Useful Tips for Your Data Science Final Report

Rafael Garcia & Lacey Hartigan

6/23/2022

I suggest starting with this code chunk:

I'll explain what the options mean here:
* include=FALSE ## Runs chunk but omits it from final Knitted file
* knitr::opts_chunk$set ## Sets default options for each chunk that follows
* echo=FALSE ## Do not display code in output (look at the knit file - you won't see the code there)
* warning=FALSE ## Do not display warning in output
* message=FALSE ## Do not display messages in output
* fig.align='center' ## Center align any figures (this includes graphs and kables)

When you save something as an object, it will not (by default) display the values for the object in your knit file.

```
#I am forcing the code to display so you can see it. I do this using echo=TRUE. The reason I have to do

test<-summary(cars) #"test" (the summary results) will not be displayed in the knit file
```

But calling the object by name *will*. This is what we refer to as "printing" in r-speak.

```
#I am forcing the code to display so you can see it.

summary(cars) #This WILL be displayed
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

This is why we **DO NOT** call an entire data file name after we have imported it or modified it, but instead use something like `head()` or `tail()` to view only a few cases. For your final reports, however, you should not need to do even that.

The table above looks crude though. We can "pretty" it up using `kable()`. We can do that a few different ways:

```
#I am forcing the code to display so you can see it.

kable(summary(cars)) #This WILL be displayed
```

speed	dist
Min. : 4.0	Min. : 2.00
1st Qu.:12.0	1st Qu.: 26.00
Median :15.0	Median : 36.00
Mean :15.4	Mean : 42.98
3rd Qu.:19.0	3rd Qu.: 56.00
Max. :25.0	Max. :120.00

Here's another way you can "kable" your tables using a pipe (%>%) and tidyverse:

