

Biostats week 3: Describing relationships among variables

Jenine Harris

Fall 2018

Outline

This packet reviews *bivariate* statistics, or statistics examining relationships between two variables:

1. One categorical variable & one continuous variable
2. Two categorical variables
3. Two continuous variables

Packages you will need in this packet:

- `vcd`
- `ggplot2`

1. One categorical variable & one continuous variable

Open the local health department data at <http://tinyurl.com/zn46s6e> and use the summary command to examine the data set and the variables in it.

```
# open local health department (LHD) data
lhd <- read.csv("http://tinyurl.com/zn46s6e")
summary(lhd)
```

```
##           id           numserved           state           expenditures
## AL032  : 1   Min.      : 5152   OH       : 6   Min.      : 145255
## AR025  : 1   1st Qu.: 20226   AR       : 3   1st Qu.: 856084
## AR038  : 1   Median : 41133   IL       : 3   Median : 2384860
## AR066  : 1   Mean    : 105177  MA       : 3   Mean    : 5155926
## CA015  : 1   3rd Qu.: 107329  IA       : 2   3rd Qu.: 6175087
## CT021  : 1   Max.    :1135992  ID       : 2   Max.    :51278368
## (Other):44                                (Other):31   NA's    :15
##           revenues           immunization hivscreen cancerscreen diabetescreen
## Min.      : 180455   no : 5           no :18   no :25           no :26
## 1st Qu.: 807600     yes:45          yes :30   yes :22          yes :20
## Median : 2238886                                NA's: 2   NA's: 3           NA's: 4
## Mean      : 3861264
## 3rd Qu.: 6046167
## Max.      :16224295
## NA's      :20
##           services
## Min.      :0.00
## 1st Qu.:3.00
## Median :4.00
## Mean      :4.04
## 3rd Qu.:6.00
## Max.      :6.00
##
```

The code book for the data is at the end of this packet.

As a reminder:

- *categorical* variables: Variables with *categories* like marital status, color, sex, alma mater, religion, ethnicity, etc.
- *continuous* variables: Variables with values that can take *any* value along some *continuum* like age, height, weight, distance, blood pressure, temperature, etc.
- *discrete* variables: Variables that are also along a continuum, but can only have certain values like the number of siblings or pets you have, the result of rolling dice, cars on a street, people in a state, etc.

There is quite a bit of evidence that health departments serving more people have more resources and are able to provide more services to their constituents.

In this data set we have a discrete variable showing number of people served (**numserved**) and a categorical variable showing whether or not the health department provides HIV screening (**hivscreen**). We can use bivariate analyses to compare the number of people served by health departments that provide HIV screening with the number of people served by health departments with no HIV screening program.

The mean number of people served by a health department shown in the summary statistics above is 105,177 people. Of the 50 health departments in this sample, 30 provide HIV screening and 18 do not.

Our research question might be:

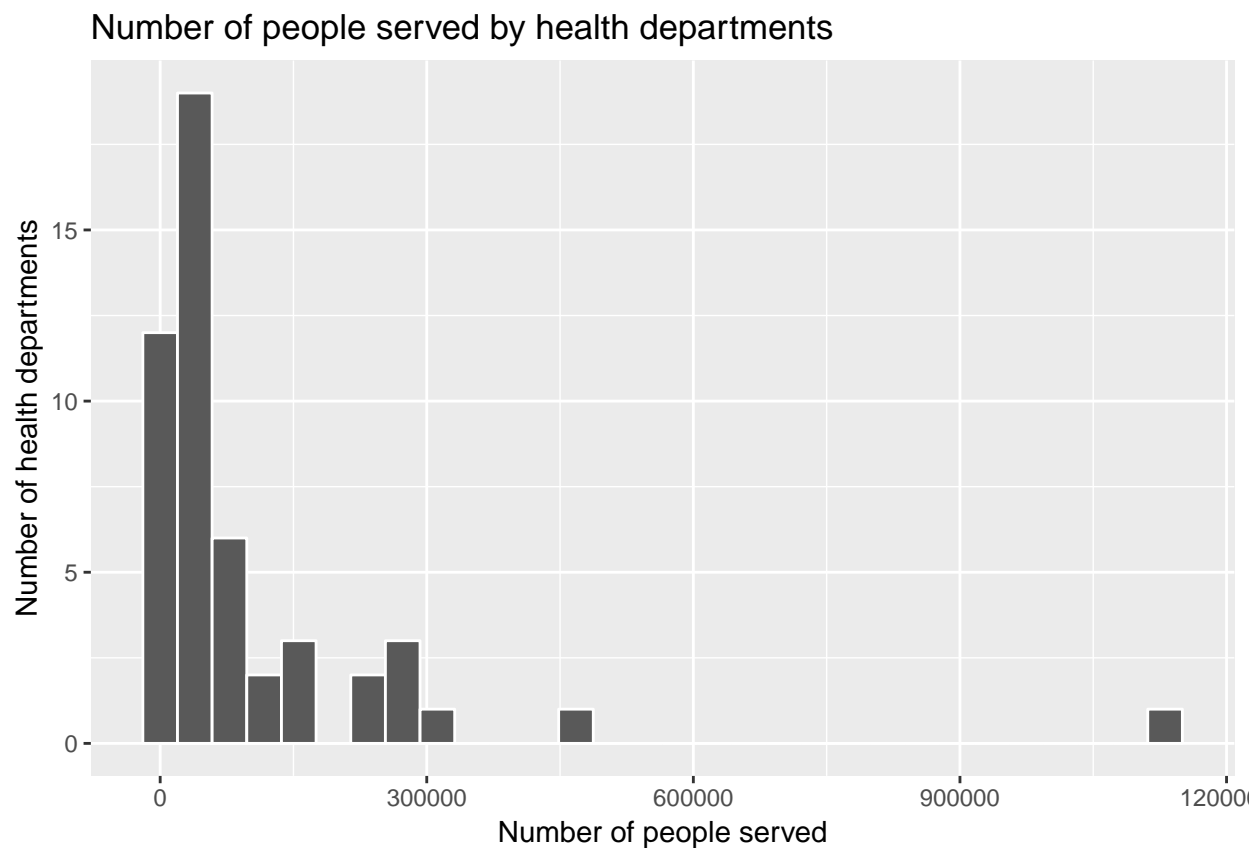
Do health departments that provide HIV screening also serve more people?

