ALY6000, WINTER 2022

Module 3 Project - Executive Summary Report 3

Submitted by: Abhinav Jain

NUID: 002938209

Submitted To: Richard Zhi

Date: 02/2/2022

Introduction to Inchbio in R:

Presently, the world is generating data at a fast pace, therefore it's difficult to manage and visualize data and demonstrate it in the graphical representation to gain useful insights. Data visualization is the technique for obtaining the data and communicating to the audience with a deep understanding of visualization. Demonstrate how to load a library and locate package-specific functions. We are loading libraries and displaying the data in serval tabular and graphical representations in this report. This report mainly focuses on FSA, FSAdata, magrittr, dplyr, tidyr, tidyverse libraries, and various data functions like head(), tail(), structure(), create an object, display level of species, number of records in the dataset, create a subset, create a table of a variable, convert the table into data frames, calculate the percentage of variable, apart from this plotting a bar chart of Fish Count and Fish Relative Frequency. Many more functions rearrange g the data in descending and ascending forms of a relative frequency. After plotting the bar chart, we will discuss Pareto charts with various implementations.

Task 1: In this task, installing the given packages to execute the various function

```
> library(tidyr)
> library(plyr)
> 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_conflicts() --
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()
> fishR()
```

Task 2. In this task imported the dataset to execute the functions from the Inchbio dataset which shows   netid,,   fishId,   species   ,   tl,   w   ,   tagscale   [1]

```
> bio <- read.csv("C:\\NEU BOSTON\\ALY6000 Submission\\ALY6000-Final\
> bio
   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
6      12      65 Bluegill  54  2.1      FALSE
7      12      66 Bluegill  27   NA      FALSE
8      13      68 Bluegill  36  0.5      FALSE
9      13      69 Bluegill  59  2.0      FALSE
10     13      70 Bluegill  39  0.5      FALSE
11     13      71 Bluegill  34  0.5      FALSE
12     13      73 Bluegill  40  1.0      FALSE
13     13      74 Bluegill  35  0.5      FALSE
14     13      75 Bluegill  32  1.0      FALSE
15     13      76 Bluegill  37  0.5      FALSE
16     13      77 Bluegill  38  1.0      FALSE
17     13      78 Bluegill  69  7.0      FALSE
18     13      80 Bluegill  39  1.0      FALSE
```

Task 3. In this task, to get the data in tabular form <bio> which shows the first six rows of the dataset, while tail(bio) shows the last six rows of the dataset, and to look at the structure of the dataset which shows that the data frame has 676 observations and 7 variables. [2]

```
> head(bio)
  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
6     12      65 Bluegill 54 2.1     FALSE
> tail(bio)
    netID fishID        species  tl    w  tag scale
671   121    808 Black Crappie 323 509 1050  TRUE
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
> 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 ...
```

Task 4. In this task, created an object which shows the list of all the species record, whereas the length is 676 in bio$species [3]

```
> counts <- bio
> counts <- count(bio$species)
> counts <- cbind (length (bio$species), list(bio$species))
> View(counts)
> counts
     [,1] [,2]
[1,] 676  character,676
>
```

Task 5. In this task we display the unique values of the dataset with species

```
> unique(bio$species, incomparables = NULL)
[1] "Bluegill"        "Bluntnose Minnow" "Iowa Darter"      "Largemouth Bass"  "Pumpkinseed"      "Tadpole Madtom"   "Yellow Perch"
[8] "Black Crappie"
>
```

Task 6. In this task after creating an <tmp> object of the different species that displays the different number of records in the species and the number of records [4]

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

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

Task 7. In this task create a subset <tmp2> of just the species variable and displayed the first five records [5]

```
> tmp2 <- subset(bio, select = species)
> head(tmp2, 5)
    species
1 Bluegill
2 Bluegill
3 Bluegill
4 Bluegill
5 Bluegill
>
```

Task 8. In this task create a table <w> of the specified variable and displayed the class of w [8]

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

  Black Crappie        Bluegill Bluntnose Minnow      Iowa Darter  Largemouth Bass      Pumpkinseed   Tadpole Madtom     Yellow Perch
             36             220              103               32              228               13                6               38
> class(w)
[1] "table"
>
```

Task 9. In this task first converted the <w> to a data frame and named it <t> and display the results which show the variable Var1 and Freq [7]

```
> 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
> |
```

Task 10. Extract In this task after extracting the data for the class(t) and displaying data frame <t> which show the Var1 and Freq [8]

```
> class(t)
[1] "data.frame"
> View(t)
> 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
```

Task 11. In this task, created the table named <cSpec> and get the data from bio$species and find all attributes records of the table.

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

    Black Crappie         Bluegill Bluntnose Minnow     Iowa Darter  Largemouth Bass      Pumpkinseed    Tadpole Madtom     Yellow Perch
               36              220              103               32              228               13                6               38
> class(cSpec)
[1] "table"
> |
```

Task 12. In this task created a which show the records in the percentage of <bio$species> which show the Var1 and Freq in Percentage and covnverted that data in tabular form [9]

```
> cSpecPct <- prop.table(cSpecPct)*100
> cSpecPct

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

    Black Crappie         Bluegill Bluntnose Minnow     Iowa Darter  Largemouth Bass      Pumpkinseed    Tadpole Madtom     Yellow Perch
         5.325444        32.544379        15.236686         4.733728        33.727811         1.923077         0.887574         5.621302
> u <- as.data.frame(cSpec)
> View(u)
> u
                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
```

