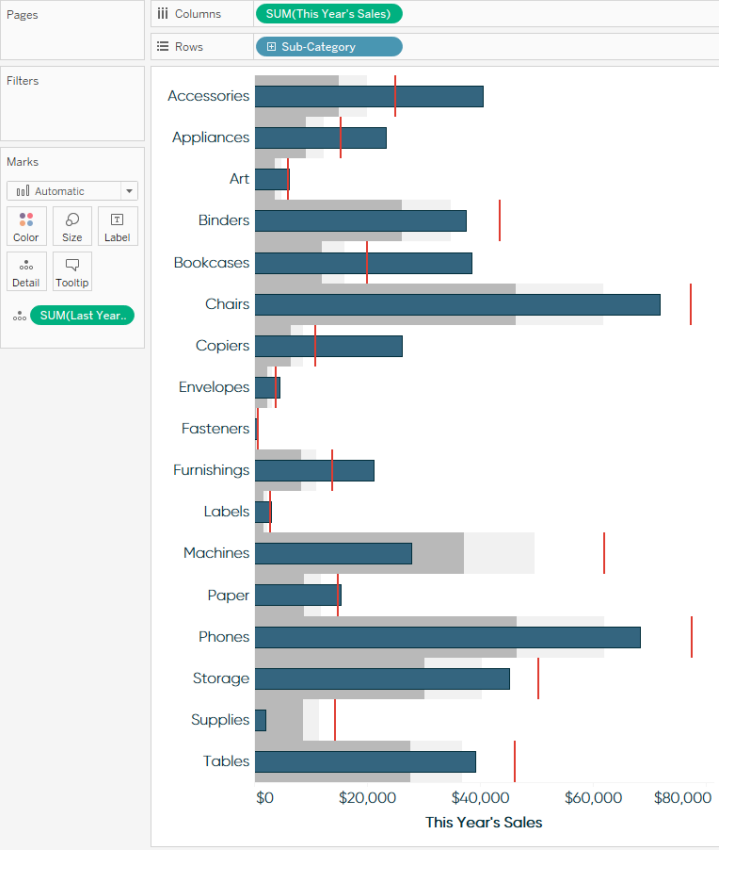
   At this point, you have already added a comparison point to your bar chart that shows whether each category is outperforming or underperforming last year's sales. To take this a step further, you can add a reference distribution to show how close this year's sales are to last year's for the underperforming sub-categories.

   This reference type is slightly more complicated, but still easy to do in Tableau. To create a reference distribution for your bullets, do the following:

5. As you did before, right-click the x-axis and select Add Reference Line.

6. Select Distribution at the top.
7. Change the Scope from Per Pane to Per Cell.
8. This is the trickiest part. You need to change the Computation Value from This Year's Sales to Last Year's Sales. You can do this by clicking the down arrow on the Computation → Value box. Then where it says "Percent of:", make the appropriate selection. Notice that the default percentages are 60% and 80%. This means that it will show shading for 0%–60% of last year's sales and 61%–80% of last year's sales.
9. Change Label to None.
10. Check the box that says Fill Below. This will provide the correct shading when you apply the reference distribution.

After hitting OK, you should see a well-done bullet graph that looks like this:



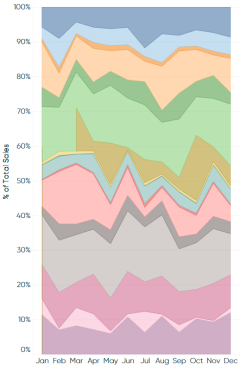
Now that you have a reference distribution, you can quickly determine not only whether or not each category is on pace with last year, but you can see how far behind pace underperforming sub-categories are relative to a year prior.

Another great application of bullet graphs is to use them to show progress toward goals. In that case, the bars would be current year's sales, and the reference lines and distributions would be your goals. By using year-end goals, you can track movement throughout the year and determine which sub-categories are progressing the fastest. This type of context may be enough to help you pivot resources to help underachievers catch up—actionable insight that may not have come from a bar chart alone.

