

Communicating How Much?

Dr. Austin Brown

Kennesaw State University

Introduction

- ▶ In this section, we are going to be learning about existing methods which help us visually communicate, “How Much?”
- ▶ “How much?” types of visualizations make use of quantitative variables, which are *measured* like:
 - ▶ “How much money did we earn this past quarter?”
 - ▶ “How much carbon dioxide is in the atmosphere?”
 - ▶ “How much time do undergraduate students in Georgia spend on social media sites per day?”
- ▶ In other words, we will be focusing on questions which involve a single (or univariate) quantitative variable.

Communicating “How Much”

- ▶ Quantitative variables could be characteristics like revenue, weight, distance, and time, among many, many others.
- ▶ Generally, here, our data source would need to be in aggregate form, so that a single record exists. For instance:
 - ▶ A sales order generating revenues of \$95
 - ▶ A shipment weighing 5.2 pounds
 - ▶ A flight covering a distance of 2,408 miles

Communicating “How Much”

- ▶ Let's check out an example of this using the NYC Garbage Data.
- ▶ These data represent the amount of garbage (the less fancy term for “refuse”), in tons, collected each month of the year, in each of the boroughs and community districts in the city.
- ▶ Suppose we wanted to know, “How does the amount of garbage/refuse (in tons) that the NYC Department of Sanitation reportedly collected from each borough compare during September 2011?”
 - ▶ Let's see how we can accomplish this!

Communicating “How Much”

- ▶ First, we have to subset the data to our specific time parameters (September 2011). Don't worry about the warning message.

```
library(tidyverse)
library(readxl)
## To answer this question, let's first
## read in the data ##
nyc <- read_xlsx("NYC Trash Data.xlsx")
## Subset to September 2011 ##
nyc_sept11 <- nyc |>
  filter(MONTH == 9 & YEAR == 2011)
## Sum up REFUSETONSCOLLECTED variable by Borough ##
trash_tot <- nyc_sept11 |>
  group_by(BOROUGH) |>
  summarize(Sum_Trash = sum(REFUSETONSCOLLECTED))
```

Communicating “How Much”

```
## Take a glimpse of the result ##  
print(trash_tot)
```

```
## # A tibble: 5 x 2  
##   BOROUGH      Sum_Trash  
##   <chr>         <dbl>  
## 1 Bronx          37735.  
## 2 Brooklyn       71074.  
## 3 Manhattan      38066.  
## 4 Queens         64365.  
## 5 Staten Island  17885.
```

Communicating “How Much”

- Now, how do we go about visualizing these data? There are lots of ways we can do so using ggplot2. Here are just a few:

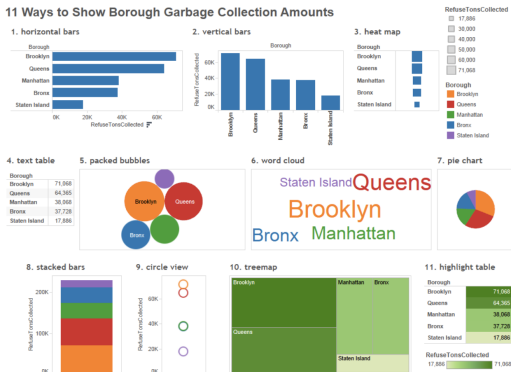


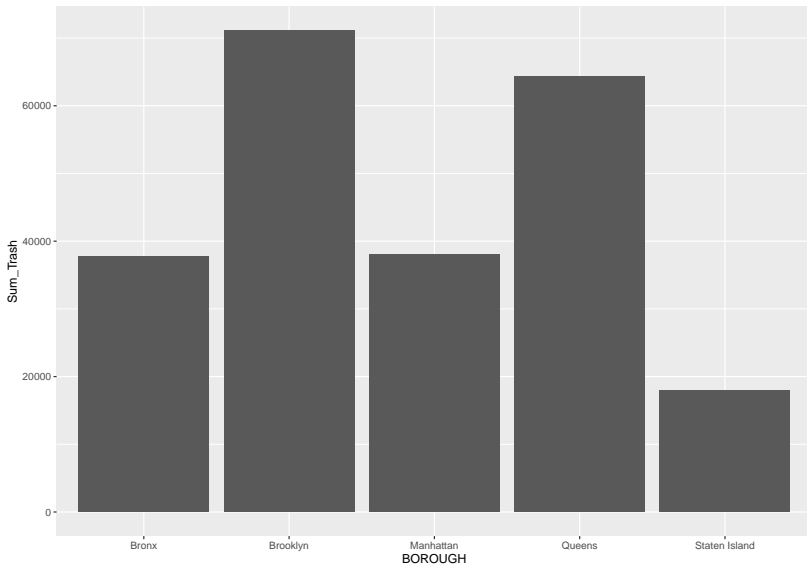
Figure 3-3. Eleven ways to show a simple quantitative (noncounting noun) comparison

