

COMPSCIX 415.2 Homework 5/Midterm

Santosh Kanutala

7/6/2018

Contents

Github location	1
RStudio and R Markdown	1
Question: 1	1
The tidyverse packages	2
Question: 1	2
Question: 2	2
R Basics	2
Question: 1	2
Question: 2	2
Question: 3	2
Data import/export	3
Question: 1	3
Question: 2	3
Visualization	4
Question: 1	4
Question: 2	4
Question: 3	4
Data munging and wrangling	5
Question: 1	5
Question: 2	6
Question: 3	6
EDA	6
Question: 1	6
Question: 2	6
Question: 3	6
Question: 4	6
Question: 5	6
Question: 6	6
Git and Github	6

Github location

My homework assignments can be found at <https://github.com/santumagic/compscix-415-2assignments.git>

RStudio and R Markdown

Question: 1

As part of this question, I have loaded the required packages and added instructions for table of contents etc in the YAML header.

```
# Load the required packages
library(tidyverse)
```

```
library(mdsr)
library(nycflights13)
```

The tidyverse packages

Question: 1

Plotting - **ggplot2**

Data munging/wrangling - **dplyr** and **tidyr**

Reshaping (speading and gathering) data - **tidyr**

Importing/exporting data - **readr**

Question: 2

Plotting - **ggplot()** and **aes()**

Data munging/wrangling - **select()** and **filter()**

Reshaping (speading and gathering) data - **separate()** and **extract()**

Importing/exporting data - **read_csv()** and **read_delim()**

R Basics

Question: 1

```
My_data.name___is.too00ooLong <- c( 1 , 2 , 3 )
```

Answer: Just with one change (removal of '!'), the code works.

Question: 2

```
# this is a character vector
my_string <- c('has', 'an', 'error', 'in', 'it')
my_string
```

```
## [1] "has"   "an"    "error" "in"    "it"
```

Question: 3

```
my_vector <- c(1, 2, '3', '4', 5)
my_vector
```

```
## [1] "1" "2" "3" "4" "5"
```

Answer: This is a numeric vector and with or without the single or double quotes, vector takes values.

Data import/export

Question: 1

```
# Download and import the file rail_trail.txt
rail_trail.txt <- read.delim("/Users/skanutal/Documents/Santosh/Learning/Berkeley/rail_trail.txt", sep=
#glimpse the data from txt file
glimpse(rail_trail.txt)

## Observations: 90
## Variables: 10
## $ hightemp <int> 83, 73, 74, 95, 44, 69, 66, 66, 80, 79, 78, 65, 41,...
## $ lowtemp <int> 50, 49, 52, 61, 52, 54, 39, 38, 55, 45, 55, 48, 49,...
## $ avgtemp <dbl> 66.5, 61.0, 63.0, 78.0, 48.0, 61.5, 52.5, 52.0, 67....
## $ spring <int> 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, ...
## $ summer <int> 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, ...
## $ fall <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, ...
## $ cloudcover <dbl> 7.6, 6.3, 7.5, 2.6, 10.0, 6.6, 2.4, 0.0, 3.8, 4.1, ...
## $ precip <dbl> 0.00, 0.29, 0.32, 0.00, 0.14, 0.02, 0.00, 0.00, 0.0...
## $ volume <int> 501, 419, 397, 385, 200, 375, 417, 629, 533, 547, 4...
## $ weekday <int> 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, ...
```

Question: 2

```
# Export the .txt file as csv into a different location
rail_trail_csv <- write_delim(
  rail_trail.txt, delim = '|', path = "/Users/skanutal/Documents/Santosh/Learning/Berkeley/3. Intro to D
)
# Load the newly created csv file
rail_trail_csv_final <- read.csv(
  "/Users/skanutal/Documents/Santosh/Learning/Berkeley/3. Intro to DS/Assignments/rail_trail.csv", sep=
)
# glimpse the data from the final csv file
glimpse(rail_trail_csv_final)

## Observations: 90
## Variables: 10
## $ hightemp <int> 83, 73, 74, 95, 44, 69, 66, 66, 80, 79, 78, 65, 41,...
## $ lowtemp <int> 50, 49, 52, 61, 52, 54, 39, 38, 55, 45, 55, 48, 49,...
## $ avgtemp <dbl> 66.5, 61.0, 63.0, 78.0, 48.0, 61.5, 52.5, 52.0, 67....
## $ spring <int> 0, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, ...
## $ summer <int> 1, 1, 0, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 1, ...
## $ fall <int> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, ...
## $ cloudcover <dbl> 7.6, 6.3, 7.5, 2.6, 10.0, 6.6, 2.4, 0.0, 3.8, 4.1, ...
## $ precip <dbl> 0.00, 0.29, 0.32, 0.00, 0.14, 0.02, 0.00, 0.00, 0.0...
## $ volume <int> 501, 419, 397, 385, 200, 375, 417, 629, 533, 547, 4...
## $ weekday <int> 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 1, 1, 1, 0, ...
```

