Northeastern University
Toronto

Date: 5th February 2022

Executive Summary Report 3

ALY6000 Introduction to Analytics
[CRN 22279]

Poonam Dighe [NUID 002965616]

Professor Mohammad Shafiqul Islam

INTRODUCTION

The following project report will provide a thorough understanding of R and RStudio as well as some hands-on experience. The objective of the project is to learn loading and installing several libraries, dataset observations, dataset structure and summary.
We will also study how to generate a scatterplot, histogram, over dense plot, regression line, and legend with a given set of specifications.

KEY FINDINGS

A. Following an introduction, provide an analysis of descriptive characteristics of the data set provided by your instructor. This includes pertinent statistics including counts, cumulative counts, and frequency, percentages, etc. Include R console screen snippets to support your observations and conclusions.

```
> head(bio,5)
  netID fishID  species tl   w tag scale
1    12     16 Bluegill 61 2.9     FALSE
2    12     23 Bluegill 66 4.5     FALSE
3    12     30 Bluegill 70 5.2     FALSE
4    12     44 Bluegill 38 0.5     FALSE
5    12     50 Bluegill 42 1.0     FALSE
> tail(bio,5)
    netID fishID       species  tl   w  tag scale
672   121    809 Black Crappie 282 352 1700  TRUE
673   121    812 Black Crappie 142  37       TRUE
674   110    863 Black Crappie 307 415 1783  TRUE
675   129    870 Black Crappie 279 344 1789  TRUE
676   129    879 Black Crappie 302 397 1792  TRUE
> headtail(bio,5)
    netID fishID       species  tl     w  tag scale
1      12     16      Bluegill  61   2.9      FALSE
2      12     23      Bluegill  66   4.5      FALSE
3      12     30      Bluegill  70   5.2      FALSE
4      12     44      Bluegill  38   0.5      FALSE
5      12     50      Bluegill  42   1.0      FALSE
672   121    809 Black Crappie 282 352.0 1700  TRUE
673   121    812 Black Crappie 142  37.0       TRUE
674   110    863 Black Crappie 307 415.0 1783  TRUE
675   129    870 Black Crappie 279 344.0 1789  TRUE
676   129    879 Black Crappie 302 397.0 1792  TRUE
> str(bio)
'data.frame':   676 obs. of  7 variables:
 $ netID  : int  12 12 12 12 12 12 12 13 13 13 ...
 $ fishID : int  16 23 30 44 50 65 66 68 69 70 ...
 $ species: chr  "Bluegill" "Bluegill" "Bluegill" "Bluegill" ...
 $ tl     : int  61 66 70 38 42 54 27 36 59 39 ...
 $ w      : num  2.9 4.5 5.2 0.5 1 2.1 NA 0.5 2 0.5 ...
 $ tag    : chr  "" "" "" "" ...
 $ scale  : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
>
```

fig. A

- The data frame provided consists of 676 observations along with 7 variables.
- There are multiple data types observed for each of the variables in the mentioned dataset. example Integer, character, etc.
- The head and tail give us the first and the last records depending on the iteration that is the value of n (n=5, in the above figure A which gives the first and last five records in the dataset).

```
> tmp <- table(bio$species)
> tmp

   Black Crappie         Bluegill Bluntnose Minnow      Iowa Darter  Largemouth Bass
              36              220              103               32              228
     Pumpkinseed   Tadpole Madtom     Yellow Perch
              13                6               38
> |
```

```
> cSpecPct <- (table(bio$species)*100)/length(bio$species)
> cSpecPct         #display species & their percentage

   Black Crappie         Bluegill Bluntnose Minnow      Iowa Darter  Largemouth Bass
        5.325444        32.544379        15.236686         4.733728        33.727811
     Pumpkinseed   Tadpole Madtom     Yellow Perch
        1.923077         0.887574         5.621302
> class(cSpecPct)
[1] "table"
```

fig. B

Explanation

- The above figure B indicates the number of records for a particular species in the dataset. From the records, there is a high frequency of records for Bluegill and Largemouth Bass species followed by the rest of them. A total of approximately 66.27% are from Bluegill and Largemouth Bass Species. The Tadpole Madtom is the only species with a fewer number of records.
- The cSpecPct is the percentage of each species according to the number of records present in the dataframe.