Communicating “How Much”: The Vertical Bar Chart

- ▶ Let's start by building a vertical bar chart.

```
## Let's start by building a vertical bar chart! ##  
  
## To do so, we will use the geom_bar function ##  
## The stat='identity' argument tells ggplot2  
## to use the data as is ##  
trash_tot |>  
  ggplot(aes(x=BOROUGH,y=Sum_Trash)) +  
  geom_bar(stat='identity')
```


Communicating “How Much”: The Vertical Bar Chart



Communicating “How Much”: The Vertical Bar Chart

- ▶ Great! We have a visualization to compare “how much?” But what are some ways we could improve this visualization? I notice a few right away:
 1. Poor use/no use of axis titles and plot titles (How does anyone know what these data represent or what time period they represent?)
 2. The comparison between boroughs isn't as straightforward as it could be due to the bars being ordered in alphabetical rather than ascending or descending order.
 3. The default color palette is less than desirable.

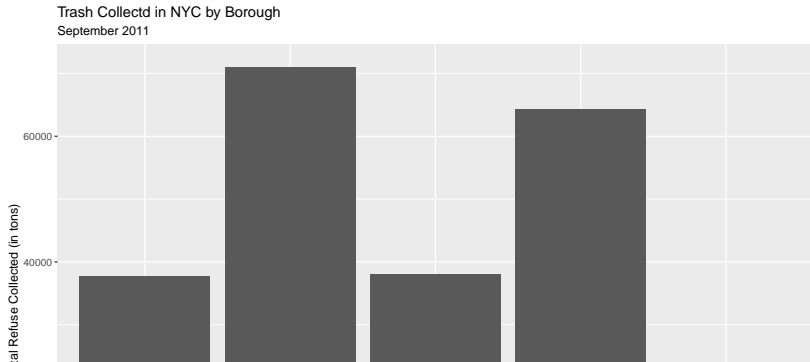
Let's see how we can solve each of these problems by adding additional code to our base visualization

Including/Modifying Axis and Overall Titles

```
## Solving the Labels Problem ##  
trash_tot |>  
  ggplot(aes(x=BOROUGH,y=Sum_Trash)) +  
  geom_bar(stat='identity') +  
  labs(x = "NYC Borough",  
       y = "Total Refuse Collected (in tons)",  
       title = "Trash Collectd in NYC by Borough",  
       subtitle = "September 2011")
```

Including/Modifying Axis and Overall Titles

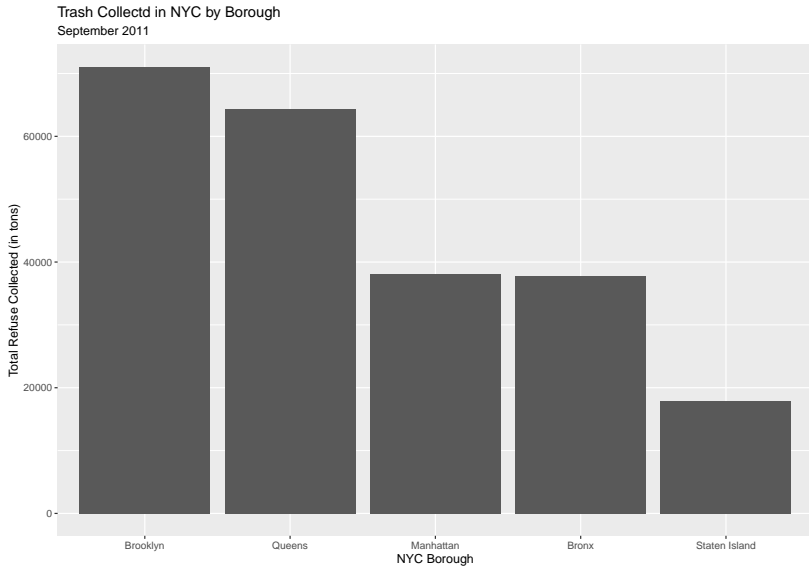
```
trash_tot |>
  ggplot(aes(x=BOROUGH,y=Sum_Trash)) +
  geom_bar(stat='identity') +
  labs(x = "NYC Borough",
       y = "Total Refuse Collected (in tons)",
       title = "Trash Collectd in NYC by Borough",
       subtitle = "September 2011")
```



