Data visualization, Barcharts

Steve Simon

Created: 2019-08-24

Barcharts, Outline

- Preparation
- Group exercises
- Colors
- Perception
- Barchart fundamentals
- Barchart recommendations

Here is an outline of what you will learn today.

Preparation, overview

- Follow these steps to get ready for lecture #2 in data visualization.
  - Download and import Titanic data
  - Draw a barchart
    - Count for pclass

The titanic data set shows mortality results (lived or died) for the 1,313 passengers onboard the Titanic. The data includes information about passenger class (third class fared poorly relative to first or second class), sex (men fared poorly compared to women), and age (children fared better than adults).

Draw a simple bar chart showing the count of the number of passengers in first, second, and third class.

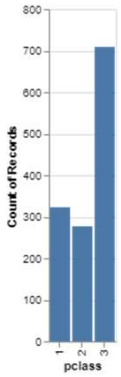
Preparation, Pyhton Code

– Python code

```
import pandas as pd
import altair as alt
df = pd.read_csv("data/titanic3.csv")
ch = alt.Chart(df).mark_bar().encode(
    x='pclass:N',
    y='count()'
)
ch.save("images/python/basic-barchart.html")
```

Here is the Python code to read in the data and produce a bar chart showing the number of passengers in each passenger class.

Preparation, Python output



Barchart in Python

Here is the Python output.

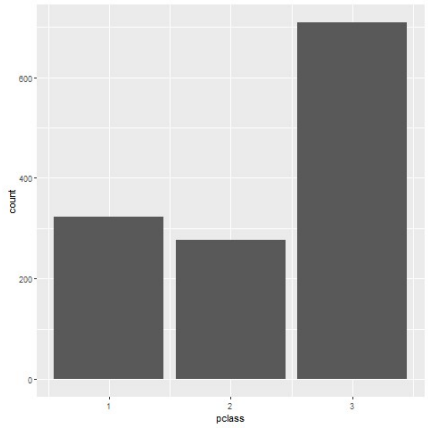
Preparation, R code

– R code

```
ti <- read.csv(file="data/titanic3/csv")
ggplot(ti, aes(pclass)) +
  geom_bar()
```

Here is the R code to read in the data and produce a bar chart showing the number of passengers in each passenger class.

Preparation, R output



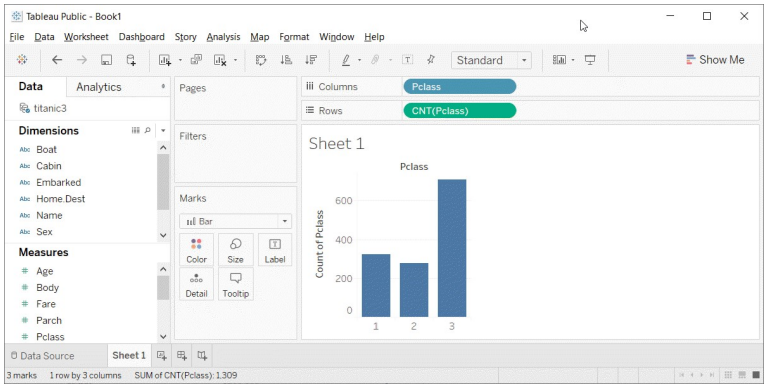
Here is the R output.

Preparation, Tableau steps

- Import titanic3.csv
- Drag pclass to columns
  - Change to Discete Dimension
- Drag pclass to rows
  - Change to Measure(Count)
- If needed, change Marks to Bar

Here are the steps in Tableau to create a barchart.

Preparation, Tableau output



Barchart in Tableau

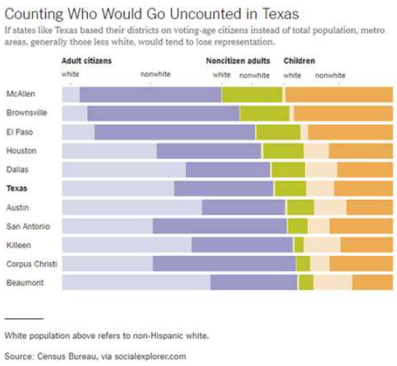
Here is the Tableau output.

Group exercise

– These bar charts come from recent newspaper articles.
  - People Who Can't Vote Still Count Politically in America. What if That Changes?
  - Attacks by White Extremists Are Growing. So Are Their Connections.
  - The Best VPN Service.

The following images are taken from various newspaper articles or press releases. Look at the graph and read/skim the article.

Graphs in the news, Voters in Texas

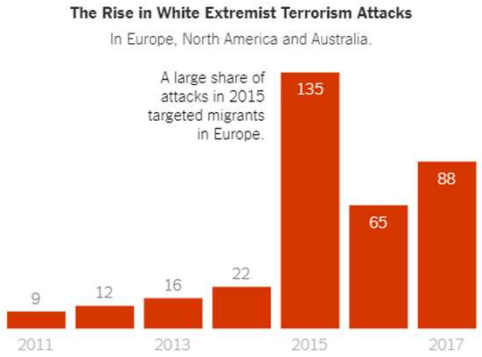


Bar chart of demographics of selected Texas counties

This graph was published in a newspaper article,

Badger, E. (2019). People Who Can't Vote Still Count Politically in America. What if That Changes? Retrieved June 24, 2019, from The New York Times website: https://www.nytimes.com/2019/06/22/upshot/america-who-deserves-representation.html

Graphs in the news, White extremist terrorism attacks

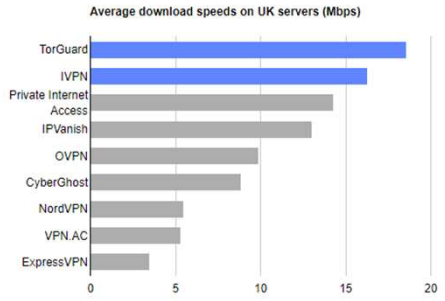


Bar chart counting white extremist terrorism attacks over time

This graph was published in a newspaper article,

Weiyi Cai, Simone Landon. Attacks by White Extremists Are Growing. So Are Their Connections. The New York Times, April 3, 2019. Retrieved August 8, 2019 from https://www.nytimes.com/interactive/2019/04/03/world/white-extremist-terrorism-christchurch.html

Graphs in the news, VPN speeds

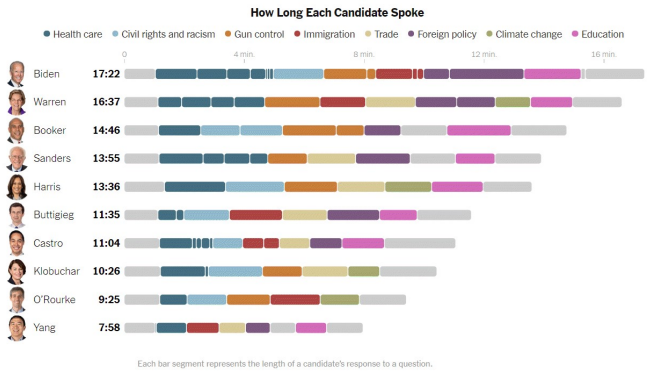


Bar chart showing speeds of various VPN providers

The graph was published in a web news article,

Mark Smirniotis. The Best VPN Service. Wirecutter, February 8, 2019. Retrieved August 14, 2019 from https://thewirecutter.com/reviews/best-vpn-service/.

Graphs in the news, Debate speaking time



Bar chart showning speaking time at Democratic debate

This graph was published in a newspaper article,

Weiyi Cai, Jasmine C. Lee, Jugal K. Patel. Speaking Time in the Democratic Debate. The New York Times, Sept. 12, 2019. Retrieved September 12, 2019 from https://www.nytimes.com/interactive/2019/09/12/us/elections/debate-speaking-time.html.

Graphs in the news, what is the message?

- Your group will be assigned one particular graph and newspaper article
- Read/skim the article and examine the graph
- What is the message?
  - Summarize in 25 words or less.

The following images are taken from various newspaper articles or press releases. Look at the graph and read/skim the article.

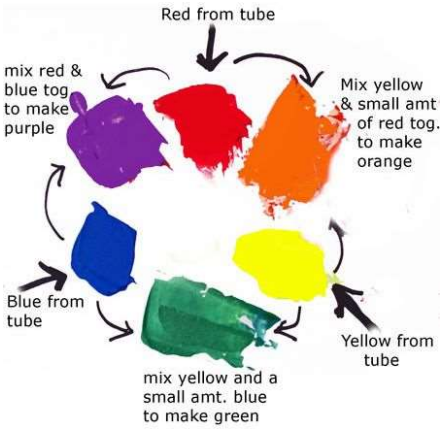
What message do you think the journalist is trying to convey with this graph. Summarize this message in 25 words or less.

Colors, introductory tutorial

- Kindergarten view of colors
- Review hexadecimal codes
- Color systems
  - RGB
  - HSV
  - HCL
  - CMYK

There are four major color systems used on computers, RGB, HSV, HCL, and CYMK. I'll discuss RGB in detail, and touch briefly on the other systems. But you need to realize that how colors work on a computer was not the way you learned in Kindergarten. A solid understanding of the RGB color system also requires you to work comfortably with hexadecimal codes, so I have a brief review of that as well.

Colors, Everything I know about colors, I learned in Kindergarten.



It was probably in Kindergarten where you learned the basic way to combine primary colors. Yellow plus red equals orange, Yellow plus blue equals green. Red plus blue equals purple/violet.

It doesn't work that way on a computer screen because screens use light to create colors and lights blend in different ways than paints or crayons.

Before you tackle ths computer system for colors, you need to review binary and hexadecimal number systems.

Colors, Binary codes

- "There are 10 types of programmers in the world, those who undestand binary and those who don't."
  - 1001 1100 (base 2)
  - $= 1 * 2^7 + 0 * 2^6 + 0 * 2^5 + 1 * 2^4 + 1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 0 * 2^0$
  - = 128+16+8+4 = 156
- Eight binary digits represent the numbers 0-255

Let me share a bad joke about binary numbers. Actually, you need to read it because the digits "1"-"0" represent something entirely different in the world of binary numbers.

Here's an example of how to decode the binary number 1001 1100.

Binary numbers use powers of two: 1, 2, 4, 8, 16, etc. Eight binary digits can represent any number from 0 to 255.

Colors, Hexadecimal codes

- Hexadecimal digits (base 16)
  - 0-9, A=10, B=11, C=12, D=13, E=14, F=15
  - 1001 1100 (base 2)
  - = 9C (base 16)
  - = $9 * 16^1 + 12 * 16^0$
- Two hexadecimal digits represent the numbers 0-255.
  - 00 (base 16) = 0, FF (base 16) = 255
- A # prefix implies hexadecimal in most computer languages.

Binary representations get unwieldy very quickly, and the hexadecimal format provides a much tighter representation that still has the spirit of binary. It is fairly easy to switch back and forth between binary and hexadecimal.

Hexadecimal means powers of 16 and you can represent numbers up to (but not including) 16 using four binary digits. The hexadecimal digits start out like normal digits, 0, 1, 2, etc. but once you get up to nine you run out of single digits. So the value of 10 is represented by the letter A, the value of 11 by B, etc. through the letter F which represents 15. At sixteen, you roll over to the next place.

So you count in hexadecimal like 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F, 10, 11, …

Split the eight digit binary number 1001 1100 into groups of four binary digits. The binary 1001 is equal to 9 and the binary 1100 is equal to 12 or C. So the hexadecimal equivalent is 9C. You can represent this using powers of 16, 9 times 16 to the first power plus twelve times 16 to the zeroth power. That works out to 156.

Two hexadecimal digits, just like eight binary digits, can represent the numbers 0 to 255, with 00 hexadecimal equaling 0 and FF hexadecimal equaling 255 (15*16+15).

Use the prefix of a pound sign (hash tag) to tell the computer that you are speaking to it in hexadecimal.

Colors, Codes for colors

- #rrggbb format
  - #000000 is pure black
  - #FFFFFF is pure white
  - #FF0000 is pure red
  - #00FF00 is pure green
  - #0000FF is pure blue
- You can mix and match to get 16,777,216 colors
  - #800080 is purple, #FF69B4 is pink, #40E0D0 is turquoise

The RGB format uses six hexadecimal digits to represent colors. A hexidecimal of all zeros is pure black and at the other extreme, a hexidecimal of all F's is pure white.

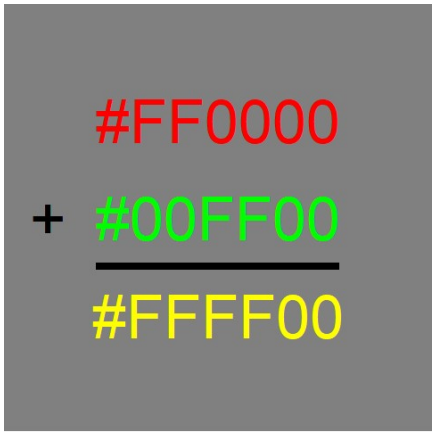
The first two hexidecimal digits represent the red channel. The highest value FF for the red channel combined with zeros for the other two channels (#FF0000) equals pure red.

The next two digits represent the green channel. #00FF00, giving the maximum to the green channel and the minimum to the other two channels produces a pure green.

The last two digits represent the blue channel, and #0000FF represents pure blue.

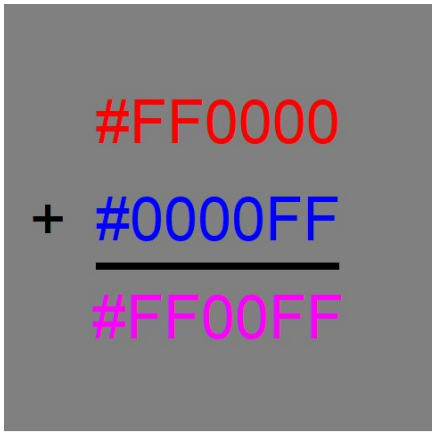
You can combine these in a variety of ways. You end up with an almost unlimited number of colors. Six hexadecimal digits allow you to produce 16^6 or 16,777,216 different colors.

Colors, Red plus green equals yellow



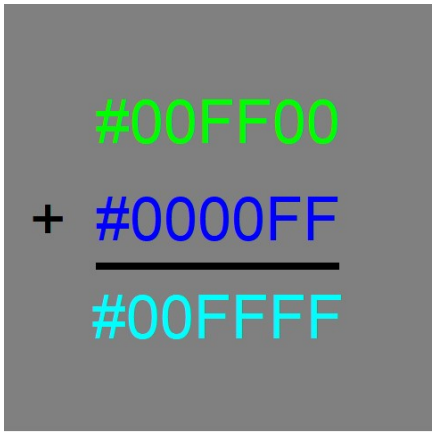
Whne you combine colors in the RGB system, they become lighter in color. So if you add red light (FF in the red channel) to green light (FF in the green chanel), you get yellow, which is FF in both the red and green channels.

Colors, Red plus blue equals magenta

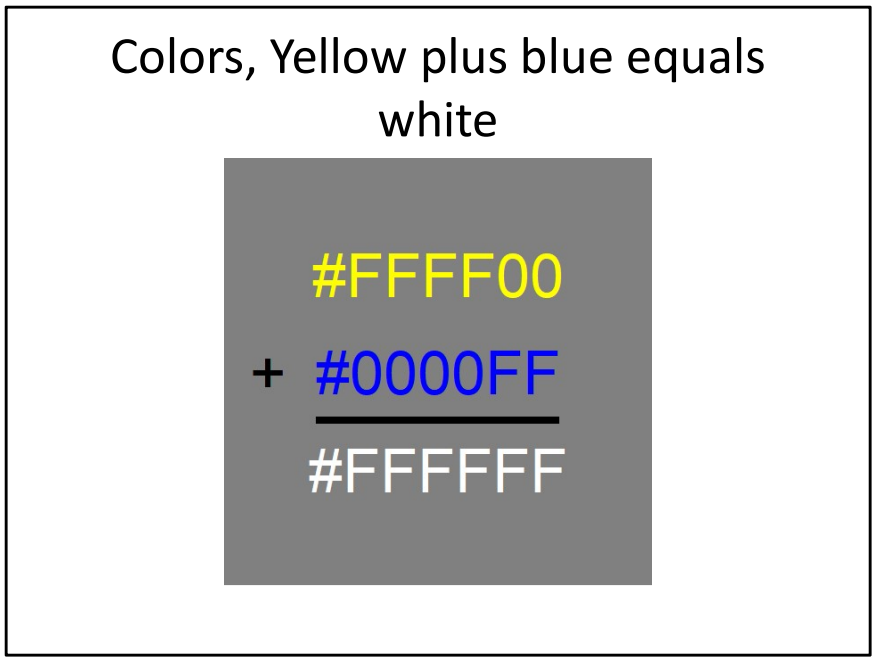


Red plus blue gives you #FF00FF, which is magenta, a light purplish red.

Colors, Green plus blue equals cyan



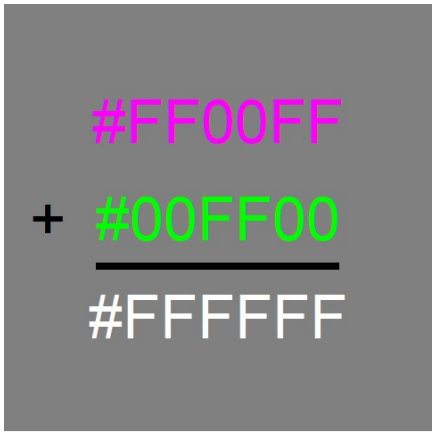
Green plus blue gives you #00FFFF, which is cyan, a greenish blue color.



Yellow has FF in the red and green channels and 00 in the blue chanel. If you combine yellow with blue, you fill up all three channels and produce white.

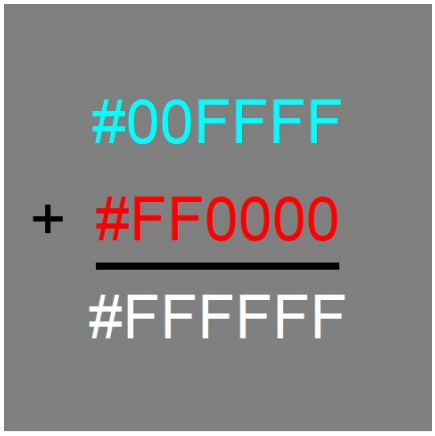
Notice that this is not at all what happens with crayons or paints. In those settings, yellow plus blue gives you green.

Colors, Magenta plus green equals white



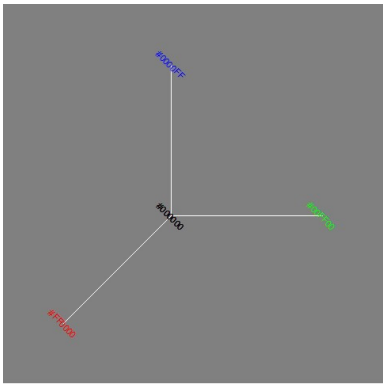
Similarly, magenta has FF in the red and blue channles. Add green to fill in the last channel and you get white.