First, what measure of central tendency should we use for number of people served? Let's graph it and find out:

```
# open ggplot2 for graphing
library(ggplot2)

## Warning: package 'ggplot2' was built under R version 3.5.1

# make a histogram
ggplot(data = lhd, aes(x = numserve)) +
  geom_histogram(color="white") +
  xlab("Number of people served") +
  ylab("Number of health departments") +
  ggtitle("Number of people served by health departments")
```

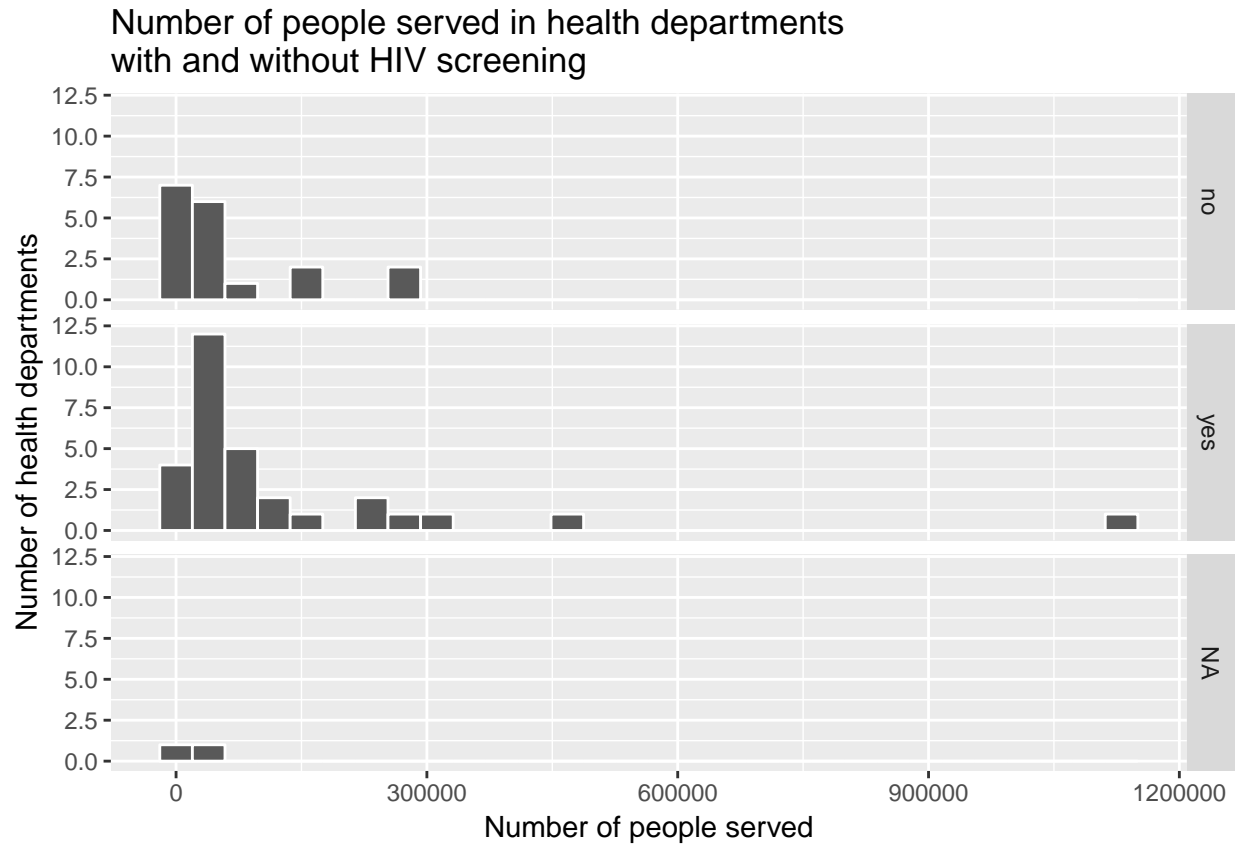


It looks right skewed, so the mean is probably not the best measure of central tendency. To be sure, see if number served is skewed within each HIV screening group. There are multiple ways to do this, we will use `facet_grid()`.

Adding `facet_grid()` to a `ggplot` allows you to plot different groups in separate sections of a plot, like this:

```
# graph with facets to see health depts
# with and without HIV programs
ggplot(data = lhd, aes(x=numserve)) +
  geom_histogram(color="white") +
  facet_grid(rows = vars(hivscreen)) +
  xlab("Number of people served") +
```

```
ylab("Number of health departments") +
ggtitle('Number of people served in health departments\nwith and without HIV screening')
```



Both groups look right skewed, so we should use the median rather than the mean. One way to get the median for each group is to use the `by` command.

The `by` command takes three arguments to get the median for each group:

- `data`: The data argument is the data to be analyzed, in this case the `numserved` variable from the `lhd` data frame
- `INDICES`: These are the groups the data should be divided up by, in this case the `hivscreen` variable
- `FUN`: The function argument is what procedure you want to do to the data from each group. In this case, we want the median.

```
# get the median for numserved by hivscreen
by(data = lhd$numserved,
    INDICES = lhd$hivscreen,
    FUN = median)
```

```
## lhd$hivscreen: no
## [1] 33088
## -----
## lhd$hivscreen: yes
## [1] 48049
```

Intepreting the results: The health departments providing HIV screening serve a median of 48,049 people, while the health departments not providing HIV screening serve a median of 33,088 people.

The answer to our original research question is *yes*. In this data, health departments with HIV screening

serve more people.

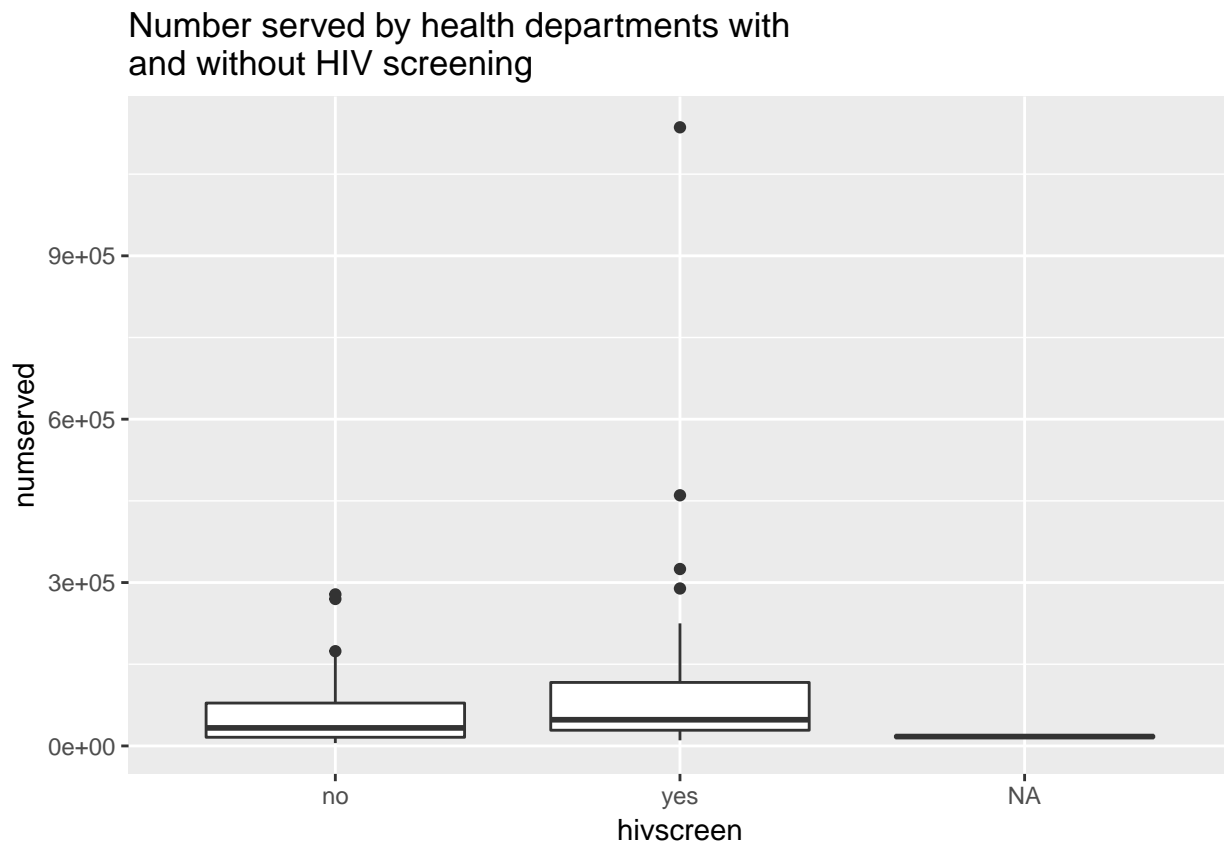
The `by` command works with many different kinds of functions. For example, use `summary` instead of `median` to get more information:

```
# get the median for numserve by hivscreen
by(data = lhd$numserve,
    INDICES = lhd$hivscreen,
    FUN = summary)

## lhd$hivscreen: no
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##      5152  15880   33088   72042   78700  278246
## -----
## lhd$hivscreen: yes
##      Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##     10326   28818   48049  130926  116594 1135992
```