```
#I am forcing the code to display so you can see it.
summary(cars) %>%
  kable() #Dropping it on the end also works.
```

speed	dist
Min. : 4.0	Min. : 2.00
1st Qu.:12.0	1st Qu.: 26.00
Median :15.0	Median : 36.00
Mean :15.4	Mean : 42.98
3rd Qu.:19.0	3rd Qu.: 56.00
Max. :25.0	Max. :120.00

You may have noticed that I did a direct code quote when I referenced `kable()`. I did it again just now! You can replicate this by using the backtick (it is the same symbol that starts the r chunks, located above the `TAB` key, not to be confused with the apostrophe).

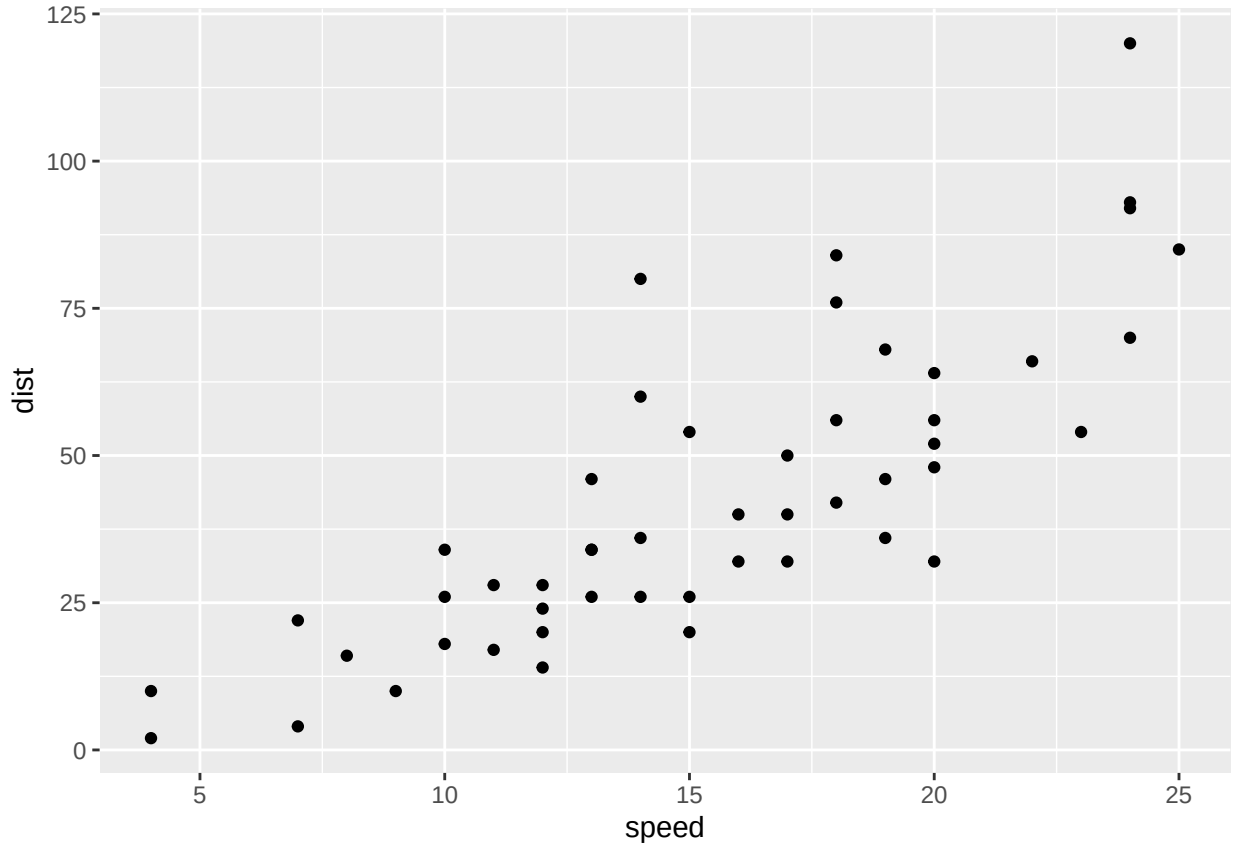
There may be a time you want to reference in text a specific value from some object you have saved. You can do that by using:

r object$value

The code above needs to be enclosed in backticks (‘). See the examples below. Modify the "object" to be your dataframe name & the "value" to be the variable name.

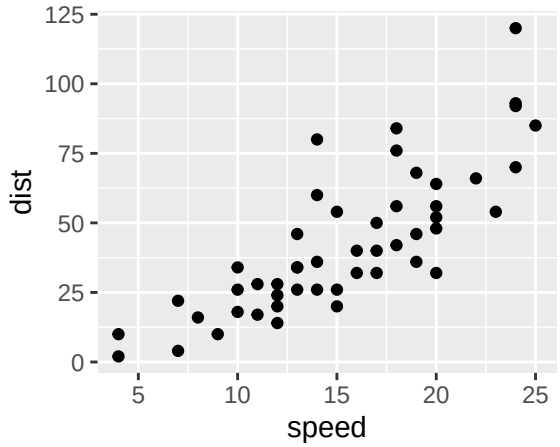
