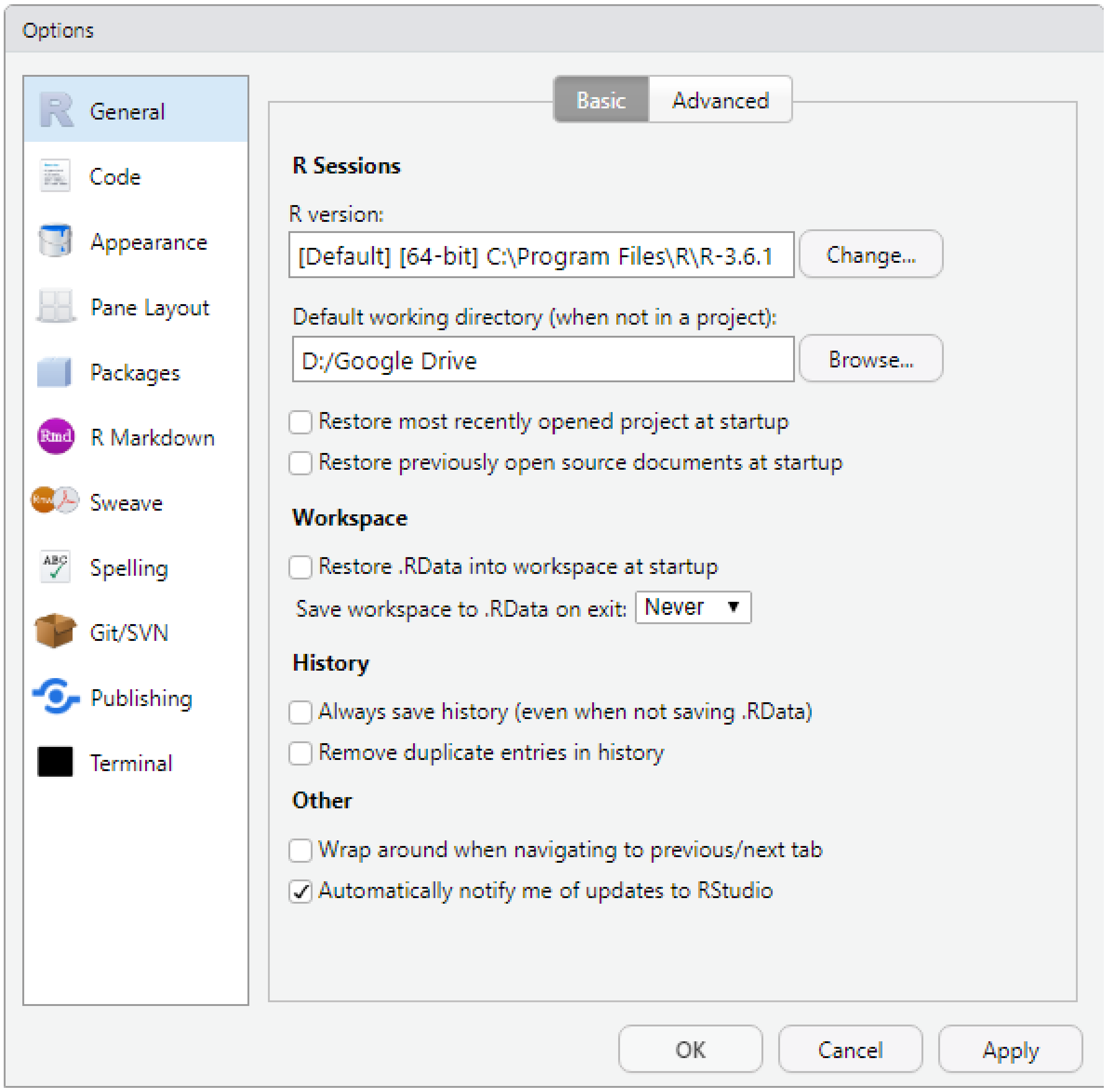
Lab 1

Adam Shen

September 16, 2020

(Optional) Modify RStudio Options



R Sessions

R version: [Default] [64-bit] C:\Program Files\R\R-3.6.1

Default working directory (when not in a project): D:/Google Drive

☐ Restore most recently opened project at startup

☐ Restore previously open source documents at startup

Workspace

☐ Restore .RData into workspace at startup

Save workspace to .RData on exit: Never

History

☐ Always save history (even when not saving .RData)

☐ Remove duplicate entries in history

Other

☐ Wrap around when navigating to previous/next tab

☑ Automatically notify me of updates to RStudio

Figure 1: Tools > Global Options...

Basics

Source and Console

The `Source` pane (top left) is where you will be writing your code. This pane is usually minimized until you create a new `.R` script. The `Console` pane (bottom left) is where your code is evaluated. It is good practice to write your code in a script rather than doing it directly in the console, for reproducibility. It is also easier to write multi-line code in a script and send it to the console for evaluation than to type it all in the console.

You can send a line of code from your script to the Console for evaluation using `Ctrl + Enter` (or `Cmd + Enter`).