Numbers are ok for conveying information, but visuals are often easier to interpret and can be more impactful. Try a box plot to visualize differences among groups:

```
# boxplot of numserve by hivscreen
ggplot(data = lhd, aes(x = hivscreen, y = numserve)) +
  geom_boxplot() +
  ggtitle("Number served by health departments with\nand without HIV screening")
```



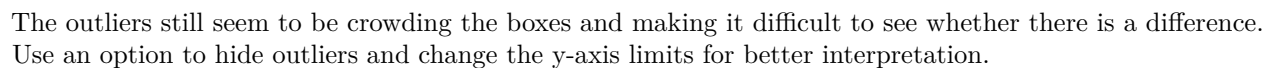
It is a little hard to see with all of the dots representing outliers and the NA box. We may want to hide the missing values box since it is not really important to our research question.

```
# take a subset of the lhd data where hivscreen
# is not NA
lhd.noNA <- subset(x = lhd, hivscreen!='NA')

# check the hivscreen variable in the new data frame
summary(lhd.noNA$hivscreen)

## no yes
## 18 30

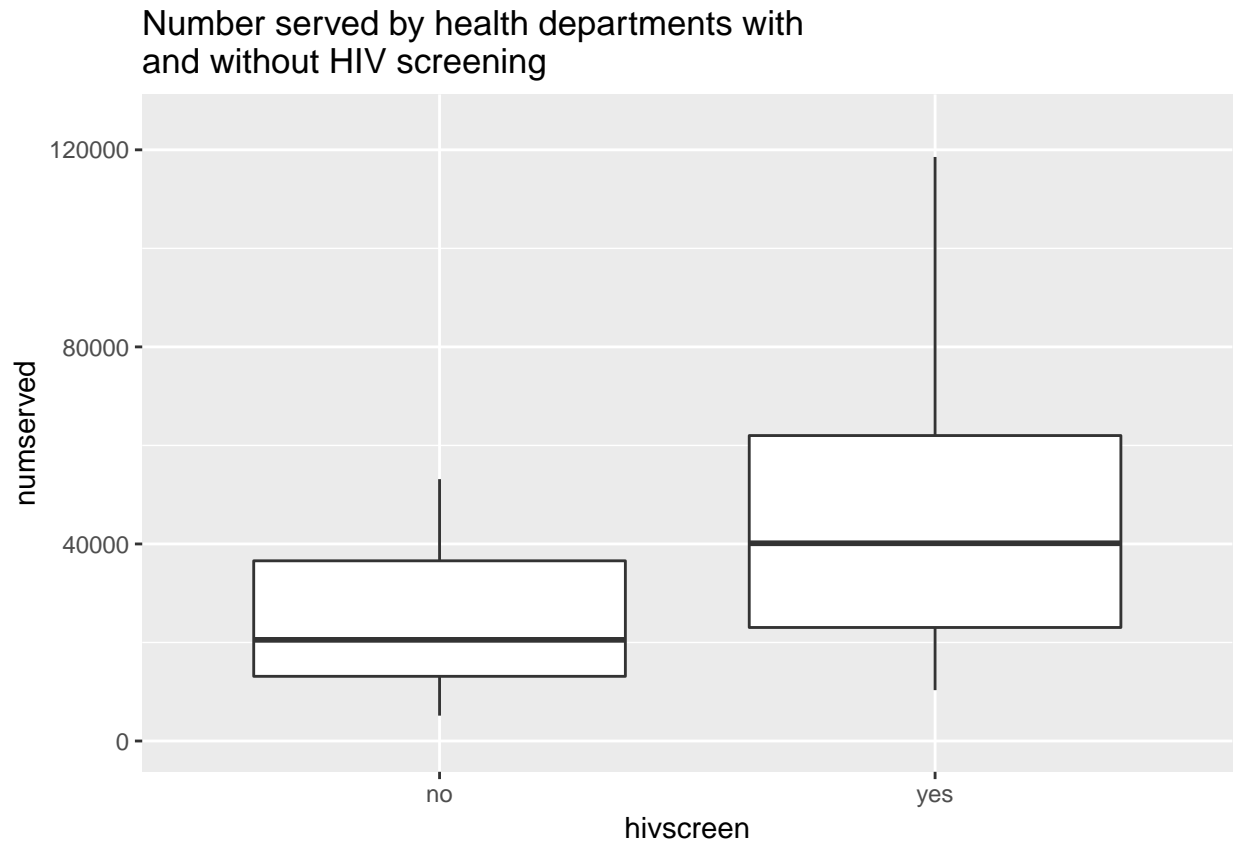
# plot it again
ggplot(data = lhd.noNA, aes(x = hivscreen, y = numserve)) +
  geom_boxplot() +
  ggtitle("Number served by health departments with\and without HIV screening")
```



```
# plot it again without the outliers
# add ylim to change the y-axis limits
ggplot(data = lhd.noNA, aes(x = hivscreen, y = numserviced)) +
  geom_boxplot(outlier.shape = NA) +
  ylim(0, 125000) +
```

```
ggtitle("Number served by health departments with\nand without HIV screening")
```