Task 13. In the task converted the data the table, <cSpecPct>, to a data frame named <u> and confirm that <u> is a data frame

```
> u <- as.data.frame(cSpecPct)
> View(u)
> u
              Var1      Freq
1    Black Crappie  5.325444
2         Bluegill 32.544379
3 Bluntnose Minnow 15.236686
4      Iowa Darter  4.733728
5  Largemouth Bass 33.727811
6      Pumpkinseed  1.923077
7   Tadpole Madtom  0.887574
8     Yellow Perch  5.621302
> |
```

Task 14. In this task created the barplot which is the "Fish Count" and comparison between the species and the counts from the bio$species with the following instructions: [10]
• Title: Fish Count
• Y axis is labeled "COUNTS"
• Y axis limits of 0 to 250
• Color the bars Light Blue
• Rotate X axis label to be vertical
• Set the X axis font magnification to 60% of nominal

```
> barplot(cSpec, main = "Fish Count", ylab = "COUNTS", ylim = c(0,250),
+          col = "Light Blue", cex.axis = .6, las =2, cex.names = .6)
> |
```
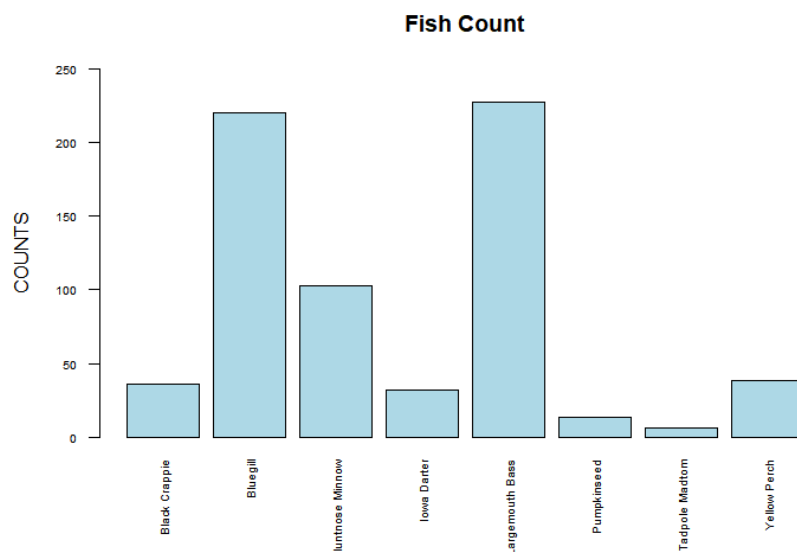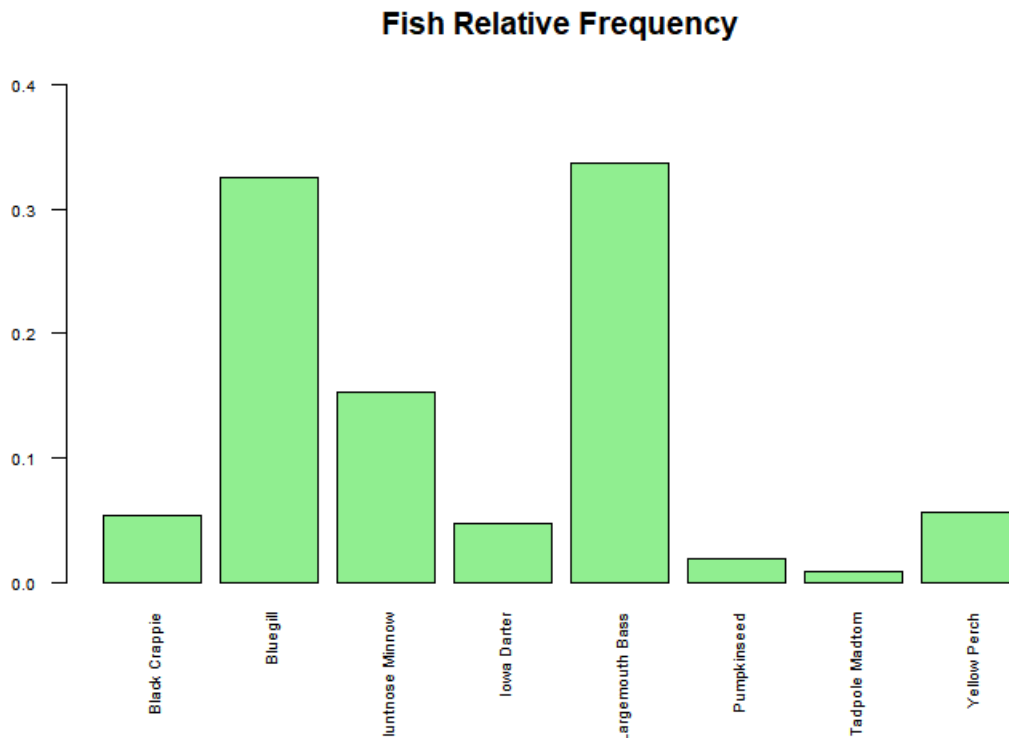

Task 15. In this task plotted the barchart of "Fish Relative Frequency" <cSpecPct>, with the following instructions:
• Y axis limits of 0 to 0.4

