Comprehensive Analysis of PGA Tour Statistics

Team Members:
The project was collaboratively completed by the following team members:
- **Darshan Gudivada**: Orator
- **Hanoonah Sheikh**: Creator
- **Prateeksha Mehta**: Interpreter & Orator
- **Sri Sai Charan Donepudi**: Deliverer

Scenario:
The study focuses on analyzing PGA Tour statistics to identify the best predictors of a player's scoring average. The dataset includes performance metrics from the top 125 players by earnings in 2008.

Objective:
To use statistical models, primarily Multiple Linear Regression, to determine the variables influencing scoring average, such as driving distance, greens in regulation, and putts per round.

Approach:
The project employs a systematic data analysis methodology, including Exploratory Data Analysis (EDA), regression modeling, and transformation techniques for enhanced accuracy.

Analytics of PGA Tour Statistics

1. Introduction

The project aims to analyze PGA Tour statistics for the top 125 players based on total earnings in 2008 to determine which performance measures are the best predictors of a player's average score (scoring average). We will employ linear regression models to explore the relationship between scoring average and key performance variables, including **driving distance**, **driving accuracy**, **greens in regulation**, **sand saves**, **putts per round**, **scrambling**, and **bounce back**. The goal is to identify which of these factors are most strongly associated with a player's scoring average and provide insights that could help the PGA Tour and players improve performance and strategy.

2. Look at the data

Rank	Player	Money ($)	Scoring Aver	DrDist	DrAccu	GIR	Sand Saves	PPR	Scrambling	Bounce Back
1	Vijay Singh	6601094	70.27	297.8	59.45	68.45	45.11	29.47	58.92	17.31
2	Phil Mickelson	5188875	70.28	295.7	55.27	65.81	62.5	28.74	60.42	26.21
3	Sergio Garcia	4858224	70.6	294.6	59.39	67.06	57.02	29.61	57.59	21.05
4	Kenny Perry	4663794	70.21	296	61.97	67.47	50	29.25	57.57	20.37
5	Anthony Kim	4656265	70.22	300.9	58.34	65.78	50.35	28.85	59.32	21.78
6	Camilo Villegas	4422641	70.6	293.3	58.15	64.6	54.61	28.97	53.52	22.58
7	Padraig Harrington	4313551	70.7	296.3	59.37	60.67	58.06	28.04	61.02	23.49
8	Stewart Cink	3979301	70.65	296.9	55.27	66.94	51.13	29.16	55.6	23.25
9	Justin Leonard	3943542	70.41	281.4	67.72	66.61	55.17	28.85	60.07	16.8
10	Robert Allenby	3606700	70.64	291.7	65.64	70.4	46.49	30.07	55.26	19.11
11	Jim Furyk	3455714	70.56	280.4	69.37	66.78	50.68	29.43	60.32	18.75
12	Ryuji Imada	3029363	71.13	278.6	59.64	61.39	57.24	28.43	60.07	17.62
13	Mike Weir	3020135	70.68	284.8	62.46	64.62	62.09	28.63	62.27	19.05
14	Geoff Ogilvy	2880099	71.38	292.1	58.18	61.89	54.17	28.86	59.91	16.13
15	K.J. Choi	2683442	71.01	286.1	61.38	65.48	51.16	29.27	57.24	16.58
16	Ben Curtis	2615798	70.96	284.7	67.2	63.45	57.43	28.92	59.2	17.33
17	Kevin Sutherland	2581311	70.22	291	61.93	68.2	54.6	29.42	60.43	21.14
18	Trevor Immelman	2566199	71.85	291.3	62.45	63.07	42.99	29.68	52.88	17.17
19	Ernie Els	2537290	71.44	291.6	56.88	61.33	54.37	29.28	56.61	14.53
20	Carl Pettersson	2512538	70.84	286	59.87	63.54	53.13	28.8	59	16.93
21	Stuart Appleby	2484630	70.86	290.9	58.19	61.9	56.3	28.55	60.24	15.87
22	Steve Stricker	2438304	70.83	283.6	56.25	63.81	52.34	28.76	61.83	13.78
23	Chad Campbell	2404770	70.37	289.9	65.68	68.44	43.41	29.5	54.68	14.86
24	Boo Weekley	2398751	71.12	291.7	64.75	67.87	50.39	30.19	57.08	16.33
25	D.J. Trahan	2304368	70.89	291.3	65.31	66.25	42.48	29.52	55.69	23.55
26	Stephen Ames	2285707	70.67	283.8	62.72	65.04	50.76	28.99	58.72	20.61
27	Ken Duke	2238885	70.61	284.9	62.27	64.8	50.96	28.79	57.82	18.35
28	Dudley Hart	2218817	70.84	275.5	61.18	66.11	63.71	28.83	61.12	22.6
29	Hunter Mahan	2208855	70.78	289.9	66.02	69.61	45.97	30.14	53.55	17.41
30	Brian Gay	2205513	70.11	270.5	71.74	63.71	56.71	28.34	64.82	20