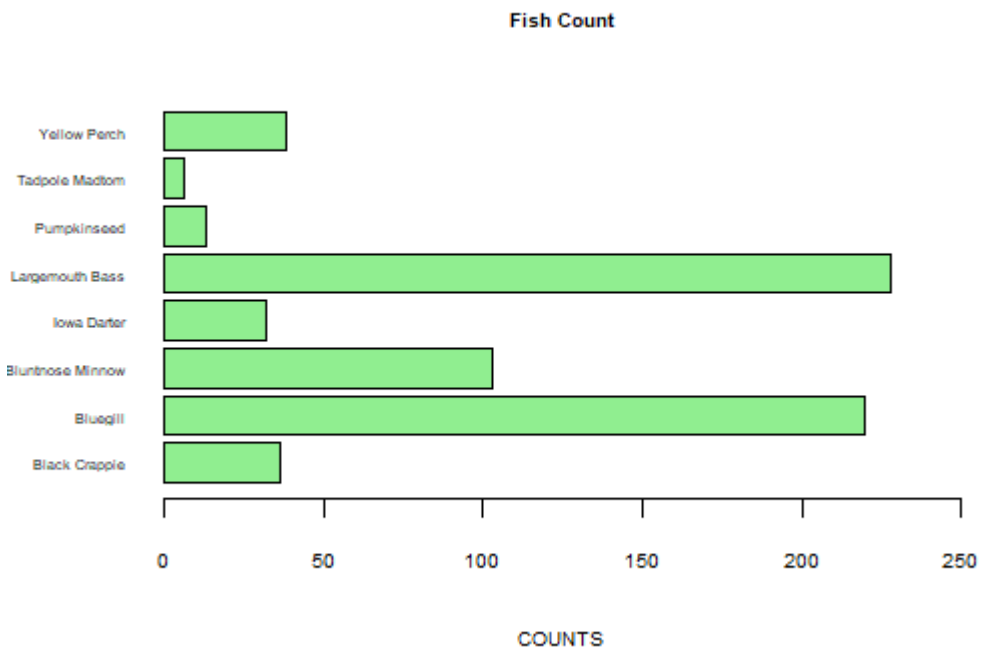
```
> d
           Species   RelFreq   cumFreq counts cumcounts
5  Largemouth Bass 33.727811  33.72781    228       228
2         Bluegill 32.544379  66.27219    220       448
3 Bluntnose Minnow 15.236686  81.50888    103       551
8     Yellow Perch  5.621302  87.13018     38       589
1    Black Crappie  5.325444  92.45562     36       625
4       Iowa Darter  4.733728  97.18935     32       657
6      Pumpkinseed  1.923077  99.11243     13       670
7   Tadpole Madtom  0.887574 100.00000      6       676
> |
```

fig. C

Explanation

- To create a Pareto chart, we must first determine the count, followed by the cumulative count for all species.
- We estimated cumulative counts after identifying the counts of each species in the above output by adding the next count to the one before it (cumcounts & cumfreq).
- This is further used in the Pareto Chart.

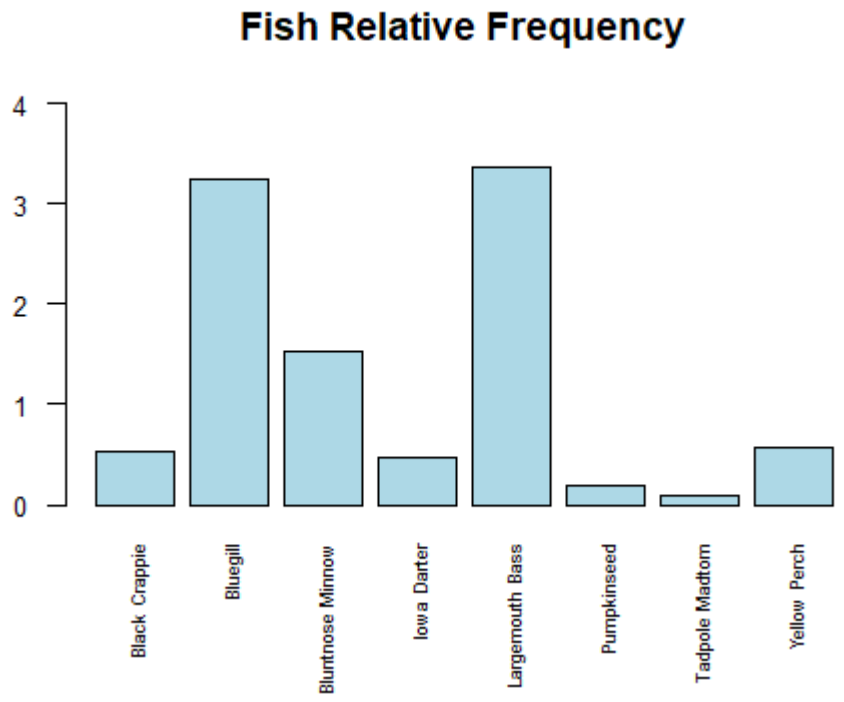
B. Provide the executive with visualizations (at least 6) in that help them see the key characteristics you want to highlight. They can be boxplots, histograms, frequency and probability distributions, barplots (bar charts) or pareto. Not only is the goal to present your visual results, but also to explain the significance of what the visuals are displaying.



Plot A : Fish Count

Explanation

- The Bar Plot above shows the fish count of each fish available. The Y-axis depicts the different species of fish available. The x-axis depicts the number of fishes available. As we can observe Bluegill and largemouth bass are the most abundant fish found. Tadpole Madtom and Pumpkinseed have the lowest fish count among the lot.