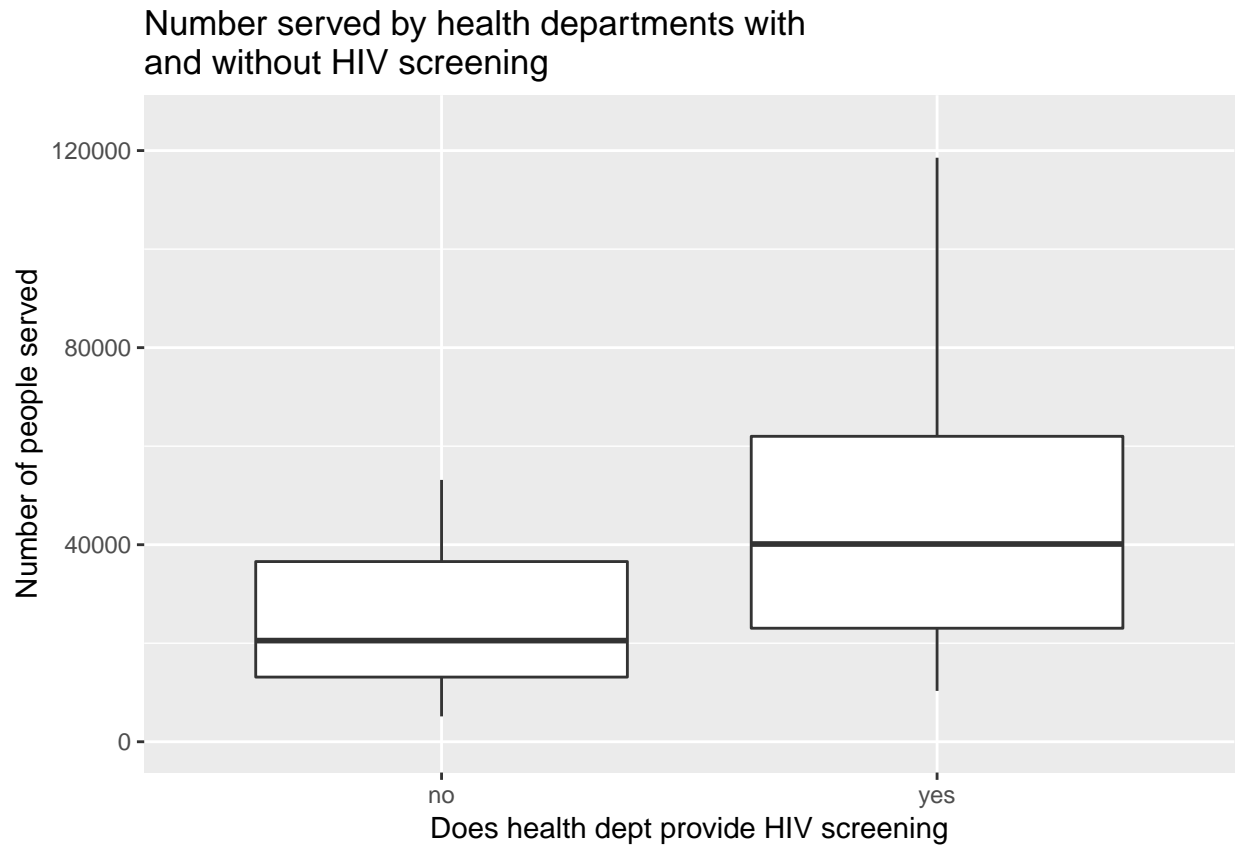
```
## Warning: Removed 11 rows containing non-finite values (stat_boxplot).
```



The warning message tells us that we removed 11 outlier values! The boxplots look great but for anyone but the analyst to know what the heck it means, it needs better labels on the axes.

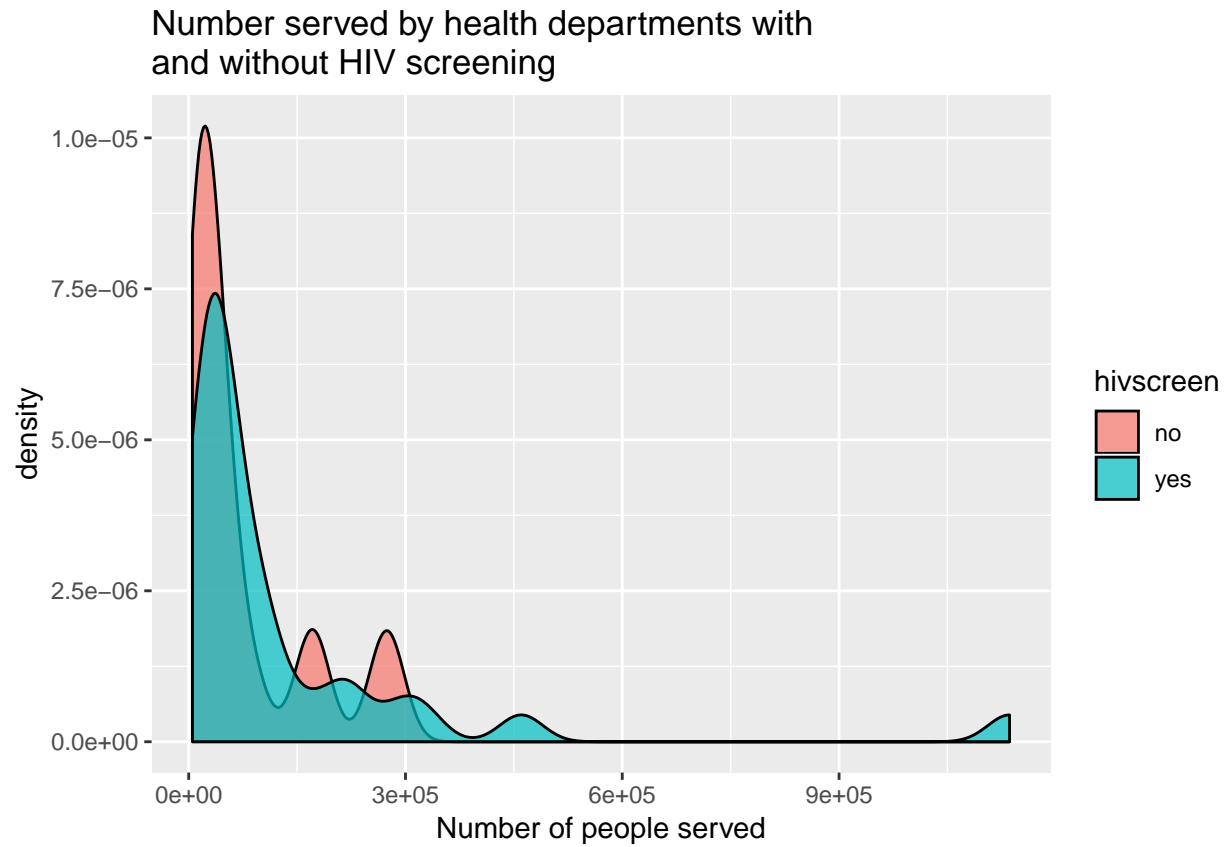
```
# add axis labels to plot
ggplot(data = lhd.noNA, aes(x = hivscreen, y = numserviced)) +
  geom_boxplot(outlier.shape = NA) +
  ylim(0, 125000) +
  ylab("Number of people served") +
  xlab("Does health dept provide HIV screening") +
  ggtitle("Number served by health departments with\nand without HIV screening")
```

```
## Warning: Removed 11 rows containing non-finite values (stat_boxplot).
```



Another option for visualizing the distribution of a continuous or discrete variable across groups is the petal plot. Try it:

```
# use the density geom to make a petal plot
# add alpha for transparency
ggplot(data = lhd.noNA,
       aes(x = numserve, fill = hivscreen)) +
  geom_density(alpha=I(.7)) +
  xlab("Number of people served") +
  ggtitle("Number served by health departments with\nand without HIV screening")
```

You try it!

