R preparation: Calculating Correlations

Quantitative Reasoning

Preparation for class on 2020-09-28

In Chapter 6, we will be going through the scatter plot as well as correlation. We will be using data from the survey you completed prior to the mid-semester break in order to practice calculating correlations. As a preparation for our next lesson, let's calculate the correlation between height and shoe size.

The formula of correlation is shown below:

$$r = \frac{\sum(x-\bar{x})(y-\bar{y})}{\sqrt{(x-\bar{x})^2(y-\bar{y})^2}} = \frac{\sum(x-\bar{x})(y-\bar{y})}{(n-1)S_xS_y}$$

Import `survey.csv` and Inspect Variables

To begin, lets import `survey.csv` and inspect our variables of interest: `height` and `shoe`.

```r
survey <- read.csv("survey.csv")
str(survey)
```

```
## 'data.frame':    213 obs. of  9 variables:
##  $ gender     : Factor w/ 3 levels "Female","Male",..: 1 3 1 1 1 1 2 2 2 2 ...
##  $ nationality: Factor w/ 3 levels "Non-Singaporean",..: 3 1 3 1 1 1 1 1 3 1 ...
##  $ height     : num  164 178 170 167 163 160 172 177 NA 186 ...
##  $ phone      : int  42 63 22 94 41 60 53 12 72 NA ...
##  $ facebook   : int  NA 7 27 33 185 10 27 0 NA 212 ...
##  $ youtube    : num  8.90e+04 4.50e+04 1.10e+06 2.44e+05 1.79e+08 ...
##  $ shoe       : num  37 28.7 25.4 39 39 23.9 28 28 24.7 29.4 ...
##  $ postcode   : num  1 3 1 0 NA 4 1 0 1 0 ...
##  $ boxoffice  : num  7.91e+07 2.58e+08 2.88e+07 6.15e+07 3.55e+08 ...
```

As you should recall if you have already watched the R tutorial that introduces `cor()`, missing values can pose a problem when calculating correlations. Before getting started, let's also inspect our two variables of interest to see if there are missing values.

```r
table(is.na(survey$height))
```

```
## 
## FALSE  TRUE 
##   209     4
```

```r
table(is.na(survey$shoe))
```

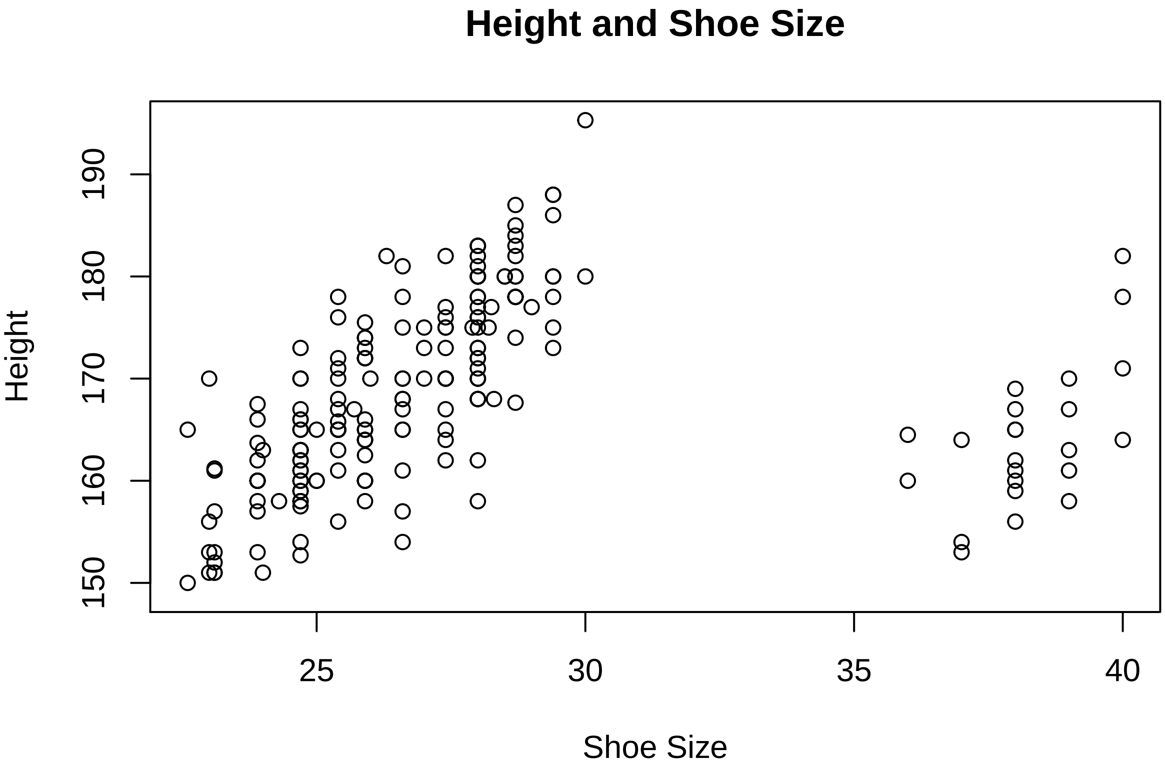
```
## 
## FALSE  TRUE 
##   207     6
```

