V S N Sai Krishna Mohan Kocherlakota

Exploratory Data Analysis (EDA) of Two Data Sets

3rd October 2023

INTRODUCTION

In this project, I worked with two datasets and performed various functions and operators. I also created plots and graphs as part of the project. Exploring packages such as tidyverse, janitor, and ggplot expanded my toolkit and enhanced the efficiency of data analysis. My main objective in completing this project was to gain a thorough understanding of all the packages and operators I used. Graphical visualization played a pivotal role in presenting my findings effectively.

KEY-FINDINGS

Part-1: - Countries GPD

Problem 5: -

In this code, I have loaded the janitor package, in which I have used the "clean_names" function. Janitor is used for initial data exploration and cleaning, particularly for restructuring the data frame and addressing problematic variable names.

```
> data_2015 <- clean_names(data_2015) #Clean name is used for better structur
e of data frame.
> data_2015
# A tibble: 158 × 13
   country     region happiness_rank happiness_score standard_error economy_g
dp_per_capita family health_life_expectancy
   <chr>       <chr>           <dbl>           <dbl>          <dbl>
<dbl>  <dbl>                  <dbl>
 1 Switzerland Weste…              1            7.59         0.0341
1.40   1.35                   0.941
 2 Iceland     Weste…              2            7.56         0.0488
1.30   1.40                   0.948
 3 Denmark     Weste…              3            7.53         0.0333
1.33   1.36                   0.875
 4 Norway      Weste…              4            7.52         0.0388
1.46   1.33                   0.885
 5 Canada      North…              5            7.43         0.0355
1.33   1.32                   0.906
 6 Finland     Weste…              6            7.41         0.0314
1.29   1.32                   0.889
 7 Netherlands Weste…              7            7.38         0.0280
1.33   1.28                   0.893
 8 Sweden      Weste…              8            7.36         0.0316
1.33   1.29                   0.911
 9 New Zealand Austr…              9            7.29         0.0337
1.25   1.32                   0.908
10 Australia   Austr…             10            7.28         0.0408
1.33   1.31                   0.932
# i 148 more rows
# i 5 more variables: freedom <dbl>, trust_government_corruption <dbl>, genero
sity <dbl>, dystopia_residual <dbl>,
#   gff_stat <dbl>
# i Use `print(n = ...)` to see more rows
```

Problem 11: -

In this code, I have utilized the pipeline operator "%>%" to efficiently execute multiple operators. This operator is a component of the dplyr package. Additionally, I have employed the "summarize" and "mutate" functions within the code.

```
> happy_summary <- happy_df %>% summarise(mean_happiness = mean(happiness_sco
re), max_happiness = max(happiness_score), mean_freedom = mean(freedom), max_
```

```
freedom = max(freedom)) #Summary of all the columns of calculated on top and
putting it into data frame.
> happy_summary
# A tibble: 1 × 4
  mean_happiness max_happiness mean_freedom max_freedom
           <dbl>         <dbl>        <dbl>       <dbl>
1           5.38          7.59        0.429       0.670
```

Problem 13: -

In this section of the code, I have calculated a new variable called "Europe and Africa" and then separately computed their respective GDPs based on the economy's GDP per capita.