Colors, Cyan plus red equals white



Finally, cyan has FF in the green and blue channels. Adding red to fill the last channel produces white.

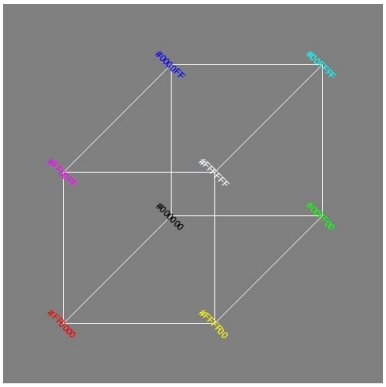
Colors, The color cube (1/2)



The three axes of the color cube

The color cube shows the three dimensions (red, green, and blue) of the RGB color system as axes in a three dimensional space.

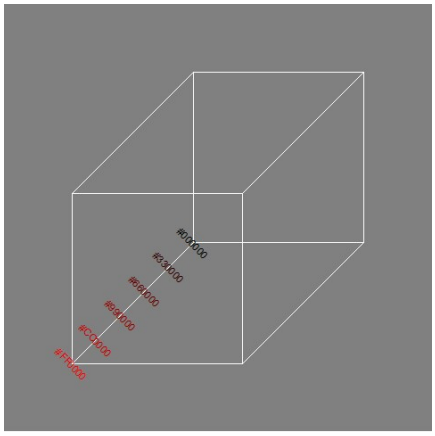
Colors, The color cube (2/2)



The full color cube with a front white vertex

The combination colors of yellow, magenta, cyan, and white represent different vertices of the color cube.

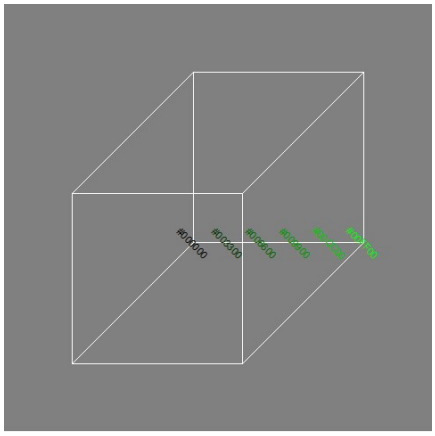
Colors, Gradients of black to red



You can develop gradients, gradual and continuous changes in color, by varying the channels from lowest to highest. If you let the red channel range from 00 to FF and keep the other channels at zero, you get a gradient from black through dark shades of red to pure red.

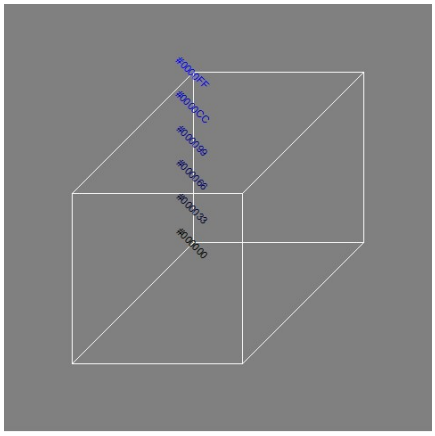
The figure only shows six values for the gradient (red channels ranging from 00 to 33, to 66 to 99 to CC), but you can produce 255 different shades between black and red.

Colors, Gradients of black to green



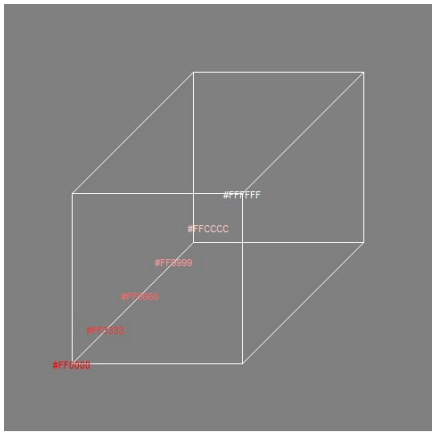
Similarly, if you let the green channel vary from 00 to FF and hold the red and blue channels at zero, you get a gradient that goes from black through dark shades of green to pure green.

Colors, Gradients of black to blue



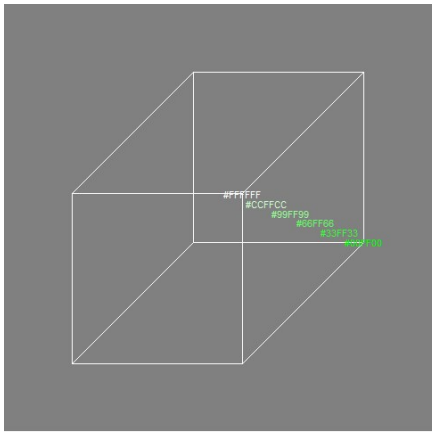
Varying the blue channel from 00 to FF while keeping the red and green channels at zero will produce a gradient of black through dark shades of blue to pure blue.

Colors, Gradients of red to white



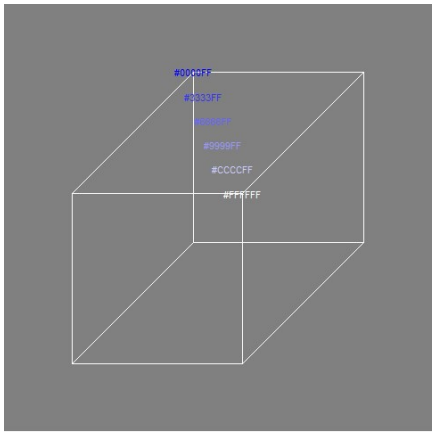
To create a gradient from red through the lighter shades of red (pinkish, actually) to white, keep the red channel at full strength (FF) and vary the green and blue channels from 00 to FF.

Colors, Gradient of green to white



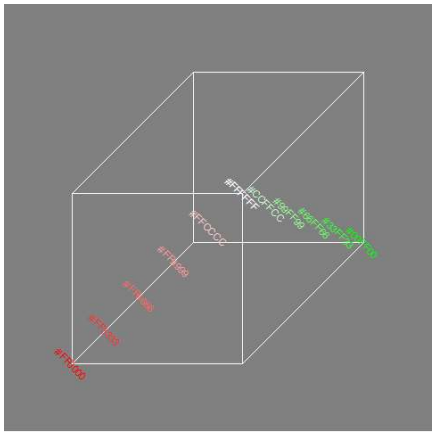
Similary, you can create a gradient from pure green through the lighter shades of green to white.

Colors, Gradient of blue to white



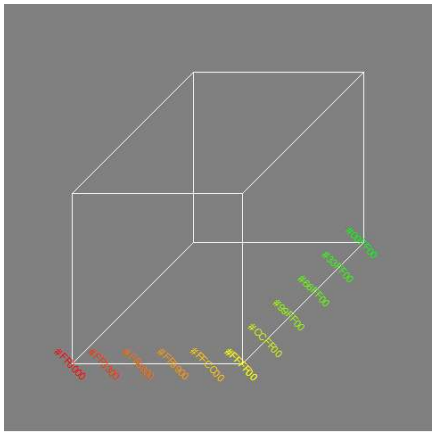
And just as easily you can develop a gradient that goes from pure blue through the lighter shades of blue to white.

Colors, Gradients of red to green via white



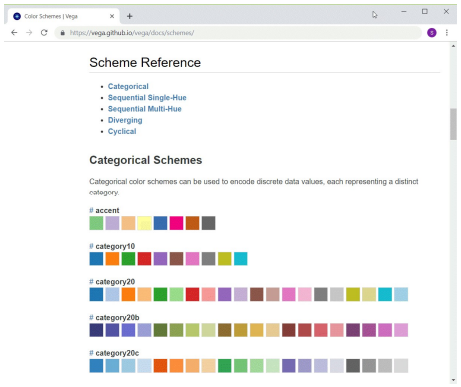
If you want to go from one pure color to another, there is more than one route that works. You can run red to white and then white to green.

Colors, Gradients of red to green via yellow



A common gradient for red to green relies on the fact that yellow is a combination of red and green. So start with red at full blast (FF), green at zero (and blue at zero). Move towards yellow by mixing in a bit more green. Once you get to pure yellow (FF in the green and red channels), reverse this process by keeping the green channel at full blast (FF) and gradually reduce the red channel to zero.

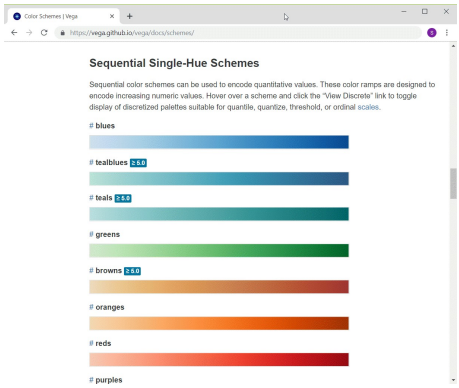
Colors, Python categorical schemes



Listing of first few Altair color schemes

If you want to use colors to represent categories, you want every color value to be readily distinguishable from every other color value. Python/Altair offers some pretty good choices if you have 10 or less levels for your categories. If you have more than 10 categories, you have to live with some colors being only slightly different than the others.

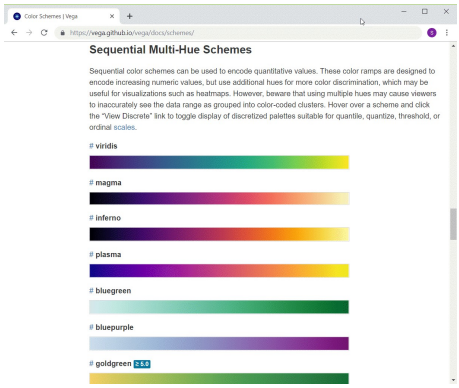
Colors, Python single hue schemes



Listing of first few Altair color schemes

Python/Altair has several single hue gradients that go from a very light to very dark in a variety of hues.

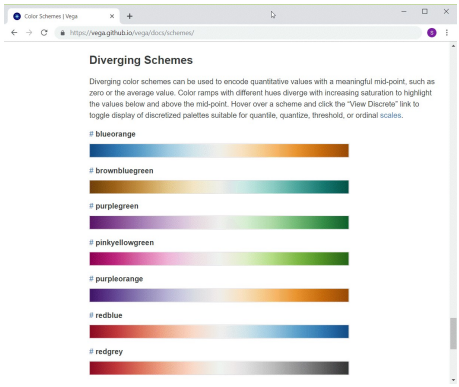
Colors, Python multi-hue schemes



Listing of first few Altair color schemes

The multi-hue color schemes in Python/Altair transition from one hue to another hue. The colors are more or less equally intense throughout the entire range.

Colors, Python diverging schemes



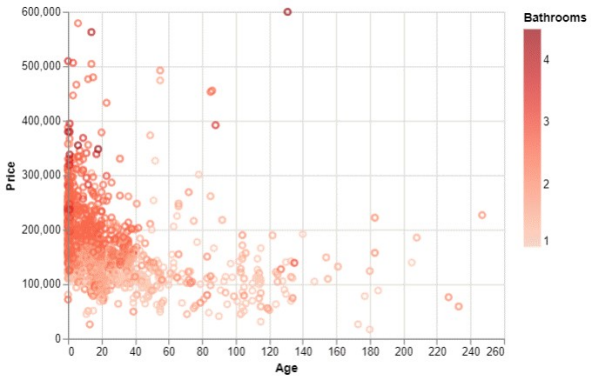
Listing of first few Altair color schemes

The diverging schemes also transition from one hue to another but in the middle the colors lose all their intensity, fading away to white or gray.

Colors, Python code to change color schemes

```
ch = alt.Chart(df).mark_point().encode(
    alt.Color('Bathrooms',
scale=alt.Scale(scheme='reds')),
    x='Age',
    y='Price'
)
```

Colors, Python output



Python scatterplot using red gradient

Colors, R qualitative schemes

- **Qualitative palettes** employ different hues to create visual differences between classes. These palettes are suggested for nominal or categorical data sets.

Display of various discrete color schemes

You can get color schemes from various sources in R. One of the best and most popular is called Color Brewer. It has color schemes in three areas: qualitative, sequential, and diverging. Here are the qualitative color schemes.

Notice that R does not try to present a categorical color scheme with more than 12 values. These schemes do a fairly good job of keeping each of the colors reasonably different from the others.

Colors, R sequential schemes

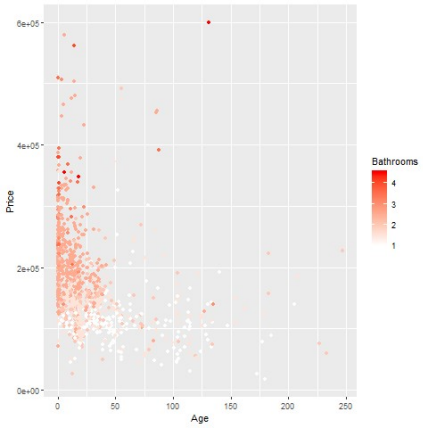
- **Sequential palettes** progress from light to dark. When used with interval data, light colors represent low data values and dark colors represent high data values.

Display of various sequential color schemes

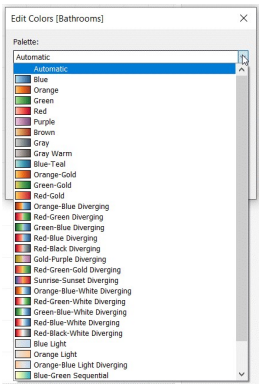
Colors, R code for changing color schemes

```
ggplot(saratoga_houses, aes(Age, Price)) +
  geom_point(aes(color=Bathrooms)) +
  scale_color_gradient(low="#FFFFFF",
high="#FF0000")
```

Colors, R output



Colors, Tableau gradient schemes

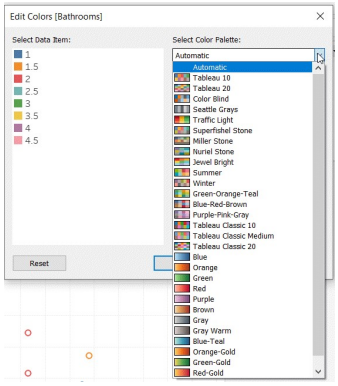


Gradient drop-down menu in Tableau

Tableau has quite a few choices for a continuous color gradient, but they are not as neatly categorized as the ones in Python/Altair and R.

https://help.tableau.com/current/pro/desktop/en-us/viewparts_marks_markproperties_color.htm

Discrete color options in Tableau



Discrete color drop-down menu in Tableau

The discrete choices (for representing categories using color) are also quite diverse, but they seem hard to group into any pattern.

Colors, Tableau output

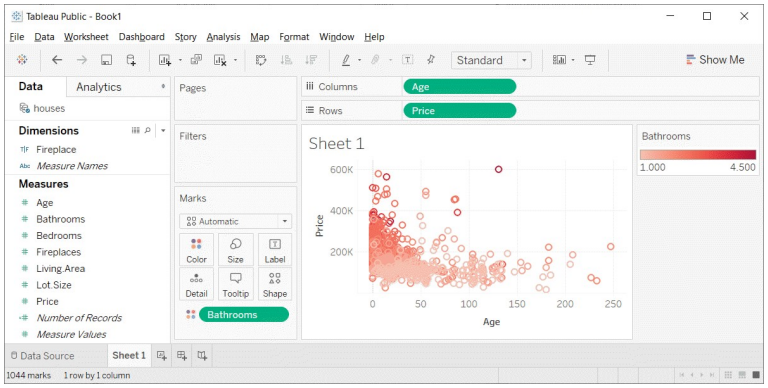


Tableau scatterplot with red gradient

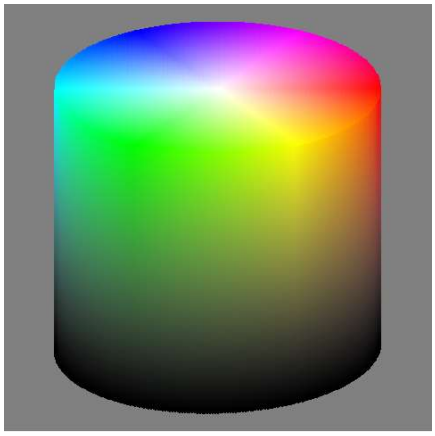
Here's what the Tableau output looks like.

Colors, HSV system

- H = Hue
  - Color wheel
  - Red, yellow, green, cyan, blue, and magenta
- Saturation
  - How colorful
  - Low saturation produces white or various shades of gray
- Value
  - How dark
  - Low values produce dark color
  - 0 produces black

The HSV system has three parameters, hue, saturation, and value. Hue is a color wheel including red, yellow, green, cyan, blue, and magenta, or some mixture of these colors. Saturation is a measure of intensity or colorfulness. Unsaturated colors produce white or various shades of gray. Value is a measure of darkness and a value of zero is totally black.

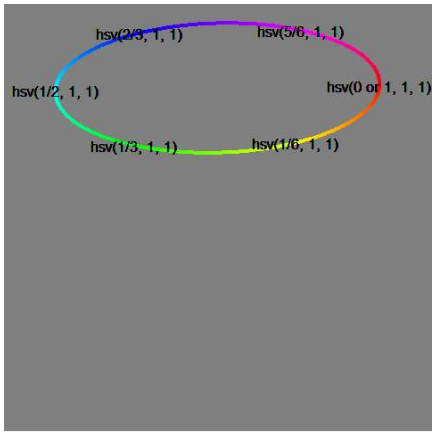
Colors, HSV system



This image shows a cylinder. It is a geometric representation of the HSV color system.

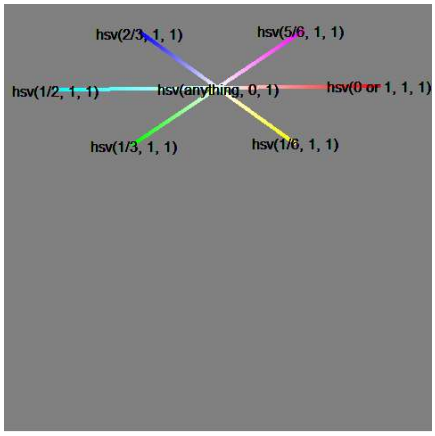
The advantage of the cylindrical HSV system over the RGB cube is that you can more readily visualize gradients, gradual transitions from one color to another.

Colors, HSV system, hue only



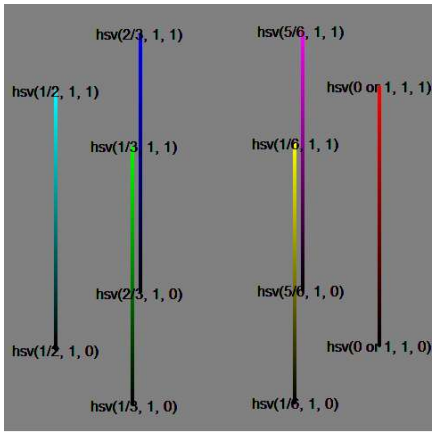
The outside ring at the top of the cylinder gives the hue. The hue is arranged in a circle from 0 to 1. The colors cycle between red on the far right, and (moving clockwise) yellow, green, cyan, blue, magenta, and back to red again.