Use numbers and a graph to compare expenditures for health departments with and without cancer screening programs. What can you conclude?

2. Two categorical variables

Sometimes we may want to know how two categorical variables are related. For example, are the health departments with no HIV screening also the health departments with no cancer screening? To figure this out, we want to know how many:

- health departments with both HIV and cancer screening (yes for both)
- health departments with HIV screening but no cancer screening
- health departments with cancer screening but no HIV screening
- health departments with neither type of screening (no for both)

Using the table function in R to find these numbers...

```
# table of hiv and cancer screening
table(lhd$hivscreen, lhd$cancerscreen)
```

```
##
##      no yes
## no  15   3
## yes 10  19
```

Well, those are the numbers! But, which is which? Are the rows HIV screening and the columns cancer screening? There must be a way that shows us more information. Try adding the variable names to the table command:

```
# add variable info to table
table(hiv = lhd$hivscreen, cancer = lhd$cancerscreen)
```

```
##      cancer
## hiv  no yes
## no  15   3
## yes 10  19
```

Better! But still not perfect. It would be great to know the percentages instead to be able to compare how people are distributed across categories.

```
# find percentages
prop.table(table(hiv = lhd$hivscreen,
                  cancer = lhd$cancerscreen))
```

```
##      cancer
## hiv  no      yes
## no  0.31914894 0.06382979
## yes 0.21276596 0.40425532
```

Interpretation: Just over 40% of health departments in the sample have both services, while 31.9% have neither service. Of the health departments 21.3% have HIV screening but no cancer screening and 6.4% have cancer screening but no HIV screening.

Visually, a facet_grid, grouped bar plot, or mosaic works well to display the relationship between two categorical variables. Install the package vcd to use the mosaic plot commands.

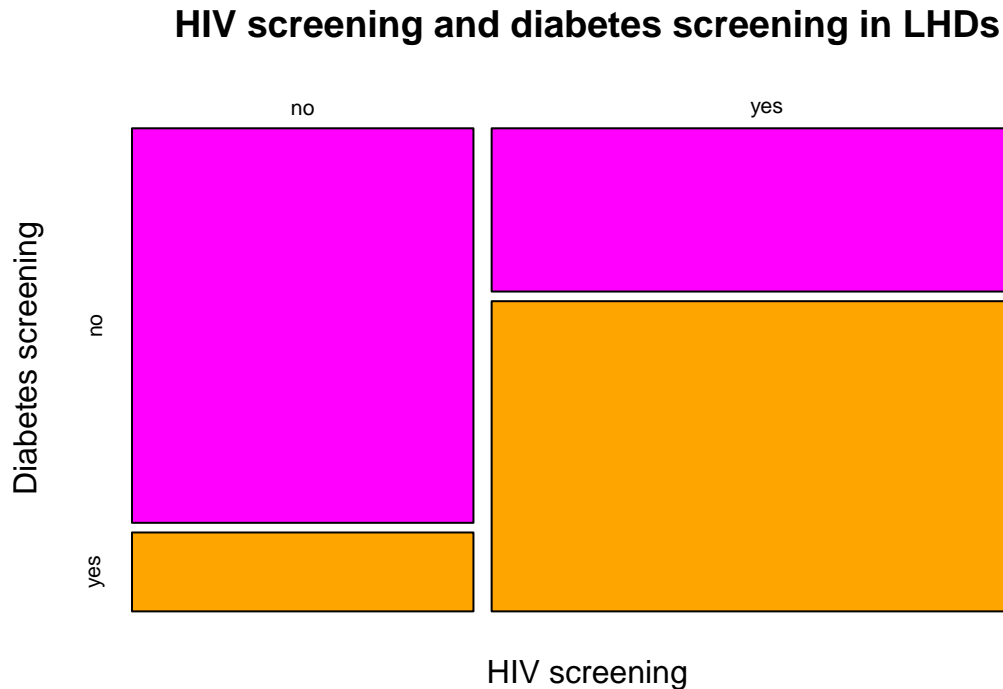
```
# open vcd package
library(vcd)
```

```
## Warning: package 'vcd' was built under R version 3.5.1
```

```
## Loading required package: grid
```

```
# make the mosaic plot
mosaicplot(~hivscreen+cancerscreen,
```

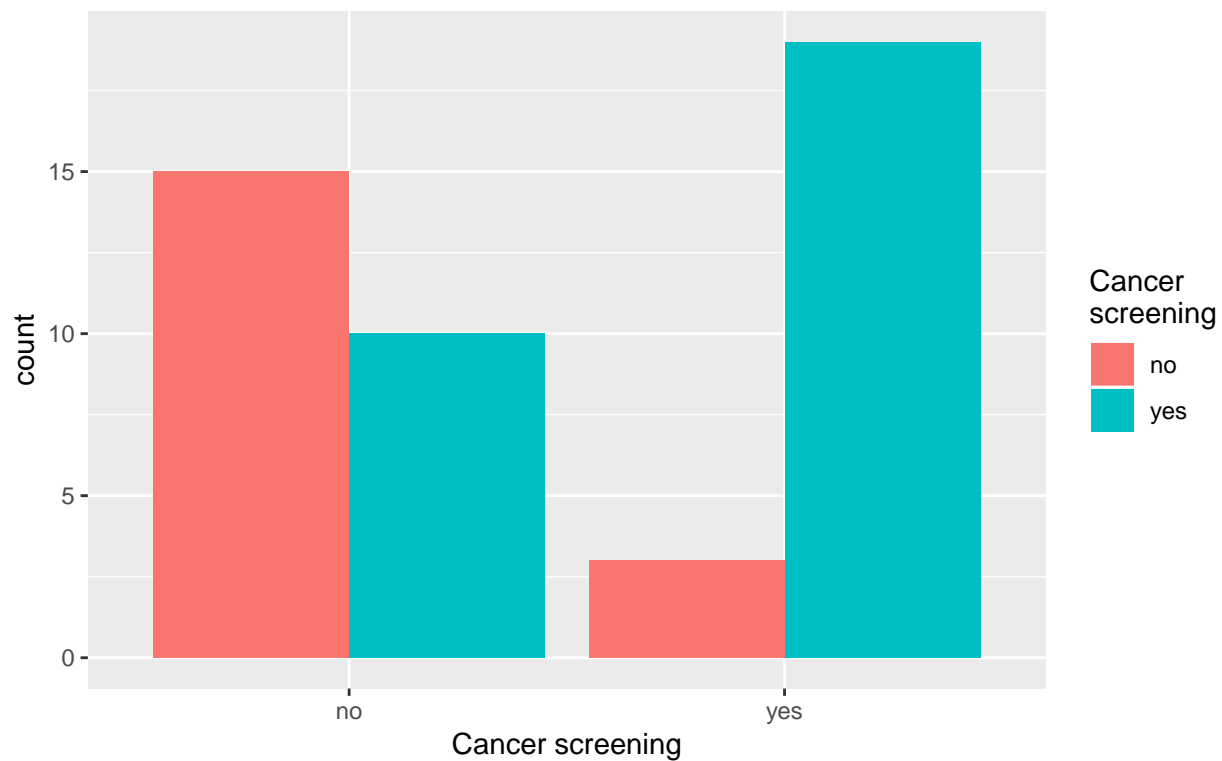
```
data=lhd, color=c("magenta","orange"),
xlab="HIV screening", ylab="Diabetes screening",
main="HIV screening and diabetes screening in LHDs")
```