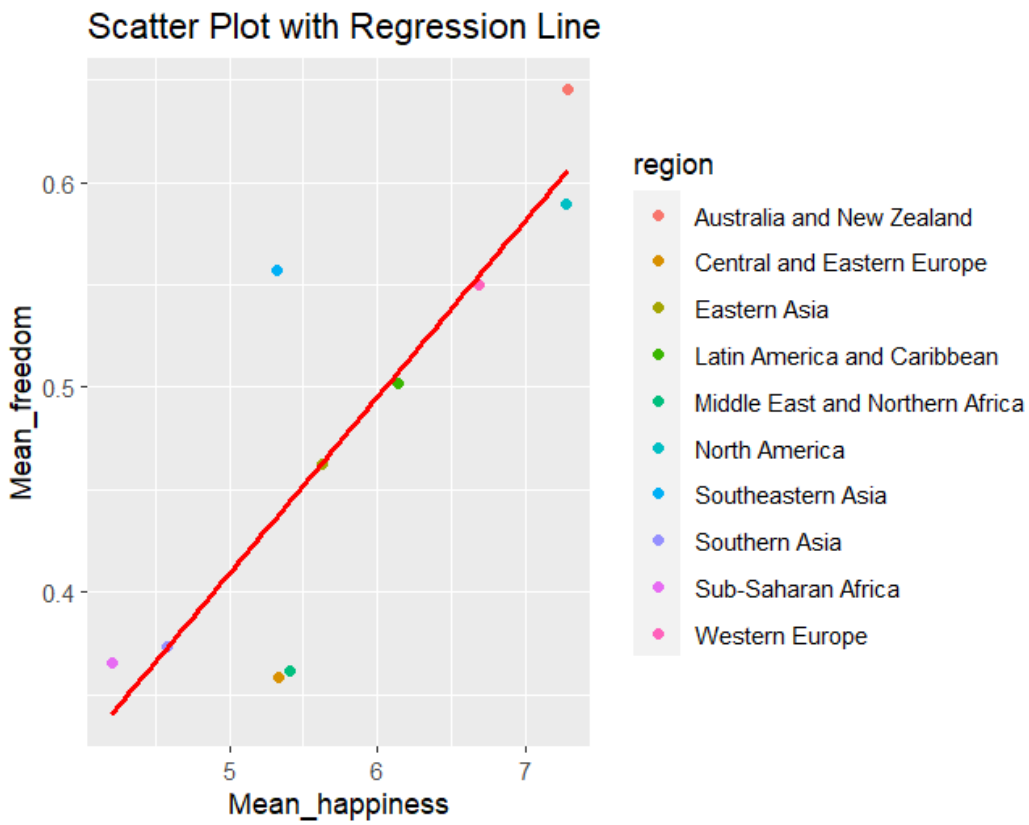
```
> europe <- data_2015 %>% group_by(region) %>%
+   filter(region %in% "Western Europe") %>%
+   arrange(desc(happiness_rank)) %>%
+   slice(1:10)
> europe  #Creating Europe data frame to calculate mean of least Europe count
ries mean
# A tibble: 10 × 13
# Groups:   region [1]
   country     region happiness_rank happiness_score standard_error economy_g
dp_per_capita family health_life_expectancy
   <chr>       <chr>           <dbl>           <dbl>          <dbl>
<dbl>  <dbl>                  <dbl>
 1 Greece      Weste…            102            4.86         0.0506
1.15  0.929                  0.882
 2 Portugal    Weste…             88            5.10         0.0480
1.16  1.14                   0.875
 3 Cyprus      Weste…             67            5.69         0.0558
1.21  0.893                  0.924
 4 North Cypr… Weste…             66            5.70         0.0564
1.21  1.07                   0.924
 5 Italy       Weste…             50            5.95         0.0391
1.25  1.20                   0.954
 6 Malta       Weste…             37            6.30         0.0421
1.21  1.30                   0.887
 7 Spain       Weste…             36            6.33         0.0347
1.23  1.31                   0.956
 8 France      Weste…             29            6.58         0.0351
1.28  1.26                   0.946
 9 Germany     Weste…             26            6.75         0.0185
1.33  1.30                   0.892
10 United Kin… Weste…             21            6.87         0.0187
1.27  1.29                   0.909
# ℹ 5 more variables: freedom <dbl>, trust_government_corruption <dbl>, genero
sity <dbl>, dystopia_residual <dbl>,
#   gff_stat <dbl>
>
> eurpoe_gdp  = round(mean(europe$economy_gdp_per_capita),digits = 2)
> eurpoe_gdp #Mean of Europe GDP
[1] 1.23
>
> africa <- data_2015 %>% group_by(region) %>%
+   filter(region %in% "Sub-Saharan Africa") %>%
+   slice(1:10)
> africa #Creating Africa data frame to calculate mean of least Africa countr
ies mean
# A tibble: 10 × 13
# Groups:   region [1]
   country     region happiness_rank happiness_score standard_error economy_g
dp_per_capita family health_life_expectancy
   <chr>       <chr>           <dbl>           <dbl>          <dbl>
<dbl>  <dbl>                  <dbl>
```

```
 1 Mauritius    Sub-S…            71            5.48         0.0720
1.01    0.985                 0.710
 2 Nigeria      Sub-S…            78            5.27         0.0419
0.654   0.904                 0.160
 3 Zambia       Sub-S…            85            5.13         0.0699
0.470   0.916                 0.299
 4 Somaliland… Sub-S…            91            5.06         0.0616
0.188   0.952                 0.439
 5 Mozambique   Sub-S…            94            4.97         0.0790
0.0831  1.03                  0.0913
 6 Lesotho      Sub-S…            97            4.90         0.0944
0.375   1.04                  0.0761
 7 Swaziland    Sub-S…           101            4.87         0.0874
0.712   1.07                  0.0757
 8 South Afri… Sub-S…           113            4.64         0.0458
0.920   1.18                  0.277
 9 Ghana        Sub-S…           114            4.63         0.0474
0.546   0.680                 0.401
10 Zimbabwe     Sub-S…           115            4.61         0.0429
0.271   1.03                  0.335
# ℹ 5 more variables: freedom <dbl>, trust_government_corruption <dbl>, genero
sity <dbl>, dystopia_residual <dbl>,
#   gff_stat <dbl>
>
> africa_gdp =round(mean(africa$economy_gdp_per_capita), digits = 3)
> africa_gdp #Calculation of mean of Africa GDP
[1] 0.523
>
> gdp_df <- tibble(europe_gdp,africa_gdp)
> gdp_df #Insertion of European and Africa GDP
# A tibble: 1 × 2
  europe_gdp africa_gdp
       <dbl>      <dbl>
1       1.23      0.523
```