Colors, HSV system, saturation only



As you move into the center of the cylinder, the color gradually gets sucked away. The degree to which a color is present or not is the saturation. At the very center of the cylinder, you get white…but only at the top of the cylinder. With darker colors, reducing saturation produces various shades of gray. At the very bottom, everything is black.

Colors, HSV system, value only



The darkness of a color is known as its value, which is the V in the HSV acronym. The top of the cylinder shows the pure colors, and heading straight down provides a continuous gradient from that pure color to pure black.

Colors, HCL system

- H = hue
  - arranged on a wheel 0-360 degrees
  - 0 = red, 120 = green, 240 = blue
  - 60 = yellow, 180 = cyan, 240 = magenta
- C = chroma
  - colorfulness relative to a gray of equal luminance
  - not quite same as saturation
- L = luminance
  - brightness, lightness
- Not all combinations of HCL work

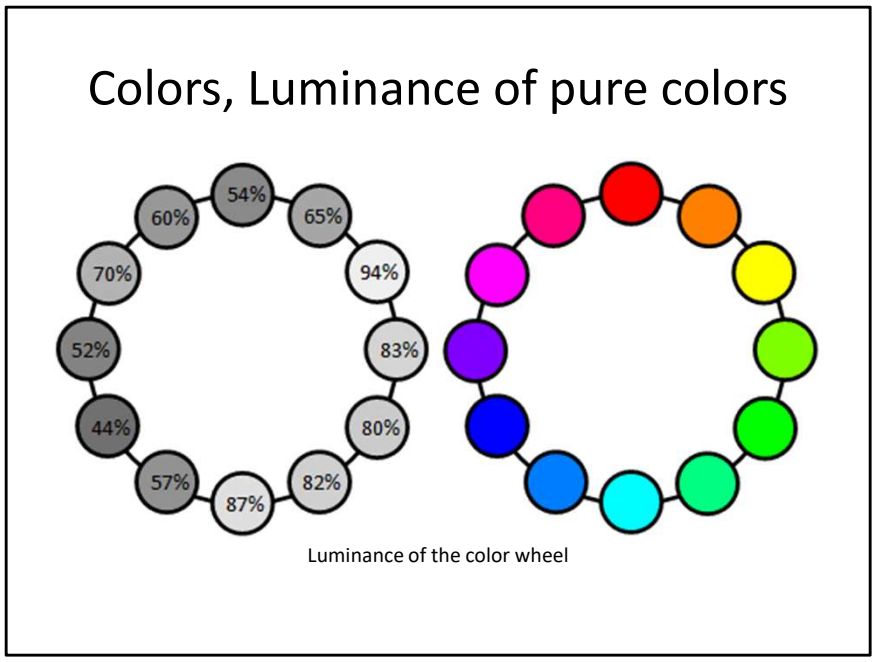
Another color system is the HCL system.

It has a hue value, just like the hsl system.

Chroma is a measure of how intense the color is. It is like saturation, but represents a relative amount of saturation, relative to a grey value of the same luminance.

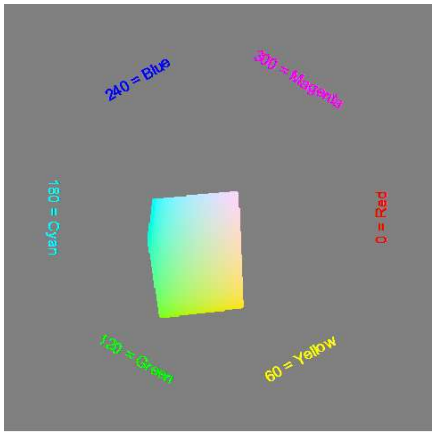
Luminance is a meausre of brightness or lightness. The luminance is similar to the V (value) in the hsv system. It does a better job of matching brightness, because certain colors, such as yellow and green have a naturally higher level of brightness.

For a given value of luminance, there may be restrictions in how far you can move towards the pure colors. For high values of luminance, you can't get near the pure red or pure blue colors, as they don't have enough brightness, especially compared to the yellows and greens. For values of low luminance, the opposite occurs. Pure yellow and pure green are just too bright, but you can get reasonable values for something close to pure red and pure blue.



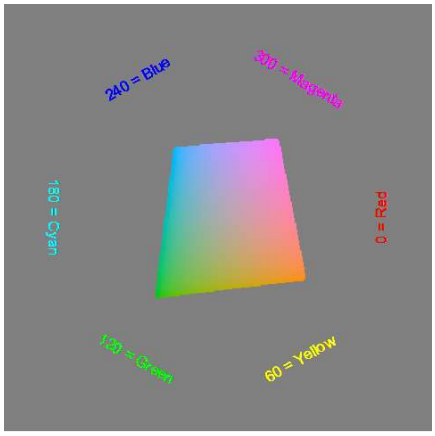
This image, borrowed from the excellent website, workwithcolor.com, shows that the pure colors are not equally bright. Yellow is the brigthest color. At 94% luminance, it is almost white. Cyan, at 87% luminance, and green, at 80% luminance, are almost as bright. The remaining colors have much less luminance. Magenta has 70% luminance, red has 54% luminance, and blue, at 44% luminance, is closer to black than it is to white.

Colors, Luminance = 90%



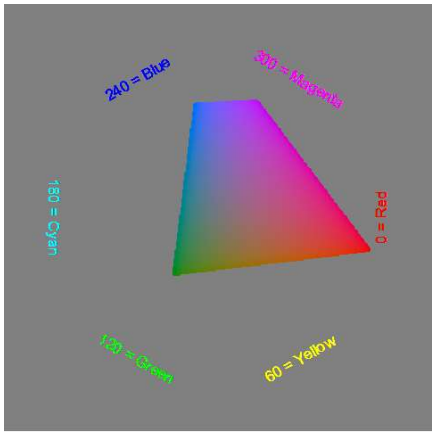
These are the color values with a luminance of 90%. It favors the green and yellow side of the color wheel.

Colors, Luminance=70%



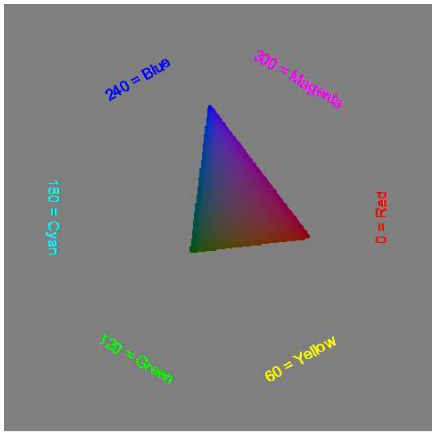
A lower value of luminance, like 70% shown here, will allow you to extend to some of the less bright dark colors like red and blue.

Colors, Luminance=50%



At a luminance value of 50%, you can no longer get very many shades of yellow or green, as they are too bright.

Colors, Luminance=30%

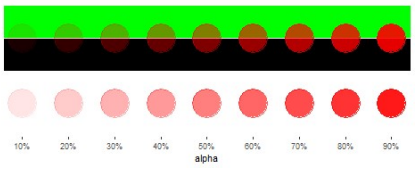


At 30% luminance, you are pretty dark and are left mostly on the blue-magenta-red hues.

Colors, Opacity (1/3)

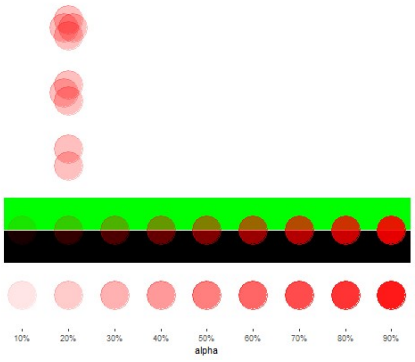
There is one more feature of color that is important, opacity. Opacity is the extent to which you can peek through a color to see the background beheath.

Colors, opacity (2/3)



A low degree of opacity does not impy a lighter color. It's only lighter when the background is white, because more of the white is showing through. When the background is black, a low degree of opacity means a dark red. When the background is green, a low degree of opacity produces a darker green with a tint of red.

Colors, Opacity (3/3)



If two points with high opacity overlap, then the overlap is more opaque. It becomes even more so when three or four points overlap.

Colors, CMYK system

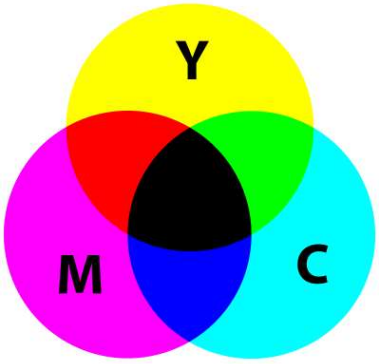


Illustration of subtractive color scheme

When you are printing, more of something makes things darker. This is the exact opposite of a computer monitor, where more of something makes things lighter. Also, printing is typically done on a sheet of white paper, so you don't need to put any ink down in areas that are pure white and you put very little ink down in areas that are nearly white.

Colors, CMYK system, combinations

- Building blocks Cyan (C), Magenta (M), Yellow (Y), Black (K).
  - Subtractive system
- Cyan plus Magenta equals Blue
- Cyan plus Yellow equals Green
- Magenta plus Yellow equals Red
- All three combined equals Black

The building blocks of the CMYK are the colors cyan, magenta, yellow and K=black. You don't want the letter B for black because it might get confused with the B for blue in an RGB system.

The combination of colors in a CMYK system works in a complementary way to RGB. The combinations of cyan and magenta produces blue, cyan plus yellow produces green, and magenta plus yellow produces red.

Colors, CMYK system, Why you need black

- Only in theory does, C+M+Y = Black
  - Too much ink
  - Dull muddy color
- On screen versus print

The CYMK color system adds a fourth color, black, to the mix. In theory, you don't need black, because you can get it with cyan plus magenta plus yellow. But that wastes a lot of expensive ink, and you can also run into trouble by putting some much in a single spot, especially when the image is very dark.

Also, in practice, the combination of cyan, magenta, and yeloow produces a dull muddy color that is not close enough to black. So adding a black ink helps. It also helps to mix in a bit of black ink for some of the other colors close to black on the color spectrum.

The translation of colors from an additive system like RGB to a subtractive system like CMYK is complicated and it is difficult to get something on the computer screen to match perfectly with what gets printed. If you work with printed color materials, you have to do a lot of extra work to get things to look right.

Colors, review

- Kindergarten view of colors
- RGB color system
  - Gradients
- HSV color system
  - Discrete color palettes
- HCL color system
  - Equal luminance
- Opacity
- CYMK system

The kindergarten view of colors (red plus yellow equals orange) doesn't work on a computer screen because the screen uses mixtures of light to combine colors. We covered the RGB system including how to create gradients. The HSV system provides a more intuitive description of colors, while CMYK is used when printing, where a subtractive rather than additive color system is needed.

Perception, introductory tutorial

- Visual tasks
  - Projection,
  - Superimposition,
  - Scanning,
  - Anchoring

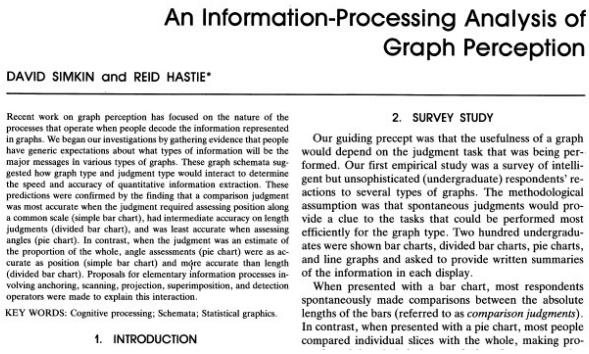
Understanding perception is the key to making effective visualizations. There are several key visual tasks: projection, superimposition, scanning, and anchoring. You want to design visualizations that facilitate these perceptual tasks.

Perception, introductory tutorial

- Hierarchy of comparisons
  - Position
  - Length
  - Angle
  - Area
  - Volume
  - Color

Comparisons are the heart of most visualizations. Comparisons follow a hierarchy from easy to difficult. Position is the easiest comparison. Does one bar in a bar chart reach higher than another? Other comparisons: length, angle, and area comparisons become increasingly more difficult. Comparisons involving volume (requiring estimation in a three dimentional space) and color (comparing the relative brightness of two colors) are the most difficult comparisons to make.

Perception, A framework for graph perception



An Information-Processing Analysis of Graph Perception

DAVID SIMKIN and REID HASTIE*

Recent work on graph perception has focused on the nature of the processes that operate when people decode the information represented in graphs. We began our investigations by gathering evidence that people have generic expectations about what types of information will be the major messages in various types of graphs. These graph schemata suggested how graph type and judgment type would interact to determine the speed and accuracy of quantitative information extraction. These predictions were confirmed by the finding that a comparison judgment was most accurate when the judgment required assessing position along a common scale (simple bar chart), had intermediate accuracy on length judgments (divided bar chart), and was least accurate when assessing angles (pie chart). In contrast, when the judgment was an estimate of the proportion of the whole, angle assessments (pie chart) were as accurate as position (simple bar chart) and more accurate than length (divided bar chart). Proposals for elementary information processes involving anchoring, scanning, projection, superimposition, and detection operators were made to explain this interaction.

KEY WORDS: Cognitive processing; Schemata; Statistical graphics.

1. INTRODUCTION

2. SURVEY STUDY

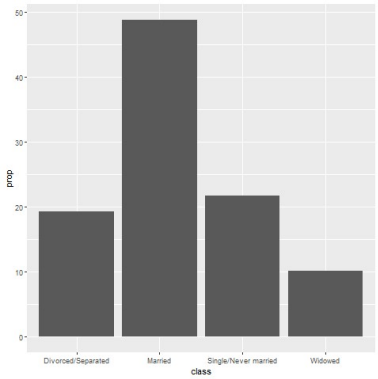
Our guiding precept was that the usefulness of a graph would depend on the judgment task that was being performed. Our first empirical study was a survey of intelligent but unsophisticated (undergraduate) respondents' reactions to several types of graphs. The methodological assumption was that spontaneous judgments would provide a clue to the tasks that could be performed most efficiently for the graph type. Two hundred undergraduates were shown bar charts, divided bar charts, pie charts, and line graphs and asked to provide written summaries of the information in each display.

When presented with a bar chart, most respondents spontaneously made comparisons between the absolute lengths of the bars (referred to as *comparison judgments*). In contrast, when presented with a pie chart, most people compared individual slices with the whole, making pro-

Part of first page from Simkin and Hastie journal article

Much of the work on the psychology of perception that I will be discussing next is drawn from this 1987 article by Simkin and Hastie.

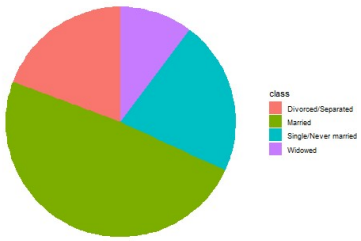
Perception, Which is better? A bar chart…



Bar chart

You have already seen this question, but it is worth examining in a bit more detail.

Perception, … or a pie chart



Pie chart

The pie chart gets very little respect, and the question is why? It can do reasonably well, depending on the question being asked. Being able to recognize when it does well and when it does not requires you to understand the psychology of perception.

Perception, Answer. It depends.

– What question are you trying to answer?
  - What proportion of the patients are single?
  - Are there more single or divorced patients?

The answer really depends on what question you are asking. There are a variety of questions that you might ask. Two are illustrated above.

You can run an experiment (people have done this) where randomize and show half of them a bar chart and half of them a pie chart. Then you ask a question, like one of the questions above. Then you note the speed and accuracy of the response. Depending on the question, sometimes pie charts give faster and more accurate answers. Sometimes bar charts give faster and more accurate answers. It turns out that the results match up nicely with what we know about the psychology of perception.

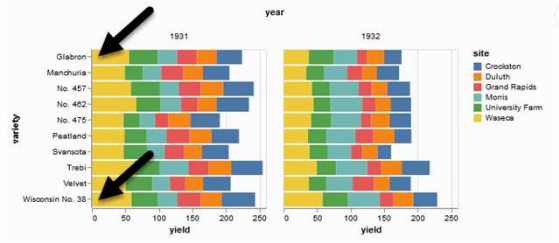
Perception, Visual processing (1 of 3)

- Projection
  - Shifting an object in a horizontal or vertical direction
- Superimposition
  - Shifting in other directions (e.g., diagonal shifts, rotation) in order to make a comparison
  - Much harder than projection
- Distance affects speed and accuracy of both projection and superimposition

Your eye will try to move objects like bars in a bar chart to try to make comparisons. If these movements are solely in a horizontal or vertical direction, you are making a projection. If these movements are in a diagonal direction or if it involves rotation, you are making a superimposition.

Projections are easier than superimpositions. Both are affected by distance. The further the distance that you have to project, for exapmle, the more time it takes and the less accurate you get, on average.

Perception, Projection (first yellow bar versus last yellow bar)

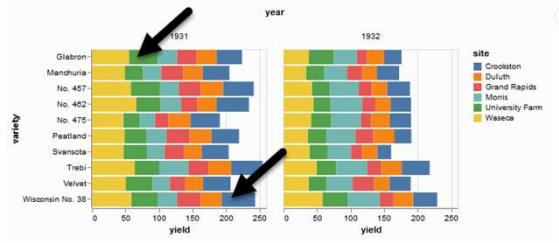


Stacked bar chart of crop yields

The position means the vertical or horizontal location. Does the first yellow bar in 1931 (Glabron seeds planted in Wasica) extend further to the right than the last yellow bar (Wisconsin No. 38 seeds planted in Wasica)?

This requires a projection. You slide the top yellow bar down to compare it to the bottom yellow bar.

Perception, Superimposition (first green bar versus last blue bar)



Stacked bar chart of crop yields

The length means either the width or the height. Does the first green bar in 1931 (Glabron seeds planted in University Farm) extend further to the right than the last yellow bar (Wisconsin No. 38 seeds planted in Crookston)?

This requires you to shift in a diagonal direction. This is a superimposition, a task that is harder than a projection. Expect that this comparison will take longer and be less accurate, on average.

Perception, Scanning and Anchoring

- Scanning
  - Quantifying distance throug the use of a mental tape measure
  - Shorter distances are easier
- Anchoring
  - Implicit or explicit development of reference points
  - Assists with scanning

Scanning is a visual assessment of distance. Think of it as a mental tape measure. Scanning is easiest for short distances. You are slower and less accurate when your mental tape measure has to cover a large distance.

Anchoring is the implicit or explicit development of reference points. They can assist with scanning by shortening the distance of your mental tape measure.

Perception, Scanning

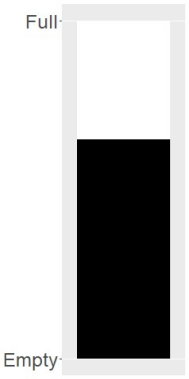


Image of a fuel gauge, 65% full

To understand scanning, think of a gas gauge. Usually it is a semicircular dial, but let's set up the gas gauge as a rectangle. If the level is at the top, you have a full tank. If the level is at the bottom, you have an empty tank. This gauge shows a tank that is 65% full. Trust me, I drew the gauge. It is at 65%. Now how would you estimate the gas level?