So it appears that there are occasional missing values in our two variables of interest. This is important to keep in mind as we go forward.

Scatterplot of Height and Shoe Size

Before we calculate the correlation between height and shoe, let's also plot a scatter plot of the data to get an intuitive sense of the relationship between the two variables.

```
plot(height~shoe, data=survey,
     main="Height and Shoe Size",
     ylab="Height",
     xlab="Shoe Size")
```



Calculate Correlation Manually

Based on the formula for correlation shown at the start of this sheet, let's manually calculate the correlation between height and shoe using R. Since we know that there are missing values in both of our variables of interest and that missing values can cause issues when calculating the correlation coefficient, lets create a new data frame that only contains the columns height and shoe, and from which we purge all observations for which height or shoe or both are missing values. We will call this new data frame svy.narm.

```
svy.narm <-na.omit(survey[, c("height","shoe")]) # create new data frame
```

Let's quickly check to see if our two data frames have different numbers of rows. Since we omitted any observations with missing values from survey when we created svy.narm, then we should expect svy.narm to have fewer rows than survey. Let's check using nrow().

```
nrow(survey)
```

```
## [1] 213
```

```
nrow(svy.narm)
```

```
## [1] 205
```

Indeed, `svy.narm` has 205 rows while `survey` has 213 rows. We will use `svy.narm` for the next few steps.

OK, so lets now manually calculate the correlation between `height` and `shoe`. In order to do so, we need to first get the mean and standard deviation of both `height` and `shoe`.

```
height_mean <- mean(svy.narm$height)
```

```
## [1] 168.3309
```

```
shoe_mean <- mean(svy.narm$shoe)
```

```
## [1] 27.68415
```

```
height_sd <-sd(svy.narm$height)
```

```
## [1] 9.005957
```

```
shoe_sd <-sd(svy.narm$shoe)
```

```
## [1] 4.173487
```

Let's just quickly double check to see that we have 205 non-missing values for each of our variables of interest.

```
sum(!is.na(svy.narm$height) # number of observations with height
```

```
## [1] 205
```

```
sum(!is.na(svy.narm$shoe)) # number of observations with shoe size
```

```
## [1] 205
```

Great! Now lets use the formula from the beginning of the sheet and plug in the descriptive stats we just calculated above in order to manually calculate the correlation.

```
sum((svy.narm$height-height_mean)*(svy.narm$shoe-shoe_mean)/(height_sd*shoe_sd))/(nrow(svy.narm)-1)
```

```
## [1] 0.1736641
```

`cor()`: return the correlation

If we wish to be a bit more efficient in how we calculate correlations, then perhaps we might use the `cor()` command in R. Let's run it using the data frame `survey`.

```
cor(survey$height, survey$shoe)
```

```
## [1] NA
```

Ah! We get an NA value! To recall, we didn't purge the missing values from `survey`, so they are passed to `cor()`, and we get NA as our output. Rather than substitute `svy.narm` for `survey`, we might try to use the argument `use` to address this issue and purge any incomplete observations with missing values. Let's try that:

```
cor(survey$height,survey$shoe,use="complete.obs")
```

```
## [1] 0.1736641
```

So, it appears that when we specify the "complete.obs" option for the argument `use` in `cor()`, we get the exact same value for the correlation as when we calculated is manually using `svy.narm` from which we had purged all of the missing values. Much more efficient!