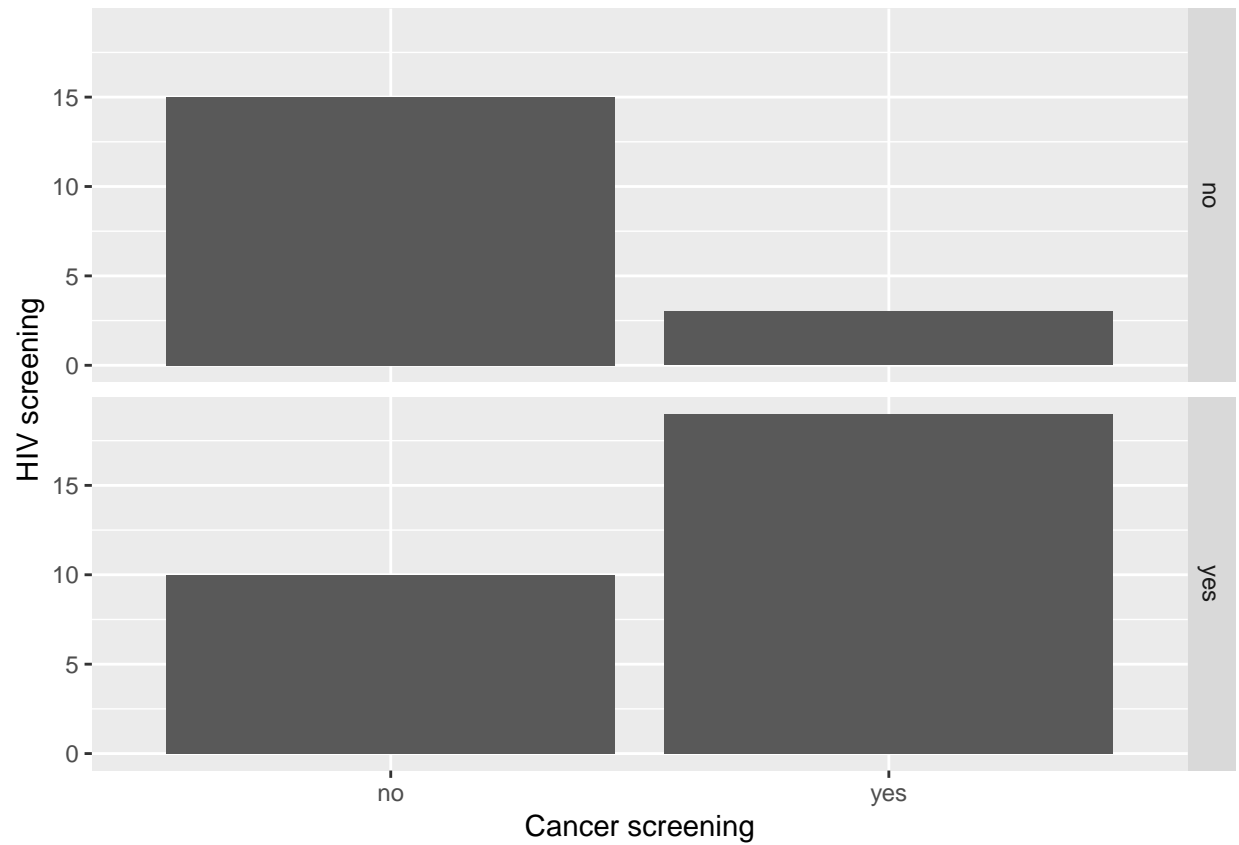
```
# try bar plots with facets or groups
# subste data so there are no NA values in cancer screen
lhd.noNA.screen <- subset(lhd.noNA, !is.na(cancerscreen))

# bar plot with groups
ggplot(data = lhd.noNA.screen,
       aes(x = cancerscreen, fill = hivscreen)) +
  geom_bar(position = "dodge") +
  xlab("Cancer screening") +
  labs(fill = "Cancer\nscreening") +
  ggtitle("Cancer screening at health departments with\nand without HIV screening")
```

Cancer screening at health departments with and without HIV screening



```
# bar plot with facets
ggplot(data = lhd.noNA.screen,
       aes(x = cancerscreen)) +
  geom_bar() +
  facet_grid(rows = vars(hivscreen)) +
  xlab("Cancer screening") +
  ylab("HIV screening")
```



```
ggtitle("Cancer screening at health departments with\nand without HIV screening")
```

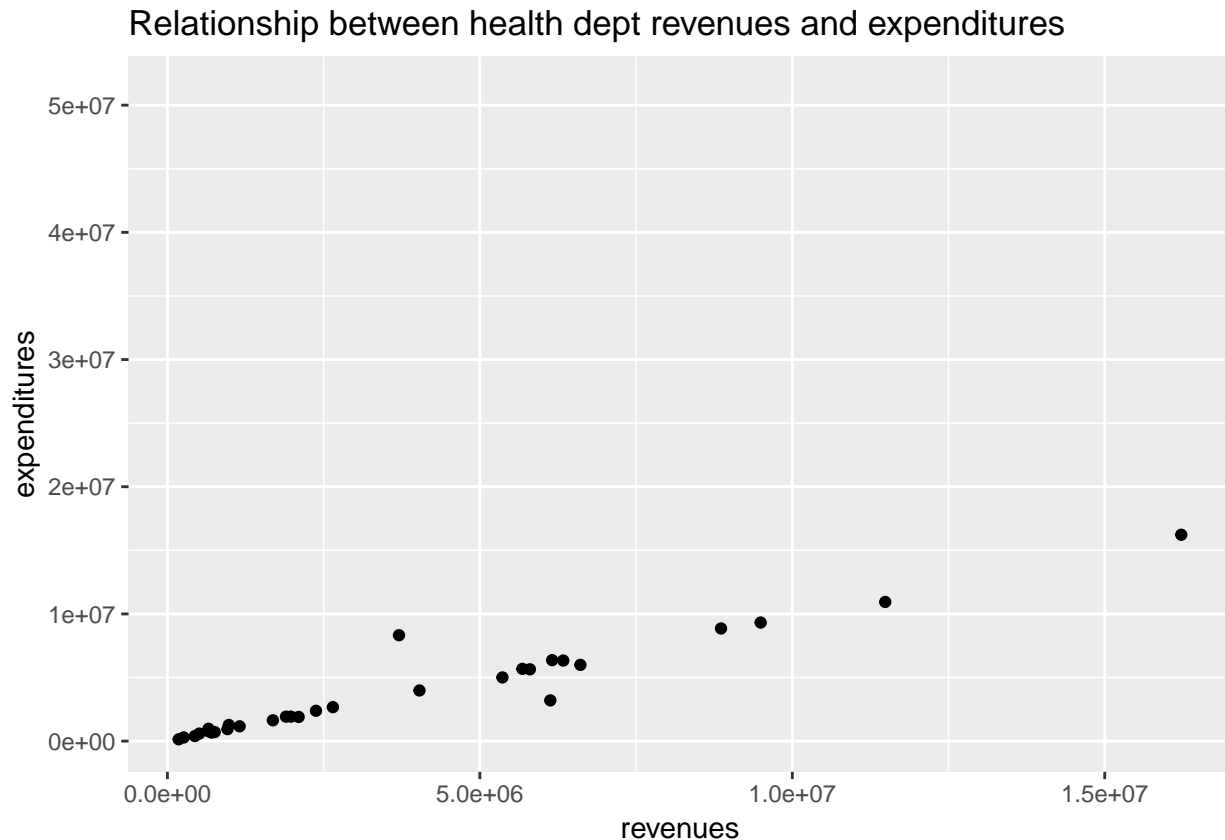
```
## $title
## [1] "Cancer screening at health departments with\nand without HIV screening"
##
## $subtitle
## NULL
##
## attr("class")
## [1] "labels"
```

