Foundations of Data Analysis

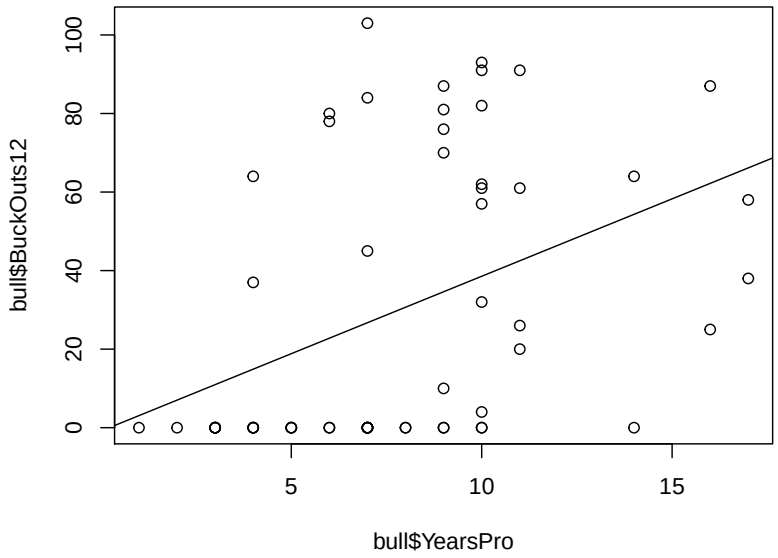
The University of Texas at Austin

R Tutorials: Week 3

Correlation

In this R tutorial, we're going to learn how to calculate the correlation between two numeric variables. And for this video, we're going to be using the same variables we used in the previous video, where we made scatter plots of some of the variables in the bull-riding data set. Specifically, we were looking at the relationship between how many years these bull-riders have been professional riders, and how many buck outs they've completed in this season.

```
plot(bull$YearsPro, bull$BuckOuts12)
abline(lm(bull$BuckOuts12 ~ bull$YearsPro))
```



Also, we looked at how many events they've participated in in relation to the number of buck outs, or attempts to ride the bulls. So we can calculate the correlation between any pair of variables using a simple function in R, and that function is called simply "**cor()**". So for this "**cor()**" function, we can first start off by just giving it two variables. If we wanted the correlation between years pro and buck outs, I could simply take those variables – and I'm just going to pull them from the plot function since I've already written them out.

```
cor(bull$YearsPro, bull$BuckOuts12)
```

```
## [1] 0.4095777
```

R will see these variables. And if you submit, so we could see that the correlation between years pro and buck outs is negative 0.410. That makes sense because in our plot, we saw a positive sloping line. Again, more about that in the linear regression part of the course.

Now we could also see the same thing with our events and buck outs.

```
cor(bull$Events12, bull$BuckOuts12)
```

```
## [1] 0.9936924
```

If we put that in our cor function, we'll see this strong, positive correlation of 0.994, which again matches what we saw in the scatter plot. Now besides giving the "**cor()**" function pairs of variables separated by a comma, we could also ask it to give us a correlation matrix.

So we can use a trick here where we essentially make a vector of variable names. So I can name this vector whatever I want. And I'll call this one *myvars*, just because it's a set of variables that I'm denoting. Then I can use the assignment key and give it a vector of variable names. So to create a vector, remember, we're going to use the "**c()**" function. So if I want to look at years pro and events and buck outs, I can put all variable names here that I want into a single vector called myvars.

```
myvars <- c("YearsPro", "Events12", "BuckOuts12")
```

So I'm going to submit that. Now if I then ask for cor of the bull data frame, and then I can index this by saying I want certain rows and columns within my data frame. So remember, we're going to use the square brackets for indexing. I need to give it row comma columns, so I want every row because I want to include every case in my correlation. So I'm going to put basically nothing there. And then the columns I want to take are the ones that match myvars. So this is basically going to go in and look at where these columns are and give me a correlation across all three.

```
cor(bull[, myvars])
```

```
##             YearsPro  Events12 BuckOuts12
## YearsPro   1.0000000 0.4314794  0.4095777
## Events12   0.4314794 1.0000000  0.9936924
## BuckOuts12 0.4095777 0.9936924  1.0000000
```

And so what R does with that is produces a correlation matrix which has each of the variables in the rows and columns. And then the cells within this table correspond to the correlation between these two variables. So you'll always see ones down the diagonal because years pro is perfectly correlated with itself. And then we see our years pro correlation with buck outs – positive 0.410 – and events correlated with buck outs – positive 0.994. This correlation matrix also gave us this additional correlation here, which was the one we didn't run before, which was years pro correlated with events. And we see moderate positive relationship here.