Problem 14: -

I utilized the ggplot package to create a scatter plot, providing information about the mean happiness and mean freedom of each region. I achieved this by utilizing the "regional_stats_df" data frame, which was generated through the group by function applied to the "region" variable in the original data frame.

```
> #14
> ggplot(regional_stats_df, aes(x = mean_happiness, y = mean_freedom,color =
region)) +
+   geom_point() +  # Scatter plot
+   geom_smooth(method = "lm", color = "red", se = FALSE) +  # Regression lin
e
+   labs(title = "Scatter Plot with Regression Line", x = "Mean_happiness", y
= "Mean_freedom", color = "region")
`geom_smooth()` using formula = 'y ~ x'
```



Part-2: - Baseball analysis

Problem 6-9

I have used mutate and the formulae to calculate batting average and on-base percentage.

```
> #6 & #7
> baseball <- baseball %>% mutate(BA = round((H)/(AB), digits = 3))
> baseball # Creating Batting average and rounding off the value of BA in Bas
eball data frame.
          Last  First Age   G  PA  AB   R   H X2B X3B HR RBI SB CS  BB  SO
BA   OBP
1        Acker    Jim  27  21  28  28   1   3   1   0  0   0  0  0   0  21 0.
107 0.107
2       Adduci    Jim  26   3  13  11   2   1   1   0  0   0  0  0   1   2 0.
091 0.167
3       Aguayo   Luis  27  62 146 133  17  28   6   1  4  13  1  1   8  26 0.
211 0.255
4     Aguilera   Rick  24  32  57  51   4   8   0   0  2   6  0  0   3  12 0.
157 0.204
5      Aldrete   Mike  25  84 256 216  27  54  18   3  2  25  1  3  33  34 0.
250 0.349
6    Alexander  Doyle  35  18  45  38   2   8   1   0  0   5  0  0   0   8 0.
211 0.211
7     Allanson   Andy  24 101 324 293  30  66   7   3  1  29 10  1  14  36 0.
225 0.261
8        Almon   Bill  33 102 230 196  29  43   7   2  7  27 11  4  30  38 0.
219 0.323
9      Amelung     Ed  27   8  11  11   0   1   0   0  0   0  0  0   0   4 0.
091 0.091
```