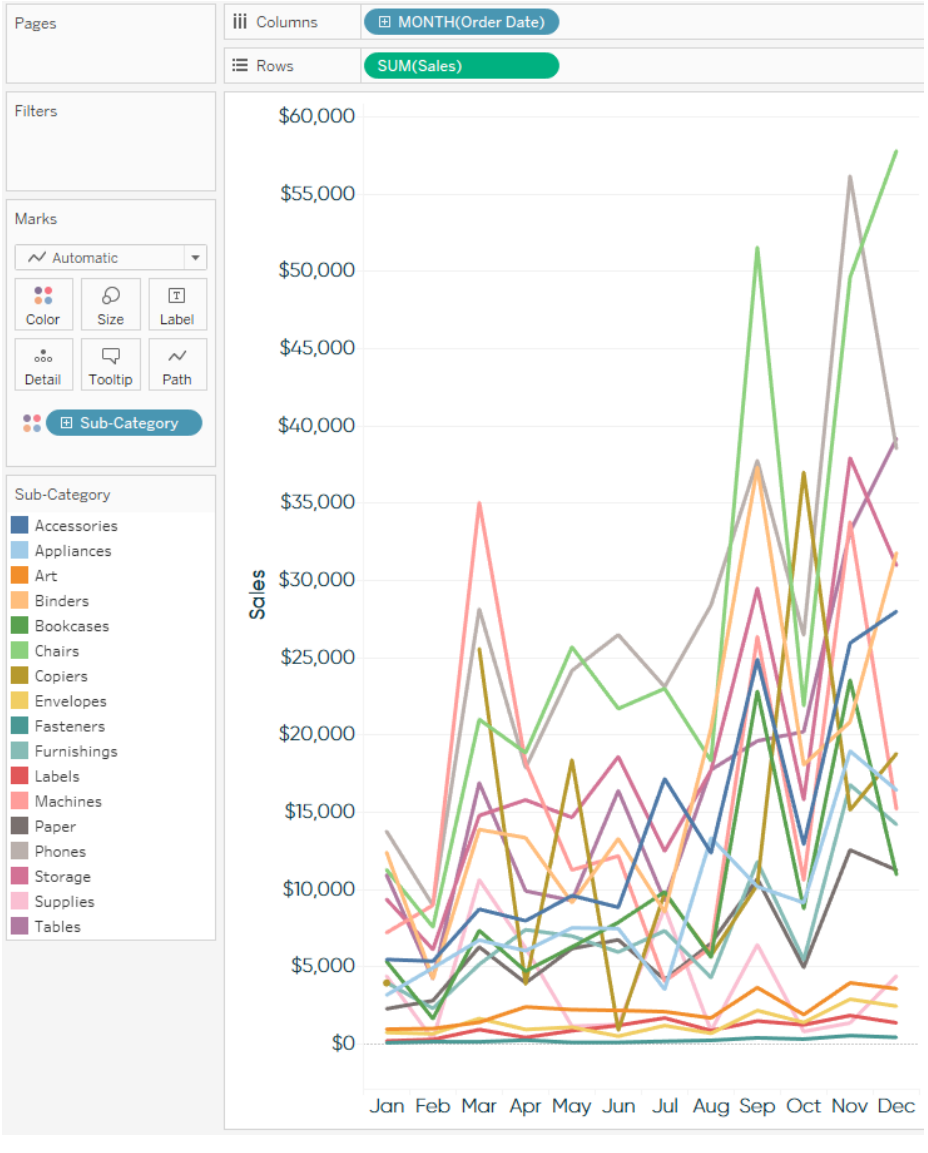
CHAPTER 27

How to Make a Stacked Area Chart

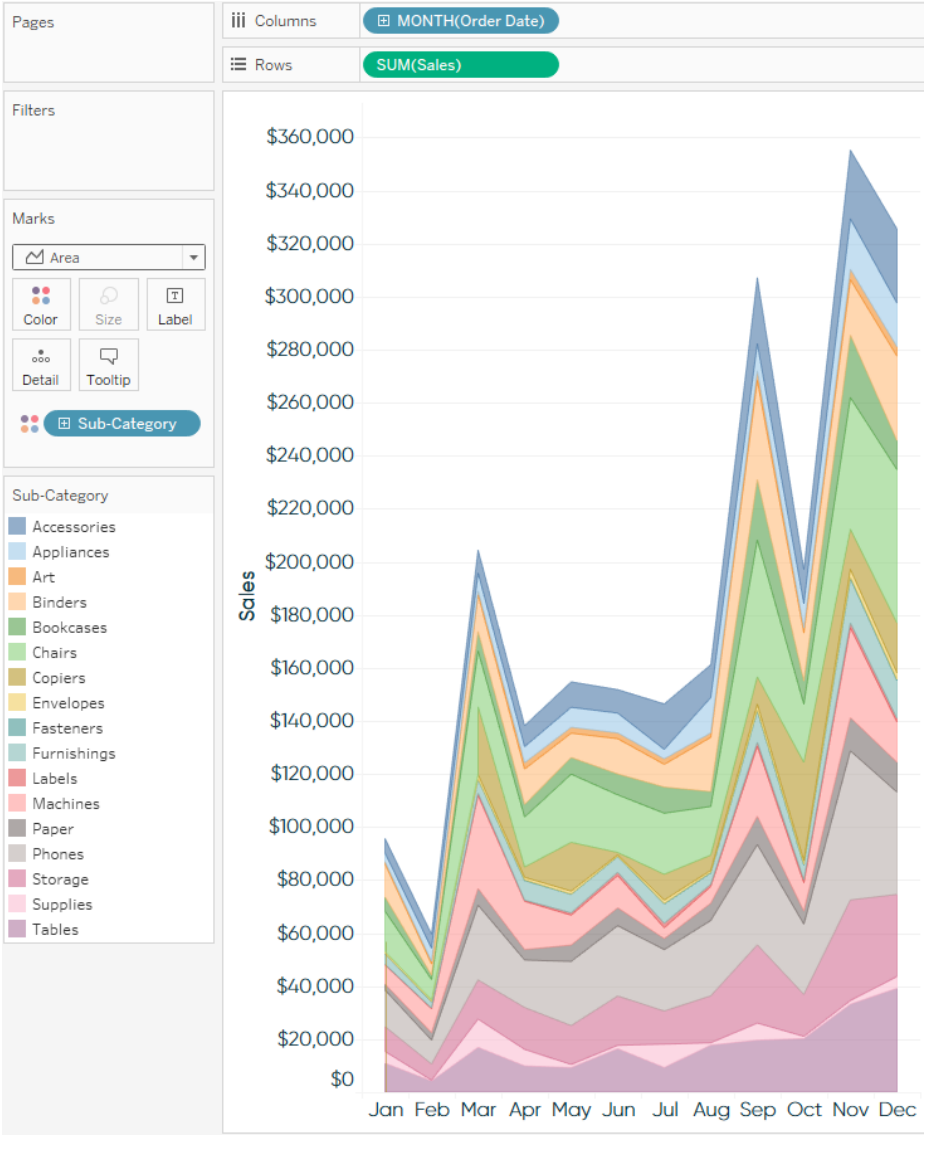
A stacked area chart "stacks" trends on top of each other to illustrate how a part-to-whole distribution changes over time. Combined with a table calculation that computes the percent of total for each dimension member in the visualization, stacked area charts are an effective way to evaluate distributions. In this chapter, we will be re-creating this visualization:



To create a stacked area chart, I will start with this line graph showing Sales by Sub-Category by discrete Month of Order Date in the Sample – Superstore dataset:



To create a stacked area chart in Tableau, simply change the mark type from Automatic, which is currently set to Line, to Area:



As stacked area charts are one of the foundational Show Me options and are extremely easy to create in Tableau, this chapter will focus more on the best applications of this chart type and a trick for using them most effectively.

As you can see in the previous image, when the mark type was changed from Line to Area, the values for each of the dimension members for Sub-Category were stacked on top of each other. This provided a new insight to us that was not available when the mark type was Line: the monthly total across all 17 sub-categories. If it was more important to know the monthly total versus the exact contribution of each sub-category, the stacked area chart is the right tool for the job.