• Color the bars Light Green
• Title of "Fish Relative Frequency"

```
> barplot(cSpecPct,  main = "Fish Relative Frequency", ylim = c(0,0.4),
+          col = "Light Green", cex.axis = .6, las = 2, cex.names = .6)
> |
```

Fish Relative Frequency

Task 16. In this task arranged the data in the descending order rearrange the <u> cSpec Pct data frame in descending order of relative frequency [11]

```
> d <- u %>% arrange(desc(Freq))
> d
             Var1 Freq
1  Largemouth Bass  228
2         Bluegill  220
3 Bluntnose Minnow  103
4     Yellow Perch   38
5    Black Crappie   36
6      Iowa Darter   32
7      Pumpkinseed   13
8   Tadpole Madtom    6
> |
```

Task 17. In this task rename the variable of columns of <d> with species and RelFreq [12]

```
> colnames <- c("Species", "RelFreq")
> print(d)
               Var1 Freq
1  Largemouth Bass  228
2         Bluegill  220
3 Bluntnose Minnow  103
4     Yellow Perch   38
5    Black Crappie   36
6      Iowa Darter   32
7      Pumpkinseed   13
8  Tadpole Madtom    6
> names(d) <- c("Species", "RelFreq")
> print(d)
           Species RelFreq
1  Largemouth Bass     228
2         Bluegill     220
3 Bluntnose Minnow     103
4     Yellow Perch      38
5    Black Crappie      36
6      Iowa Darter      32
7      Pumpkinseed      13
8  Tadpole Madtom       6
> |
```

Task 18. In this task called new variable which has NULL value that is why it is not visible in the table to <d> which are named as cumfreq, counts, and cumcounts

```
> d$cumfreq
NULL
> d$counts
NULL
> d$cumcounts
NULL
> d
           Species RelFreq
1  Largemouth Bass     228
2         Bluegill     220
3 Bluntnose Minnow     103
4     Yellow Perch      38
5    Black Crappie      36
6      Iowa Darter      32
7      Pumpkinseed      13
8  Tadpole Madtom       6
> |
```

Task 19. In this task created the parameter with vector <def()> which defines the default value.
```
> def_par <- par()
> def_par
$xlog
[1] FALSE

$ylog
[1] FALSE

$adj
[1] 0.5

$ann
[1] TRUE

$ask
[1] FALSE

$bg
[1] "white"

$bty
[1] "o"

$cex
[1] 1
```

Task 20. In this task create a barplot, <pc>, with the following specifications: [13]
• d$counts of width 1, spacing of .15
• no boarder
• Axes: F
• Y axis limit: 0, 3.05*max
• d$counts na.rm is true
• y label is "Cumulative Counts"
• scale x axis to 70%
• names.arg: d$Species

• Title of the barplot is "Species Pareto"
• las: 2

```
> pc <- barplot(d$counts, width=1, space= .15, border= NA, axes=F, ylim=c(0,3.05*max(d$counts, na.rm=T)),
+              ylab="Cumulative Counts", cex.names=.5, names.arg=d$Species, main="Species Pareto ", las=2)
```
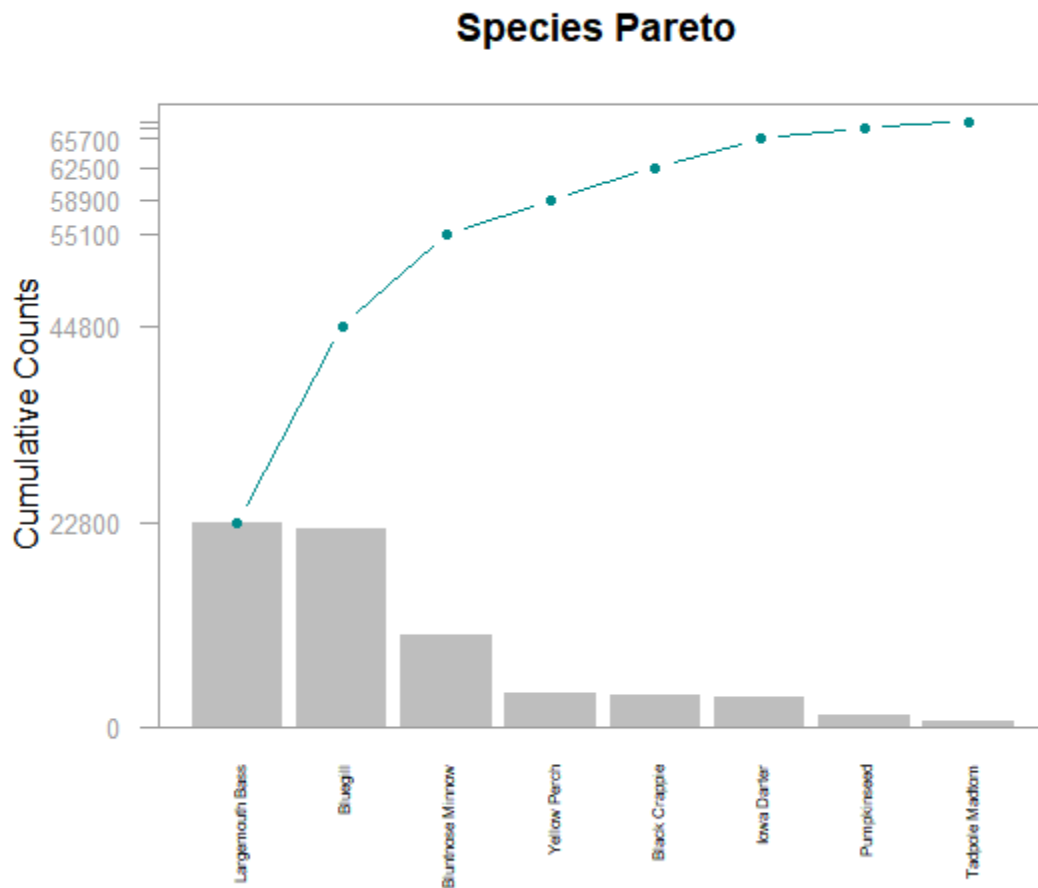