10	Andersen	Larry	33	48	7	6	0	0	0	0	0	0	0	0	0	3	0.000 0.000
11	Anderson	Dave	25	92	241	216	31	53	9	0	1	15	5	1	22	39	0.245 0.315
12	Anderson	Rick	29	15	12	11	1	1	0	0	0	0	0	0	0	4	0.091 0.091
13	Armas	Tony	32	121	453	425	40	112	21	4	11	58	0	3	24	77	0.264 0.303
14	Asadoor	Randy	23	15	60	55	9	20	5	0	0	7	1	2	3	13	0.364 0.397
15	Ashby	Alan	34	120	361	315	24	81	15	0	7	38	1	0	39	56	0.257 0.339
16	Assenmacher	Paul	25	61	8	6	0	0	0	0	0	0	0	0	2	3	0.000 0.250
17	Backman	Wally	26	124	440	387	67	124	18	2	1	27	13	7	36	32	0.320 0.378
18	Bailey	Mark	24	57	182	153	9	27	5	0	4	15	1	1	28	45	0.176 0.304
19	Baines	Harold	27	145	618	570	72	169	29	2	21	88	2	1	38	89	0.296 0.340
20	Baker	Dusty	37	83	271	242	25	58	8	0	4	19	0	1	27	37	0.240 0.316
21	Baker	Doug	25	13	30	24	1	3	1	0	0	0	0	0	2	7	0.125 0.192
22	Balboni	Steve	29	138	562	512	54	117	25	1	29	88	0	0	43	146	0.229 0.288
23	Baller	Jay	25	36	6	5	0	0	0	0	0	0	0	0	0	1	0.000 0.000
24	Bando	Chris	30	92	290	254	28	68	9	0	2	26	0	1	22	49	0.268 0.326
25	Barfield	Jesse	26	158	671	589	107	170	35	2	40	108	8	8	69	146	0.289 0.363
26	Bargar	Greg	27	22	2	2	0	0	0	0	0	0	0	0	0	2	0.000 0.000
27	Barrett	Marty	28	158	713	625	94	179	39	4	4	60	15	7	65	31	0.286 0.354
28	Bass	Kevin	27	157	640	591	83	184	33	5	20	79	22	13	38	72	0.311 0.353
29	Bathe	Bill	25	39	112	103	9	19	3	0	5	11	0	0	2	20	0.184 0.200
30	Baylor	Don	37	160	687	585	93	139	23	1	31	94	3	5	62	111	0.238 0.311
31	Beane	Billy	24	80	194	183	20	39	6	0	3	15	2	3	11	54	0.213 0.258
32	Bedrosian	Steve	28	68	6	5	0	1	0	0	0	0	0	0	1	1	0.200 0.333
33	Bell	Buddy	34	155	655	568	89	158	29	3	20	75	2	8	73	49	0.278 0.360
34	Bell	George	26	159	690	641	101	198	38	6	31	108	7	8	41	62	0.309 0.350
35	Bell	Jay	20	5	16	14	3	5	2	0	1	4	0	0	2	3	0.357 0.438
36	Bell	Terry	23	8	5	3	0	0	0	0	0	0	0	0	2	1	0.000 0.400
37	Belliard	Rafael	24	117	350	309	33	72	5	2	0	31	12	2	26	54	0.233 0.293
38	Benedict	Bruce	30	64	183	160	11	36	10	1	0	13	1	0	15	10	0.225 0.291
39	Beniquez	Juan	36	113	395	343	48	103	15	0	6	36	2	3	40	49	0.300 0.373
40	Berenguer	Juan	31	46	10	7	0	1	0	0	0	0	0	0	0	1	0.143 0.143
41	Berenyi	Bruce	31	14	12	11	0	0	0	0	0	0	0	0	0	3	0.000 0.000

```
42     Bergman    Dave  33  65 151 130  14  30   6   1  1   9  0  0  21  16 0.
231 0.338
43   Bernazard    Tony  29 146 636 562  88 169  28   4 17  73 17  8  53  77 0.
301 0.361
44       Berra    Dale  29  42 121 108  10  25   7   0  2  13  0  0   9  14 0.
231 0.291
45   Biancalana  Buddy  26 100 209 190  24  46   4   4  2   8  5  1  15  50 0.
242 0.298
46    Bielecki    Mike  26  31  54  48   3   3   0   0  0   1  0  0   2  26 0.
062 0.100
47   Bilardello   Dann  27  79 212 191  12  37   5   0  4  17  1  0  14  32 0.
194 0.249
48    Bittiger   Jeff  24   3   4   3   1   1   0   0  1   1  0  0   0   1 0.
333 0.333
49        Blue   Vida  36  28  53  43   3   4   1   0  1   3  0  0   6  20 0.
093 0.204
50      Bochte  Bruce  35 125 473 407  57 104  13   1  6  43  3  2  65  68 0.
256 0.358
51       Bochy  Bruce  31  63 142 127  16  32   9   0  8  22  1  0  14  23 0.
252 0.326
52      Bockus  Randy  25   6   1   1   0   0   0   0  0   0  0  0   0   1 0.
000 0.000
53      Boever    Joe  25  11   2   2   0   1   0   0  0   0  0  0   0   0 0.
500 0.500
54       Boggs   Wade  28 149 693 580 107 207  47   2  8  71  0  4 105  44 0.
357 0.455
55       Bonds  Barry  21 113 484 413  72  92  26   3 16  48 36  7  65 102 0.
223 0.328
 [ reached 'max' / getOption("max.print") -- omitted 671 rows ]
> #8 & #9
> baseball <- baseball %>% mutate(OBP = round((H+BB)/(AB+BB),digits = 3))
> baseball #Calculating on-base percentage and rounding off it's value
          Last  First Age   G  PA  AB   R   H X2B X3B HR RBI SB CS  BB  SO
BA   OBP
1        Acker    Jim  27  21  28  28   1   3   1   0  0   0  0  0   0  21 0.
107 0.107
2       Adduci    Jim  26   3  13  11   2   1   1   0  0   0  0  0   1   2 0.
091 0.167
3       Aguayo   Luis  27  62 146 133  17  28   6   1  4  13  1  1   8  26 0.
211 0.255
4     Aguilera   Rick  24  32  57  51   4   8   0   0  2   6  0  0   3  12 0.
157 0.204
5      Aldrete   Mike  25  84 256 216  27  54  18   3  2  25  1  3  33  34 0.
250 0.349
6    Alexander  Doyle  35  18  45  38   2   8   1   0  0   5  0  0   0   8 0.
211 0.211
7     Allanson   Andy  24 101 324 293  30  66   7   3  1  29 10  1  14  36 0.
225 0.261
8        Almon   Bill  33 102 230 196  29  43   7   2  7  27 11  4  30  38 0.
219 0.323
9      Amelung     Ed  27   8  11  11   0   1   0   0  0   0  0  0   0   4 0.
091 0.091
10    Andersen  Larry  33  48   7   6   0   0   0   0  0   0  0  0   0   3 0.
000 0.000
11    Anderson   Dave  25  92 241 216  31  53   9   0  1  15  5  1  22  39 0.
245 0.315
12    Anderson   Rick  29  15  12  11   1   1   0   0  0   0  0  0   0   4 0.
091 0.091
13       Armas   Tony  32 121 453 425  40 112  21   4 11  58  0  3  24  77 0.
264 0.303
14     Asadoor  Randy  23  15  60  55   9  20   5   0  0   7  1  2   3  13 0.
364 0.397
15       Ashby   Alan  34 120 361 315  24  81  15   0  7  38  1  0  39  56 0.
257 0.339
```