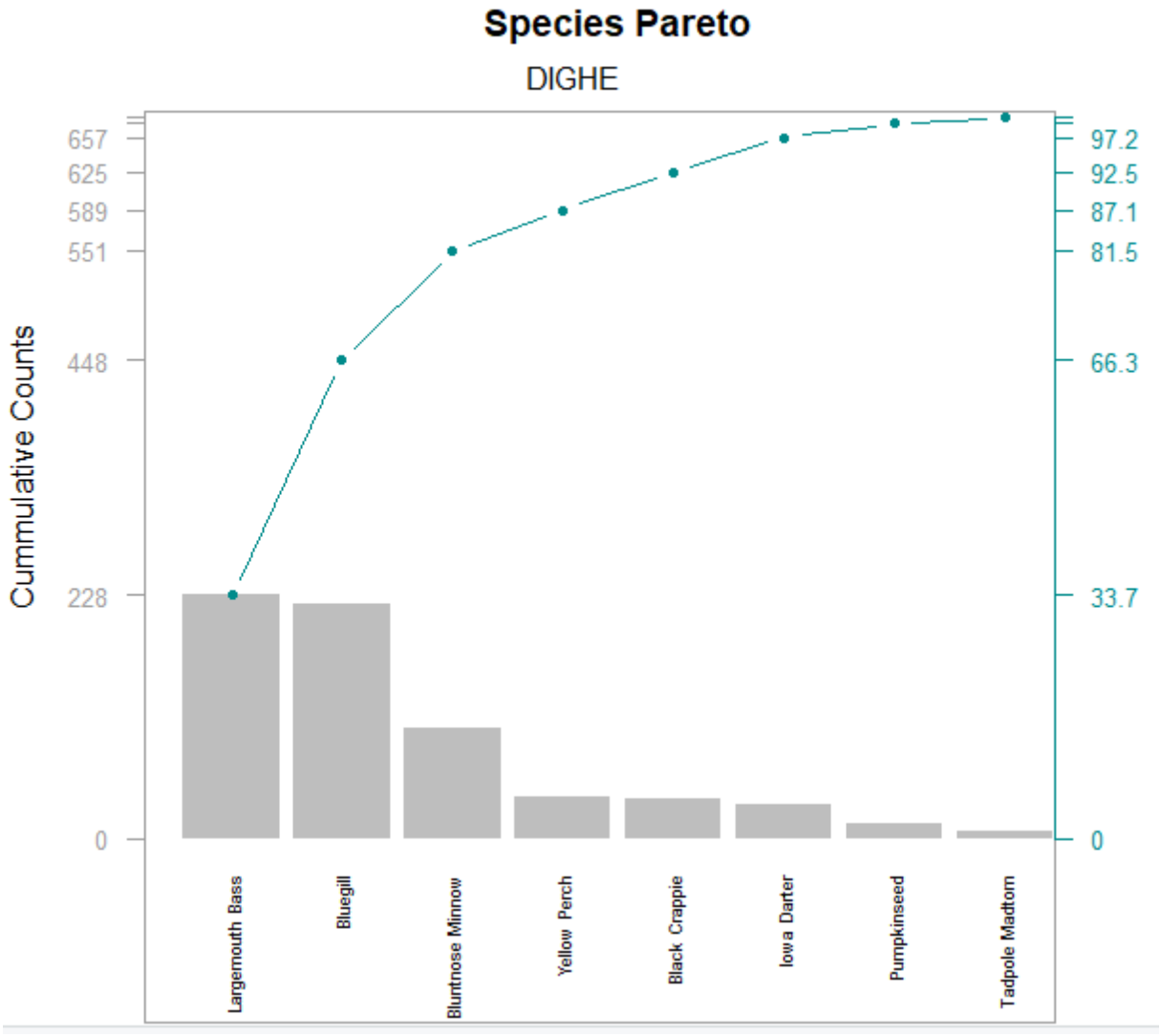


Plot B : Fish Relative Frequency

Explanation

- There is a difference between the normal frequency and relative frequency in statistics.
- Normal Frequency indicates the number of records that can be extracted from the dataset whereas Relative Frequency can be calculated with the help of the number of records (Frequency in the dataset).
- Relative Frequency(fi) = Frequency/length (Frequency) = ni / N
- The above graph indicates the relative frequency graph which has been calculated for every species.
- The main aspect that can be questioned after observing the above two graphs is the scarcity of the Tadpole Madtom and Pumpkinseed species. What exactly is the reason for the species count to go low and what is causing them to disappear.
- Are there any environmental changes that are causing the rise and fall of such species?

- A detailed survey, as well as some research, can help in identifying the root cause for such an issue.



Plot C : Species Pareto

Explanation

- The Pareto Plot majorly consists of the cumulative frequency calculation which shows us the link between the cumulative frequency and individual factors listed by the frequency of occurrence.