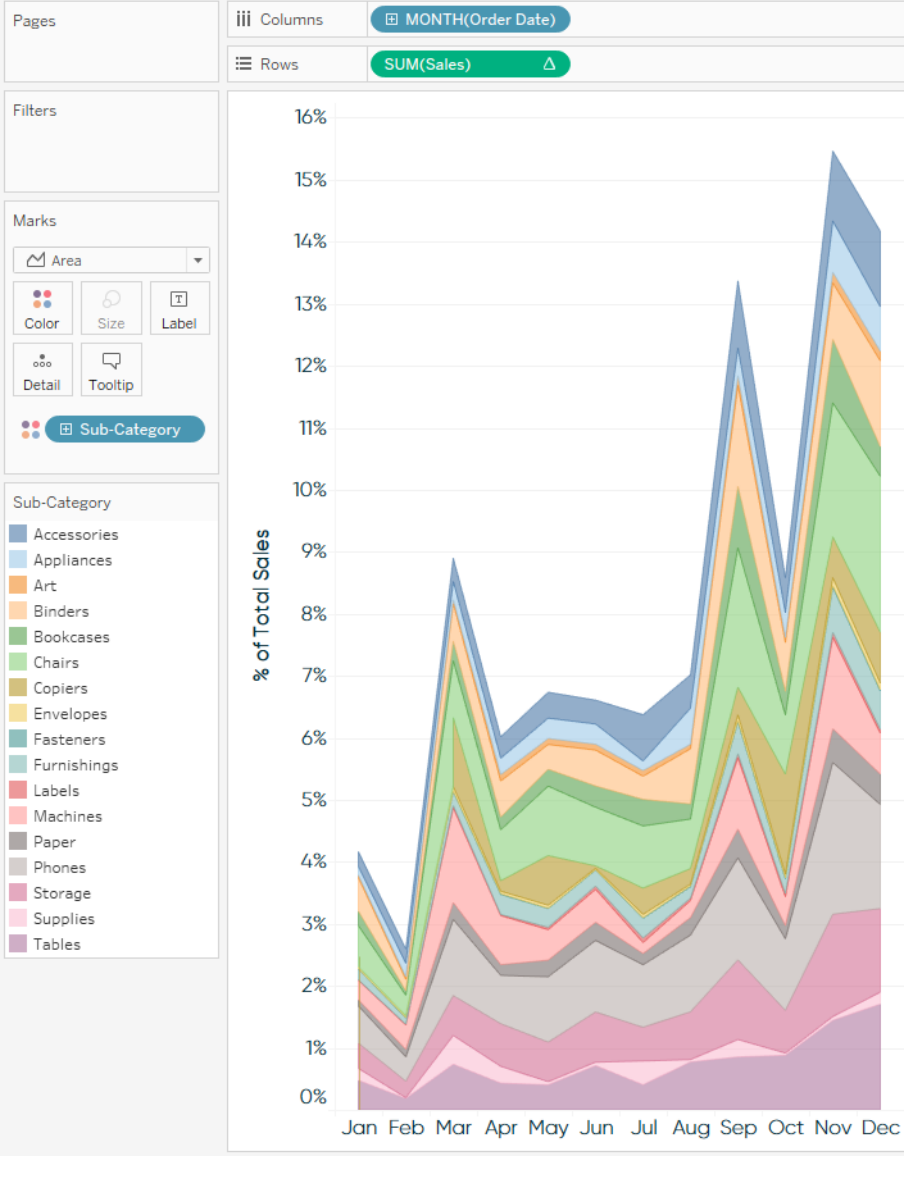
However, if it is more important to know the individual trends of each sub-category, that becomes difficult to assess with a stacked area chart. The reason is that after the trend on the bottom of the stack, each subsequent trend inherits the trend below it. So by the time you get to the slice on top, the value equals the value of the first 16 slices combined, plus the value of the 17th and final dimension member. You can turn off the stacks by navigating to Analysis → Stack Marks and choosing Off, but at that point, you're better off with the line graph we started with.

One other big word of caution if you are using stacked area charts: never use them to stack rates, such as a click-through rate, open rate, or another type of conversion that divides a numerator by a denominator. This is again because the trends are stacked on top of each other, so if you have three dimension members, each with a conversion rate of 5%, the top slice would display the value of the first two stacks combined (10%) plus the value of the third and final dimension member (5%). In this scenario, the slice on top would display a value of 15%, far from its true 5% performance.