Species Pareto
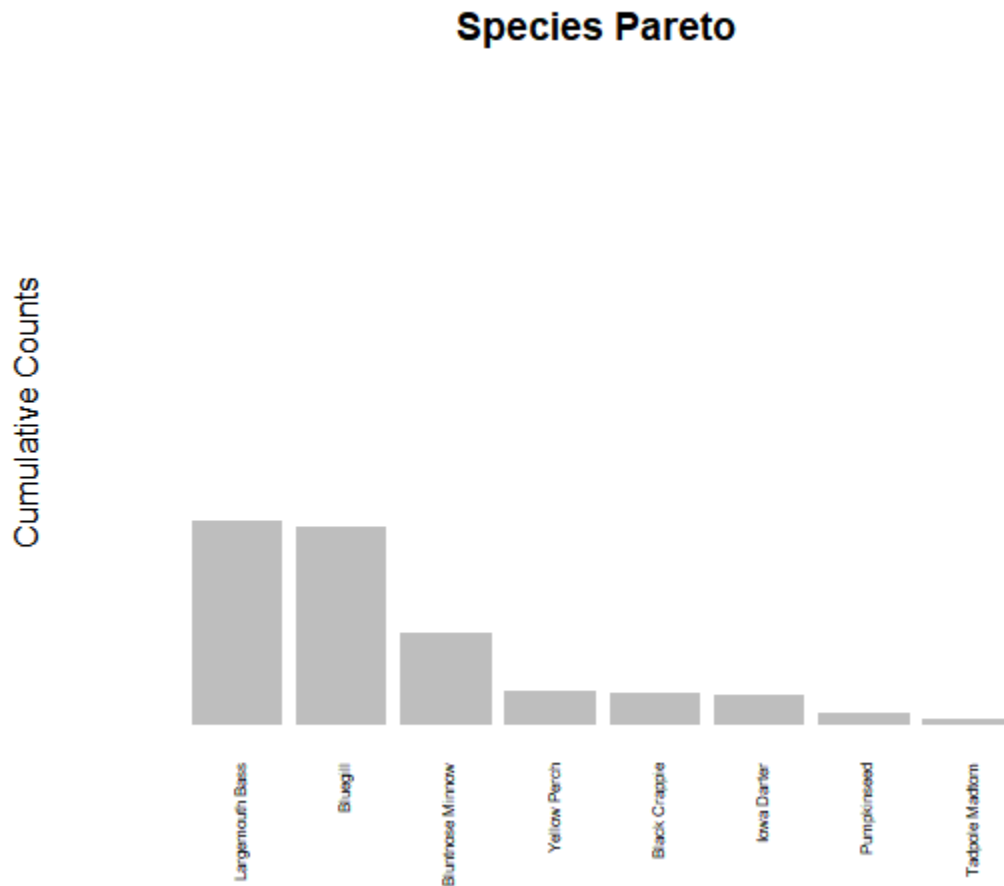
Task 21. In this task as per the instruction, plotted a cumulative counts line to the <pc>[14]