- It can also be observed that the ecological changes in the human environment can also lead to such situations as natural calamities, etc.
- The Pareto Chart is a combination of bar chart as well as line chart and it is plotted in a descending frequency calculation fashion. Therefore, it can be observed that the Pareto Chart consists of three-axis (2 Y-axis and 1 X-axis).
- The X-axis consists of all the species names arranged according to the highest frequency first and so on. From the graph, Largemouth Bass, Bluegill, and Bluntnose Minnow as the most dominant species.
- Largemouth Bass makes up 33.72 percent of the total species, followed by Bluegill (32.54 percent) and Bluntnose Minnow (both 32.54 percent) (15.23 percent). The remaining species account for 18.51 percent of all species, which is less than 20% of the total.

- From further reading about these species, the Largemouth Bass and Bluegill prey on other species for their living whereas the Pumpkinseed and Tadpole Madtom are dependent on algae for their food.
  - By some of this analysis, we can conclude that these species may become extinct or become dominant over others based on the data provided.

C.Finally, provide a clear two to three sentence paragraph summary of the key points that you want the audience to walk away with regarding your analysis. This summary should present accurate analysis and be supported by the data presented in the rest of the report.

To summarize the above analysis, we can say the "inchbio" dataset provided and renamed as "bio" provided us with an understanding of the species which are going to be extinct or become dominant over the other. The Largemouth Bass and Bluegill are the most dominant of all whereas the Pumpkinseed and Tadpole Madtom are on the verge of extinction. This analysis can be done based on the calculation of relative frequency and cumulative frequency depending on the dataset provided.

BIBLIOGRAPHY

1. “Bar Plot in R Using Barplot() Function.” DataMentor, 22 Nov. 2017, https://www.datamentor.io/r-programming/bar-plot/.

2. “Calculate Percentage in R (Example) | Convert Vector to Probability Table.” Statistics Globe, https://statisticsglobe.com/calculate-percentage-in-r/.
Accessed 3 Feb. 2022.

3. Coerce Lists, Matrices, and More to Data Frames — As_tibble. https://tibble.tidyverse.org/reference/as_tibble.html.
Accessed 3 Feb. 2022.

4. Government of Canada, Statistics Canada. Statistics: Power from Data! Analytical Graphing: Cumulative Frequency. 12 Dec. 2002, https://www150.statcan.gc.ca/n1/edu/power-pouvoir/ch10/5214862-eng.htm.

5. “How to Change Font Size of Text and Axes on R Plots.” Edureka Community, https://www.edureka.co/community/2000/how-to-change-font-size-of-text-and-axes-on-r-plots.
Accessed 3 Feb. 2022.

6. “PLOT in R ⭕ [Type, Color, Axis, Pch, Title, Font, Lines, Add Text, Label Points].” R CODER, 17 Apr. 2020, https://r-coder.com/plot-r/.

7. Quick-R: Bar Plots. https://www.statmethods.net/graphs/bar.html.
Accessed 3 Feb. 2022.

8. R Programming - Cumulative Frequency and Cumulative Relative Frequency via Table Command. www.youtube.com, https://www.youtube.com/watch?v=rZj4lVyxckw.
Accessed 3 Feb. 2022.

9. “Rename Data Frame Columns in R.” Datanovia, https://www.datanovia.com/en/lessons/rename-data-frame-columns-in-r/
Accessed 3rd Feb. 2022

10. Kabacoff, R. I. (2015). R in action: Data analysis and graphics with r. Manning Publications Co.
Accessed 3rd Feb. 2022.