Paths

A path is a string (must be surrounded by quotes) indicating the location of a folder or file. On Windows it may look something like:

```
"D:/Google Drive/Stat 3553/"
"C:/Users/Adam/Desktop/boeing.txt"
```

On Mac, it may look something like:

```
"/Google Drive/Stat 3553/"
"/Users/Adam/Desktop/boeing.txt"
```

You can find the path of a specific file on your computer through a file explorer window using:

```
file.choose()
```

The Working Directory

The working directory is the location where R will read/write files. When you open RStudio, the working directory is set to your default (which can be changed – see Page 1). When you open a script using RStudio through your file explorer, RStudio will automatically set your working directory to the folder where the script was located.

Your current working directory is written at the top of the Console pane. You can also find your current working directory by typing in the console:

```
getwd()
```

```
## [1] "D:/Google Drive/Resources/Stat 3553/Lab 1"
```

You can change your working directory using `setwd()` and a valid path (based on your operating system):

```
setwd("C:/Users/Desktop/A New Folder/")
```

You can also specify paths relative to where you currently are. If my current working directory is

```
"D:/Google Drive/Resources/Stat 3553/"
```

and I want to access

```
"D:/Google Drive/Resources/Stat 3553/Lab 99/"
```

I can use the path string

```
"./Lab 99/"
```

You can also go up one level (the preceeding folder of where you are) using `"../"`. If my current working directory is

```
"D:/Google Drive/Resources/Stat 3553/Lab 1/"
```

and I want to access

```
"D:/Google Drive/Resources/Stat 3553/Lab 99/"
```

I can use the path string

```
"../Lab 99/"
```

Creating Variables

Values can be assigned to variables using `<-` or `=`, however it is recommended to use `<-` for assignment and `=` for setting parameter values of a function. E.g.

```
# Set `z` to 1.645 using `<-`
z <- 1.645

# Find area to the right of `z` on the standard normal distribution. Set function parameters
# using `=`. This output is not stored in a variable.
pnorm(z, lower.tail=FALSE)
```

```
## [1] 0.04998491
```

Notice that when we assign values to a variable, there is no output. You may wish to see output to verify that you have assigned the correct value to your variable. This can be done by surrounding your assignment with brackets. E.g.

```
(num1 <- 3+4*10)
```

```
## [1] 43
```

```
(num2 <- (3+4)*10)
```

```
## [1] 70
```

Function Documentation

If you are unsure what a function does or you want to see the parameters of the function, append a `?` before your function and type it in the console. The documentation will show up in the Help tab of the bottom right pane. E.g.

```
?pnorm
?read.table
```

Boeing Data

Load the data

Avoid calling your data variable `data`. I don't recommend using `attach`/`detatch`...

```
airline <- read.table("./boeing.txt", header=TRUE)
```

Preview the data

We can preview the first few rows of the data using:

```
head(airline)
```

```
##   Passengers Cost
## 1         61 4.28
## 2         63 4.08
## 3         67 4.42
## 4         69 4.17
## 5         70 4.48
## 6          74 4.30
```

We can obtain the names of the variables in the data set using:

```
names(airline)
```

```
## [1] "Passengers" "Cost"
```

If we wish to access the `Cost` variable in the `airline` data set, we can do so in one of many ways:

```
airline$Cost
```

```
##  [1] 4.28 4.08 4.42 4.17 4.48 4.30 4.82 4.70 5.11 5.13 5.64 5.56
```

```
airline[,2] # All rows of column 2
```

```
##  [1] 4.28 4.08 4.42 4.17 4.48 4.30 4.82 4.70 5.11 5.13 5.64 5.56
```

```
airline[,"Cost"] # All rows of column "Cost"
```

```
##  [1] 4.28 4.08 4.42 4.17 4.48 4.30 4.82 4.70 5.11 5.13 5.64 5.56
```

Fit a simple linear regression model

```
m1 <- lm(Cost ~ Passengers, data=airline)
```

Commonly used functions for linear regression models

```
summary(m1)
```

```
##
## Call:
## lm(formula = Cost ~ Passengers, data = airline)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.28171 -0.14938  0.04101  0.13162  0.22741
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 1.569793   0.338083   4.643 0.000917 ***
## Passengers  0.040702   0.004312   9.439 2.69e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1772 on 10 degrees of freedom
## Multiple R-squared:  0.8991, Adjusted R-squared:  0.889
## F-statistic: 89.09 on 1 and 10 DF,  p-value: 2.692e-06
```

```
anova(m1)
```

```
## Analysis of Variance Table
##
## Response: Cost
##            Df  Sum Sq Mean Sq F value    Pr(>F)
## Passengers  1 2.79803 2.79803  89.092 2.692e-06 ***
## Residuals  10 0.31406 0.03141
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
fitted(m1)
```

```
##        1        2        3        4        5        6        7        8 
## 4.052590 4.133993 4.296800 4.378203 4.418905 4.581711 4.663114 4.866622 
##        9       10       11       12 
## 5.070130 5.273638 5.436445 5.517848
```

```r
resid(m1)
```

```
##           1           2           3           4           5           6 
##  0.22740971 -0.05399349  0.12320012 -0.20820308  0.06109532 -0.28171107 
##           7           8           9          10          11          12 
##  0.15688573 -0.16662226  0.03986975 -0.14363825  0.20355536  0.04215216
```

```r
confint(m1)
```

```
##                  2.5 %     97.5 %
## (Intercept) 0.81649700 2.32308856
## Passengers  0.03109358 0.05030962
```