3. Two continuous variables

It would make sense that health departments that bring in more revenues would use that money and have higher expenditures. Let's see if that is the case:

```
# graph revenues by expenditures
ggplot(data = lhd, aes(x = revenues, y = expenditures)) +
  geom_point() +
  ggtitle("Relationship between health dept revenues and expenditures")
```

```
## Warning: Removed 20 rows containing missing values (geom_point).
```



It looks like, as revenues go up, expenditures go up!

We can check this numerically using several different methods. The first method is covariance, which measures whether two variables vary together (co-vary) using this formula:

$$cov_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{n - 1}$$

If the covariance is a positive number, there is a positive relationship between the two variables. Revenues and expenditures looked like it had a positive relationship; the covariance can help quantify it. Note that the *covariance* command is like the *mean* command in that it needs to know what to do with missing values. In this case, instead of *na.rm*, the *cov* command uses *use* to determine which values to use. See the help documentation for information on all the possible options for *use*.

```
# covariance of revenues and expenditures
cov(lhd$revenues,lhd$expenditures, use='complete')
```

```
## [1] 1.439844e+13
```

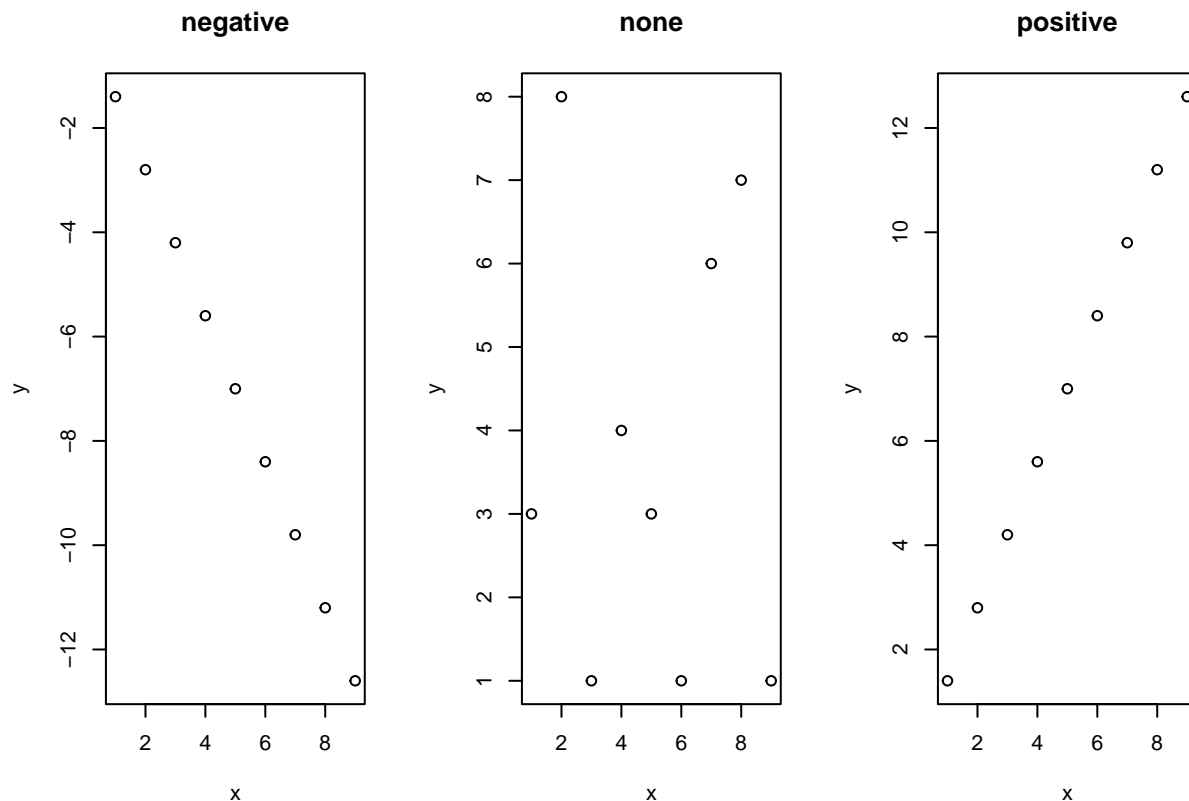
Well, this seems a little strange. The e+13 is scientific notation indicating that the decimal place should be moved 13 places to the right of where it currently is, so the variance is actually 1,440,543,000,000. A number this big is not really very useful beyond being able to say that the relationship is positive.

Because the covariance measure is highly influenced by the size of the numbers used to compute it (we had very large numbers for revenues and expenditures, so we had a very large covariance), it is not all that useful. Instead we use a version of the covariance divided by the standard deviations of x and y. This is called a correlation coefficient and is computed:

$$r_{xy} = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{(n-1)s_x s_y}$$

This version of the correlation coefficient is called Pearson's r and can range from -1 (a perfect linear negative relationship) to 0 (no relationship) to 1 (a perfect linear positive relationship).

- *Negative correlations* are when one variable goes up, the other goes down
- *No correlation* is when there is no discernable pattern in how two variables vary
- *Positive correlations* are when one variable goes up, the other also goes up (or when one goes down the other does too); both variables move together in the same direction