You would take a mental tape measure, starting at the bottom and measure up to where the black box ends.

Now if you were smart, you'd start at the top and scan downwards. Less distance means that you can do this faster and more accurately.

Now if you were Albert Einstein, you'd split the gauge at the halfway point and measure from the halfway point to the top of the black box. Actually, there's a little of Albert Einstein in all of us. That halfway point is something that all of us do subconciously. You did, because you recognized almost immediately that the tank was more than half full.

Perception, Assisting scanning with anchors

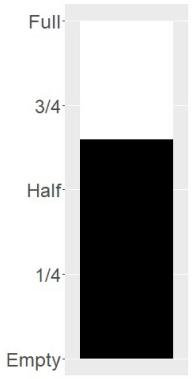


Image of a fuel gauge, anchors at 1/4, half, 3/4

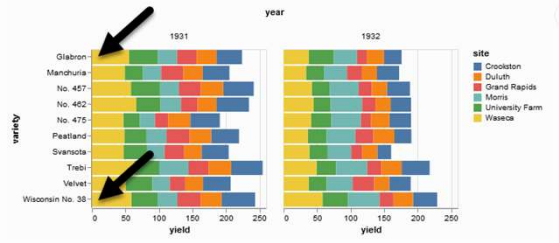
Here's the same gas gauge, still at 65% full, but now we have added anchors at 1/4, half, and 3/4. You can read this gauge faster and more accurately, because you can scane from half up to 65% or from 3/4 down to 65%.

Perception, Visual processing (3 of 3)

- Visually simple tasks
  - Position
  - Length
  - Angle/slope
- Visually demanding tasks
  - Area
  - Volume
  - Density/Saturation/Hue

There are a variety of perceptual tasks that you use when making comparisons within an image. These are arranged on this slide roughly in order of difficulty, with the easiest tasks at the top.

Perception, Position (first yellow bar versus last yellow bar)

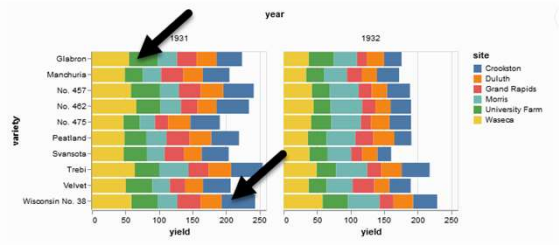


Stacked bar chart of crop yields

The comparison of the two yellow bars is a comparison of position. Which yellow bar extends further to the right?

We've seen this before, and described it as a projection. Projections are easy which explains why relative position is a simple visual comparison.

Perception, Length (first green bar versus last blue bar)



Stacked bar chart of crop yields

The comparison of the green and blue bars is a length comparison. The two bars start at different spots, so the position can't help you.

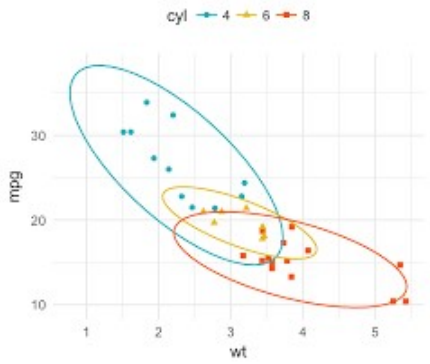
Length is harder to judge than position, because it involves a superimposition rather than a projection. It is still easy, however, compared to some other visual tasks.

Perception, Angle/slope (first month decline versus last month decline)

Sales trend shown with a line graph

This graph shows sales trends over a twelve month span. If you want to assess whether the first month decline (the dip in sales between January and February) was worse than the last month decline (the dip in sales between November and December), you would probably do this by judging the angle of the first line segment to the angle of the second line segment. This is not quite as easy as a position or length judgement, but it isn't too bad either.

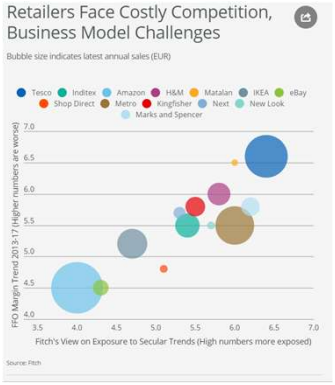
Perception, Area



Three ellipses

There are three ellipses here. Is the one above and to the left bigger than the one lower and off to the right? This is an area comparison. It's not too hard here because the one ellipse is markedly larger, but area judgements in general are more difficult than length judgements.

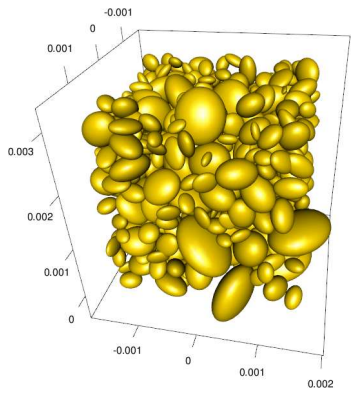
Perception, Area (or maybe length)



Bubble plot

This graph shows exposure and margin trends on the x and y axes. Larger values are worse on both axes. The size of the bubble is the size of the company (annual sales). There is some ambiguity here–is size measured in terms of diameter or in terms of area. Which is bigger, the brown circle (Metro) or the dark blue circle (Tesco)? Either way, it is more difficult than the relative position comparison.

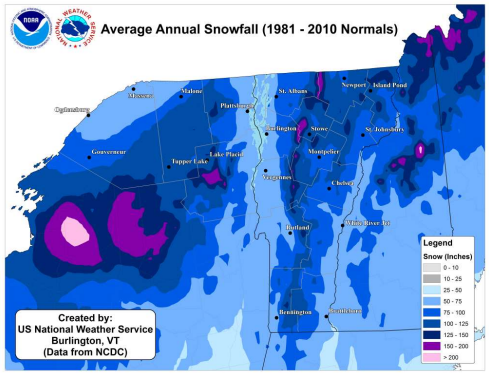
Perception, Volume



Ellipsoid plot

Volume comparisons are especially difficult. This graphic image, busy beyond belief, shows three dimensional ellipsoids of various orientations. Picking out the biggest among these ellipsoids is not easy, even if you restricted it to ellipsoids that are fully visible.

Perception, Color



Weather map of annual snowfall

What city gets more snowfall, Remington, in the southwester corner of the state, or Newport, in the northeast corner of the state? This is a comparison of color, more specifically the lightness or darkness of a color. This is a very difficult task, unless the colors have a marked variation. So Th pink areas of upstate New York are not to difficult to distinguish from the light and dark blue areas of New Hampshire and Vermont. But the comparison of Remington to Newport involves a region of the map that is mostly shades of blue.

Perception, summary

- Visual tasks
  - Projection, Superimposition, Scanning, Anchoring
- Hierarchy of comparisons
  - Position, Length, Angle, Area, Volume, Color

Your visualization should try to facilitate the visual tasks of projection, superimposition, scanning, and anchoring. If you want to make comparisons, they rank from easiest to hardest as comparisons of position, length, angle, area, volune, and color.

(NOte to myself) Another important visual task is “look up” or the ability to rapidly identify a particular component of a visualization. A table is the best tool for look up, but moving a legend to a more prominent location or replacing it with labels placed directly on the graphs can often help. Alphabetical lists in legends and in bar placements can also improve the speed and accuracy of lookup. The use of an accent color helps, if you know in advance what component your viewers are going to want to look up.

Fundamentals

- Fewer in numbers than points
- Usually a summary statistic
  - Count
  - Percent
  - Average
  - Total
- A bar chart is NOT a histogram

Bar charts are different than point charts. There are usually only a few bars. These bars usually represent a summary statistic, like a count, percent, average, or total.

There is a technical distinction between a bar chart and a histogram. Histograms are a great diagnostic tool, but usually ends up on the cutting room floor. So I won't be talking about it much, if at all, in this workshop.

Fundamentals, Aesthetics for bars

- Review
  - Location
  - Size
  - Color
  - NOT shape!!!

Recall that aesthetics are visual attributes associated with a geometry/mark. You can map variables to the location, size, and/or color of bars, but you cannot assign a variable to the shape of a bar.

There is a bit of potential ambiguity on bar charts. If you are looking at a vertical bar chart (the bar chart shown earlier is a vertical bar chart), then the location on the x-axis represents which category. For the vast majority of bar charts, the bars are anchored on the x-axis, though there are a few exceptions.

The location of the Y-axis, then, is the measure of the bar's height. You could, if you wanted to, call this the bar's size, but most people think of the bar's size as the width of the bar. For the vast majority of cases, the width of a bar is constant, but there are a few exceptions.

The color of a bar can refer to the interior of the bar or the border of the bar. Usually these two colors are the same, but there are some exceptions.

We won't talk about the exceptions in this lecture, but you are welcome to ask about them, if you are curious.

You cannot vary the shape of a bar. A bar is a bar.

Fundamentals, Basic barchart

– Python code

```
ch = alt.Chart(df).mark_bar().encode(
    x='pclass:N',
    y='count()'
```

– R code

```
ggplot(ti, aes(x=pclass)) +
  geom_bar()
```

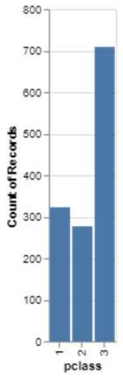
– Tableau

- Drag and drop

The Python/Altair code uses the mark_bar function instead of mark_point. You encode the passenger class on the x axis, but you have to remind Python/Altair that the values of 1, 2, and 3 represent categories of first class, second class, and third class. The y axis represents the count.

The code in R uses the geom_bar function. You can get away with specifying only a single location, the x-axis, because the default in geom_bar is to use the count as the location on the y-axis.

Fundamentals, Python output for basic barchart



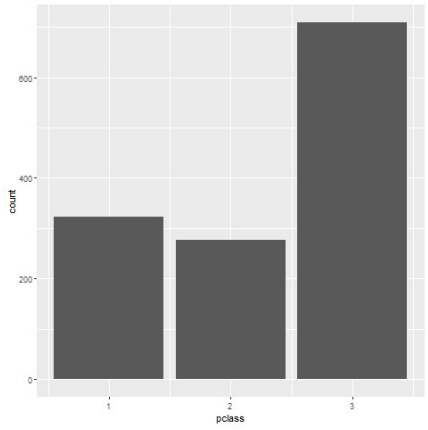
Barchart in Python

This is the Python version of the barchart.

This is data from the Titanic, a ship that was as massive as its name. It was considered unsinkable, but on its maiden voyage in 1912, the ship hit an iceberg and sank. It sank during an era where people really believed in women and children first, and this shows quite clearly in the mortality statistics. There was a difference, though, between third class passengers like Leonardo diCaprio and first class passengers like Kate Winslet.

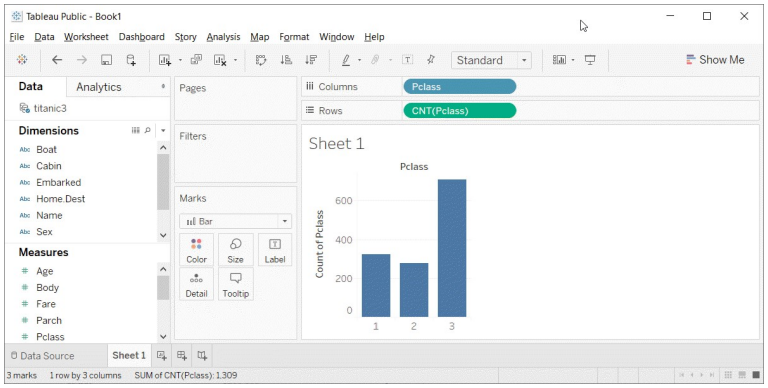
This shows a simple bar chart. The x-axis is passenger class and the y-axis is count. Note that there are about the same number of first and second class passengers, but the two combined does not come close to the number of third class passengers.

Fundamentals, R output for basic barchart



This is the R version of the barchart.

Fundamentals, Tableau output for basic barchart



Barchart in Tableau

Here is the Tableau output. Notice the columns (the x-axis) is a blue pill. That tells Tableau that passenger class is a discrete dimenion (or categorical). The rows (y-axis) is the count summary function.

Exercise, Gender barchart

– Draw a bar chart showing the number of people in each gender (use the Sex variable).

Now try to draw a different bar chart, one that shows the count for each Gender.

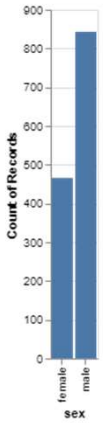
Exercise, Python code

– Here's the Python code

```
ch = alt.Chart(df).mark_bar().encode(
    x='sex',
    y='count()'
)
```

Here is the Python code. You insert sex and count into the encode function.

Exercise, Python output



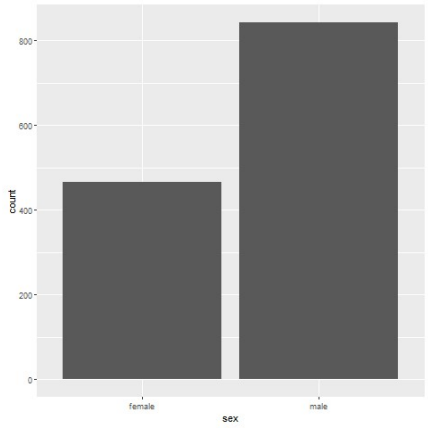
Here is the Python output.

Exercise, R code

– Here's the R code

```
ggplot(titanic, aes(sex)) +
  geom_bar()
```

Exercise, R output



This the R code. You put the sex variable inside the aesthetic (aes) function. The count is implied as the default.

Exercise, Tableau steps

– Drag Sex to the columns AND the rows.
  - Change Sex in the columns to Dimension, Discrete (blue pill)
  - Change Sex in the rows to Measure, Count.

You use drag and drop to create a gender barchart. Drag Sex to both the rows and the columns. In the columns, change it to a blue pill, representing a discrete dimension. In the rows, change it to a count summary.

Exercise, Tableau output

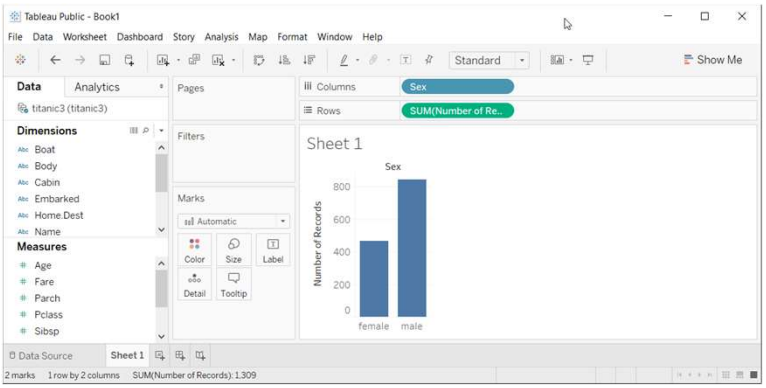


Tableau barchart

Here is the Tableau output.

Fundamentals, changing the default

– Python code

```
ch = alt.Chart(df).mark_bar(
    color='#FF0000',
    size=5
).encode(
    x='pclass:N',
    y='count()'
)
```

– R code

```
ggplot(titanic, aes(pclass)) +
  geom_bar(fill="#FF0000", width=0.5)
```

– Tableau

- Point and click

Here is the code to change the color and the size of all the bars. In Altair/Python, you place color and size arguments inside the mark_bar function. The size represents the number of pixels (not counting a one pixel border), so a value of 5 will produce a 7 pixel barrather narrow bar. The default in Python is to leave a 3 pixel gap between each bar.

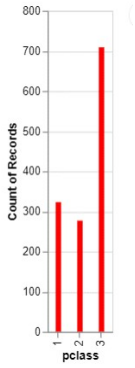
In R, the arguments are different. You specify the color of a bar with the fill argument. If you tried to use the color argument, it would change the color of the border, and leave the interior of the bar as the default color.

R also uses the width argument rather than size. This might not be the best choice, but width is less ambiguous than size, which could just as easily refer to the height as the width of the bar.

In R, width is a relative measure, and a value of 1 would mean that the bars fill up the entire space without any gaps. The value of 0.5 shown here will make the bars fill 50% of the space, making the bar widths equal to the width of the gaps. The default in R is 0.8.

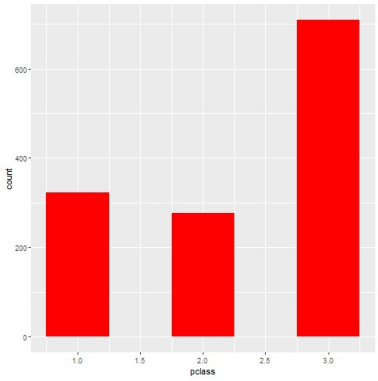
In Tableau, you click on the color button to change the color for every bar. You click on the size button to change the width of the bars. Slide all the way to the right to have the bars touching and all the way to the left to get a single pixel width bar.

Fundamentals, Python output for changing the default



Bar chart with red bars

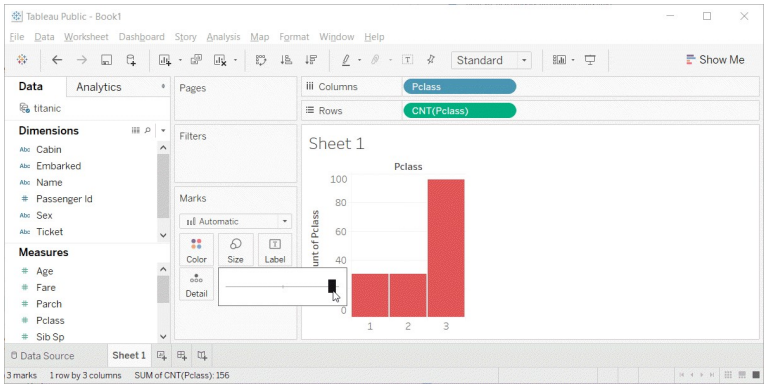
Fundamentals, R output for changing the default



Bar chart of passenger class count

Here is the output in R. Notice that the gaps between the bars are equal to the widths of the bars themselves.

Fundamentals, Tableau output for changing the default



Barchart with red bars

This is what Tableau produces. I am showing the slider bar that appears when you click on the size button. It is almost all the way to the right, so the bars are very fat and there is only a very tiny gap between the bars.

Fundamentals, Color

- Use for emphasis in simple bar charts
- Very important for stacked or side-by-side bar charts

For simple bar charts like all the ones we've seen so far, color is not really needed. You already can distinguish between the passenger classes using the location. If you do use color in a simple bar chart, it is often to emphasize a point. In the previous bar chart, I used red for third class, because Kate Winslet, a rich first class passenger, found true love in third class, with the adorable Leonardo di Caprio.

Color becomes very important in just a minute when we add another layer of complexity.

