# Suggestions for the improvements of RStudio

*senjiew*

*11/26/2017*

## Introduction

Throughout this semester, I have familiarized myself with the language R as both a programming language and a statistical analysis tool. As someone with little to no coding experience prior to the class, I found R to be relatively intuitive and that I will most likely continue to use it for my coding/data analysis needs in the future. However, that is not to say that R is perfect. The interface of RStudio can sometimes to troublesome to navigate. Therefore, there are many improvements I want to point out that the developers can easily fix going into the future to make R a better language for beginners like myself. Therefore, in the post, I will mainly discuss how I feel like R can improve based on my experiences with other programming languages. It will cover topics such as spelling, line debugging, and hints for good coding habits. I feel like it really is just the small things that RStudio can improve to be able to go a long way.

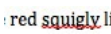
## Spelling

Right off the bat, I want to talk about how R and RStudio don't correct the user's spelling. Its console lets the user type in anything he/she wants without providing intelligent feedback. For example:

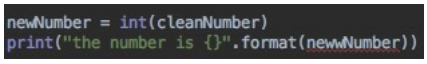
```
Hello <- c(1,2,3,4)

#print(Helo)
```

```
Error in print(Helo) : object 'Helo' not found
```

if I assign 'Hello' a vector, and for some reason printed 'Helo' in the end, it will not provide the results I wanted, 1 2 3 4, but instead I will get an error message telling me that 'Helo' not found. In a longer code, this could cause a beginner like myself to waste some serious time trying to find the bug where it would have been easier if the English dictionary was built in and show the red squiggly line that is present in text editors such as Word and IDEs like IntelliJ.




Example of spelling error in Word




Example of spelling error in IntelliJ

I would like to point out here that the spreadsheet Excel developed by Microsoft doesn't correct for spelling either, despite it being developed by the same company that created Word and PowerPoint, both of which will check the user's spelling.

More resources regarding basic interactions with RStudio can be found here, and more resources regarding basic interactions with IntelliJ for different coding languages such as Python and Java can be found here

More information about Microsoft Office can be found here

## Debugging

Another issue I have consistently had with RStudio is its often times unclear error messages, which can lead to difficult debugging. The error could sometimes be either very vague about what my mistake is or it could be just completely wrong in terms of what it really wants. For example:

```
> ![](pictures/error1.jpg)
Error: unexpected '[' in "!["
>
```

Here you can see that my code in R markdown is trying to run the error1 picture from my picture folder using a relative path. However, my working directory at the time was not in the right place, the code didn't not work. It couldn't find a picture folder in the then current working directory. It is clear that the error displayed here is very misleading, as it would lead an uninformed coder to believe that something is wrong with the bracket and that instead of [], it should have been !].

Furthermore, often times when I finish a project and attempt to knit it into a form suitable for github, a message like this will appear to halt the knitting.

```
Line 38 Error in filtr(dat, player == "Stephen Curry") : could not find function "filtr"
        Calls: <Anonymous> ... handle -> withCallingHandlers -> withVisible -> eval ->
        eval Execution halted
```

It shows that I have an error in line 38 with the spelling of "filtr" when it should have been "filter" when in fact it was in line 39.

```
37 ```{r}
38   dat <- read.csv("data/nba2017-players.csv",stringsAsFactors = FALSE)
39   filtr(dat, player == "Stephen Curry")
40 ```
```

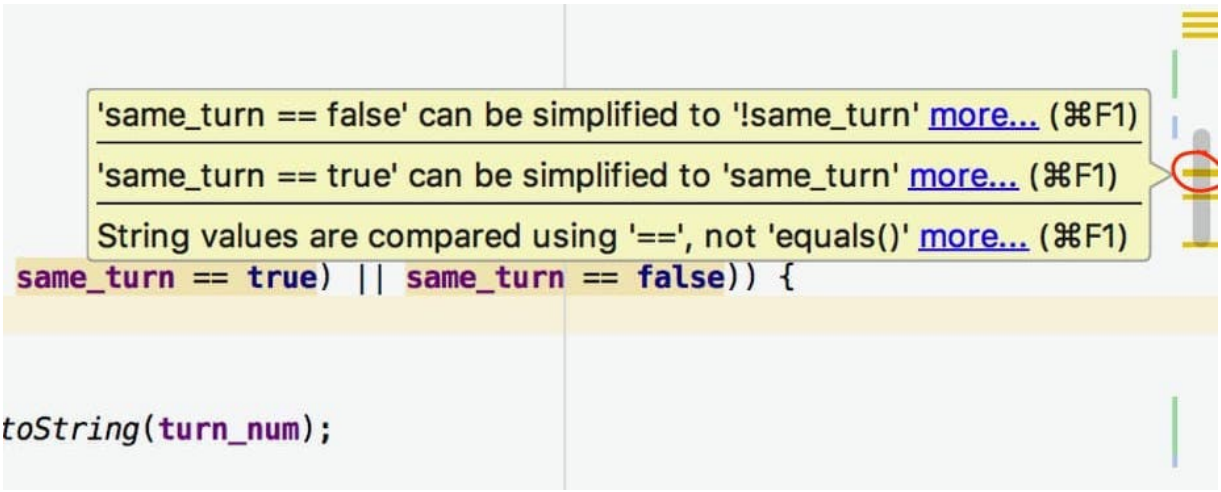
As this is only an example, the code shown here is relatively simple and error is easy to catch. However, sometimes the line error given can be off by a lot, which then can confuse beginners such as myself.

Here is some more resources concerning the debugging process in RStudio that might serve as better examples.

## Hints for good coding habits

Finally, I believe one last thing RStudio can improve on is incorporating little hints that encourage good coding habits, which will either make codes more efficient, or less prone to errors.

IntelliJ has this feature shown here:




As you can see, the yellow bars on the side indicate that I could have written my code in a better way. In the example shown above, my 'same_turn' == true and 'same_turn' == false terms can be simplified to '!same_turn' and just simply 'same_turn', which is less code for the same results, allowing my program to run more efficiently. In the long run, for beginners such as myself, it is more likely that I will be able to pick up these habits and adapt my style of coding for the better. Therefore, if R Studio can implement something like this and learn from other IDEs, I think R definitely can become a more user friendly language.

Here is a link for more information regarding IDEs, and here is a link for more information regarding IntelliJ, a specific IDE I have been talking about in this post

# Conclusion

With all that said, I want to point that developers for RStudio have consistently been updating the app and make it better in small but meaningful ways. An example of that can be found on this website. I hope that this post is not viewed as a criticism of R or Rstudio as I have genuinely enjoyed my time coding in R. I believe it is a beginner friendly language that can serve as a good introduction into the world of coding. This post is merely some of the things I picked up that I found to be inconvenient in RStudio that I though can be improved. Overall, I believe that R is a great data analysis tool that provides fast results and I will be looking forward to learn more about it in the future.

For more resources on R and other languages, a great place to start is Stack Overflow. It provides a platform for people all over the world to discuss coding related questions and it is completely free.

# References

1.http://www.r-tutor.com/r-introduction
2.https://examples.javacodegeeks.com/desktop-java/ide/intellij-idea-tutorial-beginners/
3.https://www.gcflearnfree.org/subjects/office/
4.https://support.rstudio.com/hc/en-us/articles/205612627-Debugging-with-RStudio
5.http://searchsoftwarequality.techtarget.com/definition/integrated-development-environment
6.https://www.jetbrains.com/idea/
7.https://blog.rstudio.com/2017/09/13/rstudio-v1.1---the-little-things/
8.https://stackoverflow.com/