- 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")
> |
```
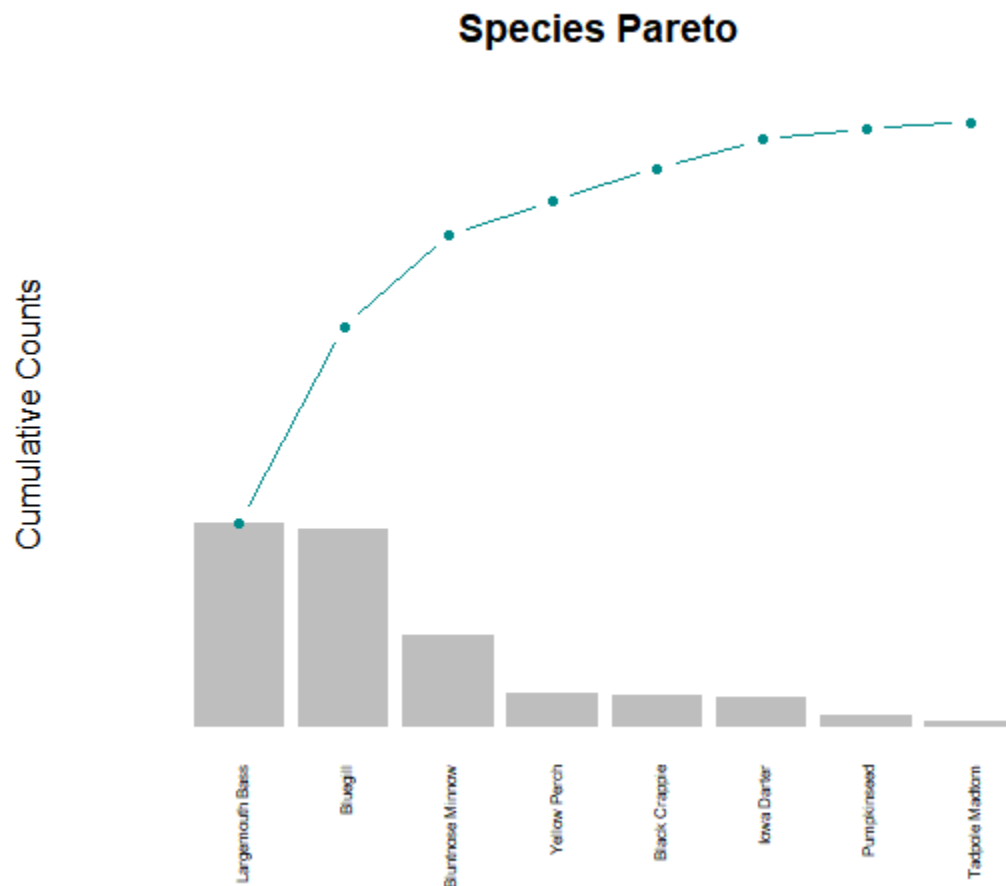

Task 22. In this task placed the graph in a grey box border to the pareto plot [16]

Species Pareto

Task 23. Ain this Task added the axis on the left side with the instruction given below•
Horizontal values at tick marks at cumcounts on side 2 [17]
• Tickmark color of grey62
• Color of axis is grey62
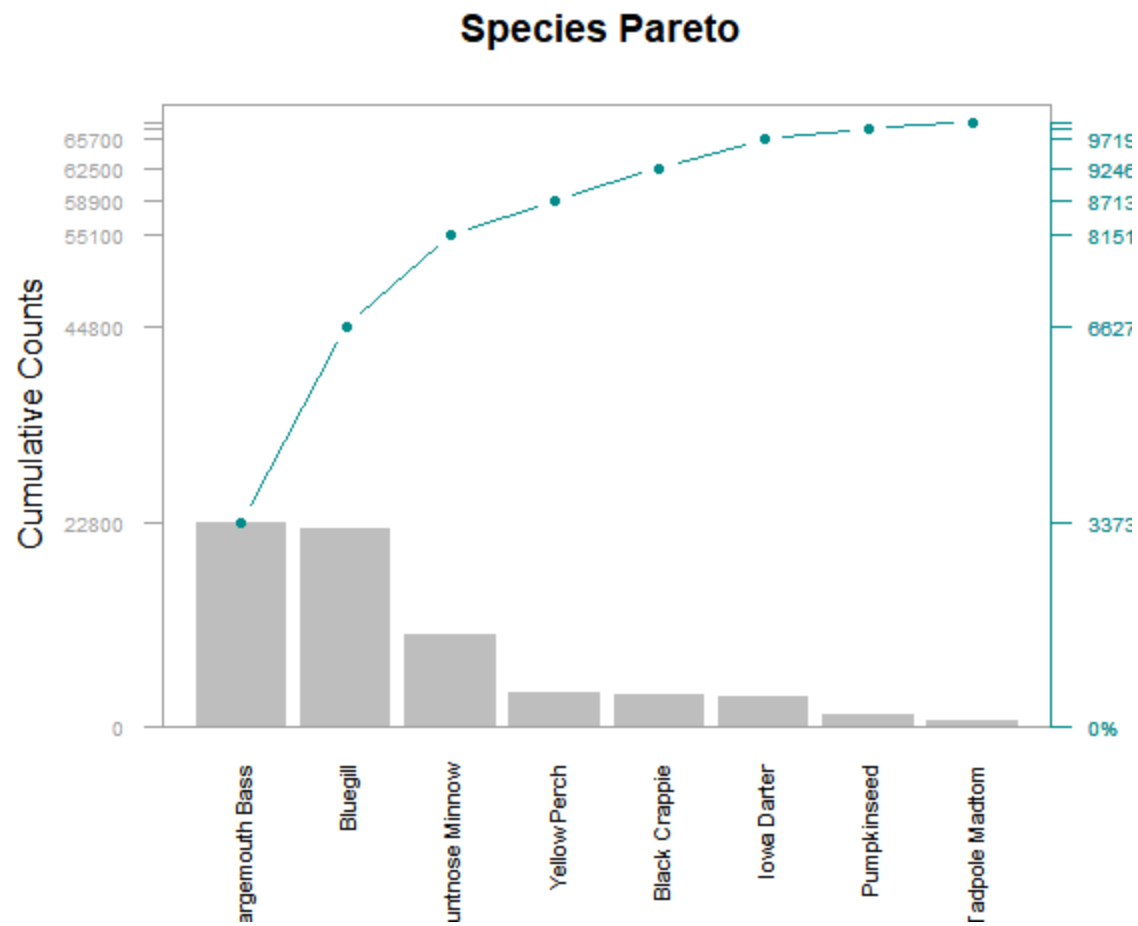• Axis scaled to 80% of normal