My Github repository link –

https://github.com/PoonamDighe/ALY_6000_Module3_Project3.git

M3_Assignment_3.R

poonam

2022-02-05

```
r = getOption("repos")
r["CRAN"]="http://cran.us.r-project.org"
options(repos=r)

#1.Print your name at the top of the script.Import libraries including: FSA,
FSAdata, magrittr, dplyr, tidyr plyr and tidyverse

Name <- ("Poonam Dighe")
Name

## [1] "Poonam Dighe"

install.packages("FSA")

## Installing package into 'C:/Users/poonam/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'FSA' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\poonam\AppData\Local\Temp\RtmpqMvEGF\downloaded_packages

install.packages("FSAdata")

## Installing package into 'C:/Users/poonam/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'FSAdata' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\poonam\AppData\Local\Temp\RtmpqMvEGF\downloaded_packages

install.packages("magrittr")

## Installing package into 'C:/Users/poonam/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'magrittr' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'magrittr'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying C:
## \Users\poonam\Documents\R\win-library\4.1\00LOCK\magrittr\libs\x64\magritt
r.dll
## to C:\Users\poonam\Documents\R\win-library\4.1\magrittr\libs\x64\magrittr.
```

```
dll:
## Permission denied

## Warning: restored 'magrittr'

##
## The downloaded binary packages are in
##  C:\Users\poonam\AppData\Local\Temp\RtmpqMvEGF\downloaded_packages

install.packages("dplyr")

## Installing package into 'C:/Users/poonam/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'dplyr' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'dplyr'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying C:
## \Users\poonam\Documents\R\win-library\4.1\00LOCK\dplyr\libs\x64\dplyr.dll
to C:
## \Users\poonam\Documents\R\win-library\4.1\dplyr\libs\x64\dplyr.dll: Permis
sion
## denied

## Warning: restored 'dplyr'

##
## The downloaded binary packages are in
##  C:\Users\poonam\AppData\Local\Temp\RtmpqMvEGF\downloaded_packages

install.packages("tidyr")

## Installing package into 'C:/Users/poonam/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'tidyr' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'tidyr'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying C:
## \Users\poonam\Documents\R\win-library\4.1\00LOCK\tidyr\libs\x64\tidyr.dll
to C:
## \Users\poonam\Documents\R\win-library\4.1\tidyr\libs\x64\tidyr.dll: Permis
sion
## denied

## Warning: restored 'tidyr'

##
## The downloaded binary packages are in
##  C:\Users\poonam\AppData\Local\Temp\RtmpqMvEGF\downloaded_packages

install.packages("plyr")
```

```
## Installing package into 'C:/Users/poonam/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'plyr' successfully unpacked and MD5 sums checked

## Warning: cannot remove prior installation of package 'plyr'

## Warning in file.copy(savedcopy, lib, recursive = TRUE): problem copying C:
## \Users\poonam\Documents\R\win-library\4.1\00LOCK\plyr\libs\x64\plyr.dll to
C:
## \Users\poonam\Documents\R\win-library\4.1\plyr\libs\x64\plyr.dll: Permissi
on
## denied

## Warning: restored 'plyr'

##
## The downloaded binary packages are in
##  C:\Users\poonam\AppData\Local\Temp\RtmpqMvEGF\downloaded_packages

install.packages("tidyverse")

## Installing package into 'C:/Users/poonam/Documents/R/win-library/4.1'
## (as 'lib' is unspecified)

## package 'tidyverse' successfully unpacked and MD5 sums checked
##
## The downloaded binary packages are in
##  C:\Users\poonam\AppData\Local\Temp\RtmpqMvEGF\downloaded_packages

library("FSA")

## ## FSA v0.9.1. See citation('FSA') if used in publication.
## ## Run fishR() for related website and fishR('IFAR') for related book.

library("FSAdata")

## ## FSAdata v0.3.8. See ?FSAdata to find data for specific fisheries analys
es.

library("magrittr")
library("dplyr")

##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library("tidyr")

##
## Attaching package: 'tidyr'

## The following object is masked from 'package:magrittr':
##
##     extract

library("plyr")

## ------------------------------------------------------------------------
----

## You have loaded plyr after dplyr - this is likely to cause problems.
## If you need functions from both plyr and dplyr, please load plyr first, th
en dplyr:
## library(plyr); library(dplyr)

## ------------------------------------------------------------------------
----

##
## Attaching package: 'plyr'

## The following objects are masked from 'package:dplyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following object is masked from 'package:FSA':
##
##     mapvalues

library("tidyverse")

## -- Attaching packages --------------------------------------- tidyverse 1.
3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflict
s() --
## x plyr::arrange()    masks dplyr::arrange()
## x purrr::compact()   masks plyr::compact()
## x plyr::count()      masks dplyr::count()
## x tidyr::extract()   masks magrittr::extract()
## x plyr::failwith()   masks dplyr::failwith()
## x dplyr::filter()    masks stats::filter()
## x plyr::id()         masks dplyr::id()
## x dplyr::lag()       masks stats::lag()
```

```
## x plyr::mutate()     masks dplyr::mutate()
## x plyr::rename()     masks dplyr::rename()
## x purrr::set_names() masks magrittr::set_names()
## x plyr::summarise()  masks dplyr::summarise()
## x plyr::summarize()  masks dplyr::summarize()
```

```