So when is a stacked area chart a good choice?

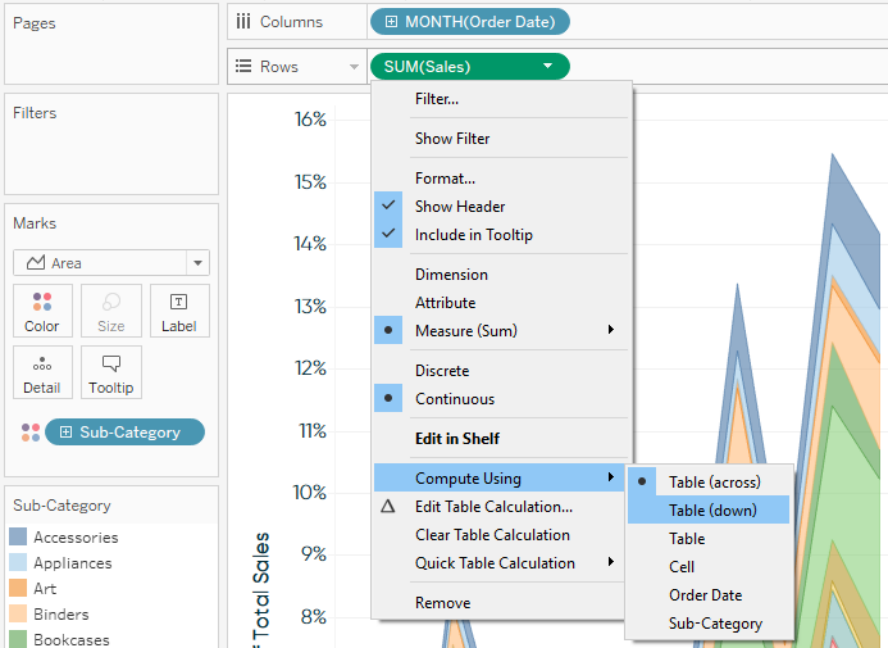
I like to use stacked area charts when the total axis equals 100% and each individual dimension member is displayed as a percentage of the total. This can be achieved by adding a quick table calculation for "Percent of total" to the measure being displayed. For a refresher on table calculations, see Chapter 13.

By default, Tableau calculates table calculations across the table. So if I were to add a quick table calculation for percent of total to the stacked area chart in the previous image, I see this:

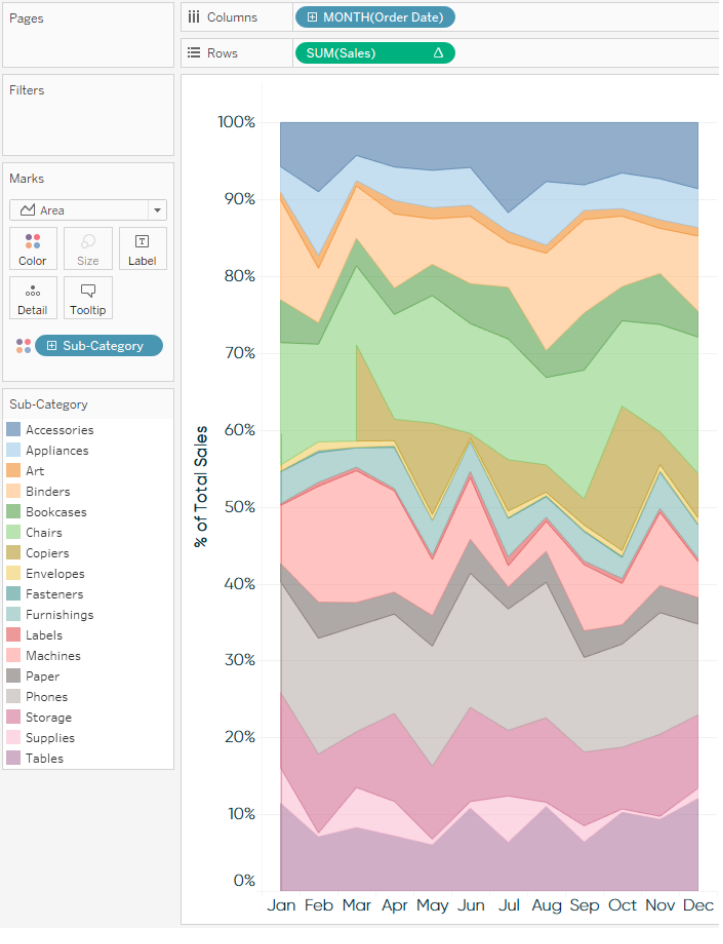


With this view, the trends look exactly the same, but the axis is being displayed as a percent of total instead of the raw revenue values. To change the view so that the axis totals 100% and the areas represent each dimension member's contribution to each