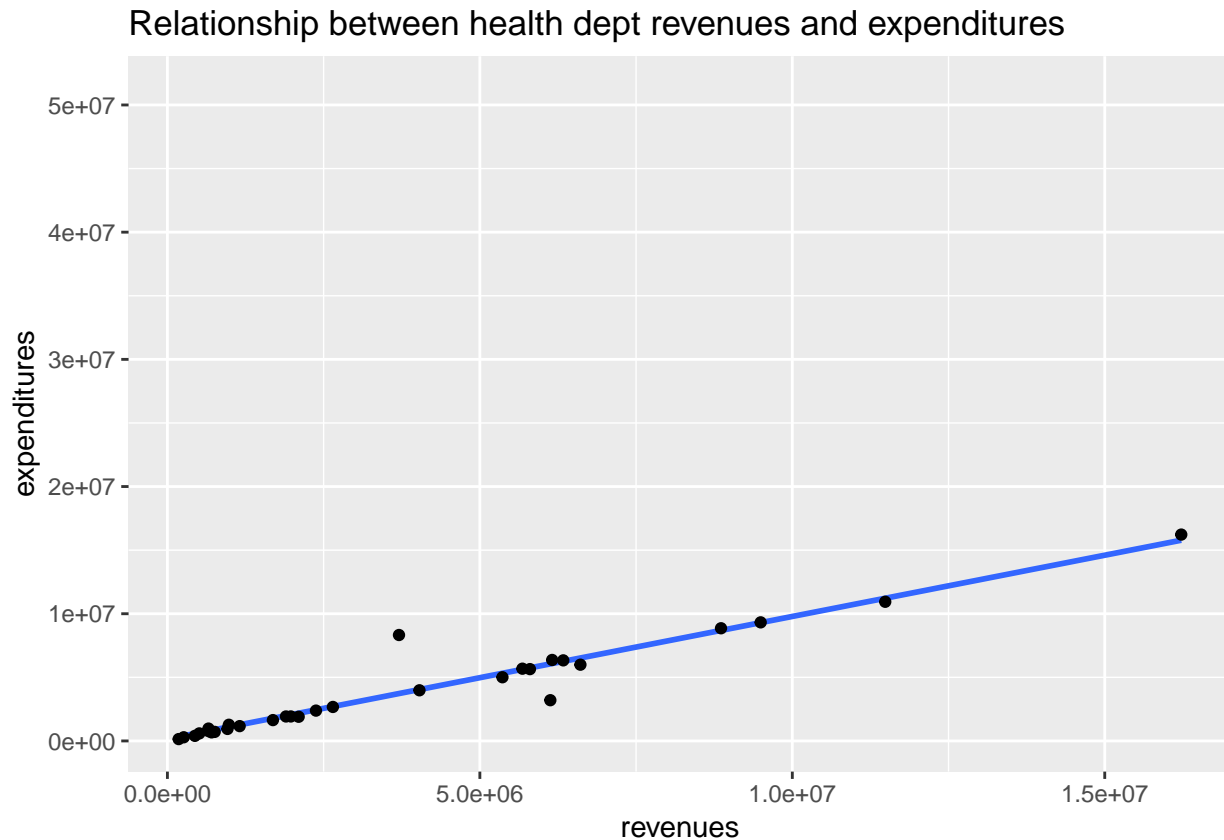


If we look again at the graph of revenues and expenditures and use the `geom_smooth` option to add a line to show the relationship, we can see that it is a strong positive relationship:

```
# scatterplot of revenues and expenditures
ggplot(lhd, aes(revenues, expenditures)) +
  geom_smooth(method = "lm", se = FALSE) +
  geom_point()+
  ggtitle("Relationship between health dept revenues and expenditures")
```

```
## Warning: Removed 20 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 20 rows containing missing values (geom_point).
```

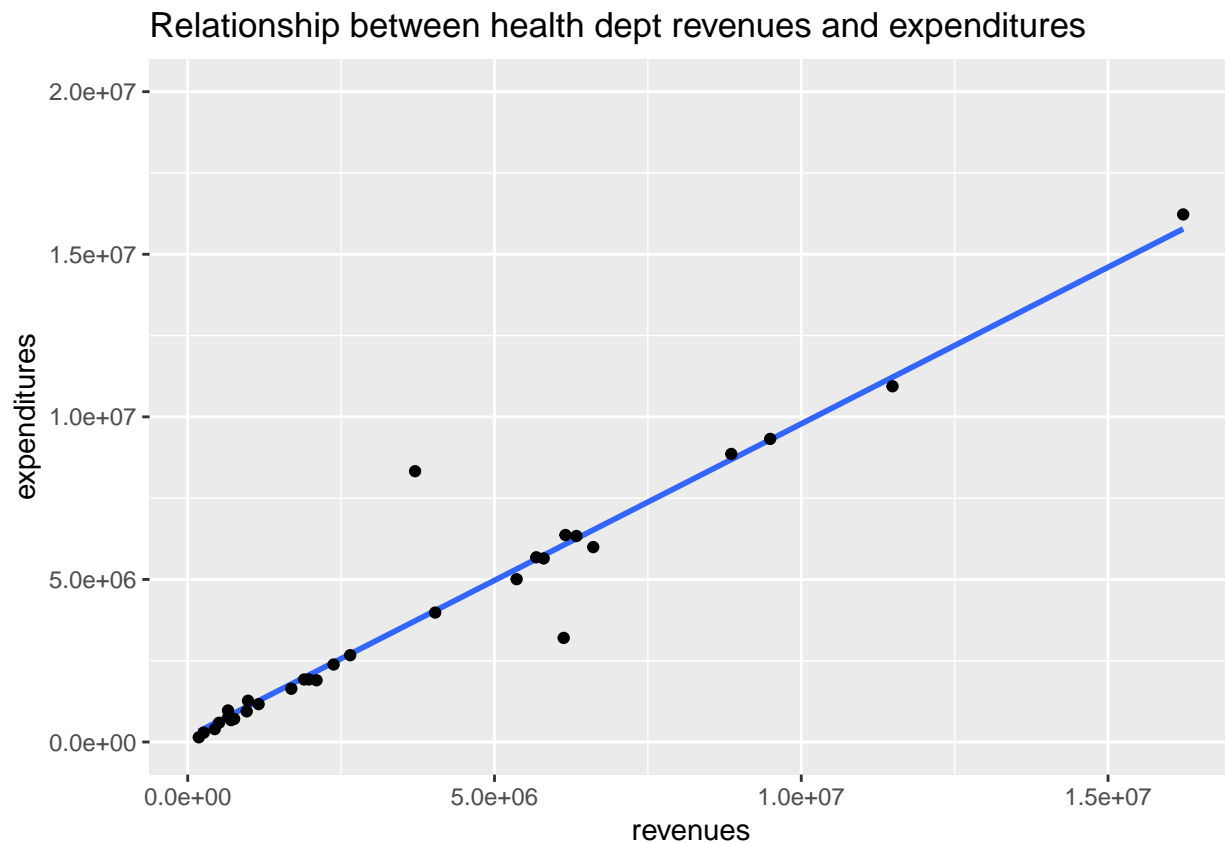


It does seem strange that the y-axis values go so much higher than the highest y value in the data set, so we can limit the y-axis for a better visualization:

```
# graph with limits on y-axis
ggplot(lhd, aes(revenues, expenditures)) +
  geom_smooth(method = "lm", se = FALSE) +
  geom_point()+
  ggtitle("Relationship between health dept revenues and expenditures")+
  ylim(0,20000000)
```

```
## Warning: Removed 20 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 20 rows containing missing values (geom_point).
```

Better!

A correlation coefficient can help us quantify the relationship:

```
# correlation between revenues and expenditures  
cor(lhd$revenues, lhd$expenditures, use='complete')
```

```
## [1] 0.9641266
```

A correlation of .96 is positive and very strong since 1.0 is the largest positive correlation possible.

Interpretation: *Health department revenues are strongly and positively correlated with health department expenditures ($r=.96$). As revenues go up, expenditures also go up.*

FoB3 Challenge

Install the `rmarkdown` package.

Download the standard or hacker version of the Challenge from GitHub and follow the instructions.

Upload whichever version you chose **before the next class meeting**.