#2. Import the inchBio.csv and name the table <bio>
bio <- read.csv("C:\\Users\\poonam\\Desktop\\inchBio.csv")
View(bio)

#3. Display the head, tail and structure of <bio>
head(bio,5)
```

```
##   netID fishID  species tl   w tag scale
## 1    12     16 Bluegill 61 2.9     FALSE
## 2    12     23 Bluegill 66 4.5     FALSE
## 3    12     30 Bluegill 70 5.2     FALSE
## 4    12     44 Bluegill 38 0.5     FALSE
## 5    12     50 Bluegill 42 1.0     FALSE
```

```
tail(bio,5)
```

```
##     netID fishID       species  tl   w  tag scale
## 672   121    809 Black Crappie 282 352 1700  TRUE
## 673   121    812 Black Crappie 142  37       TRUE
## 674   110    863 Black Crappie 307 415 1783  TRUE
## 675   129    870 Black Crappie 279 344 1789  TRUE
## 676   129    879 Black Crappie 302 397 1792  TRUE
```

```
headtail(bio,5)
```

```
##     netID fishID       species  tl     w  tag scale
## 1      12     16      Bluegill  61   2.9      FALSE
## 2      12     23      Bluegill  66   4.5      FALSE
## 3      12     30      Bluegill  70   5.2      FALSE
## 4      12     44      Bluegill  38   0.5      FALSE
## 5      12     50      Bluegill  42   1.0      FALSE
## 672   121    809 Black Crappie 282 352.0 1700  TRUE
## 673   121    812 Black Crappie 142  37.0       TRUE
## 674   110    863 Black Crappie 307 415.0 1783  TRUE
## 675   129    870 Black Crappie 279 344.0 1789  TRUE
## 676   129    879 Black Crappie 302 397.0 1792  TRUE
```

```
str(bio)
```

```
## 'data.frame':    676 obs. of  7 variables:
##  $ netID  : int  12 12 12 12 12 12 12 13 13 13 ...
##  $ fishID : int  16 23 30 44 50 65 66 68 69 70 ...
##  $ species: chr  "Bluegill" "Bluegill" "Bluegill" "Bluegill" ...
##  $ tl     : int  61 66 70 38 42 54 27 36 59 39 ...
##  $ w      : num  2.9 4.5 5.2 0.5 1 2.1 NA 0.5 2 0.5 ...
```

```
##  $ tag    : chr  "" "" "" "" ...
##  $ scale  : logi  FALSE FALSE FALSE FALSE FALSE FALSE ...
```

```
#4.Create an object, <counts>, that counts and lists all the species records
counts <- bio$species
table(counts)
```

```
## counts
##    Black Crappie         Bluegill Bluntnose Minnow      Iowa Darter
##               36              220              103               32
##  Largemouth Bass      Pumpkinseed   Tadpole Madtom     Yellow Perch
##              228               13                6               38
```

```
#5.Display just the 8 levels (names) of the species

unique(bio$species)
```

```
## [1] "Bluegill"         "Bluntnose Minnow" "Iowa Darter"      "Largemouth B
ass"
## [5] "Pumpkinseed"      "Tadpole Madtom"   "Yellow Perch"     "Black Crappi
e"
```

```
#6.Create a <tmp> object that displays the different species and the number o
f record of each species in the dataset.
#Include this information in your report.

tmp <- table(bio$species)
tmp
```

```
##
##    Black Crappie         Bluegill Bluntnose Minnow      Iowa Darter
##               36              220              103               32
##  Largemouth Bass      Pumpkinseed   Tadpole Madtom     Yellow Perch
##              228               13                6               38
```

```
#7.Create a subset, <tmp2>, of just the species variable and display the firs
t five records.

tmp2 <- subset(bio,select = species)
head(tmp2,5)
```

```
##    species
## 1 Bluegill
## 2 Bluegill
## 3 Bluegill
## 4 Bluegill
## 5 Bluegill
```

```
#8.Create a table, <w>, of the species variable. Display the class of w.
```

```
w <- table(bio$species)
w
```

```
##
##    Black Crappie         Bluegill Bluntnose Minnow      Iowa Darter
##               36              220              103               32
##  Largemouth Bass      Pumpkinseed   Tadpole Madtom     Yellow Perch
##              228               13                6               38
```

```
class(w)
```

```
## [1] "table"
```

```
#9.Convert <w> to a data frame named <t> and display the results.

t<- as.data.frame(w)
t
```

```
##               Var1 Freq
## 1    Black Crappie   36
## 2         Bluegill  220
## 3 Bluntnose Minnow  103
## 4      Iowa Darter   32
## 5  Largemouth Bass  228
## 6      Pumpkinseed   13
## 7   Tadpole Madtom    6
## 8     Yellow Perch   38
```

```
class(t)
```

```
## [1] "data.frame"
```

```
#10.Extract and display the frequency values from the <t> data frame

t$Freq
```

```
## [1]  36 220 103  32 228  13   6  38
```

```
#11.Create a table named <cSpec> from the bio species attribute (variable) an
d confirm that you created a table which displays the number of species in th
e dataset <bio>

cSpec <- table(bio$species)
cSpec
```

```
##
##    Black Crappie         Bluegill Bluntnose Minnow      Iowa Darter
##               36              220              103               32
##  Largemouth Bass      Pumpkinseed   Tadpole Madtom     Yellow Perch
##              228               13                6               38
```

```
class(cSpec)
```

```
## [1] "table"
```

```
#12.Create a table named <cSpecPct> that displays the species and percentage
of records for each species. Confirm you created a table class.
cSpecPct <- (table(bio$species)*100)/length(bio$species)
cSpecPct         #display species & their percentage
```

```
##
##    Black Crappie         Bluegill Bluntnose Minnow      Iowa Darter
##         5.325444        32.544379        15.236686         4.733728
##  Largemouth Bass      Pumpkinseed   Tadpole Madtom     Yellow Perch
##        33.727811         1.923077         0.887574         5.621302
```

```
class(cSpecPct)
```