month's total, the table calculation needs to be changed to compute using Table (down). This is changed by clicking the measure with the table calculation for percent of total, now designated with a delta symbol, hovering over Compute Using, and selecting Table (down):



After changing the direction of the table calculation, it is easier to see the monthly contribution for each individual dimension member:



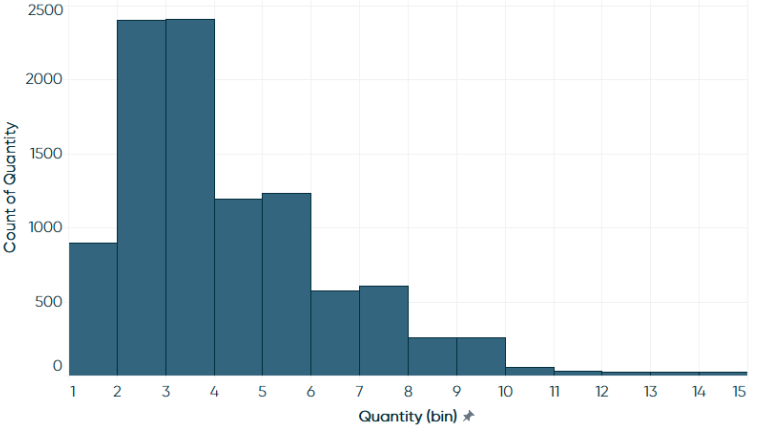
Unless the slice is on the top or the bottom, it can still be challenging to evaluate the trends for individual dimension members with a great deal of precision. At the very least, it should be easier to evaluate which dimension members are gaining or losing share from month to month. As with most analyses in Tableau, the chart selection should be informed by the business question at hand. Stacked area charts, especially when combined with table calculations, are a good visualization to have in your tool-belt to display how distributions in your business are changing over time.

CHAPTER 28

How to Make a Histogram

Histograms are one of the most effective chart types for showing a distribution of quantitative data at one point in time. Similar to a bar chart, the important distinction is that histograms are used to plot continuous, numerical data while the bar chart is used to plot discrete, categorical data. When you create a histogram, bins are created to group equally sized numerical ranges. Despite being continuous and quantitative, these bins can be thought of as the dimension that you slice and dice the count of records by to create the histogram.

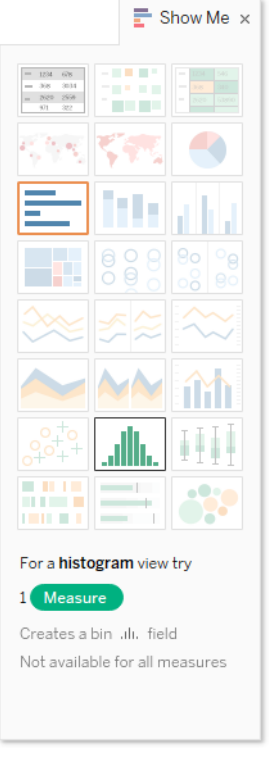
In this chapter, we will re-create the following visualization:



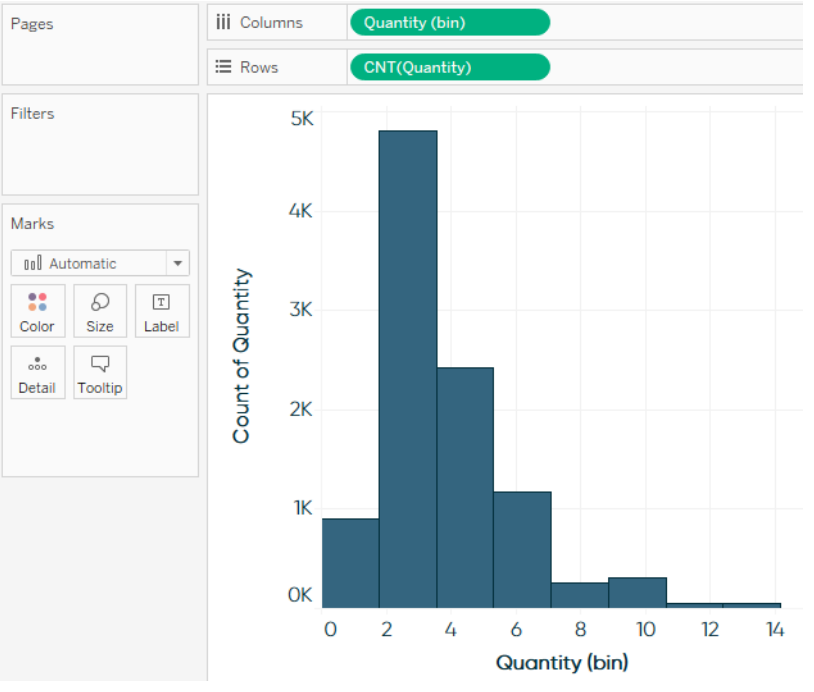
How to Make a Histogram in Tableau

Histograms are one of the 24 chart types in Tableau that can be created using the Show Me tool in the upper-right corner of the Tableau Desktop interface. I've men-

tioned I usually like to explain how a visualization is created manually in Tableau instead of relying on Show Me, but histograms are one of my few exceptions. Histograms are created in Tableau using just one measure. To create a histogram, pre-select the measure that you want to visualize the distribution for by clicking it in the Measures Shelf. Then navigate to the Show Me options and choose "histogram":

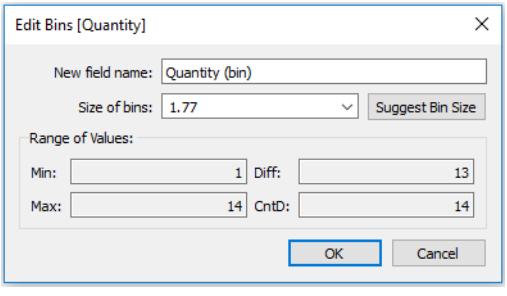


Here's how a histogram looks if you follow the steps for the Quantity measure in the Sample – Superstore dataset:

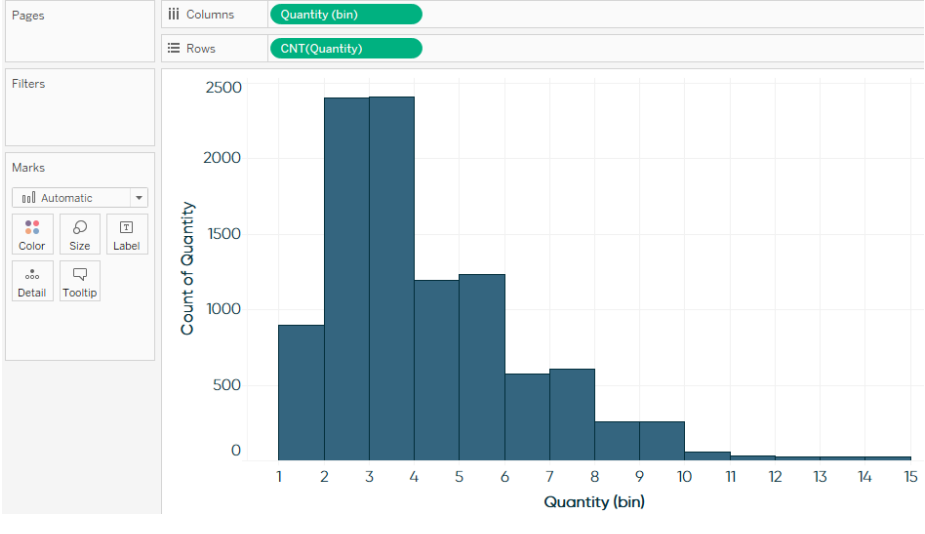


When this chart was created, Tableau conveniently created the dimension for Quantity (bin) to provide the equally sized ranges that are required to create a histogram. The bars then represent the count of records within each of these ranges. Notice that the bars have no space between them; this is how some authors like to differentiate the bars with continuous data in a histogram from the bars with categorical data in a bar chart.