Variables in the dataset:

Money	Total earnings in PGA Tour events.
Scoring Average	The average number of strokes per completed round.
DrDist (Driving Distance)	DrDist is the average number of yards per measured drive. On the PGA Tour driving distance is measured on two holes per round. Care is taken to select two holes which face in opposite directions to counteract the effect of wind. Drives are measured to the point at which they come to rest regardless of whether they are in the fairway or not.
DrAccu (Driving Accuracy)	The percentage of time a tee shot comes to rest in the fair- way (regardless of club). Driving accuracy is measured on every hole, excluding par 3's
GIR (Greens in Regulation)	The percentage of time a player was able to hit the green in regulation. A green is considered hit in regulation if any portion of the ball is touching the putting surface after the GIR stroke has been taken. The GIR stroke is determined by subtracting 2 from par (1st stroke on a par 3, 2nd on a par 4, 3rd on a par 5). In other words, a green is considered hit in regulation if the player has reached the putting surface in par minus two strokes.
Sand Saves	The percentage of time a player was able to get "up and down" once in a greenside sand bunker (regardless of score). "Up and down" indicates it took the player 2 shots or less to put the ball in the hole from a greenside sand bunker.
PPR (Putts per Round)	The average number of putts per round.
Scrambling	The percentage of time a player missed the green in regulation but still made par or better.
Bounce Back	The percentage of time a player is over par on a hole and then under par on the following hole. In other words, it is the percentage of holes with a bogey or worse followed on the next hole with a birdie or better.

We first read the dataset into R:

```
#Read the csv file
getwd()
# Set the working directory
setwd("/Users/csuftitan/Downloads")

# Save the data set in the PGATour
PGATour <- read.csv("/Users/csuftitan/Downloads/PGATour.csv")
head(PGATour,10)
```

A sample of the first 10 records is shown below:

```
> head(PGATour,10)
   Rank             Player Money.... Scoring.Average DrDist DrAccu   GIR Sand.Saves   PPR Scrambling Bounce.Back
1     1        Vijay Singh   6601094           70.27  297.8  59.45 68.45      45.11 29.47      58.92       17.31
2     2      Phil Mickelson  5188875           70.28  295.7  55.27 65.81      62.50 28.74      60.42       26.21
3     3       Sergio Garcia  4858224           70.60  294.6  59.39 67.06      57.02 29.61      57.59       21.05
4     4         Kenny Perry  4663794           70.21  296.0  61.97 67.47      50.00 29.25      57.57       20.37
5     5         Anthony Kim  4656265           70.22  300.9  58.34 65.78      50.35 28.85      59.32       21.78
6     6     Camilo Villegas  4422641           70.60  293.3  58.15 64.60      54.61 28.97      53.52       22.58
7     7  Padraig Harrington  4313551           70.70  296.3  59.37 60.67      58.06 28.04      61.02       23.49
8     8        Stewart Cink  3979301           70.65  296.9  55.27 66.94      51.13 29.16      55.60       23.25
9     9      Justin Leonard  3943542           70.41  281.4  67.72 66.61      55.17 28.85      60.07       16.80
10   10      Robert Allenby  3606700           70.64  291.7  65.64 70.40      46.49 30.07      55.26       19.11
```

3. Exploratory Data Analysis

- It gives you a sense of the distributions of the individual variables in the data
- Any potential relationships exist between variables, whether there are outliers and/ or missing values.
- How to build your model.

<u>Summary Statistics:</u>

── Variable type: numeric ──────────────────────────

	skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100	hist
1	Rank	0	1	63	36.2	1	32	63	94	125	
2	Money	0	1	1791113.	1036283.	800694	1068207	1488214	2146431	6601094	
3	Scoring.Average	0	1	71.0	0.422	70.1	70.7	71.0	71.3	72.1	
4	DrDist	0	1	288.	8.64	261.	282	288.	294.	315.	
5	DrAccu	0	1	63.4	5.16	51.1	59.9	62.9	66.5	74.0	
6	GIR	0	1	64.9	2.66	58.0	63.3	64.9	66.9	71.1	
7	Sand.Saves	0	1	50.0	5.61	36.8	45.7	50.6	53.4	63.7	
8	PPR	0	1	29.2	0.533	27.9	28.8	29.2	29.5	30.9	
9	Scrambling	0	1	57.7	2.74	48.4	56.1	57.8	59.6	64.8	
10	Bounce_Back	0	1	18.6	2.41	13.8	16.9	18.7	20.2	26.2	

We will get to know the mean, standard deviation, all quantile values and if there are any missing values present for each explanatory variable.