Predictions

```r
predict(m1, level=0.95, interval="confidence", se.fit=TRUE, newdata=data.frame(Passengers=80))
```

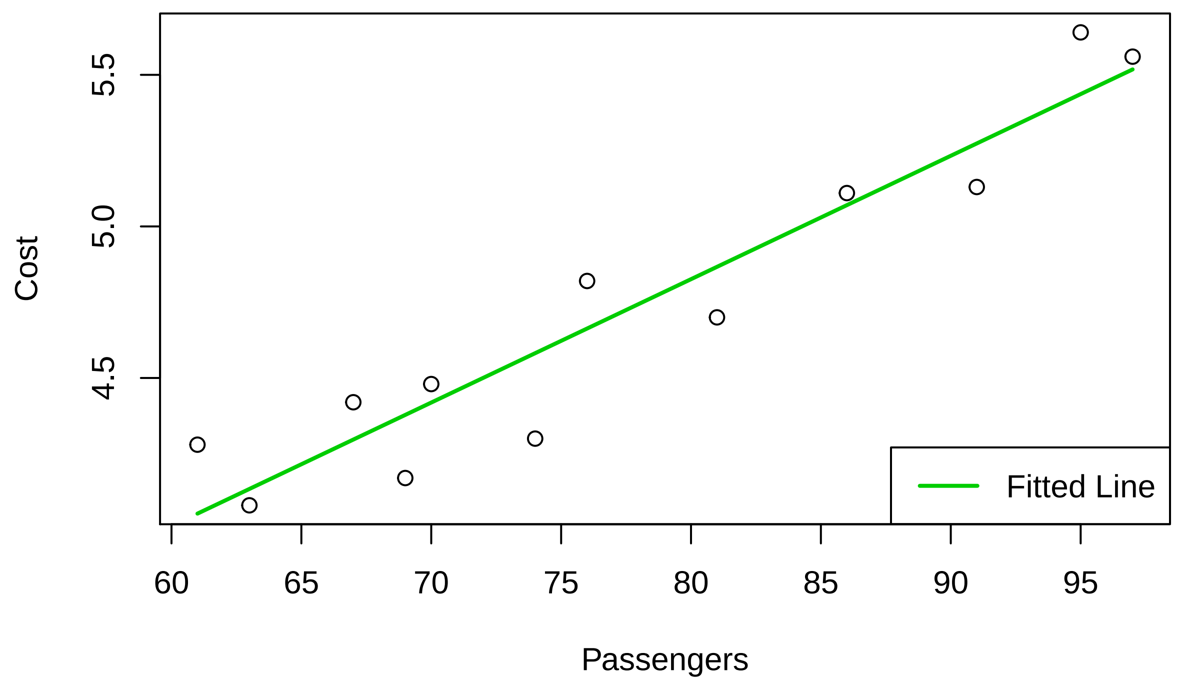
```
## $fit
##        fit     lwr      upr
## 1 4.825921 4.70943 4.942412
## 
## $se.fit
## [1] 0.05228178
## 
## $df
## [1] 10
## 
## $residual.scale
## [1] 0.1772175
```

```r
predict(m1, level=0.95, interval="prediction", se.fit=TRUE, newdata=data.frame(Passengers=80))
```

```
## $fit
##        fit      lwr      upr
## 1 4.825921 4.414231 5.237611
## 
## $se.fit
## [1] 0.05228178
## 
## $df
## [1] 10
## 
## $residual.scale
## [1] 0.1772175
```

Scatterplot with fitted line

```r
with(airline, plot(x=Passengers, y=Cost, xlab="Passengers", ylab="Cost"))
with(airline, lines(x=sort(Passengers), y=fitted(m1)[order(Passengers)], col="green3", lwd=2))
legend("bottomright", "Fitted Line", col="green3", lwd=2)
```



Export the plot

If you will be using Microsoft Word to type your solutions, you should export this plot as an image.

If you will be using LaTeX to type your solutions, you should export this plot as a pdf.

Please **do not** take a picture of your computer screen using your phone!

Fit a model without an intercept

```r
m2 <- lm(Cost ~ 0 + Passengers, data=airline)
coef(m2)
```

```
## Passengers
## 0.06049319
```

The equation of this line is:

$$\widehat{\text{Cost}} = 0.6049 * \text{Passengers}$$

Is such a model reasonable, i.e. if a plane flies with no passengers, are no costs incurred for this flight?