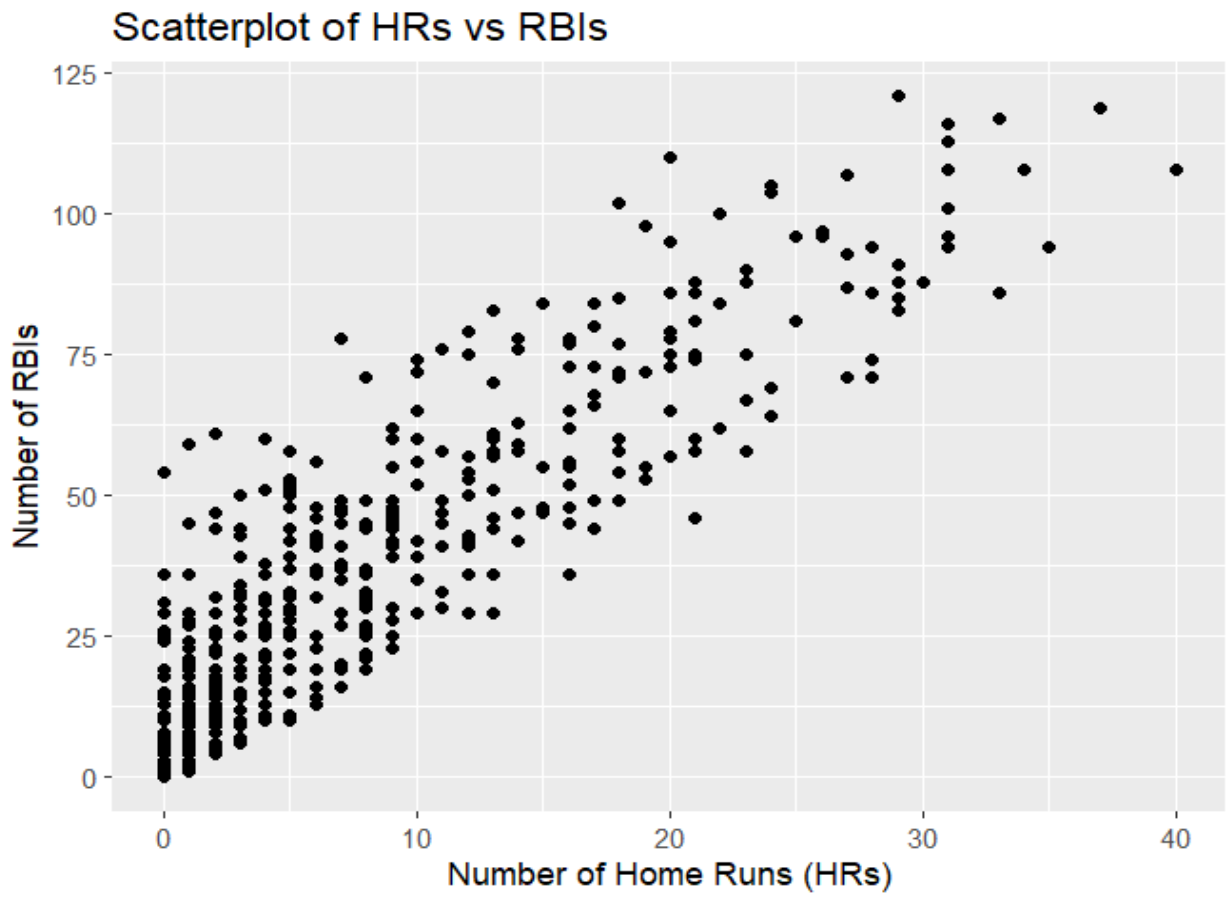
16 Assenmacher Paul 25 61 8 6 0 0 0 0 0 0 0 0 2 3 0.
000 0.250
17 Backman Wally 26 124 440 387 67 124 18 2 1 27 13 7 36 32 0.
320 0.378
18 Bailey Mark 24 57 182 153 9 27 5 0 4 15 1 1 28 45 0.
176 0.304
19 Baines Harold 27 145 618 570 72 169 29 2 21 88 2 1 38 89 0.
296 0.340
20 Baker Dusty 37 83 271 242 25 58 8 0 4 19 0 1 27 37 0.
240 0.316
21 Baker Doug 25 13 30 24 1 3 1 0 0 0 0 0 2 7 0.
125 0.192
22 Balboni Steve 29 138 562 512 54 117 25 1 29 88 0 0 43 146 0.
229 0.288
23 Baller Jay 25 36 6 5 0 0 0 0 0 0 0 0 0 1 0.
000 0.000
24 Bando Chris 30 92 290 254 28 68 9 0 2 26 0 1 22 49 0.
268 0.326
25 Barfield Jesse 26 158 671 589 107 170 35 2 40 108 8 8 69 146 0.
289 0.363
26 Bargar Greg 27 22 2 2 0 0 0 0 0 0 0 0 0 2 0.
000 0.000
27 Barrett Marty 28 158 713 625 94 179 39 4 4 60 15 7 65 31 0.
286 0.354
28 Bass Kevin 27 157 640 591 83 184 33 5 20 79 22 13 38 72 0.
311 0.353
29 Bathe Bill 25 39 112 103 9 19 3 0 5 11 0 0 2 20 0.
184 0.200
30 Baylor Don 37 160 687 585 93 139 23 1 31 94 3 5 62 111 0.
238 0.311
31 Beane Billy 24 80 194 183 20 39 6 0 3 15 2 3 11 54 0.
213 0.258
32 Bedrosian Steve 28 68 6 5 0 1 0 0 0 0 0 0 1 1 0.
200 0.333
33 Bell Buddy 34 155 655 568 89 158 29 3 20 75 2 8 73 49 0.
278 0.360
34 Bell George 26 159 690 641 101 198 38 6 31 108 7 8 41 62 0.
309 0.350
35 Bell Jay 20 5 16 14 3 5 2 0 1 4 0 0 2 3 0.
357 0.438
36 Bell Terry 23 8 5 3 0 0 0 0 0 0 0 0 2 1 0.
000 0.400
37 Belliard Rafael 24 117 350 309 33 72 5 2 0 31 12 2 26 54 0.