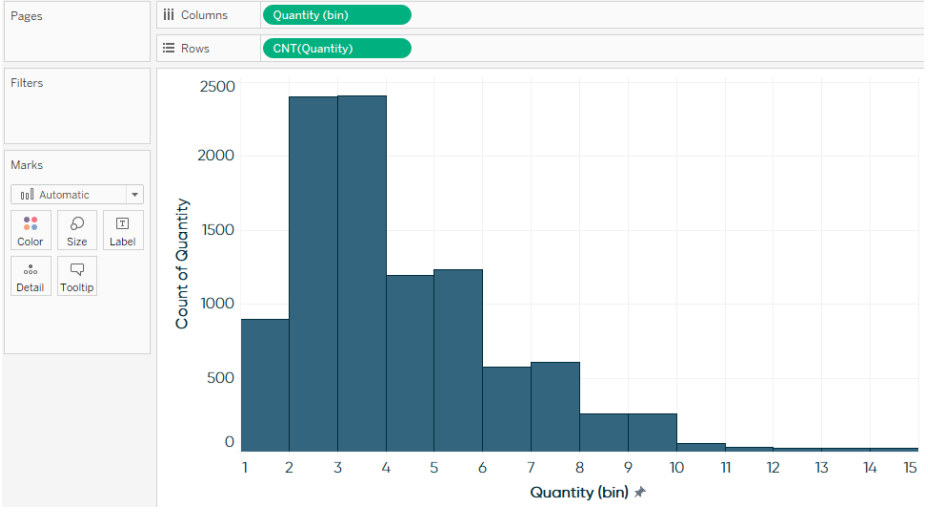
Depending on how the distribution looks, you may choose to change the size of the bins. This is easy to accomplish by locating the newly created Quantity (bin) dimension in the Dimensions area of the Data pane, right-clicking the field, and choosing Edit. A new interface will appear that shows you the current bin size and allows you to change the equal ranges:



Here, you can either type in a number, or click the drop-down arrow to have the bin size be based on a parameter. Let's say for simplicity that we want to change the bin size for the quantity measure to one. This analysis would show the end user how many records (i.e., sales) included one item, two items, three items, and so on. Here's how the histogram looks after editing the bin size:



Lastly, because the bin ranges are continuous, we have a continuous axis that begins at zero and ends at one bin past our maximum bin size. If you need to change the axis range, right-click the x-axis and choose "Edit axis." Here is how my final histogram for quantity looks after fixing the axis range from 1 to 15:



CHAPTER 29

How to Make a Box-and-Whisker Plot

The box-and-whisker plot, or box plot, is another effective visualization choice for illustrating distributions. Along with histograms and stacked area charts, box-and-whisker plots are among my favorite chart types used for this purpose. They work particularly well when you want to compare the distributions across two different dimension members side-by-side, where one set of dimension members makes up the x-axis, and the other dimension member is used as the visualization's level of detail. To help illustrate, here's the box-and-whisker plot we will create with the Sample – Superstore dataset during this tutorial:

As you can see, each set of circles corresponds to the dimension members on the x-axis for the Sub-Category dimension. The level of detail, or most granular level of the analysis, is Month of Order Date. Since the level of detail is Month of Order Date, each Sub-Category column has 12 circles, one for each month of the year.

In short, this visualization is showing how the distributions of monthly sales vary between product sub-categories. While I can easily find several insights in this visualization and believe box-and-whisker plots to be among the most effective ways to communicate distributions, I find them to be one of the most misunderstood chart types when I attempt to share them with an external audience.

For this reason, this chapter shares not only how to make box-and-whisker plots in Tableau, but how to read them. You may need to teach your users how to make sense of a visualization, an effort that can pay off when building a dashboard or other viz that they will refer to on a regular basis. Putting in this effort can also be useful if you're doing a live presentation, but it may not be the best choice if you're putting data out for a general audience.