Fundamentals, Coding colors

– Python code

```
ch = alt.Chart(df).mark_bar().encode(
    alt.Color('pclass:N'),
    x='pclass:N',
    y='count()'
)
```

– R code

```
ggplot(titanic, aes(x=pclass)) +
  geom_bar(aes(fill=pclass))
```

– Tableau

- Point and click

If you want each bar to be a different color, then in Python, you have to include the color inside the encode function.

In R, you have to place the variable inside the aes function.

In Tableau, you have to drag and drop the variable on top of the color button.

Fundamentals, Choosing your own colors

– Python code

```
ch = alt.Chart(df).mark_bar().encode(
    alt.Color(
        'pclass:N',
        scale=alt.Scale(
            range=['#00FF00', '#FFFF00',
'#FF0000']
        )
    ),
    x='pclass:N',
    y='count()'
)
```

While the default colors chosen by your software are almost always good, you should experiment with different color combinations. A common choice for a three level ordinal variable is the traffic light colors of red, yellow, and green.

In Python, you add a scale argument to encode.

Fundamentals, Choosing your own colors

- R code

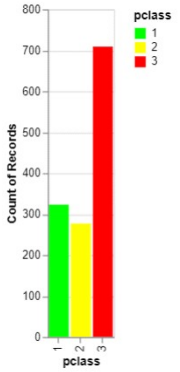
```
ggplot(titanic, aes(x=pclass)) +
  geom_bar(aes(fill=pclass)) +
  scale_fill_manual(values=c("#00FF00",
"#FFFF00", "#FF0000"))
```

- Tableau
  - Point and click

In R, you have to add another layer to the graph with the scale_fill_manual function.

In Tableau, you click on each color in the legend, then click in the upper corner to lock in that bar only. Then click on the color button to change that bar to the desired color.

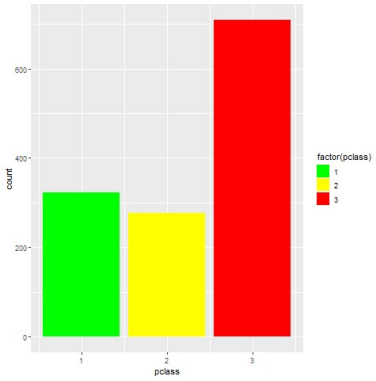
Fundamentals, Python output for coding colors



Bar chart with traffic light colors

This is what the Python bar chart looks like.

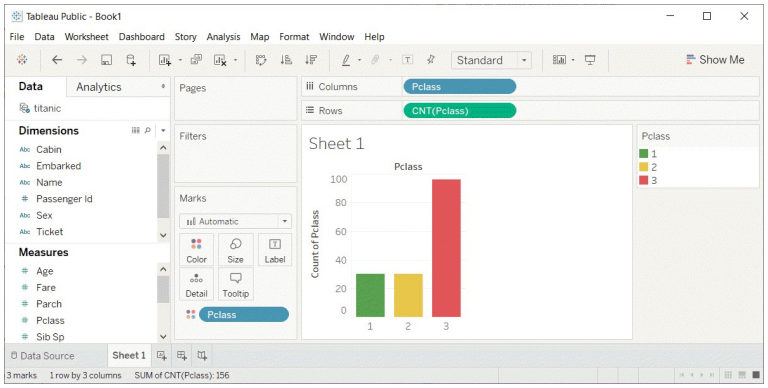
Fundamentals, R output for coding colors



Barchart with traffic light colors

This is what the R graph looks like.

Funadamentals, Tableau output for coding colors



Barchart with traffic light colors

Here is the Tableau output.

Exercise, First class red

– Make the first class bar red (#FF0000)
– Make the other two bars gray (#808080)

One way to emphasize a particular category is to make it a bright color and to set the other colors to a less prominent color like gray that lets them fade into the background. It's okay to use the same color for more than one category.

Make the first class category red and set the second and third class categories to gray.

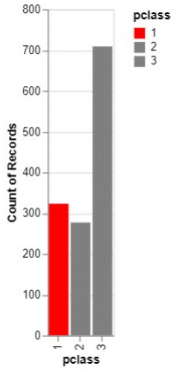
Exercise, Python code

– Here's the Python code

```
ch = alt.Chart(df).mark_bar().encode(
    alt.Color(
        'pclass:N',
        scale=alt.Scale(
            range=['#FF0000', '#808080',
'#808080']
        )
    ),
    x='pclass:N',
    y='count()'
)
```

Here is the Python code. You modify the colors with the Scale function.

Exercise, Python output



Bar chart with red accent color

Here is the Python output.

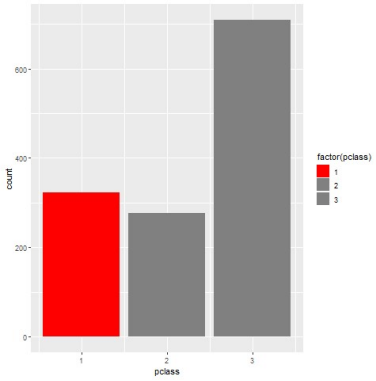
Exercise, R code

– Here's the R code

```
ggplot(titanic, aes(x=pclass)) +
  geom_bar(aes(fill=pclass)) +
  scale_fill_manual(values=c("#FF0000",
"#808080", "#808080"))
```

Here is the R code. The scale_fill_manual function allows you to specify your colors.

Exercise, R output



Barchart with red highlight

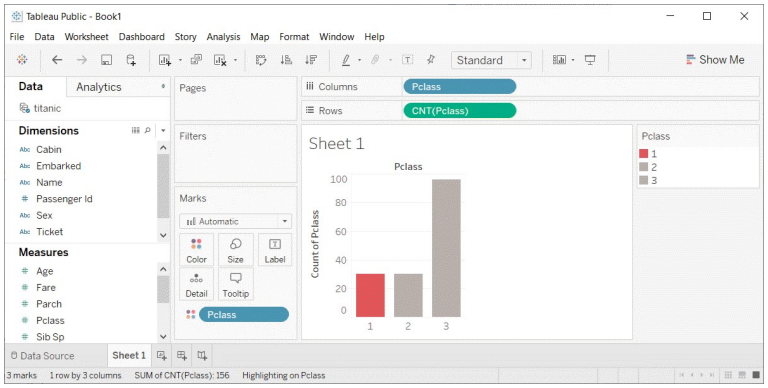
Here is the graph in R.

Exercise, Tableau steps

- Drag PClass to Columns and Rows
  - Change the PClass in Columns to Dimension, Discrete (blue pill)
  - Change the PClass in Rows to Summary, Count
- Drag PClass to the Color button
- Click on PClass 1
  - Then click on the upper right pen icon
  - Click on the color button
  - Choose a different color
- Repeat for PClass 2 and PClass 3

Here are the steps in Tableau. The tricky part is selecting just a single passenger class. It use the pen icon in the upper right corner.

Exercise, Tableau output



Barchart with red highlight

Here is the Tableau output.

Fundamentals, Stack, Dodge, Normalize

- Summarize by two categories
- Dodge
  - Side by side
- Stack
  - One on top, one on bottom
- Normalize
  - Stack and set full bar to 100%

Bar charts that represent a summary across a single categorical variable are fairly simple and easy to handle. But when you want to summarize by two categories simultaneously, things get interesting. Interesting in a good way.

You can present the extra category in side by side comparisons. This is called “dodge” in R, but is handled indirectly in Python and Tableau.

The stack option, where one category level is placed above another category level above another, is the default in all three packages, though you should take the trouble of specifying stack if you are using R.

The normalize option creates stacked bars, but then forces each stack to have a height of 1 (or 100%).

Fundamentals, code for stack

– Python code

```
ch = alt.Chart(df).mark_bar().encode(
    x='pclass:N',
    y='count()',
    color='sex'
)
```

– R code

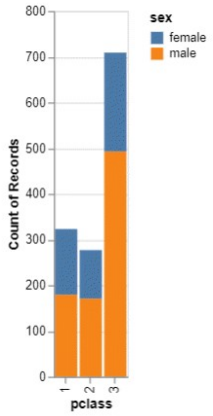
```
ggplot(titanic, aes(x=pclass, fill=sex)) +
  geom_bar(position="stack")
```

– Tableau steps

- Point and click

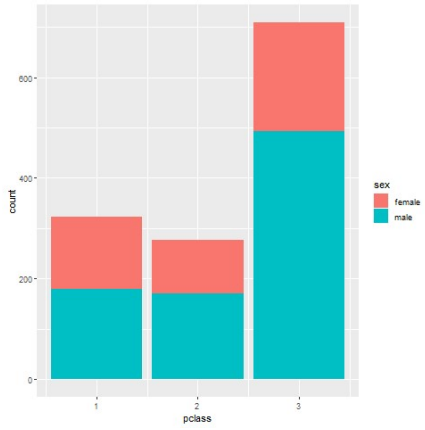
if you specify a third variable in the color argument, Python will, by default, create a stacked chart. In R, you have to specify position="stack" inside the geom_bar function.

Fundamentals, Python output for stacked bars



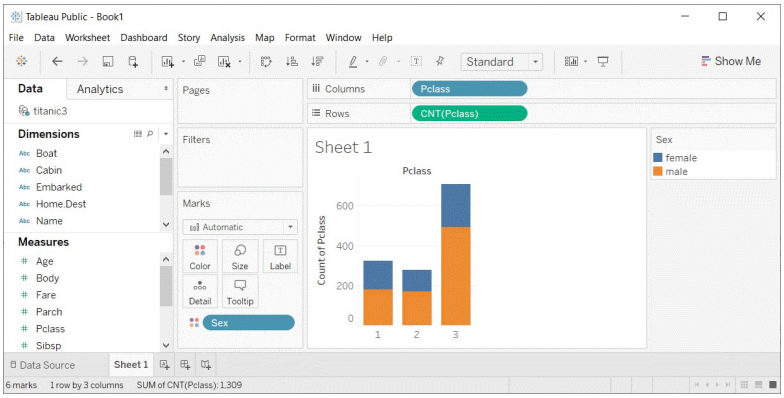
This is the Python output.

Fundamentals, R output for stacked bars



This is the R output.

Fundamentals, Tableau output for stacked bars



Here is the stacked barchart in Tableau. Drag pclass to the x-axis, count of pclass to the y-axis, and sex to the color button.

Fundamentals, code for dodge

– Python code

```
ch = alt.Chart(df).mark_bar().encode(
    x='sex',
    y='count()',
    color='sex',
    column='pclass:N'
```

– R code

```
ggplot(titanic, aes(x=pclass, fill=sex)) +
  geom_bar(position="dodge")
```

– Tableau

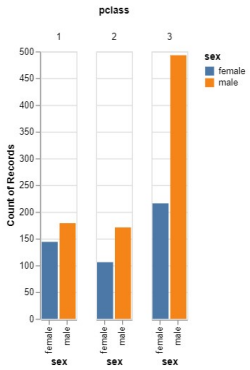
- Point and click

Here is the code in Python for dodged bars. You code sex as you x location and your color. Then you code the column argment as pclass.

In R, you change the argument inside the geom_bar function to position="dodge".

Tableau uses point and click, which I will describe below.

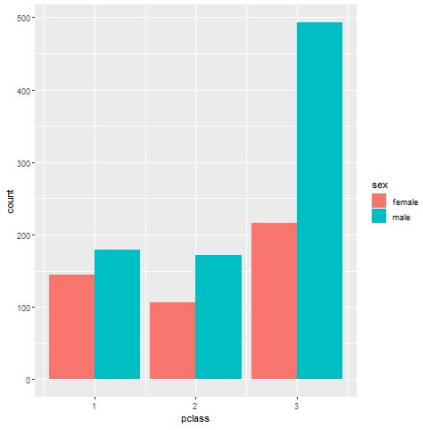
Fundamentals, Python output for dodged bars



Grouped barchart for passenger class and gender

Here is the dodged bar chart in Python.

Fundamentals, R output for dodged bars



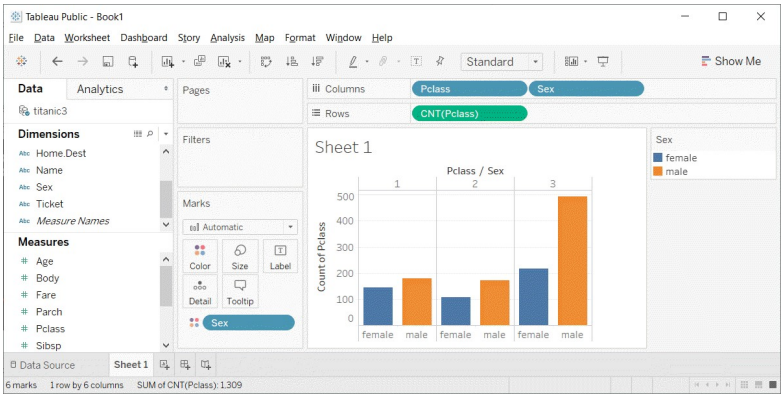
This is what the dodged bar chart looks like in R.

Fundamentals, Tableau steps for dodged bars

- Drag pclass to columns
  - Set to Dimension Discrete
- Drag pclass to columns
  - Set to measure, count.
- Drag sex to columns (to the right of pclass)
  - Set to Dimension Discrete
- Drag sex to color button
  - Set to Dimension Discrete

Here are the steps in Tableau for a dodged bar chart. You have to double up on the columns field.

Fundamentals, Tableau output for dodged bars



This is what the Tableau output looks like.

Fundamentals, Code for normalize

- Python code

```
ch = alt.Chart(df).mark_bar().encode(
    x='sex',
    y=alt.Y('count()', stack='normalize'),
    color='pclass:N'
)
```

- R code

```
ggplot(titanic, aes(survived)) +
  geom_bar(aes(fill=sex), position="fill")
```

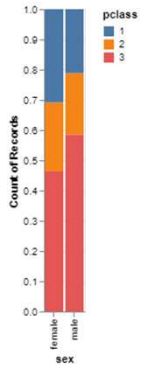
- Tableau
  - Drag and drop

If you want a stacked barchart in Altair/Python, you need to use the alt.Y function with the stack='normalize' argument.

In R, you use the position='fill' argument in the geom_bar function. Note that this argument falls outside of the aes function.

In Tableau, you right click on the count summary, and pick Add Table Calculation from the drop down menu. In the dialog box, select Percent of Total and Compute Using Cell.

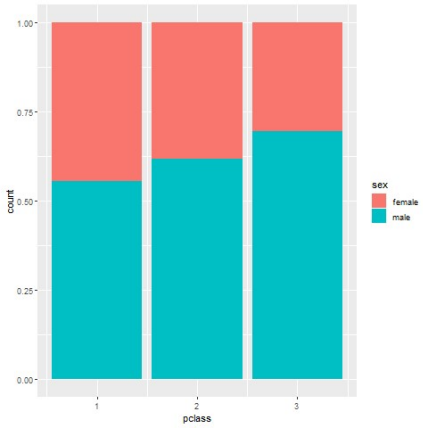
Fundamentals, Python output for normalized bars



Normalized barchart in Python

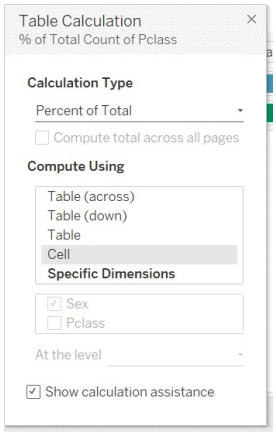
This is what the Python normalized barchart looks like.

Fundamentals, R output for normalized bars



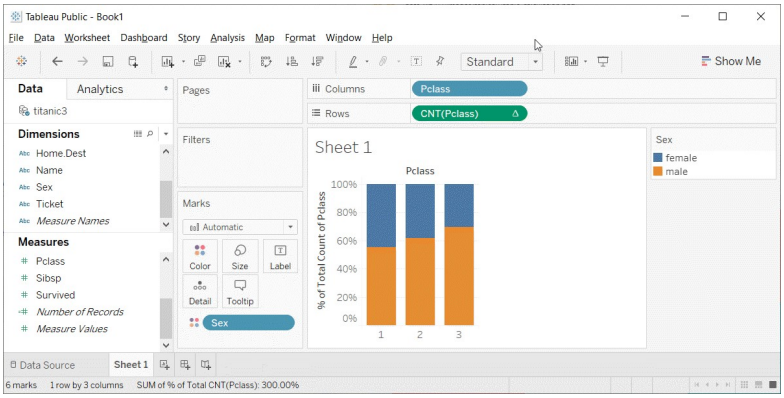
This is what the normalized bar chart looks like in R.

Fundamentals, Tableau output for normalized bars



In Tableau, you right click on the count summary, and pick Add Table Calculation from the drop down menu. In the dialog box, select Percent of Total and Compute Using Cell.

Fundamentals, Tableau output for normalized bars



This is what the visualization looks like in Tableau.

Exercise, Mortality by gender

– Draw a bar chart showing counts involving both mortality and gender. Use normalized bars.

Mortality on the Titanic reflects the fact that this was an era when people really did believe in the concept of “women and children first.” Someone like me, I’d be shoving the little kids aside so I could get on one of the lifeboats.

Anyway, examine how mortality is related to gender. It’s a spoiler alert, but on the Titanic, Kate survives, but Leonardo, I am so sad to say this, didn’t make it.

Exercise,Python code

– Python code

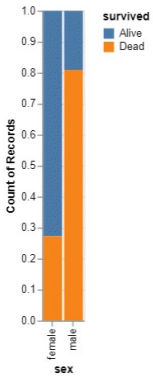
```
ch = alt.Chart(df).mark_bar().encode(
    x='sex',
    y=alt.Y('count()', stack='normalize'),
    color='survived'
)
```

– Alternative

```
ch = alt.Chart(df).mark_bar().encode(
    x='survived',
    y=alt.Y('count()', stack='normalize'),
    color='sex'
)
```

Here is the Python code. Notice that there are two different approaches.

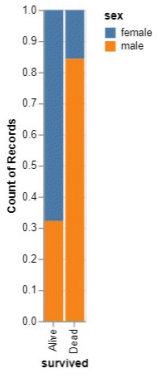
Exercise, Python output, Mortality by sex



Mortality by sex barchart

This is what the Python normalized barchart looks like.

Exercise, Python output, Sex by mortality



Sex by mortality barchart

This is what the alternate version of the Python normalized barchart.