Reordering the Bars in Descending Order Based on Numeric Value

```
## Reordering the Bars ##  
trash_tot |>  
  ggplot(aes(x=reorder(BOROUGH,-Sum_Trash),y=Sum_Trash)) +  
  geom_bar(stat='identity') +  
  labs(x = "NYC Borough",  
       y = "Total Refuse Collected (in tons)",  
       title = "Trash Collectd in NYC by Borough",  
       subtitle = "September 2011")
```

Reordering the Bars in Descending Order Based on Numeric Value



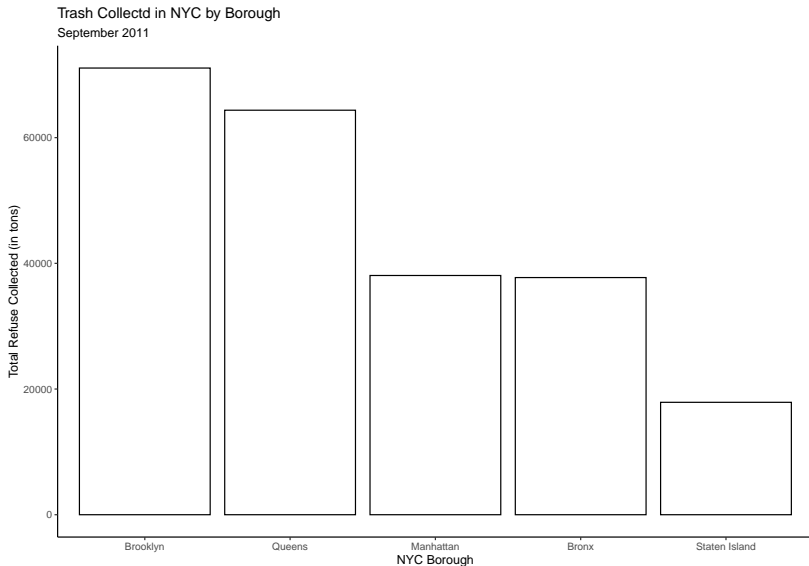
Changing the Color of the Bars and Overall Theme

```
## Solving the Colors Problem ##
```

```
trash_tot |>
```

```
  ggplot(aes(x=reorder(BOROUGH,-Sum_Trash),y=Sum_Trash)) +  
  geom_bar(stat='identity',color='black',fill='white') +  
  labs(x = "NYC Borough",  
        y = "Total Refuse Collected (in tons)",  
        title = "Trash Collectd in NYC by Borough",  
        subtitle = "September 2011") +  
  theme_classic()
```