233 0.293
38 Benedict Bruce 30 64 183 160 11 36 10 1 0 13 1 0 15 10 0.
225 0.291
39 Beniquez Juan 36 113 395 343 48 103 15 0 6 36 2 3 40 49 0.
300 0.373
40 Berenguer Juan 31 46 10 7 0 1 0 0 0 0 0 0 0 1 0.
143 0.143
41 Berenyi Bruce 31 14 12 11 0 0 0 0 0 0 0 0 0 3 0.
000 0.000
42 Bergman Dave 33 65 151 130 14 30 6 1 1 9 0 0 21 16 0.
231 0.338
43 Bernazard Tony 29 146 636 562 88 169 28 4 17 73 17 8 53 77 0.
301 0.361
44 Berra Dale 29 42 121 108 10 25 7 0 2 13 0 0 9 14 0.
231 0.291
45 Biancalana Buddy 26 100 209 190 24 46 4 4 2 8 5 1 15 50 0.
242 0.298
46 Bielecki Mike 26 31 54 48 3 3 0 0 0 1 0 0 2 26 0.
062 0.100
47 Bilardello Dann 27 79 212 191 12 37 5 0 4 17 1 0 14 32 0.
194 0.249

```
48    Bittiger   Jeff  24   3   4   3   1   1   0   0  1   1  0  0   0   1 0.
333 0.333
49        Blue   Vida  36  28  53  43   3   4   1   0  1   3  0  0   6  20 0.
093 0.204
50      Bochte  Bruce  35 125 473 407  57 104  13   1  6  43  3  2  65  68 0.
256 0.358
51       Bochy  Bruce  31  63 142 127  16  32   9   0  8  22  1  0  14  23 0.
252 0.326
52      Bockus  Randy  25   6   1   1   0   0   0   0  0   0  0  0   0   1 0.
000 0.000
53      Boever    Joe  25  11   2   2   0   1   0   0  0   0  0  0   0   0 0.
500 0.500
54       Boggs   Wade  28 149 693 580 107 207  47   2  8  71  0  4 105  44 0.
357 0.455
55       Bonds  Barry  21 113 484 413  72  92  26   3 16  48 36  7  65 102 0.
223 0.328
 [ reached 'max' / getOption("max.print") -- omitted 671 rows ]
```