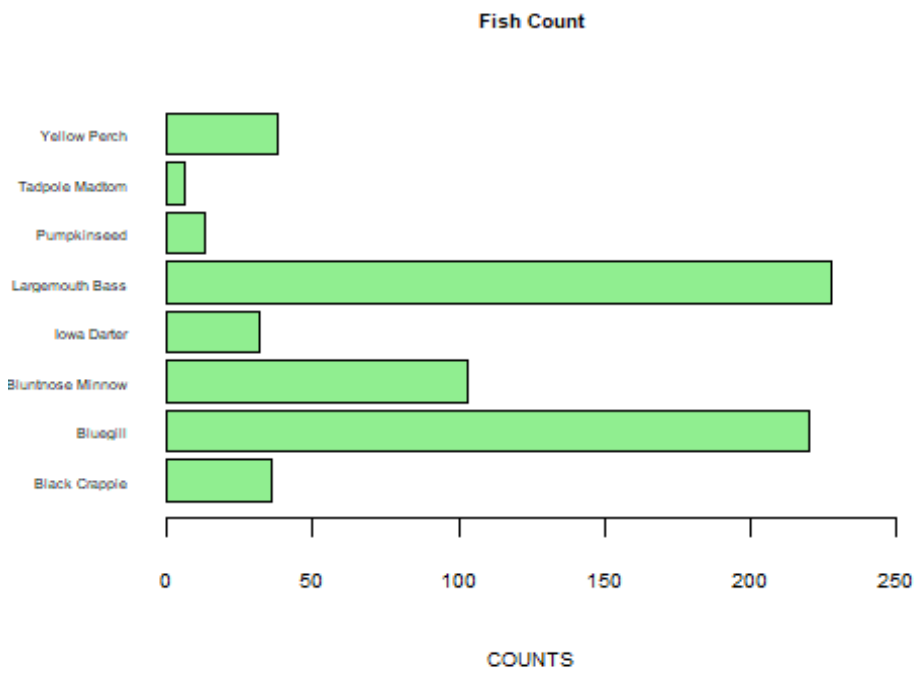
```
## [1] "table"
```

```
#13.Convert the table, <cSpecPct>, to a data frame named <u> and confirm that
<u> is a data frame
u <- data.frame(cSpecPct )  #converting cSpectPct to a data frame
class(u)
```

```
## [1] "data.frame"
```