Exercise, R code

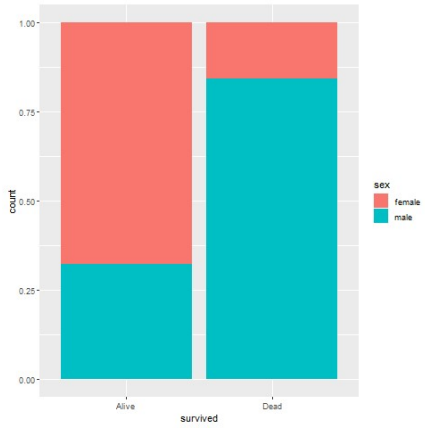
– R code

```
ggplot(titanic, aes(x=survived, fill=sex)) +
  geom_bar(position="normalize")
```

– Alternate version

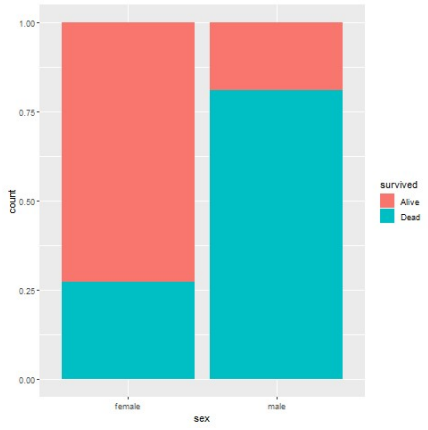
```
ggplot(titanic, aes(x=sex, fill=survived)) +
  geom_bar(position="normalize")
```

Exercise, R output, Mortality by sex



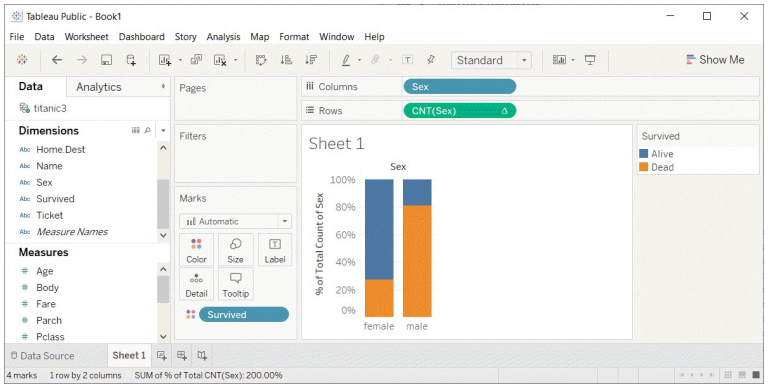
Here is the R output.

Exercise, R output, Sex by mortality



Here is the alternate version.

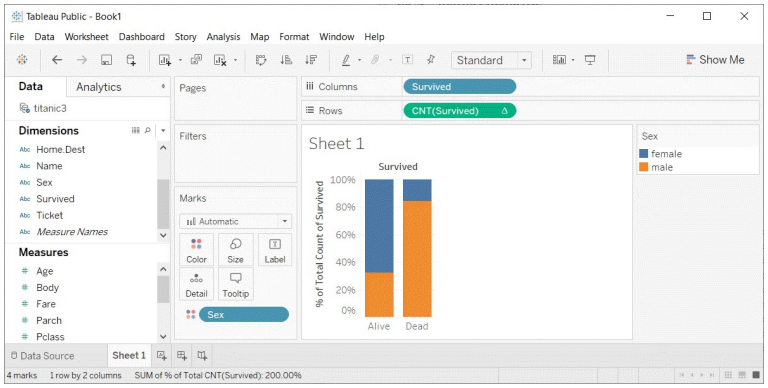
Exercise, Tableau output, mortality by sex



Mortality by sex barchart

Here is the Tableau output.

Exercise, Tableau output, sex by mortality



Sex by mortality barchart

Here is the alternate version.

Fundamentals, Summaries other than count

- Summaries vary by package
- All three include
  - Mean
  - Sum
  - Minimum
  - Maximum

There are more summaries that you can create bar charts for. The summary functions vary by package, but all of them allow you to compute a mean, sum, minimum value, and maximum value.

Fundamentals, Code for mean summary

– Python code

```
ch = alt.Chart(df).mark_bar().encode(
    x='sex',
    y='mean(age)'
)
```

– R code

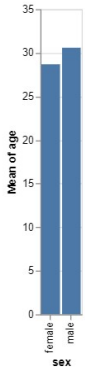
```
stat_summary(fun.y=mean, geom="bar")
```

– Tableau

- Choose Summary, Mean

In Python, you change the count function to the mean function. In R, you replace the geom_bar function with the stat_summary function. In Tableau, you change the variable from Measure, Count to Measure, Average.

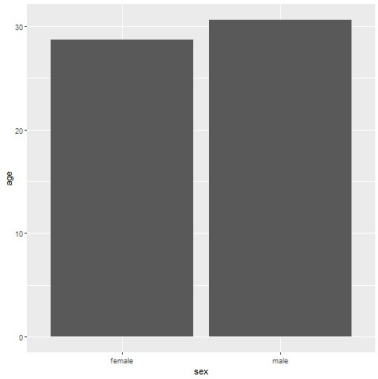
Fundamentals, Python output for mean bars



Barchart of mean ages by sex

Here is the Python output.

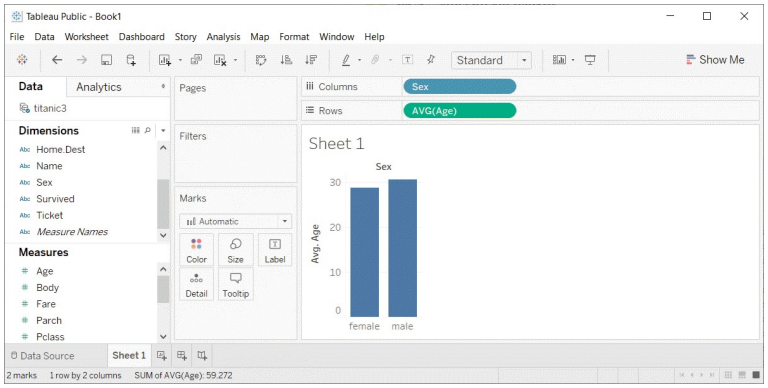
Fundamentals, Example with means



Barchart of mean ages by sex

Here is the R output.

Fundamentals, Tableau output of mean barchart



Barchart of mean ages by sex

Here is the Tableau output.

Fundamentals, Summary

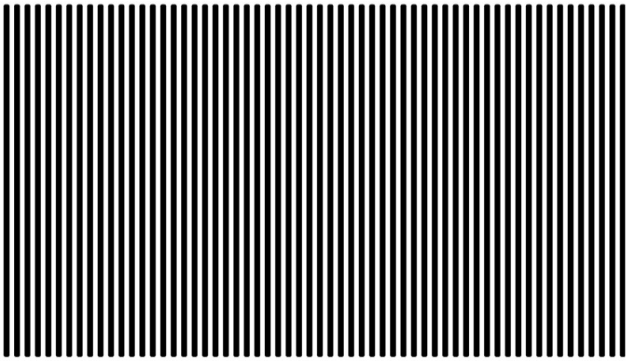
- "A mapping of data to the visual aesthetics of geometries/marks"
  - Bars are a type of geometry/mark
  - Aesthetics for bars include location, size, color
  - Stack versus dodge versus normalize

Recall the definition of data visualization, the mapping of data ro the visual aesthetics of geometries/marks. Bars are a type of geometry or mark and you can vary the location, size, and color, though size (meaning width) is usually held constant.

You can control the placement of different color bars using dodge, stack, or normalize. Note that the terminology here is not standardized across the different packages.

(Note to myself) If I have time, it might be nice to include error bars and boxplots at the end of this section.

Recommendations, Don’t make gaps equal to widths (1/2)

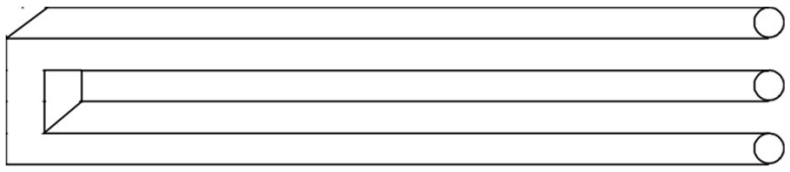


Alternate white and black lines showing a Moire effect

If the space between the bars is equal to the width of the bars themselves, you get an unsettling vibratory effect. This is because your eye is constantly shifting perspective. Sometimes it perceives the black as the foreground and the white as the background. Sometimes it perceives the white as the foreground and the black as the background.

“I’m ten years old, my life’s half over. And I don’t even know if I’m black with white stripes or white with black stripes.” Marty the Zebra in the movie Madagascar.

Recommendations, Don’t make gaps equal to widths (2/2)



Optical illusion showing two bars becoming three

If the widths are the same and the bars are empty, then you can get a differnt problem. You might get confused as to what is the bar and what is the gap, as in this optical illusion.

As a general rule, the gap between bars should be about 10 to 20 percent of the width of the bars. Most visualization software has sensible defaults, but beware when you have a very large number of bars. There’s always a bit of rounding when you place pixels on a screen or on a page. Your software is trying to fill the available plotting area, so it may have to squeeze or stretch one gap or another. This unevenness can become noticeable when the gaps are only a few pixels wide.

Barchart fundamentals, Thoughts on location

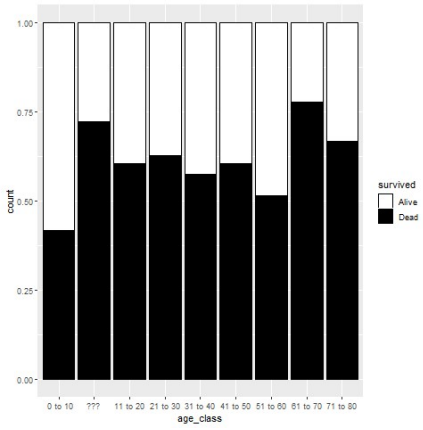
- Axis labels often fit better next to horizontal bars
- Bar charts with many bars
  - Vertical bars allow more room
  - Beware of rounding artefacts
- Do not cut off bar charts at the knees

The default for most visualization software is vertical bars, but you should give thoughtful consideration to horizontal bars. The labels often fit better when the bars are horizontal. You also often have more room left to right than you do up and down in a graph, so the bars can stretch out more, allowing you to more easily discern small and subtle differences.

If you have a very large number of bars, then a vertical format will allow those bars to fit better.

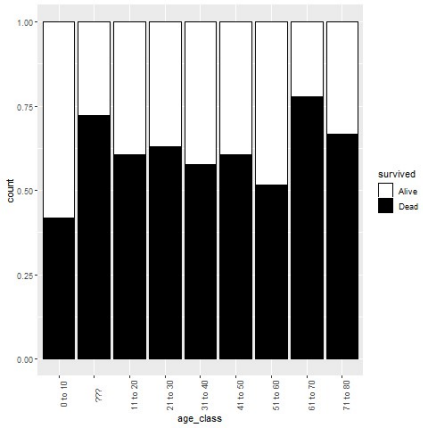
Remember the fault of default principle. Always try different ways of displaying your data. It costs nothing other than a few electrons to display a horizontal alternative to the typical vertical bar chart format, so why not indulge yourself?

Recommendations, Labels on a barchart (1/3)



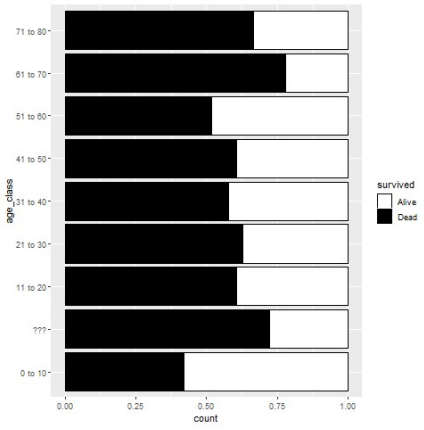
Labels will sometimes clash on a vertical bar chart.

Recommendations, Labels on a barchart (2/3)



You can turn the x-axis text by 90 degrees to make it fit better. But it is harder to read text that is turned 90 degrees.

Recommendations, Labels on a barchart (3/3)



A better solution is to turn the bars sideways. The labels fit better on a horizontal bar chart.

With a large number of bars and a limited number of pixels in your image, you might end up seeing some artefacts caused by rounding. There is always some rounding error in the placement of bars, but this is mostly imperceptible when you only have a few bars. What is a pixel here or there when your bars are a hundred pixels wide and the gaps between the bars are 10 pixels wide?

But a large number of bars will shrink the width of the bars and the size of the gaps to the point where having to add or remove a pixel becomes noticeable.

Here's an example of a barchart with 44 bars. Now for this particular example, there is no compelling reason to have 44 bars, but bear with me, as there are situations, now and then, where you really want that many bars.

I drew the image as 480 pixels by 480 pixels. After allowing for margins, it looks like there are about 340 pixels for the 66 bars. 44 does not divide evenly into 340. You get something like 7.73. If you allocate 7 pixels per bar, that uses up 308 pixels, leaving you with 32 pixels to divide among the 43 gaps between the 44 bars. So some of the gaps will be one pixel wide and some will be zero pixels wide.

Recommendations, Solutions to rounding artefacts

– Combine categories to reduce the number of bars.
– Increase the resolution of your plot
– Switch to a line graph

If you notice some rounding artefacts, you have several possible remedies.

First, if you can combine categories to reduce their numbers, this will help. This particular plot had very narrow age intervals, and if you could afford to widen the intervals to five years (0 to 4, 5 to 9, etc.) or even wider, that would help. Sometimes you can combine a large number of infrequently occuring categories into an “other” category.

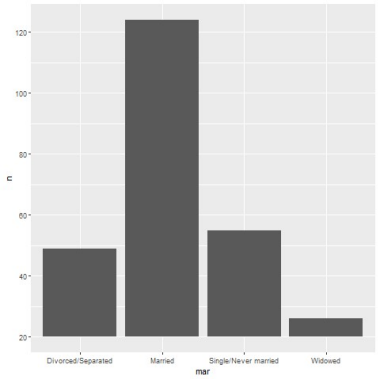
Combining categories has the risk of losing important details, so you do need to be careful.

In this particular example, narrow intervals are needed for children, because the survival probabilities change a lot between a 2 year old, a 5 year old, and a 13 year old passenger. But they don’t change as much for adults. So a chart with eighten bars for the first eighteen years of life plus a nineteenth bar showing survival for adults might be a good compromise.

I chose 480 by 480 pixels for most of the graphs in this talk because it is, for the most part, a good compromise between size of the file and quality of the image. But for a

difficult graph with 44 bars, a resolution of 960 by 960 or even higher might be needed. Be careful, though. When I import these files into another program like PowerPoint, that software may reduce the resolution in order to fit within their own page size limits. Trial and error is often the only good solution here.

Don’t cut your barchart off at the knees

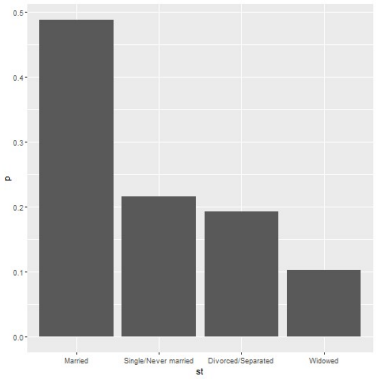


Truncated Barchart

Bar charts almost always start at zero. If you start it at a higer value, you are cutting the barchart off at the knees. This makes it impossible to make relative assessments. So in the original bar chart, it was fairly easy to tell the the Single and the Divorced/Separated columns were a bit less than half of the Married column, but on this truncated bar chart, you might briefly gain an impression that these two bars were less than a third of the married column. It’s even worse for the widowed column, which looks in this graph to be less than a quarter of the single and divorced/separated columns.

There are times where the zero value is so far away that you lose a lot of resolution by starting the bars at zero.

Recommendations, sort your bars by size



Sorted Bar chart

If you sort the bars by size, then comparisons between bars of approximately the same size are comparisons that are side by side. This shortens the distance that you have to project.

If your categories have a natural ordering, like the age groups we saw in an earlier example, you can't sort by size. A bar chart that starts with 30 to 40 year old passengers, jumps to 60 to 70 year old passengers, and then jumps again to 20 to 30 year old passengers will cause too much confusion. A Likert scale pretty much has to start with "strongly disagree" and follow the proper sequence through "disagree," "neutral," "agree," and end up at "strongly agree."

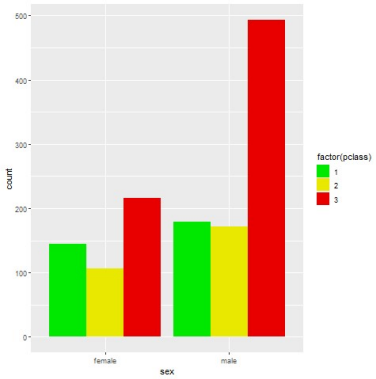
Some people like a nice orderly alphabetical order to their bars. It helps a little bit with "look up" or the process of quickly finding a particular bar of interest. But the value of look up is often overrated, as the more important visual tasks are comparisons of one bar to another rather than the rapid identification of a single bar.

Recommendations, Multiple categories

- Which bars get to snuggle?
  - Minimize distance between important comparisons
- When should you stack?
- Counts versus percentages
  - HOw to de-ephasize small categories
- What percentage to use?
  - Row, column, or cell percentages

When you have two different categories, like gender and passenger class in the Titanic example, you have the opportunity to display the bar chart several different ways. It is more work, but it is often well worth your time to try your bar chart several different ways.

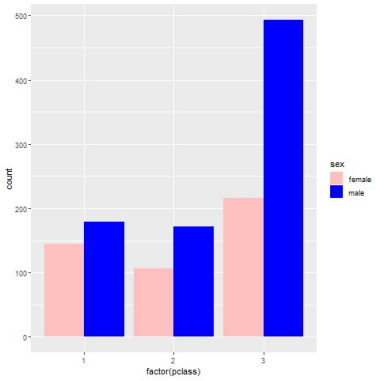
Recommendations, Which bars get to snuggle? (1/2)



Dodged barchart with passenger classes adjacent

When you place the passenger classes next to one another, it both emphasizes and facilitates comparisons among the passenger classes. The rule is to place things close that you want to be compared. So this graph tells you that males tend to be found by far more often in third class than any other class. This is where Leonardo DiCaprio travelled. The distribution among passenger classes is uneven for women as well, but not to as great an extent.

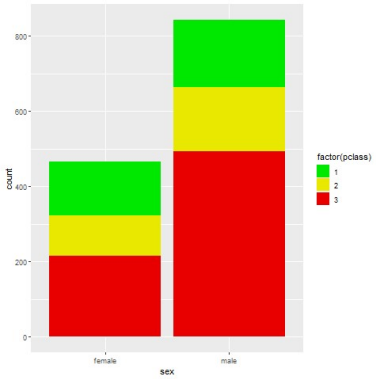
Recommendations, Which bars get to snuggle? (2/2)



Dodged barchart with sex adjacent

The emphasis changes when you place the men and women next to each other. Now the comparison that leaps out to you is that men outnumbered women in every passenger class, but down in thrid class, the gender disrepancy is the largest.

Recommendations, When should you stack? (1/3)

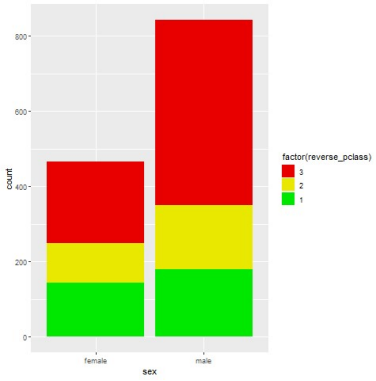


Barchart with passenger class stacked

Stacking tends to emphasize how each individual piece contribute to a whole. Here you see the relative distribution of females across the three passenger classes in the bar on the left and the relative distribution of males across the three passenger classes in the bar on the right.