Problem 11: -

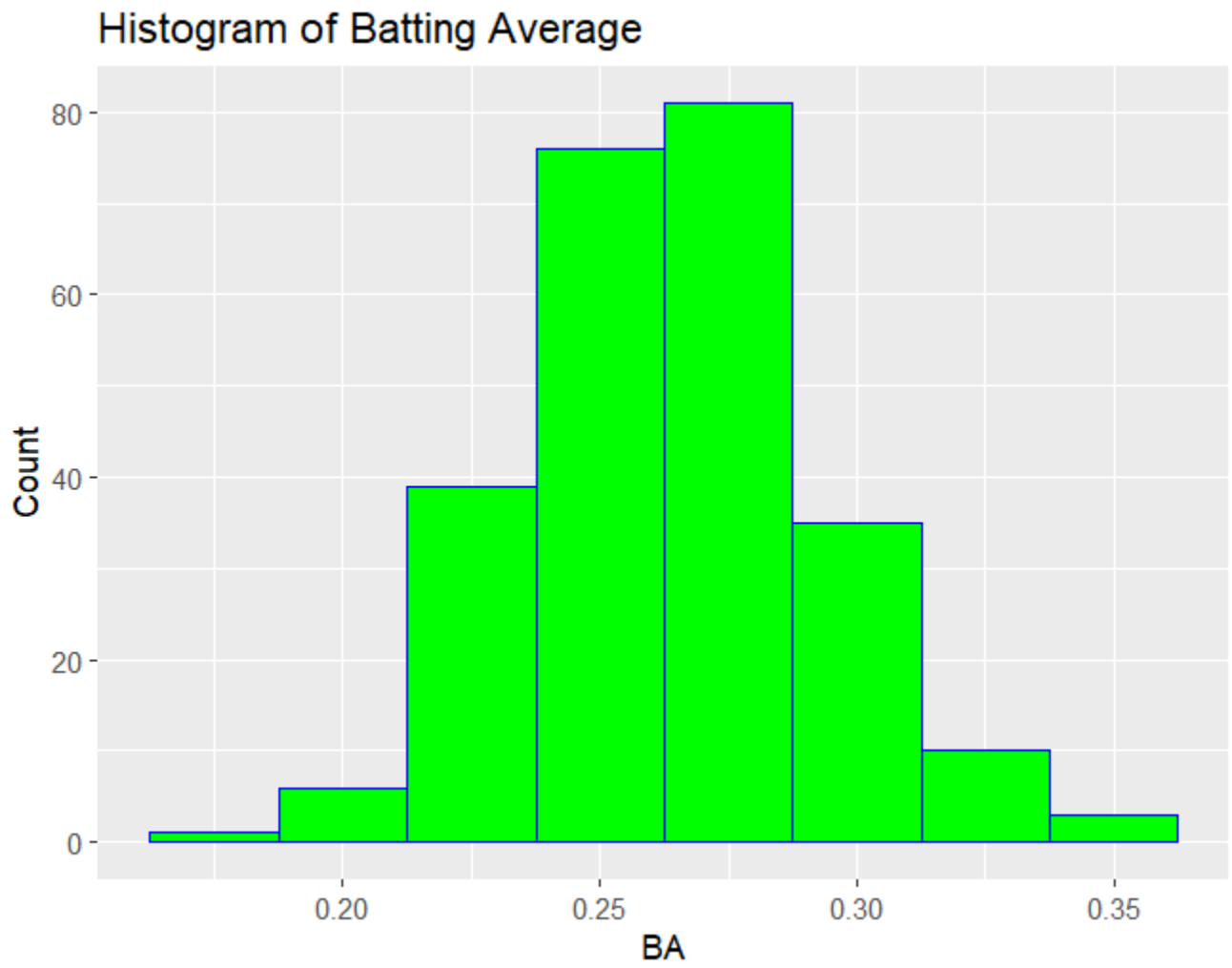
I utilized the ggplot package to create a scatter plot, providing information about the Home Runs (HR's) and RBI's of baseball data frame. I achieved this by utilizing the "baseball" data frame.

```
> #11
> ggplot(baseball, aes(x = HR, y = RBI)) +
+   geom_point() +
+   labs(x = "Number of Home Runs (HRs)", y = "Number of RBIs") +
+   ggtitle("Scatterplot of HRs vs RBIs") # Create the scatter plot
```



Problem 13: -

I generated a histogram representing batting averages with a binwidth of 0.025. The histogram is displayed in blue with a green fill.