```
> axis(side=4, at=c(0, d$cumcounts), labels=paste(c(0, round(d$cumfreq * 100)),
+              "%", sep=""), las = 1, col.axis="gray62", col="gray62", cex.axis = 0.6)
> |
```

Species Pareto
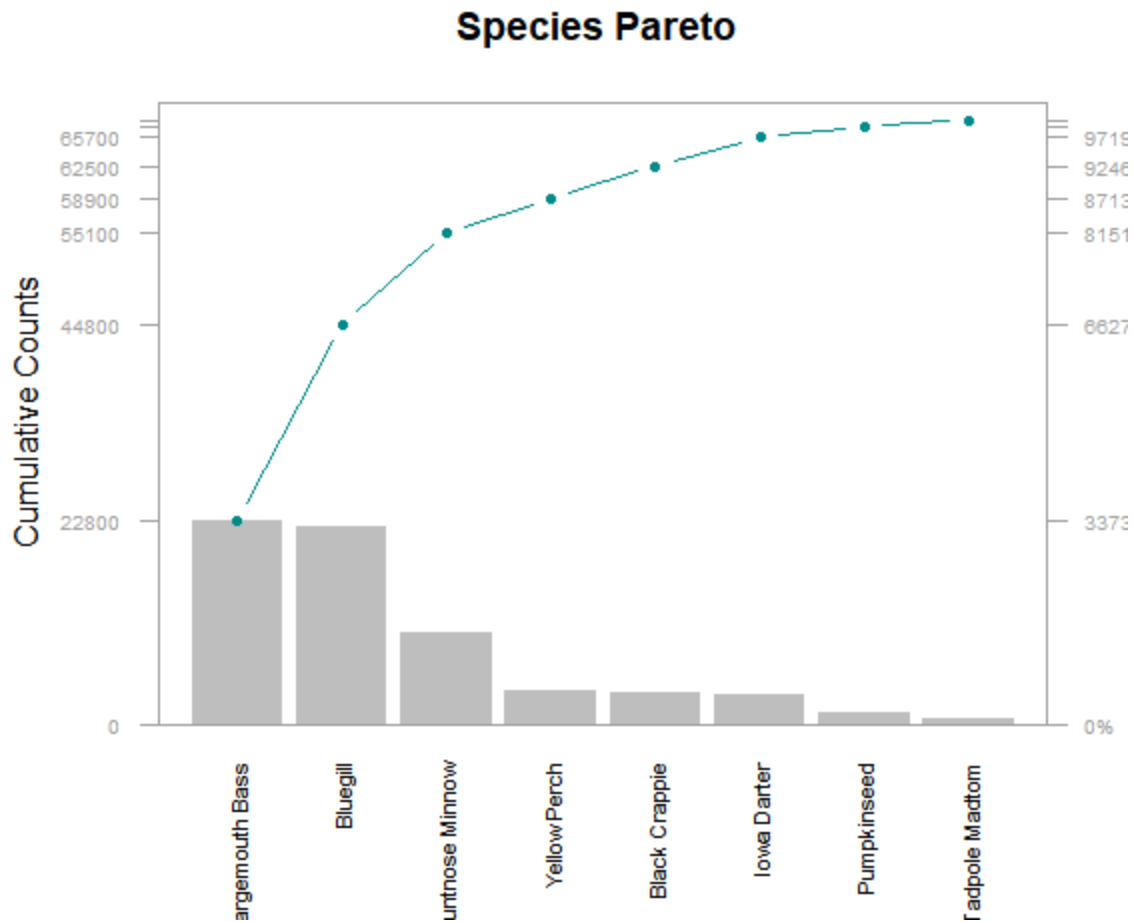
Task 24. Ain this task added the axis on the right side of the box with the instruction given below

• 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 * 100)),
+                "%", sep=""), las = 1, col.axis="cyan4", col="cyan4", cex.axis = 0.6)
```

Species Pareto

Cumulative Counts (left axis)	(right axis)
65700	9719
62500	9246
58900	8713
55100	8151
44800	6627
22800	3373
0	0%

Categories (left to right): argemouth Bass, Bluegill, untnose Minnow, Yellow Perch, Black Crappie, Iowa Darter, Pumpkinseed, Tadpole Madtom

25. Display the finished Species Pareto Plot (without the star watermarks). Have your last name on the plot

```
> mtext (text = "JAIN", side = 3)
>
```