Visualization

Question: 1

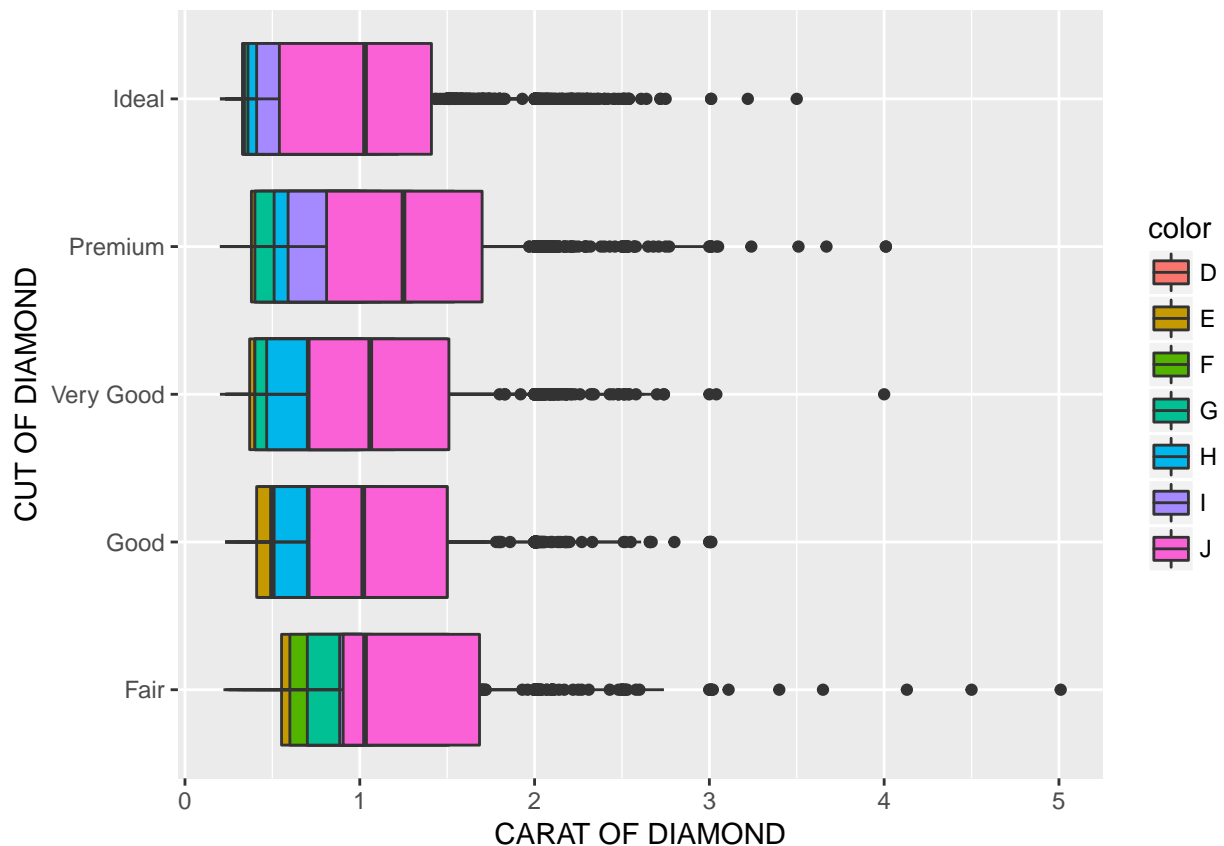
Answer:

1. Both the categories age group and gender are plotted on same axis, which is confusing at a first glance.
2. There is no clear comparison visible between the age groups and with in the genders because it shows the individual elements are compared against the responses only.
3. The graph elements are not sorted properly.

Question: 2

Reproduce the given graph

```
ggplot(data = diamonds, mapping = aes(x = cut, y = carat, fill = color)) +  
  geom_boxplot(position = "identity") +  
  coord_flip() +  
  labs(x = "CUT OF DIAMOND", y = "CARAT OF DIAMOND")
```