Problem 17 & 18 : -

I have created a MVP candidates list and used rank function to sort using HR, RBI and OBP. I have calculated total rank= (RankHR + RankRBI + RankOBP). Created two new datasets “mvp_candidates” & “mvp_candidates_abbreviated”.

```
> #17
> mvp_candidates <- eligible_df %>%
+   arrange(TotalRank)%>%
+   slice(1:20)
> mvp_candidates #Top 20 ranks of the baseball
         Last   First Age   G  PA  AB   R   H X2B X3B HR RBI SB CS BB  SO
BA   OBP RankHR RankRBI RankOBP
1   Mattingly     Don  25 162 742 677 117 238  53   2 31 113  0  0 53  35 0.3
52 0.399      7       5       8
```

```
2     Schmidt    Mike  36 160 657 552  97 160  29   1 37 119  1  2 89  84 0.2
90 0.388      2       2       16
3    Barfield   Jesse  26 158 671 589 107 170  35   2 40 108  8  8 69 146 0.2
89 0.363      1       7       45
4       Evans  Dwight  34 152 640 529  86 137  33   2 26  97  3  3 97 117 0.2
59 0.374     27      17       30
5     Puckett   Kirby  26 161 723 680 119 223  37   6 31  96 20 12 34  99 0.3
28 0.360      7      18       50
6        Rice     Jim  33 157 693 618  98 200  39   2 20 110  0  1 62  78 0.3
24 0.385     52       6       18
7     O'Brien    Pete  28 156 641 551  86 160  23   3 23  90  4  4 87  66 0.2
90 0.387     36      28       17
8        Bell  George  26 159 690 641 101 198  38   6 31 108  7  8 41  62 0.3
09 0.350      7       7       74
9  McReynolds   Kevin  26 158 641 560  89 161  31   6 26  96  8  6 66  83 0.2
88 0.363     27      18       45
10     Gibson    Kirk  29 119 521 441  84 118  11   2 28  86 34  6 68 107 0.2
68 0.365     19      34       41
11     Gaetti    Gary  27 157 661 596  91 171  34   1 34 108 14 15 52 108 0.2
87 0.344      4       7       86
12      Hayes     Von  27 158 690 610 107 186  46   2 19  98 24 12 74  77 0.3
05 0.380     61      16       21
13    Downing   Brian  35 152 631 513  90 137  27   4 20  95  4  4 90  84 0.2
67 0.376     52      22       28
14 Strawberry  Darryl  24 136 562 475  76 123  27   5 27  93 28 12 72 141 0.2
59 0.356     23      26       57
15      Evans Darrell  39 151 601 507  78 122  15   0 29  85  3  2 91 105 0.2
41 0.356     14      38       57
16      Hrbek    Kent  26 149 634 550  85 147  27   1 29  91  2  2 71  81 0.2
67 0.351     14      27       71
17      Davis    Eric  24 132 487 415  97 115  15   3 27  71 80 11 68 100 0.2
77 0.379     23      71       22
18   Winfield    Dave  34 154 652 565  90 148  31   5 24 104  6  5 77 106 0.2
62 0.350     32      12       74
19    Parrish   Larry  32 129 524 464  67 128  22   1 28  94  3  1 52 114 0.2
76 0.349     19      23       77
20     Murray   Eddie  30 137 578 495  61 151  25   1 17  84  3  0 78  49 0.3
05 0.400     74      40        6
   TotalRank
1         20
2         20
3         53
4         74
5         75
6         76
7         81
8         88
9         90
10        94
11        97
12        98
13       102
14       106
15       109
16       112
17       116
18       118
19       119
20       120
> #18
> mvp_candidates_abbreviated <- mvp_candidates %>%
+   select(First,Last,RankHR,RankRBI, RankOBP,TotalRank)
> mvp_candidates_abbreviated #Creating a new data frame based on MVP_List
     First       Last RankHR RankRBI RankOBP TotalRank
```

1	Don	Mattingly	7	5	8	20
2	Mike	Schmidt	2	2	16	20
3	Jesse	Barfield	1	7	45	53
4	Dwight	Evans	27	17	30	74
5	Kirby	Puckett	7	18	50	75
6	Jim	Rice	52	6	18	76
7	Pete	O'Brien	36	28	17	81
8	George	Bell	7	7	74	88
9	Kevin	McReynolds	27	18	45	90
10	Kirk	Gibson	19	34	41	94
11	Gary	Gaetti	4	7	86	97
12	Von	Hayes	61	16	21	98
13	Brian	Downing	52	22	28	102
14	Darryl	Strawberry	23	26	57	106
15	Darrell	Evans	14	38	57	109
16	Kent	Hrbek	14	27	71	112
17	Eric	Davis	23	71	22	116
18	Dave	Winfield	32	12	74	118
19	Larry	Parrish	19	23	77	119
20	Eddie	Murray	74	40	6	120

Problem 19: -

After a thorough analysis of the provided data, I recommend Mike Schmidt as the Most Valuable Player (MVP) for the league. Mike Schmidt demonstrated remarkable consistency in HR, RBI, and OBP rankings, which are vital statistics for evaluating a player's performance. His overall contribution to his team and individual excellence in multiple categories make him a strong candidate for the MVP award.

It is important to note that while Jesse Barfield also performed exceptionally well, Mike Schmidt's consistency and broader impact on his team give him the edge for the MVP title. The decision to exclude pitchers from MVP consideration is reasonable, as the dataset only provides information about position players.

CONCLUSION

In conclusion, this project has been a valuable journey in the realm of data analysis and visualization for me. Working with two diverse datasets allowed me to apply a wide range of functions and operators, enhancing my data manipulation skills significantly. The creation of compelling plots and graphs added depth to my analysis and made my findings more accessible to a wider audience. In essence, this project has not only deepened my expertise in statistics and applied mathematics but has also reinforced the significance of data visualization as a powerful means of communication in the world of data science.

```
> testthat::test_file("project2_tests.R")
[ FAIL 0 | WARN 0 | SKIP 0 | PASS 39 ]
```

CITATIONS

1.**Overview of janitor functions. (2022, February 2).**
https://cran.r-project.org/web/packages/janitor/vignettes/janitor.html#clean-dataframenames-with-clean_names

2.**Zach. (2023). R: A Complete Guide to ties.method in rank Function. *Statology*.**
https://www.statology.org/r-rank-ties-method/