Species Pareto

JAIN
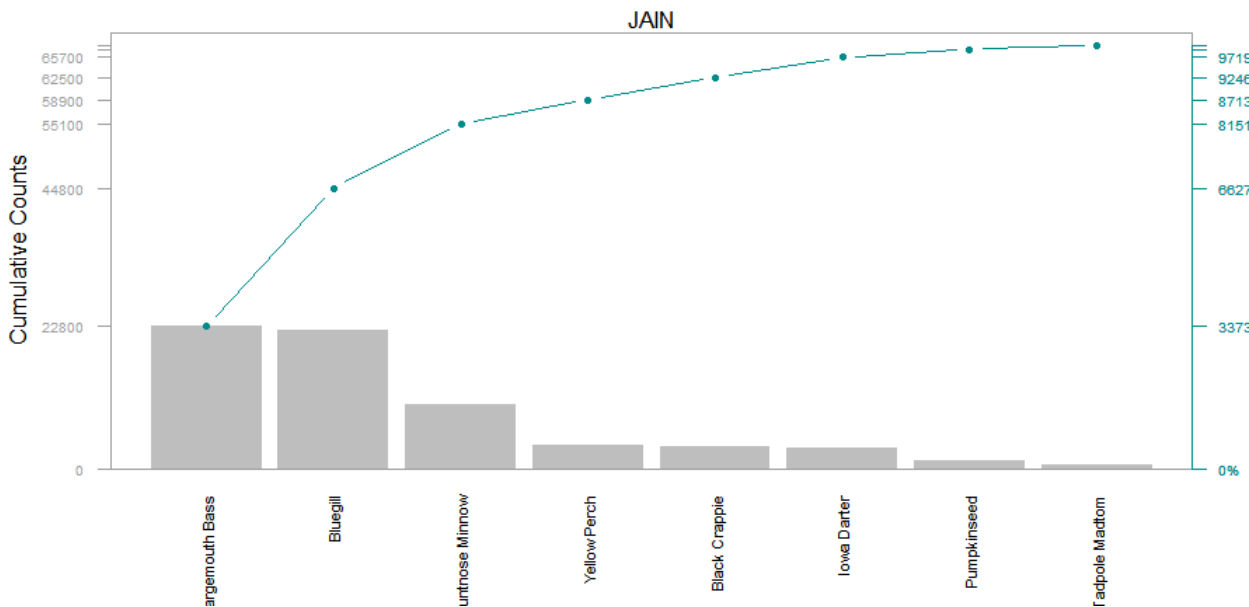
Summary Report

To visualize the data in graphical representation about the species. In this report, inch-bio data has 676 observations to analyze the different types of species surviving in this distribution. If we considered these species for measuring the life span of species like the data shows Black Crappie (36) Bluegill 220, Bluntnose Minnow 103, Iowa Darter 32, Largemouth Bass 228, Pumpkinseed 13, Tadpole Madtom 6, Yellow Perch 38, in this data Bluegill has the highest survival ration whereas Tadpole Madtom has the rare survival ratio. This dataset shows the frequency by which we can analyze the survival of species over a period of time. If we talk about the species length Largemouth Bass have the highest percentage 33.73% of the ratio of survival. and Tadpole Madtom has the lowest percentage (0.89%). R programming has an inbuilt function to change the variable name of the dataset variables. This report shows the relative frequency, cumulative frequency, counts, cumulative counts, especially of all the species records.

The main purpose of using R is to define the dataset into visualization form which helps analysts to perform better in decision-making processes. Likewise, the driven data is visualized in a bar chart which itself explains the above observations. In fact, the length of the bar signifies the life of the species and as we have arranged the data in descending order it shows the longest bar on the left side of the graph, and the shortest bar shows the survival rate of the particular species. It represents the situation of the frequency of the bar chart. Moreover, we use a Pareto chart while analyzing the data.

In conclusion, to show the frequency in the data used Pareto chart. Through, R it is easy to demonstrate the results shown in the respective graphs.

Bibliography

[1] Importing Data

robk@statmethods.net
https://www.statmethods.net/input/importingdata.html

[2] Admin & Admin

http://rfunction.com/archives/699

[3] R: Count number of objects in list

KarlKarl 5 et al.
https://stackoverflow.com/questions/1740524/r-count-number-of-objects-in-list

[4] https://stat.ethz.ch/R-manual/R-devel/library/base/html/tempfile.html

[5] Quick-R: Subsetting Data - statmethods.net

The subset ( ) function is the easiest way to select variables and observations. In the following example, we select all rows that have a value of age greater than or equal to 20 or age less then 10. We keep the ID and Weight columns. # using subset function. newdata <- subset (mydata, age >= 20 | age < 10,
https://www.statmethods.net/management/subset.html

[6] How to Create Tables in R (With Examples)

Zach
https://www.statology.org/create-table-in-r/

[7] Data Frames

https://www.tutorialspoint.com/r/r_data_frames.htm#:~:text=A%20data%20frame%20is%20a,of%20values%20from%20each%20column.&text=The%20data%20stored%20in%20a,same%20number%20of%20data%20items.

[8] Extract data frame cell value

https://campus.datacamp.com/courses/model-a-quantitative-trading-strategy-in-r/chapter-1-introduction-to-r-for-trading?ex=4

[9] How do I find the percentage of something in R?

daultongray8daultongray8 6311 gold badge11 silver badge55 bronze badges et al.
https://stackoverflow.com/questions/42379751/how-do-i-find-the-percentage-of-something-in-r/42379897

[10] Bar Plots

robk@statmethods.net
https://www.statmethods.net/graphs/bar.html

[11] Reorder Data Frame Rows in R - Datanovia

Reorder Data Frame Rows in R. This tutorial describes how to reorder (i.e., sort) rows, in your data table, by the value of one or more columns (i.e., variables). Sort a data frame rows in ascending order (from low to high) using the R function arrange () [dplyr package] Sort rows in descending order (from high to low) using arrange () in ...
https://www.datanovia.com/en/lessons/reorder-data-frame-rows-in-r

[12] Rename Data Frame Columns in R - Datanovia

In this tutorial, you will learn how to rename the columns of a data frame in R.This can be done easily using the function rename() [dplyr package].It's also possible to use R base functions, but they require more typing.
https://www.datanovia.com/en/lessons/rename-data-frame-columns-in-r

[13] Barplot in R - R CODER

In this article we are going to explain the basics of creating bar plots in R. 1 The R barplot function. 1.1 Barplot graphical parameters: title, axis labels and colors. 1.2 Change group labels. 1.3 Barplot width and space of bars. 1.4 Barplot from data frame or list. 1.5 Barplot for continuous variable.
https://r-coder.com/barplot-r