With the stacked chart, you also bring the third class females close to the third class males, the second class females close to the second class males, and the first class females close to the first class males. But notice that the comparison of the bottom components, the third class females to the third class males is the easiest comparison, because it is a projection. To compare third class females to third class males, you just slide the female blue bar horizontally to the male blue bar. For the others, the comparison is a superimposition, you have to shift one bar diagonally to compare it to another. The two salmon bars representing first class females and first class males are almost the same size, but it's hard to tell with this graph because the diagonal shift is slower and less accurate.

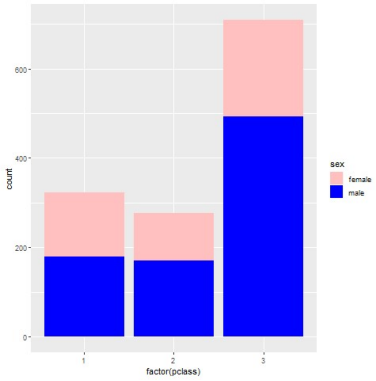
Recommendations, When should you stack? (2/3)



Barchart with passenger class stacked in reverse order

Notice how the comparison of first class females and first class males is so much easier when you place them at the bottom of the graph?

Recommendations, When should you stack? (3/3)



Barchart with sex stacked

You can switch things so that gender is stacked for each passenger class. This barchart shows that setting up a co-ed volleyball team would be a lot easier in first or second class, which have a roughly equal distribution of mean and women compared to third class.

Recommendations, Which percentage (1/8)

Table of counts

	Happy	Miserable	Total
Rich	30	10	40
Poor	90	70	160
Total	120	80	200

When you are computing percentages, you need to decide what particular percentage you want. The data set shown above is totally fictional, but it helps you understand what your options are.

This represents a sample of 200 people who are classified by income (rich or poor) and by outlook (happy or miserable). There are only 40 rich people in our sample, because there's only so much money in the world, and it is distributed unevenly. There's a more even split between happy and miserable people, but happy people, thankfully, are in the majority.

When you start calculating percentages for a table like this with two categorical variables, it gets a bit tricky.

Recommendations, Which percentage (2/8)

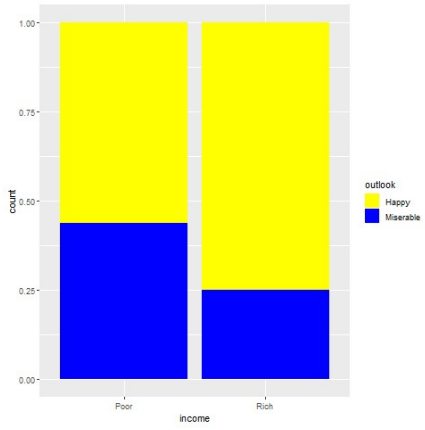
Table of row percents

	Happy	Miserable	Total
Rich	75%	25%	100%
Poor	56%	44%	100%
Total	60%	40%	100%

You can compute row percentages which divide the entry in each row by the row total. Row percents add up to 100% within each row.

The 75% in the upper left corner is a conditional probability. The probability of being happy given that you are rich is 75%. Money can buy happiness, so it seems. But the probability below it, 56%, provides a different picture. That's the probability of being happy given that you are poor. It seems that poor people can be happy also because there are other things, like family and friends that can also buy happiness.

Recommendations, Which percentage (3/8)



This arrangement of a stacked bar chart shows that the yellow regions, the regions associated with a happy outlook, are dominant in both the rich bar and the poor bar.

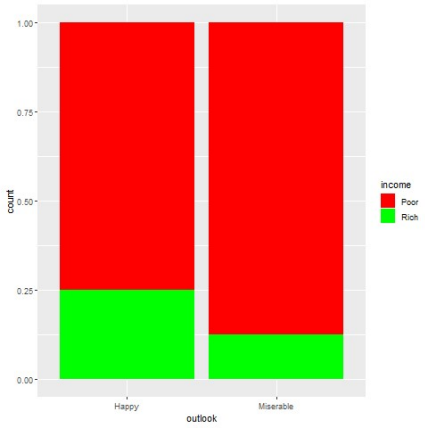
Recommendations, Which percentage (4/8)

Table of column percents

	Happy	Miserable	Total
Rich	25%	12%	20%
Poor	75%	88%	80%
Total	100%	100%	100%

You get a different percentage and a different picture, though, with the column percents. You get a column percent by dividing each entry by the corresponding column total. Column percents add up to 100% within a column, This is a very different probability. The probability in the upper left corner is the probability of being rich given that you are happy. It is only 25%. The percentage next to it, 12%, is the probability of being rich given that you are miserable. That's also quite small.

Recommendations, Which percentage (5/8)



This is a bar chart that illustrates these conditional probabilities. Notice the color coding. Green is for rich people, because green is the color of money. Red is for poor people because red is the color of indebtedness. Notice that red dominates this barchart. So if you find someone who is happy, don't presume that they are happy because they have money. It is far more likely 75% versus 25% that they are happy for a reason other than money.

Recommendations, Which percentage (6/8)

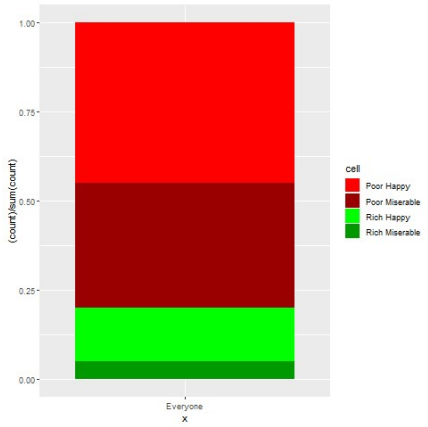
Table of cell percents

	Happy	Miserable	Total
Rich	15%	5%	20%
Poor	45%	35%	80%
Total	60%	40%	100%

There's a third percentage that you can compute, the cell percentage. That is the percentage that you get when you divided by the grand total. The cell percentages add up to 100% only when you add together all the percentages within the table.

It is unusual that you would want cell percentages in any visualization. It tells you haw frequently you see each possible cross classification. If you just randomly encountered someone, your best guess would be that they are poor but happy. That's almost half of the total. Your next best guess would be poor and miserable.

Recommendations, Which percentage (7/8)



This graph shows the relative percentages for poor happy, poor miserable, rich happy, and rich miserable people. I used the coding of red for poor, and green for rich, but made the happy outlooks a bright color and the miserable outlooks a dark color.

Now this is a rather artificial example, but it shows how different percentages and different barchart orientations provide different messages.

Recommendations, Which percentage (8/8)

- No hard and fast rules
- Classify your categorical variables
  - What is the treatment, what is the outcome?
  - What is cause, what is effect?
  - What is precedent, what is antecedent?
- Stack the outcome, effect, or antecedent.
  - Probability of outcome given the treatment

There are no absolute rules about how to stack the barchart, but here's some general guidance. There is usually a hierarchy to the categorical variables. One can usually be considered a treatment (a variable that you manipulate) and the other would be the outcome or the result of that treatment. Similarly, you could label one variable the cause and the other the effect or one the precedent (the categorical variable that comes first in time) and the antecedent (the variable that comes afterward). The second in the list, the outcome, the effect, or the antecedent is what you would normally consider placing in the stack.

This produces a conditional probability that makes sense from a pragmatic perspective, such as the probability of an outcome given a treatment.

Demographic variables are almost always treatments, causes, or precedents. You were a male or a female for your whole life, so it has to preceed what class ticket you bought or whether you lived or died.

Recommendations, Summary

- Don't make the gaps the same size as the bars
- Labels often fit better on a horizontal barchart
- With many bars, watch for rounding artefacts
- Sort your bars by size
- With two categorical variables, you have lots of options
  - Minimize distance between the most important comparisons
  - The bottom category in a stacked barchart is easiest to read
  - Stack outcomes/effects/antecedents

When you're designing a bar chart, make sure the gaps are smaller than the bars. If they are equal, you end up with a visually unpleasant effect or with potential confusion between what's a bar and what's a gap. With a lot of bars and long labels, you might try a horizontal bar chart, as the labels fit much easier. Also with many bars, watch out for rounding artefacts.

If your bars do not already have a natural ordering, sort them by size to make comparisons easy.

You have lots of options at your disposal when you are providing information about two categorical variables in a bar chart. Think about what comparisons are important and facilitate them by making them close.

The bottom category in a stacked barchart can be readily compared across stacks through projection, so it represents the easiest comparison.

It's not always clear what variable should be stack, but often it is the variable that could be considered the outcome, the effect, or the antecedent.

Always experiment with different arrangements and layouts. You should pick the chart that most clearly emphasizes the main message in the data set.

Recommendations, lessons from the world of fashion

#4 TOO MANY COLORS

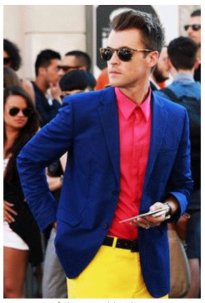


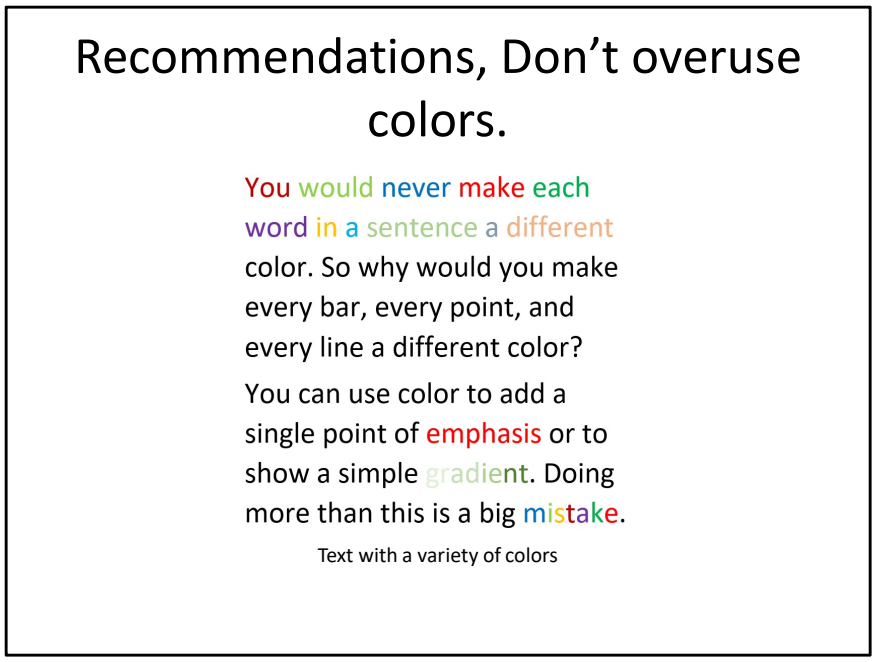
photo source: pinterest.com

Honestly, we find the best application for this quote in fashion: *"simplicity is the ultimate form of sophistication"*. Keeping it simple might be very difficult for several men, and, looking to the picture above, it really is.

Image of man wearing three bold colors

It's a well known fasion mistake to wear too many colors at the same time. Maybe this guy could get away with it, but most of us would look like idiots if we tried to dress that way.

There's a similar lesson for data visualization.



Naomi Robbins, an expert on data visualization, made an interesting observation. You would never make each word in a sentence a different color. So why would you make every bar, every point, and every line a different color?

Too many colors dilutes the impact that color can have.

You can use a second color to add emphasis. Or maybe a gradient between two different colors could work. Doing more than this is usually a big mistake.

Recommendations, The power of a single color

Clip of a red umbrella in a sea of black umbrellas

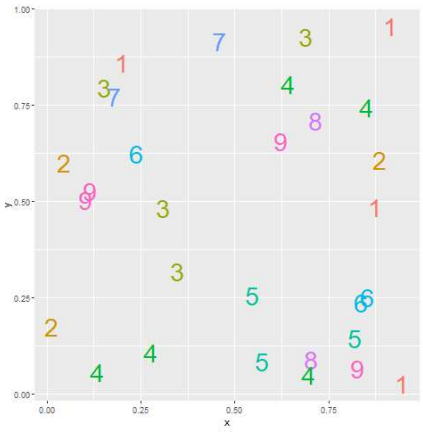
Graphic designers have known for quite a while that a restrained use of colors can be very effective. Here is an image from a YouTube video clip,

The Travelers - Look under the Umberella commercial (1986). Retrieved 2019-09-07 from https://www.youtube.com/watch?v=3zQX66jd_c0.

The single red umbrella in a sea of black umbrellas stands out. Your eye can't help but follow this umbrella as it travels across the screen from left to right. It's a very powerful image.

A small dollop of color in your visualizations can be far more effective than using a whole bunch of different colors.

Recommendations, Count the fives (1/3)

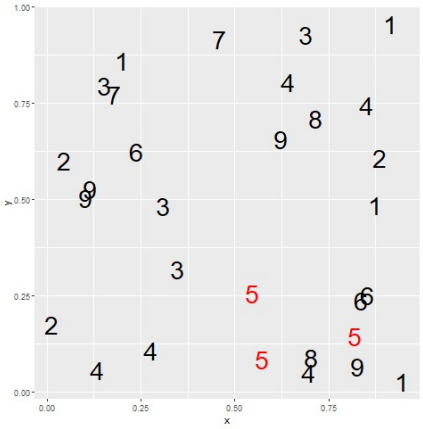


Here's an exercise that adapted from Olson and Bergen.

How many fives are there in this picture. I've used a different color for each number to make it easier for you to pick out any particular number. It takes a while, but you can see that there are three 5's, clustered in the lower right corner of the graph.

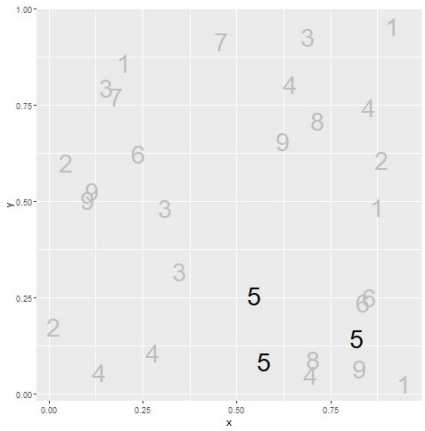
Did the colors help? Well, not all that much. It is hard to pick out nine colors and not have a few of them look very similar. In particular, the 5's and the 6's are pretty close, as are the 8's and the 9's.

Recommendations, Count the fives (2/3)



When you use a bit of restraint and only show two colors, you make the process of identifying all the fives much easier. The red fives are like the red umbrella in the Traveller's commercial.

Count the fives (3/3)



The same trick works with black versus gray. By making some features of your graph gray, you help them to fade into the background, making what remains in black more prominent.

Now why would you want to make something fade into the background? Why not just eliminate it? Sometimes that makes sense, but often you want to see the overall range that the emphasized points lie in.

Recommendations, Problems with emphasis

- Don't always know what to emphasize
  - Solution: interactivity
- Let the data speak for itself
  - But sometimes data speaks too subtly
- Potential for deception
- Bias potential in EVERY visualization choice

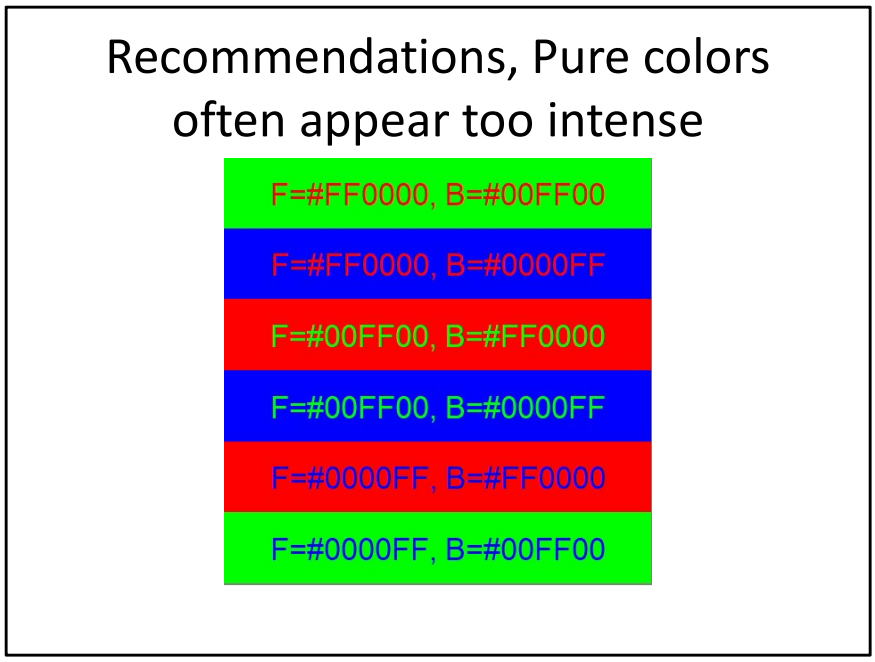
Now, if you feel a bit of unease about this example, it is a feeling I share. I used bold colors for the 5's. Red versus black or black versus gray. Doesn't this hurt the feelings of the other numbers? I hear the 6 has a fragile ego.

Seriously, the effort to emphasize one group always results in the de-empahsis of other groups. You may feel uncomfortable with this because you don't really know what group is worth emphasizing. Fair enough. If you don't know, then there are interactive features like mouseover that allow the end user to change the emphasis on the fly to what they find intersting. Mouseover is a feature where the graph changes as you let the mouse hover over certain features of the graph. It is a very effective tool, but beyond the scope of this workshop.

There's also a common refrain, "LEt the data speak for itself." It sounds nice in theory, but in practice, data doesn't speak very well. It has a soft voice. It mumbles. It's message gets lost in the noise. You job as a data visualizer is to amplify the message. If you are afraid to do this, you are in the wrong job.

The other thing to keep in mind is that if you have the power to emphasize (or de-emphasize) certain features of the data, you have the power to deceive your reader.

And this happens a lot. I won't share any deceptive plots with you, but you can find them easily with a Google search. Some of it is just plain stupidity, but you have to believe that some of it is malicious intent. The examples you see on the web are pretty easy to spot, but that doesn't mean that there aren't subtle ways to use graphics to deceive.

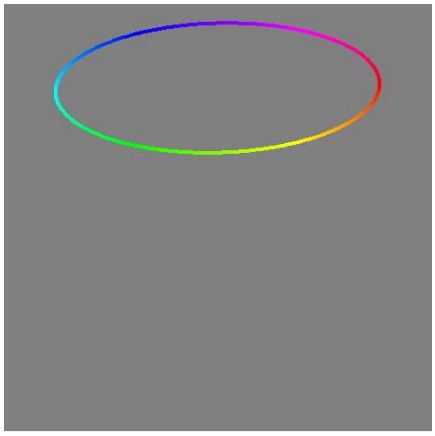


The pure colors: pure red, green, and blue are sometimes a bit harsh. This is especially true if they are placed in direct contrast. Here is an example.