Changing the Color of the Bars and Overall Theme



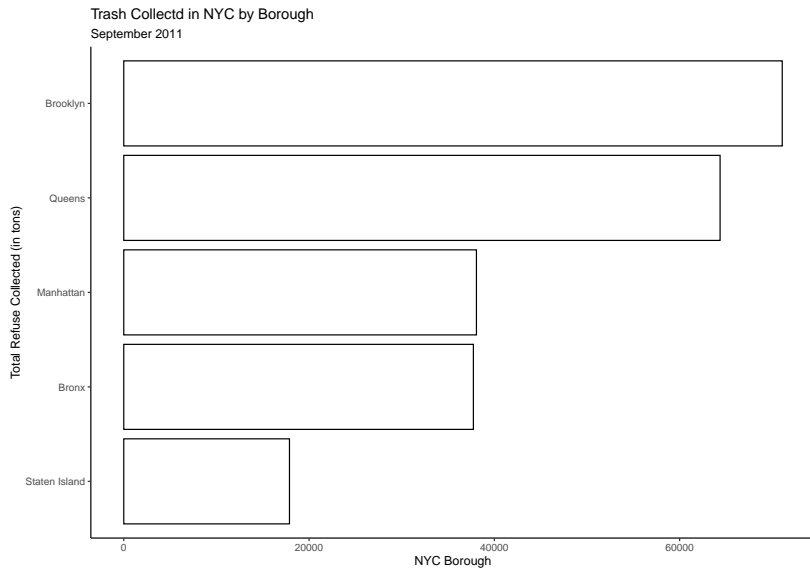
Communicating “How Much”: The Horizontal Bar Chart

- ▶ Notice how this final visualization is displaying the exact same information as our base visualization, but in a manner where audiences can more readily glean the information being conveyed (not to mention the improved aesthetics!).
- ▶ One other modification which may also make the visualization more effective is for the bars to be made *horizontal* rather than *vertical*.
- ▶ We change the x and y global aesthetics to achieve this.

Ordered Horizontal Bar Chart

```
trash_tot |>
  ggplot(aes(x=Sum_Trash,y=reorder(BOROUGH,Sum_Trash))) +
  geom_bar(stat='identity',color='black',fill='white') +
  labs(x = "NYC Borough",
       y = "Total Refuse Collected (in tons)",
       title = "Trash Collectd in NYC by Borough",
       subtitle = "September 2011") +
  theme_classic()
```

Ordered Horizontal Bar Chart



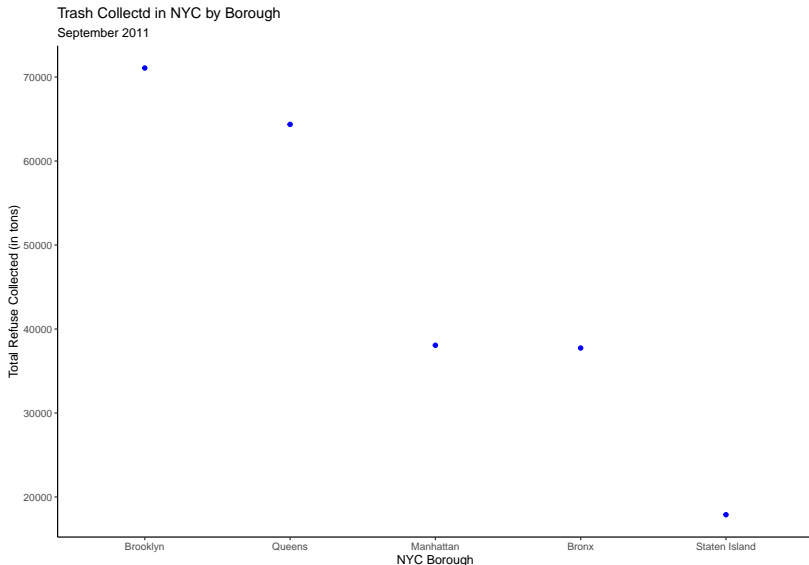
Building Dot Charts to Communicate “How Much”

- ▶ The ordered vertical and horizontal bar charts are really useful tools for comparing “how much” but certainly they aren’t the only tools.
- ▶ Another tool, called the “dot chart” is sometimes preferred to the two methods previously discussed seeing as the eye may be overwhelmed by the bars. We are really just comparing values.
 - ▶ In essence, we only need the “top” of the bars, not the whole thing.
- ▶ Using the basic `ggplot2` syntax we have already developed, we can easily change our bar charts to dot charts:

Building Dot Charts to Communicate “How Much”

```
trash_tot |>
  ggplot(aes(x=reorder(BOROUGH,-Sum_Trash),y=Sum_Trash)) +
  geom_point(color='blue',fill='blue') +
  labs(x = "NYC Borough",
       y = "Total Refuse Collected (in tons)",
       title = "Trash Collectd in NYC by Borough",
       subtitle = "September 2011") +
  theme_classic()
```