This is a weird example, but if you wanted to, say, plug in the mean of `speed` from the `cars` dataset (native to R), you could do something like: Mean = 15.4 mph. This can be very convenient if you are running an analysis where the numbers will fluctuate due to randomization (though you could also fix that problem by just fixing the seed value). (Look at the knit file to see how this displays in the output - pretty neat, huh!?)

If you want to adjust plot sizes, you can do that, too. I ran code to generate a scatter plot of `speed` vs `dist` from the `cars` dataset. Here is what it looks like with no modifications.

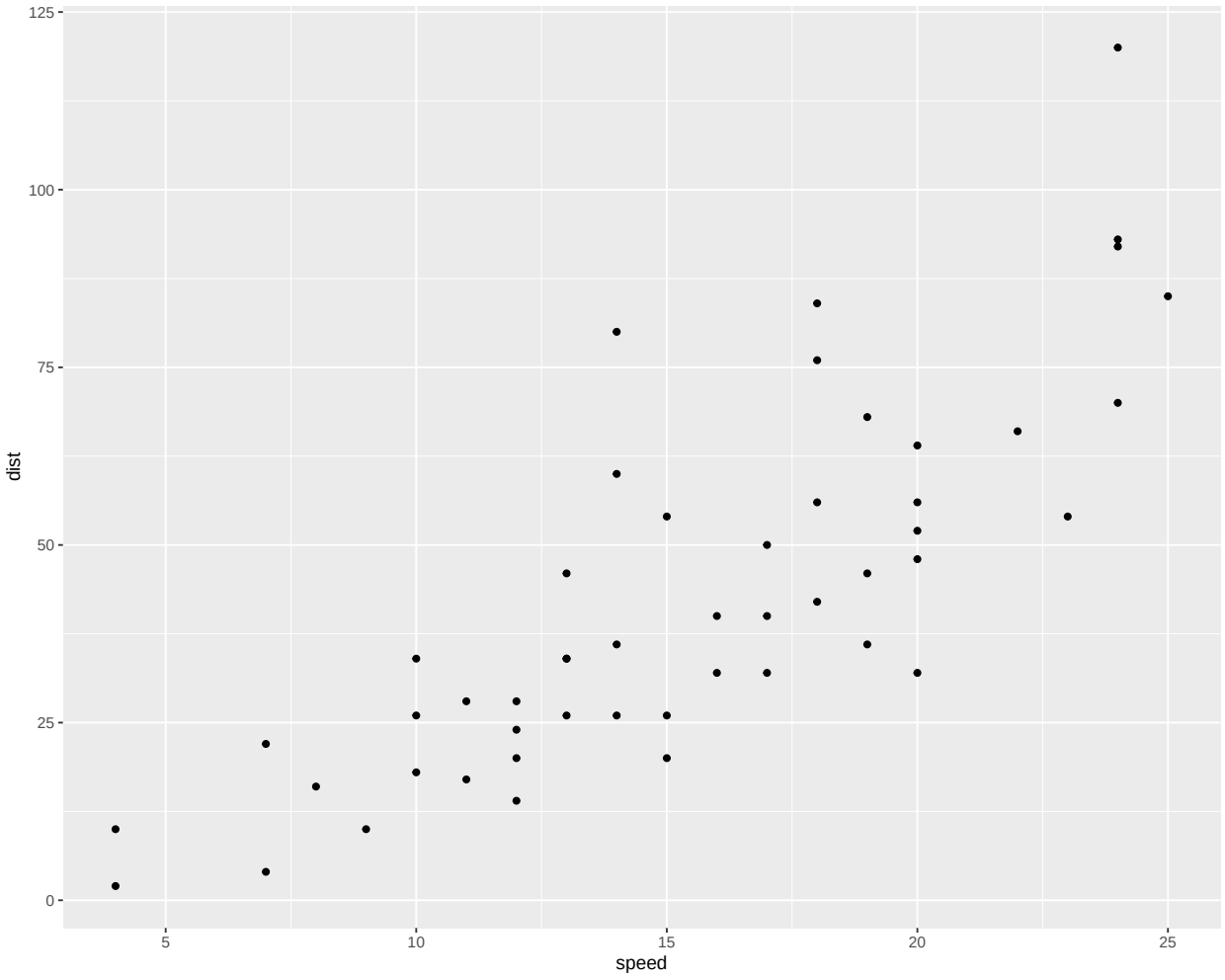


Here it is with some modifications:

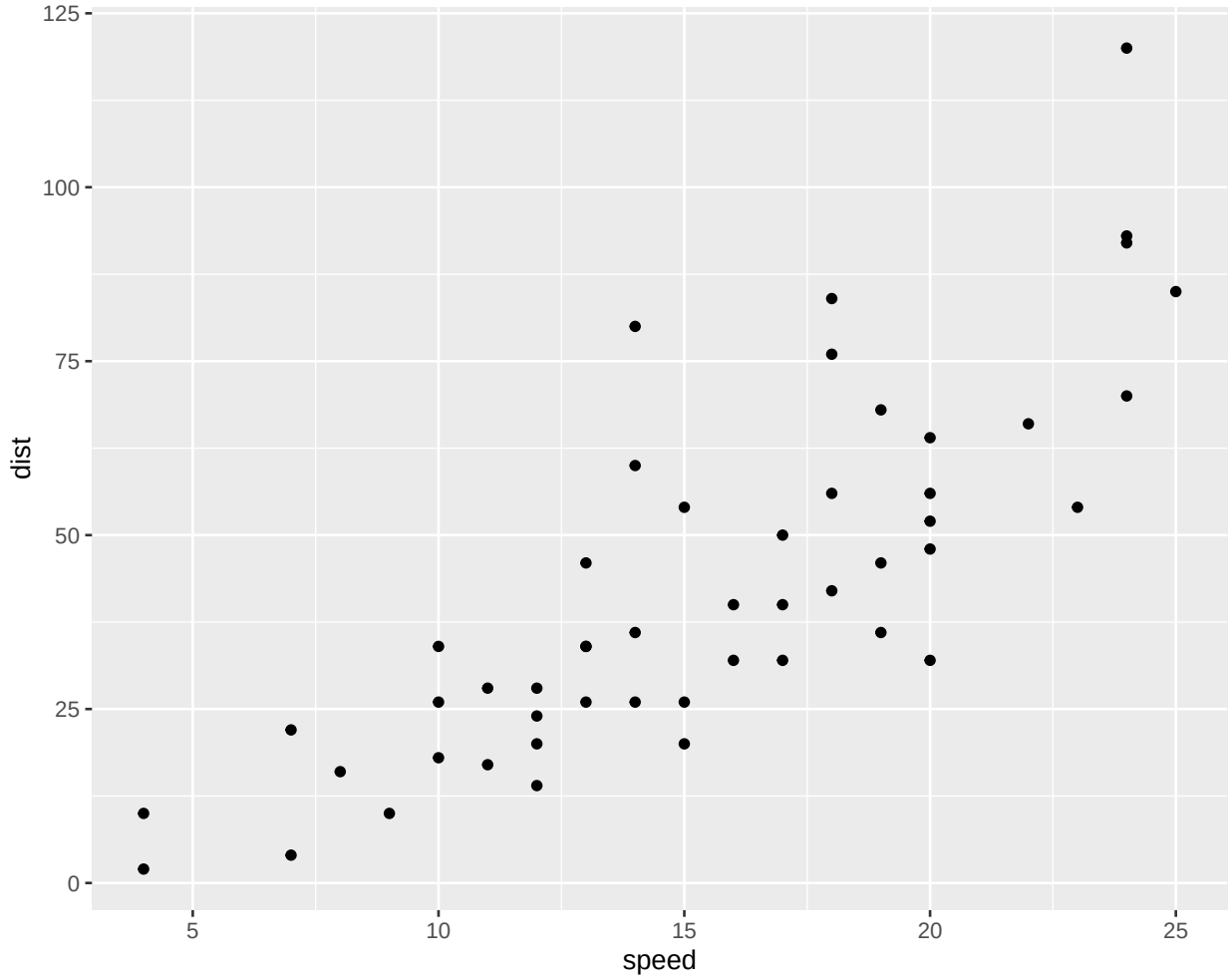
```
#I am forcing the code to display so you can see it.
# fig.asp=.8, figure.width=3
carplot
```



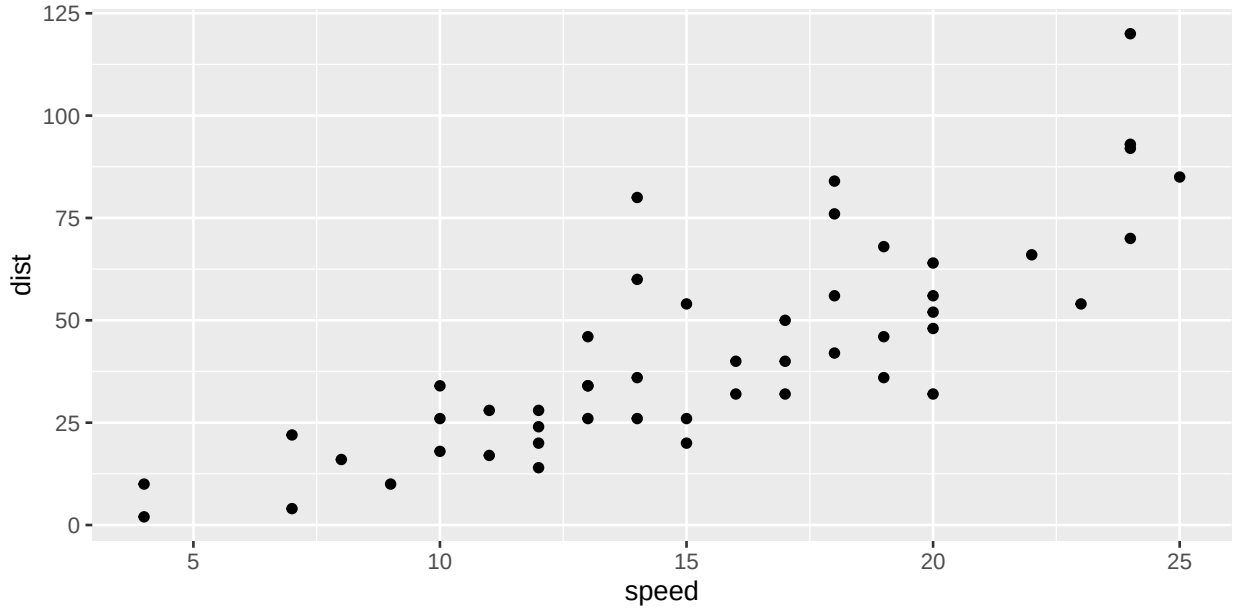

```
#I am forcing the code to display so you can see it.
# fig.asp=.8, figure.width=10
carplot
```




```
#I am forcing the code to display so you can see it.
# fig.asp=.8, figure.width=7
carplot
```




```
#I am forcing the code to display so you can see it.
# fig.asp=.5, figure.width=7
carplot
```



You will need to play around with the exact proportions to determine what you think looks best.

Other random .Rmd things to note:
* Line spacing. You'll notice that sometimes R knit files will have text that runs together (even though you put that text on multiple lines in your .Rmd file). If you end a line of text with at least 5 spaces, that makes it clear to R that you want a line break in your text.

Remember: this should look like a report that you would hand to a boss/supervisor. Don't include code or R messages as they make it look unprofessional (and your boss might not understand them anyway).

If you need help figuring out headers or other formatting conventions, there are some great guides on the course website Link 1 and Link 2.