[14] How to create a cumulative graph in R

d-cubedd-cubed 93455 gold badges3030 silver badges5353 bronze badges & thelatemailthelatemail 84.3k1212 gold badges119119 silver badges174174 bronze badges
https://stackoverflow.com/questions/38492329/how-to-create-a-cumulative-graph-in-r

[15] R - Pareto Chart

https://www.geeksforgeeks.org/r-pareto-chart/

[16] Graphical Parameters

robk@statmethods.net
https://www.statmethods.net/advgraphs/parameters.html

[17] Axes and Text

robk@statmethods.net
https://www.statmethods.net/advgraphs/axes.html

Appendix

R Script


```r
#*****************************************************************
#*                                                             *#
#*                                                             *#
#*                  Inchbio Plotting in R                      *#
#*                                                             *#
#*                                                             *#
#*****************************************************************

scriptname <- ("Inchbio Plotting in R - JAIN")

install.packages("FSA")
install.packages("FSAdata")
install.packages("magrittr")
install.packages("dplyr")
install.packages("tidyr")
install.packages("plyr")
install.packages("tidyverse")

library(FSA)
library(FSAdata)
library(magrittr)
library(dplyr)
library(tidyr)
library(plyr)
library(tidyverse)

fishR()

bio <- read.csv("C:\\NEU BOSTON\\ALY6000 Submission\\ALY6000-Final\\Module 3 Week 2\\inchbio.csv")
bio

head(bio)
tail(bio)
str(bio)

counts <- bio

counts <- count(bio$Species)
counts <- cbind (length (bio$species), list(bio$species))
view(counts)
counts

unique(bio$species, incomparables = NULL)

tmp <- table (bio$species)
tmp

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

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

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



```r
t <- as.data.frame(w)
t
class(t)
view(t)
t

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

cSpecPct <- table (bio$species)
cSpecPct

cSpecPct <- prop.table(cSpecPct)*100
class(cSpecPct)
cSpecPct <- as.data.frame(cSpecPct)
cSpecPct


u <- as.data.frame(cSpecPct)
view(u)
u

barplot(cSpec,  main = "Fish Count", ylab = "COUNTS", ylim = c(0,250),
        col = "Light Blue", cex.axis = .6, las =2, cex.names = .6)

barplot(cSpecPct,  main = "Fish Relative Frequency", ylim = c(0,0.4),
        col = "Light Green", cex.axis = .6, las = 2, cex.names = .6)


d <- u %>% arrange(desc(Freq))
d


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

d$cumfreq
d$counts
d$cumcounts
d

def_par <- par()
def_par

pc <- barplot(d$counts, width=1, space= .15, border= NA, axes=F, ylim=c(0,3.05*max(d$counts, na.rm=T)),
        ylab="cumulative Counts", cex.names=.5, names.arg=d$species, main="Species Pareto ", las=2)
```

```r
81  barplot(cSpec,  main = "Fish Count", ylab = "COUNTS", ylim = c(0,250),
82          col = "Light Blue", cex.axis = .6, las =2, cex.names = .6)
83
84  barplot(cSpecPct,  main = "Fish Relative Frequency", ylim = c(0,0.4),
85          col = "Light Green", cex.axis = .6, las = 2, cex.names = .6)
86
87
88
89  d <- u %>% arrange(desc(Freq))
90  d
91
92
93
94  colnames <- c("Species", "RelFreq")
95  print(d)
96  names(d) <- c("Species", "RelFreq")
97  print(d)
98
99  d$cumfreq
100 d$counts
101 d$cumcounts
102 d
103
104
105 def_par <- par()
106 def_par
107
108 pc <- barplot(d$counts, width=1, space= .15, border= NA, axes=F, ylim=c(0,3.05*max(d$counts, na.rm=T)),
109          ylab="Cumulative Counts", cex.names=-.5, names.arg=d$species, main="Species Pareto ", las=2)
110 lines(pc, d$cumcounts, type="b", cex=0.7, pch = 19, col="cyan4")
111
112
113
114
115 pc <- barplot(d$counts, width=1, space= .15, border= NA, axes=F, ylim=c(0,3.05*max(d$counts, na.rm=T)),
116          ylab="Cumulative Counts", cex.names=-.7, names.arg=d$species, main="Species Pareto ", las=2)
117 lines(pc, d$cumcounts, type="b", cex=0.7, pch = 19, col="cyan")
118 axis(side=2, at=c(0, d$cumcounts), las=1, col.axis="grey62", col="grey62", cex.axis=0.6)
119
120 box(col="grey62")
121
122
123 axis(side=4, at=c(0, d$cumcounts), labels=paste(c(0, round(d$cumfreq * 100)),
124          "%", sep=""), las = 1, col.axis="gray62", col="gray62", cex.axis = 0.6)
125
126 axis(side=4, at=c(0, d$cumcounts), labels=paste(c(0, round(d$cumfreq * 100)),
127          "%", sep=""), las = 1, col.axis="cyan4", col="cyan4", cex.axis = 0.6)
128
129
130
131 mtext (text = "JAIN", side = 3, adj = )
132
133
134
135 #***************END OF REPORT 4***************************
```

```
135:2  (Top Level)                                                    R Script

Console
```

****************************** END OF REPORT ******************************