Building Dot Charts to Communicate “How Much”



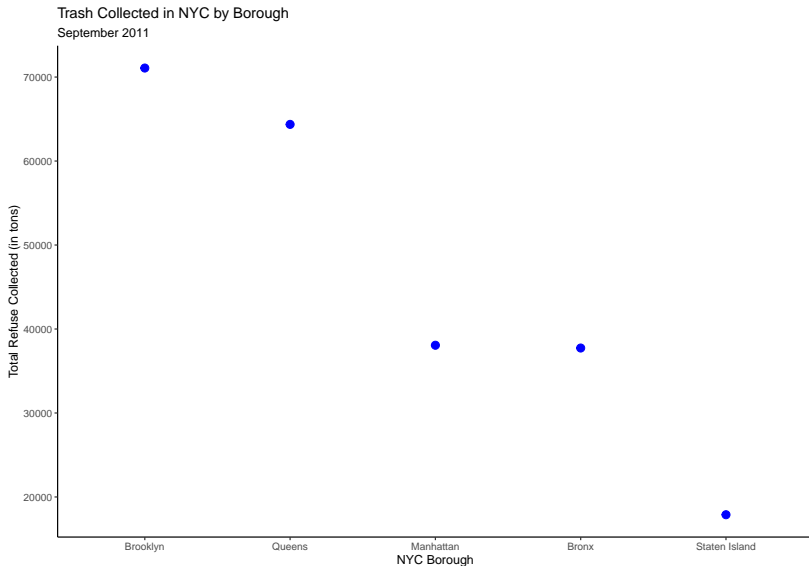
Building Dot Charts to Communicate “How Much”

- ▶ Notice something about these dots... they're tiny! How can we control the size of the dots?
- ▶ We can make use of the `size` argument within the `geom_point` function.
- ▶ The default value is 1.5. We can increase or decrease this value based on our needs! Let's try `size=3`.

Building Dot Charts to Communicate “How Much”

```
trash_tot |>
  ggplot(aes(x=reorder(BOROUGH,-Sum_Trash),y=Sum_Trash)) +
  geom_point(color='blue',fill='blue',size=3) +
  labs(x="NYC Borough",
       y="Total Refuse Collected (in tons)",
       title="Trash Collected in NYC by Borough",
       subtitle="September 2011") +
  theme_classic()
```


Building Dot Charts to Communicate “How Much”

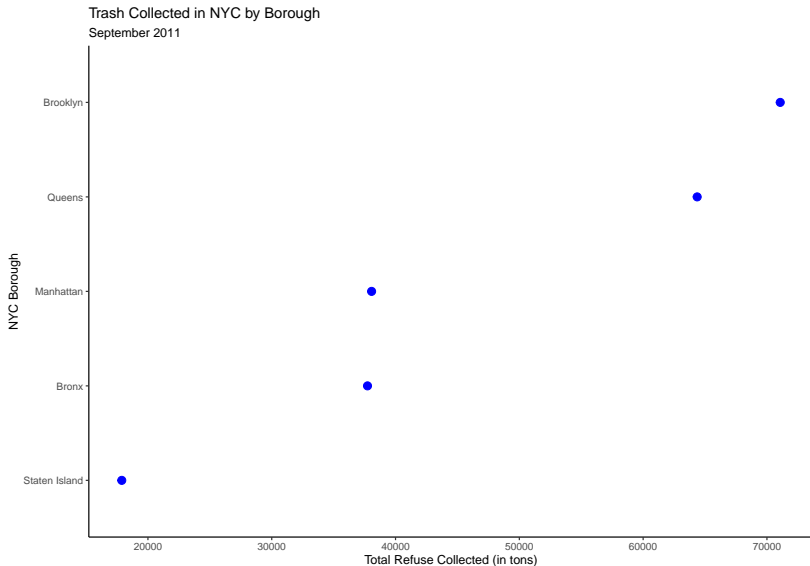


Building Dot Charts to Communicate “How Much”