One theory why these colors appear harsh is that they tend to produce afterimages. An afterimage is a short term distortion of vision. If you stare at an image long enough, so of the retinal cells in your eye become “tired” and stop firing as often. When your focus finally move away from that image, the opposite pattern appears temporarily in your field of vision because only the non-tired cells that transmit the opposite color can fire. It fades away as your tired cells recover.

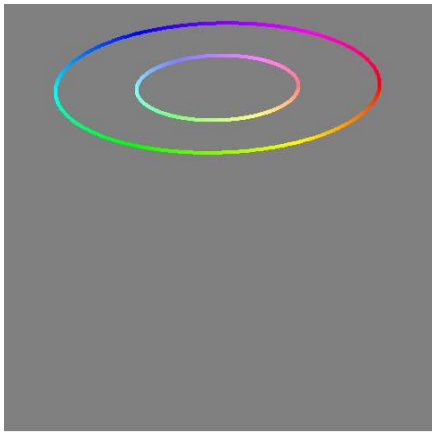
Sometimes the harshness is good. It is an attention grabber. But there are times when you want to soften the colors a bit. You can do this in two ways, making the colors a bit lighter or making the colors a bit darker.

Recommendations, Pure color circle and alternatives (1/3)



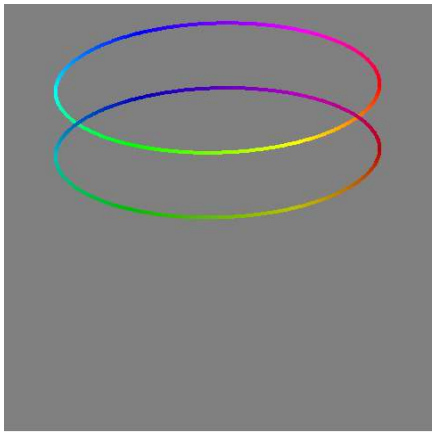
This is the circle of pure colors on the hsv cylinder that you saw earlier.

Recommendations, Pure color circle and alternatives (2/3)

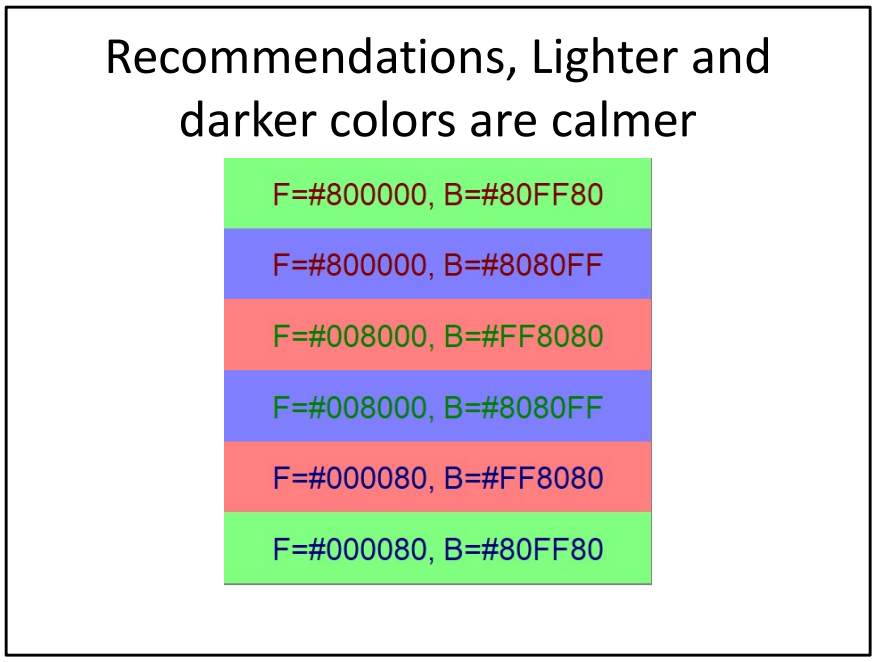


If you move the circle inward, all of the colors get lighter. They develop something of a pastel color to them. These are less harsh than the pure colors.

Colors, Pure color circle and alternatives (3/3)

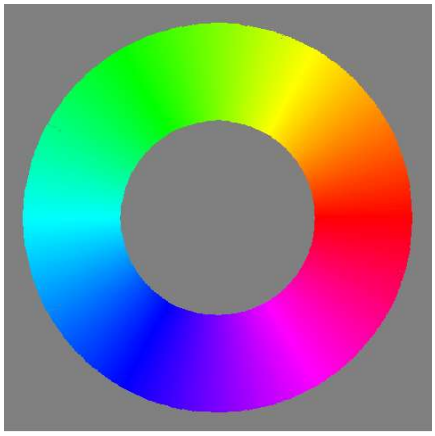


If you move the pure color circle down on the hsv cylinder, you get darker shades and these also are less harsh.



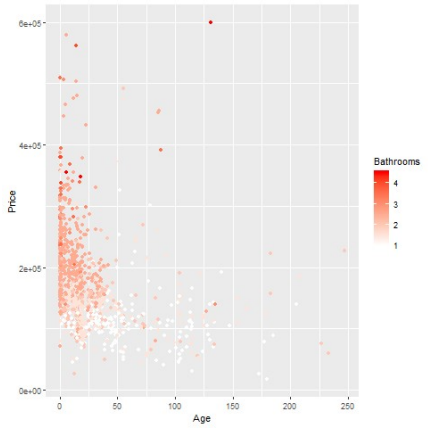
This chart shows lighter bacgrounds and darker foregrounds. The combination has a bit less contrast. The dark green foreground on the light blue background is perhaps the worst. But the softening of colors appears calmer and less harsh.

Recommendations, Avoid the rainbow gradient



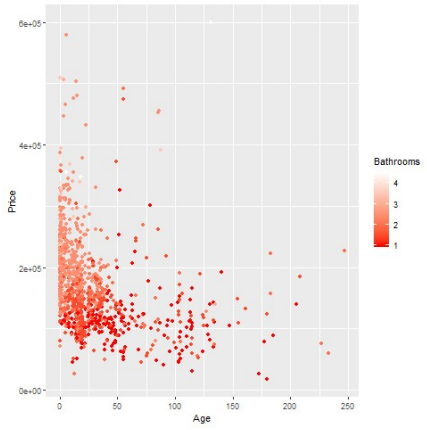
Most visualization experts recommend against the use of circular gradients. This gradient would make values at the low end of the scale look close in color to values on the high end of the scale. This almost never happens. The one prominent exception would be a variable like wind speed, where values of 0 degrees and 360 degrees are identical.

Recommendations, Simple gradient can emphasize high values



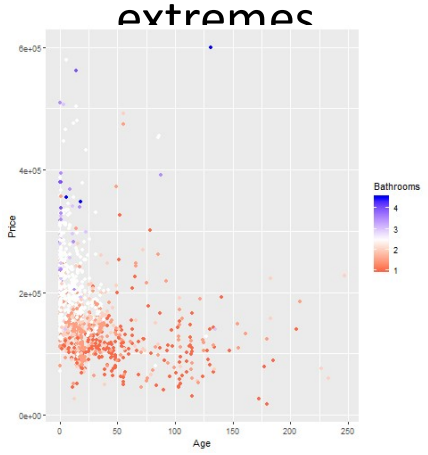
The single gradient can move from a light color to a dark color, and on a white background, this will tend to de-empahsize the small values and put greater emphasis on the large values.

Recommendations, Simple gradient can empahsize low values



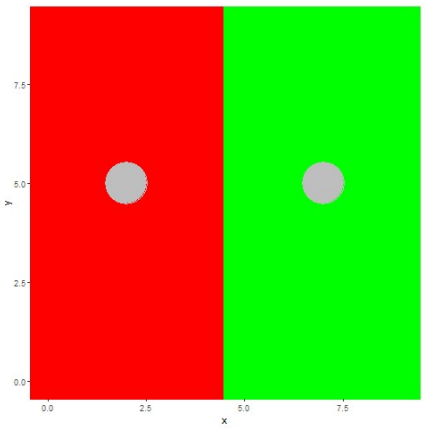
If you reverse this, and move from a dark color to a light color, then (at least on a white background), you will tend to de-empahsize the large values and put greater emphasis on the small values.

Recommendation, Divergent gradient empahsizes both extremes

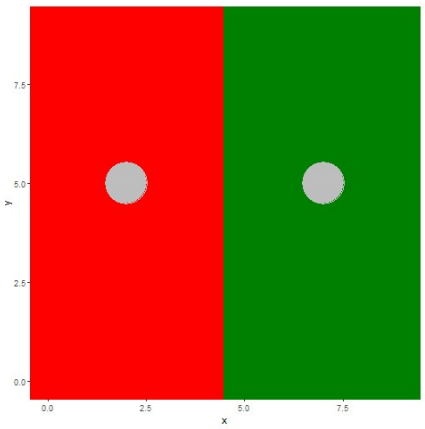


The middle of a divergent gradient typically has a muted color like gray that fades into the background. The two extremes in a divergent gradient are colors that contrast sharply with the background and with each other. This produece an emphasis at both extremes and a de-empahsis in the middle.

Recommendations, Unequal luminence causes optical illusions



Recommendations, Minimze illusions with equalized luminance



Recommendations, Disadvantages of equal luminance

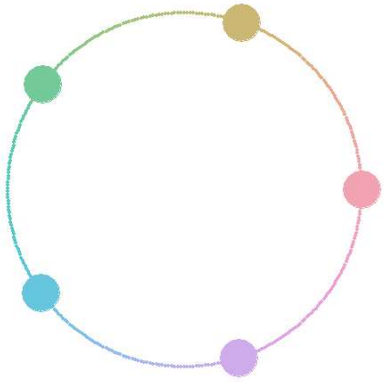
- Limited contrast
- Poor black and white reproduction
- Possible problems for color blind viewers

Equal luminance does present some disadvantages that you need to be aware of. There is less contrast among equal luminance colors that there would be among colors that have variations in luminance.

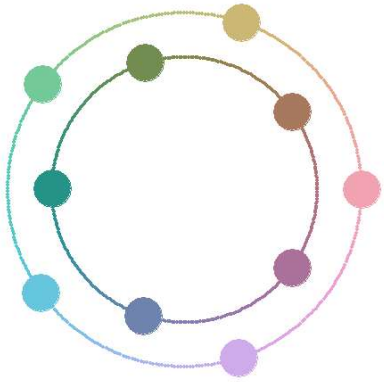
Colors with equal luminance tend to produce the exact same gray shade in black and white reproductions of color graphs.

Equal luminance can, in some situations, produce more problems for color blind viewers. We'll discuss in greater detail in just a bit.

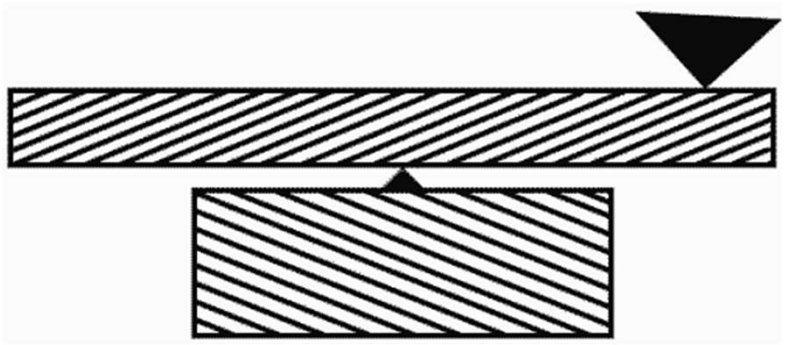
Recommendations, Give categories same luminance, equi-spaced hues



Recommendations, For many categories add second level of luminance



Recommendations, Avoid crosshatching



OPtical illusion caused by slanting lines

In the era when color was expensive, grpahic designers would use different types of crosshatching in place of colors. This is not really needed today, so you see it less often. But crosshatching can often produce optical illusions like the one shown above.

Recommendations, Consider color-blindness in your visualizations.

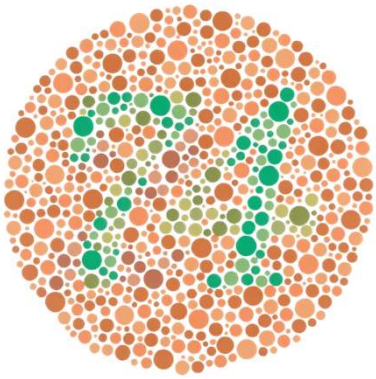
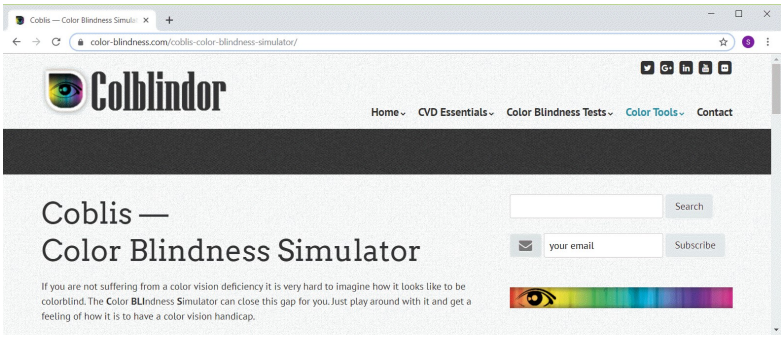


Image to test for color-blindness

Many of the color palettes used here cause difficulty for color blind people.

Recommedations, Color blindness simulator



Screenshot of color blindness simulator

This is one of many color blind simulators available on the web.

Recommendations, Other steps to control for colorblindness

- Categorical palettes are most troublesome
- Use palettes described as color blind friendly
- Use second visual cue (e.g., shape)
- Deliberately vary luminance

In general, equal luminance palettes for categorical data are the most likely to cause difficulty for color blind viewers. The single gradients and divergent gradients will usually have other changes in luminance that can help color blind viewers.

There are palettes that are known to be color blind friendly, meaning that the colors are easily distinguishable even for someone who is color blind.

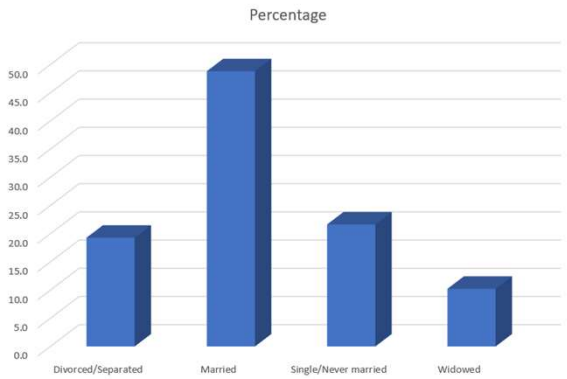
It helps all of us, but especially color blind viewers if you can find a second way to code a variable. This is fairly easy to do using shape as well as color to encode a categorical variable.

You can also ignore the previous efforts to balance luminance and deliberately vary luminance, as this is easily perceived even by color blind viewers.

Recommendations, summary

- Don’t overuse colors
- Pure colors are often too intense
  - Darker/lighter combinations are better
- Different gradients emphasize different features
- Unequal luminance causes optical illusions
- Consider colorblind viewers

Perception, Adding depth (1/4)

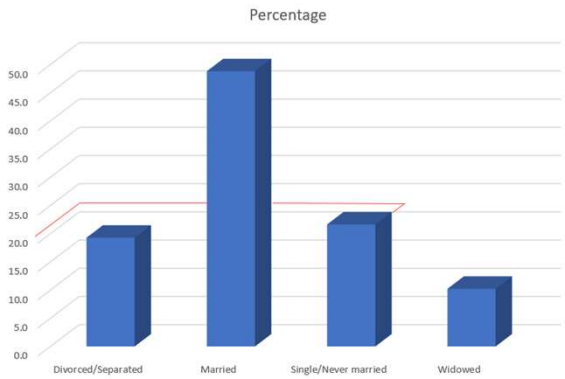


Three dimensional bar chart

Some sotware packages allow you to add some depth to your pie chart. This can catch your eye, at first, but almost all experts hate this approach. The most common complaint is that the three dimensional effects make a graph worse.

Worse means that it slows you down and it decreases the accuracy of your response. So here's a question. What percentage of your sample is single/never married? THe three dimensional effects slow you down. So for this three dimensional bar chart, do you measure the height of the single/never married bar by projecting the front of the bar to the axis on the left or by projecting the back of the bar? Actually, you need to first project to the back "wall" because the bars are placed a small distance in front.

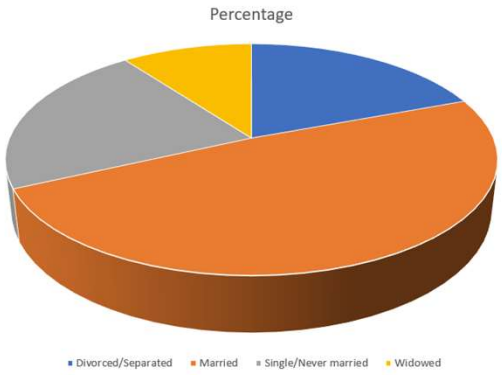
Perception, Adding depth (2/4)



Projection lines for a three dimensional bar chart

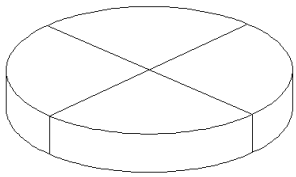
Once you realize that you need to project to the back wall, you have to do this accurately. Notice that it is not a simple horizontal or vertical projection. Instead you have to do a couple of superimpositions, shifts at a 45 degree angle. You can do this, but it degrades your accuracy.

Perception, Adding depth (3/4)



Three dimensional pie chart

Perception, Adding depth (4/4)



Three dimensional pie chart split into four equal pieces

There are several issues with the three dimensional pie chart, and you can see this best when you split this pie chart into four equal pieces. Notice that the angles are no longer 90 degrees because the perspective view distorts the angles. So you lose the big advantage (and possibly the only advantage) of the pie chart, the ability to divide it easily into four pieces.

Also notice that the wedge in the foreground looks bigger than the wedge in the background, because you can see the side of the foreground wedge, but you can't see the side of the background wedge.

Perception, review

– Visual tasks
– Hierarchy of perception

Graphs in the news, what are the aesthetics?

- What aesthetics (location, shape, size, color) are used?
- What aesthetics are not used?
- What variables are mapped to which aesthetics?

Review the same newspaper article and graph that you used earlier. Identify the various aesthetics that are used or are not used in your graph. What variables are associated with which aesthetics?

Barcharts, On your own

– While the Titanic sailed during an era of women and children first, it also sailed in an era where class distinctions were important. This shows in the mortality statistics for each passenger class. Show this using a bar chart and include information about gender and mortality as well.