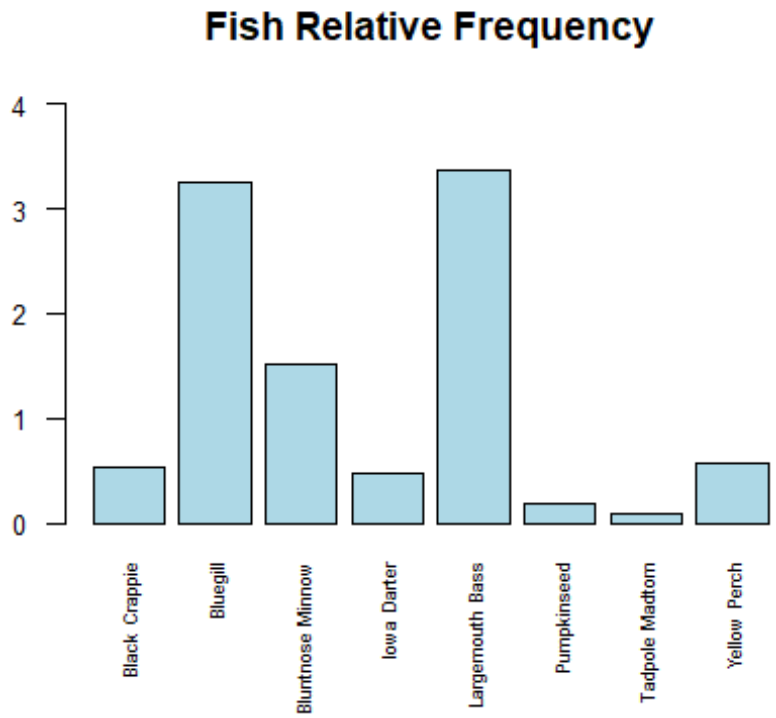
```
#14.Create a barplot of <cSpec> with the following: titled Fish Count with th
e following specifications:
#Title: Fish Count
#Y axis is labeled "COUNTS"
#Color the bars Light Green
#Rotate Y axis to be horizontal
#Set the X axis font magnification to 60% of nominal

barplot(cSpec,
        main="Fish Count",
        xlab="COUNTS",
        col="light green",
        las = 1 , horiz = TRUE,
        cex.axis=0.60,
        cex.names = 0.45,
        cex.lab = 0.60,
        cex.main = 0.60,
        xlim = c(0,250))
```




```
#15.Create a barplot of <cSpecPct>, with the following specifications:
#Y axis limits of 0 to 4
#Y axis label color of Light Blue
#Title of "Fish Relative Frequency"


barplot(cSpecPct/10,
        ylim=c(0,4),
        las=2,
        col= "Light Blue",
        main= "Fish Relative Frequency",
        cex.axis = 0.8,
        cex.names = 0.55)
```




```
#16.Rearrange the <u> cSpec Pct data frame in descending order of relative fr
equency. Save the rearranged data frame as the object <d>

d <- u[order(u$Freq,decreasing=TRUE),]
d

##                Var1      Freq
## 5  Largemouth Bass 33.727811
## 2         Bluegill 32.544379
## 3 Bluntnose Minnow 15.236686
## 8     Yellow Perch  5.621302
## 1    Black Crappie  5.325444
## 4      Iowa Darter  4.733728
## 6      Pumpkinseed  1.923077
## 7   Tadpole Madtom  0.887574

#17.Rename the <d> columns Var 1 to Species, and Freq to RelFreq

colnames(d) <- c("Species","RelFreq")
d

##            Species   RelFreq
## 5  Largemouth Bass 33.727811
## 2         Bluegill 32.544379
## 3 Bluntnose Minnow 15.236686
## 8     Yellow Perch  5.621302
## 1    Black Crappie  5.325444
```

```
## 4      Iowa Darter  4.733728
## 6      Pumpkinseed  1.923077
## 7   Tadpole Madtom  0.887574

#18.Add new variables to <d> and call them cumfreq, counts, and cumcounts.

t<- arrange(t,desc(t$Freq))
d<- mutate(d, cumFreq = cumsum(RelFreq), counts = t$Freq, cumcounts = cumsum(
t$Freq))
d

##            Species   RelFreq   cumFreq counts cumcounts
## 5  Largemouth Bass 33.727811  33.72781    228       228
## 2         Bluegill 32.544379  66.27219    220       448
## 3 Bluntnose Minnow 15.236686  81.50888    103       551
## 8     Yellow Perch  5.621302  87.13018     38       589
## 1    Black Crappie  5.325444  92.45562     36       625
## 4      Iowa Darter  4.733728  97.18935     32       657
## 6      Pumpkinseed  1.923077  99.11243     13       670
## 7   Tadpole Madtom  0.887574 100.00000      6       676

#19.Create a parameter variable <def_par> to store parameter variables.

def_par <- as.data.frame(d)
def_par

##            Species   RelFreq   cumFreq counts cumcounts
## 5  Largemouth Bass 33.727811  33.72781    228       228
## 2         Bluegill 32.544379  66.27219    220       448
## 3 Bluntnose Minnow 15.236686  81.50888    103       551
## 8     Yellow Perch  5.621302  87.13018     38       589
## 1    Black Crappie  5.325444  92.45562     36       625
## 4      Iowa Darter  4.733728  97.18935     32       657
## 6      Pumpkinseed  1.923077  99.11243     13       670
## 7   Tadpole Madtom  0.887574 100.00000      6       676

#20. Create a barplot, <pc>, with the following specifications:
#d$counts of width 1, spacing of .15
#no boarder
#Axes: F
#Yaxis limit 0,3.05*max
#d$counts na.rm is true
#y label is Cummulative Counts
#scale x axis to 70%
#names.arg: d$Species
#Title of the barplot is "Species Pareto"
#las: 2)
#25.Display the finished Species Pareto Plot (without the star watermarks).Ha
ve your last name on the plot.
subtitle="DIGHE"
pc <- barplot(d$counts,
```

```
              width = 1,
              space = 0.15,
              border = NA,
              axes = F,
              ylim = c(0,3.05*max(d$counts, na.rm = TRUE)),
              ylab = "Cummulative Counts",
              cex.axis = 0.7,
              names.arg = d$Species,
              main = "Species Pareto",
              las=2,
              cex.names = 0.55)
mtext(side=3, line=0, at=-0.17, adj= -4, cex=1, subtitle)
par(mar = c(0.1, 4, 4.5, 3))
#21.Add a cumulative counts line to the <pc> plot with the following:
#Spec line type is b
#Scale plotting text at 70%
#Data values are solid circles with color cyan4

lines(pc, d$cumcounts, type = "b", cex = 0.7, pch = 19, col="cyan4")

#22.Place a grey box around the pareto plot.
box(col = "grey62")

#23.Add a left side axis with the following specifications
#Horizontal values at tick marks at cumcounts on side 2
#Tickmark color of grey62
#Color of axis is grey62
#Axis scaled to 80% of normal

axis(side = 2,
     at = c(0, d$cumcounts),
     las = 1,
     col.axis = "grey62",
     col = "grey62",
     cex.axis = 0.8)


#24.Add axis details on right side of box with the specifications:
#Spec: Side 4
#Tickmarks at cumcounts with labels from 0 to cumfreq with %,
#Axis color of cyan5 and label color of cyan4
#Axis font scaled to 80% of nominal

axis(side = 4,
     at = c(0, d$cumcounts),
     labels = paste(c(0, round(d$cumFreq,digit=1)),sep=" "),
     las = 1,
     col.axis = "cyan4",
```

```
    col = "cyan4",
    cex.axis = 0.8)
```