- ▶ Better! Let's flip the coordinates as before to create a horizontal dot chart (notice I once again removed the “-” symbol from the reorder function so the dots render in descending rather than ascending order):

```
trash_tot |>
  ggplot(aes(x=Sum_Trash,y=reorder(BOROUGH,Sum_Trash))) +
  geom_point(color='blue',fill='blue',size=3) +
  labs(y="NYC Borough",
       x="Total Refuse Collected (in tons)",
       title="Trash Collected in NYC by Borough",
       subtitle="September 2011") +
  theme_classic()
```

Building Dot Charts to Communicate “How Much”



Building Dot Charts to Communicate “How Much”

- ▶ One potential limitation of the dot chart, as it stands, is that even though we only care about the dots themselves, the eye may need help in moving from the y-axis to the dot, especially when the values associated with the dots are large (in a relative sense)
- ▶ Instead of including a whole bar, what if we included just a dashed line going from the borough name to the dot?
- ▶ We can do this in multiple ways, but one straightforward way is to add another `geom` function! This time we can use the `geom_segment` function to draw a line segment.

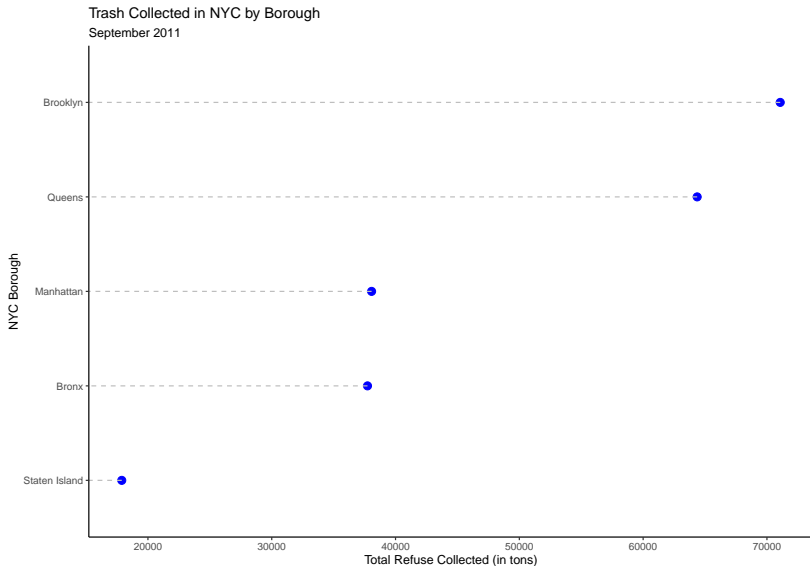
Building Dot Charts to Communicate “How Much”

- ▶ Notice in the `geom_segment` function, we have to specify when the line segments begin and end in terms of their `x` and `y` coordinates.
- ▶ We've already specified where `x` and `y` start in the global `ggplot` function: at the value of the point and the borough itself, respectively.
- ▶ Since we are drawing horizontal lines, the value of `y` doesn't change. Thus `yend=BOROUGH`.
 - ▶ Since `x` starts at the value of the point, `xend=0`
 - ▶ We can also control the `color` and `linetype` as well.

Building Dot Charts to Communicate “How Much”

```
trash_tot |>
  ggplot(aes(x=Sum_Trash,y=reorder(BOROUGH,Sum_Trash))) +
  geom_point(color='blue',fill='blue',size=3) +
  geom_segment(aes(yend=BOROUGH),xend=0,color='gray',
               linetype='dashed') +
  labs(y="NYC Borough",
       x="Total Refuse Collected (in tons)",
       title="Trash Collected in NYC by Borough",
       subtitle="September 2011") +
  theme_classic()
```

Building Dot Charts to Communicate “How Much”



In-Class Activity

- ▶ Now, using the `Lahman` package and the `Batting` and `People` datasets, suppose I want to know who the top 10 homerun hitters during the 2022 Major League Baseball regular season were and how many homeruns they hit?
 1. Create an ordered horizontal barchart to answer this question.
 2. Create an ordered vertical dot chart to answer this question.
 3. How else might this visualization be modified to better communicate the story we are trying to tell?