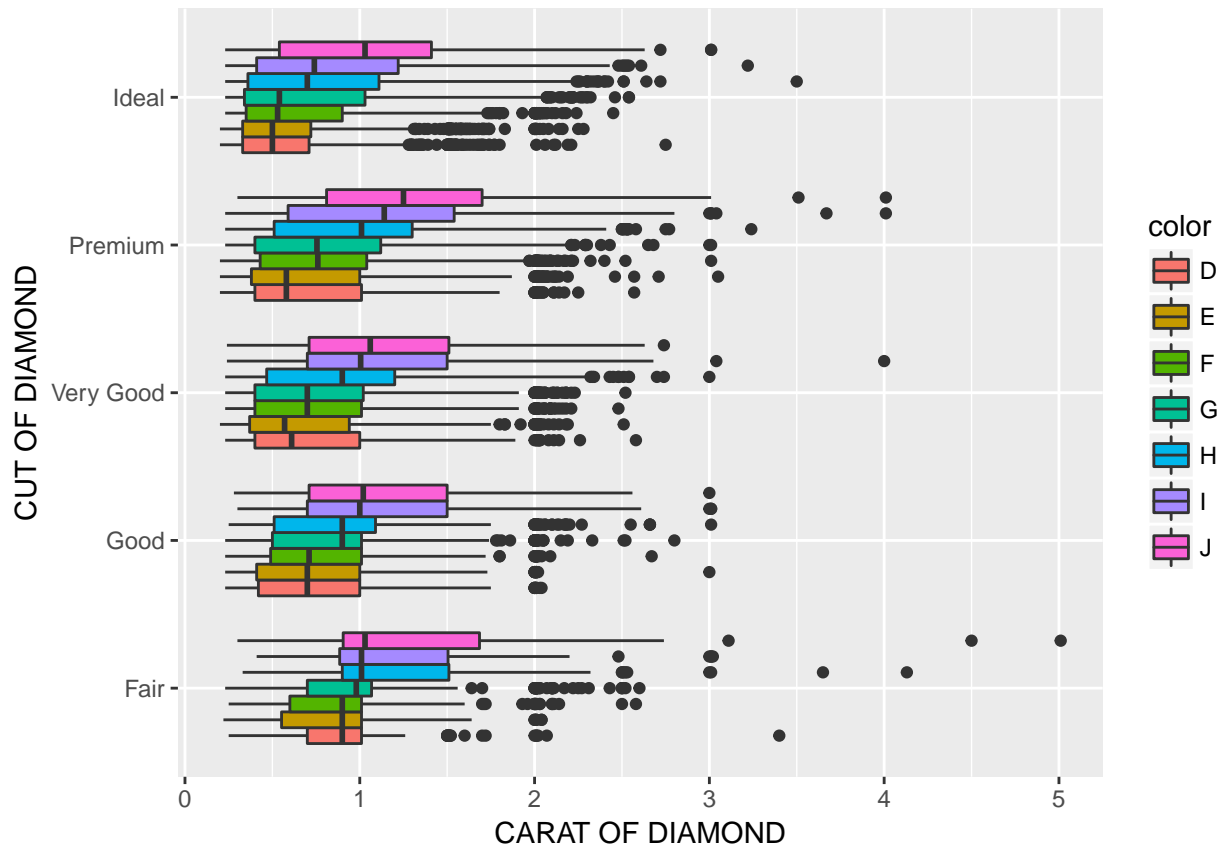


Question: 3

Enhancing the graph by changing the position to "dodge"

```
ggplot(data = diamonds, mapping = aes(x = cut, y = carat, fill = color)) +  
  geom_boxplot(position = "dodge") +
```

```
coord_flip() +
labs(x = "CUT OF DIAMOND", y = "CARAT OF DIAMOND")
```



Explanation: By using position = “dodge”, we can compare the individual values side by side.

Data munging and wrangling

Question: 1

```
# Finding the dataset tidy or not
table2
```

```
## # A tibble: 12 x 4
##   country    year type      count
##   <chr>      <int> <chr>    <int>
## 1 Afghanistan 1999 cases      745
## 2 Afghanistan 1999 population 19987071
## 3 Afghanistan 2000 cases      2666
## 4 Afghanistan 2000 population 20595360
## 5 Brazil      1999 cases      37737
## 6 Brazil      1999 population 172006362
## 7 Brazil      2000 cases      80488
## 8 Brazil      2000 population 174504898
## 9 China       1999 cases      212258
## 10 China      1999 population 1272915272
## 11 China      2000 cases      213766
```

```
## 12 China          2000 population 1280428583
# It is not a tidy data, so below code makes it a tidy dataset
table2_tidy <- spread(table2, type, count)
# Display table2 in tidy way
table2_tidy
```

```
## # A tibble: 6 x 4
##   country      year  cases population
##   <chr>      <int> <int>      <int>
## 1 Afghanistan 1999     745   19987071
## 2 Afghanistan 2000    2666   20595360
## 3 Brazil      1999   37737   172006362
## 4 Brazil      2000   80488   174504898
## 5 China       1999  212258  1272915272
## 6 China       2000  213766  1280428583
```

Question: 2

Question: 3

EDA

Question: 1

Question: 2

Question: 3

Question: 4

Question: 5

Question: 6

Git and Github