<u>Evaluating the correlation value of each explanatory variable</u>

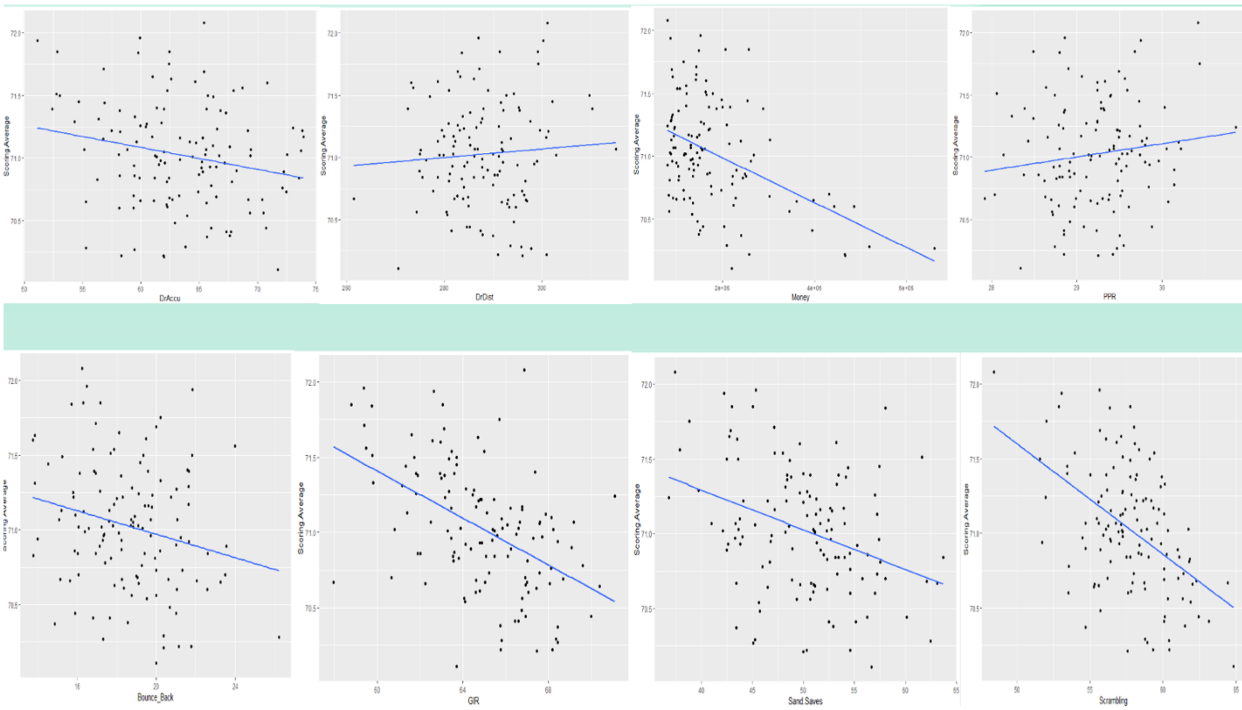
```
> PGATour %>% select(Money, DrDist, DrAccu, GIR, Sand.Saves, Bounce_Back, Scrambling, PPR) %>% cor()
```

	Money	DrDist	DrAccu	GIR	Sand.Saves	Bounce_Back	Scrambling	PPR
Money	1.0000000	0.1849729	-0.23959957	0.12176273	0.28487829	0.17105030	0.11137800	-0.11127384
DrDist	0.1849729	1.0000000	-0.61750666	0.24382623	-0.26625938	0.18924036	-0.53068387	0.38212187
DrAccu	-0.2395996	-0.6175067	1.00000000	0.27606945	-0.04174072	-0.05888865	0.27419362	0.11522450
GIR	0.1217627	0.2438262	0.27606945	1.00000000	-0.20490210	0.09449977	-0.25548831	0.73365074
Sand.Saves	0.2848783	-0.2662594	-0.04174072	-0.20490210	1.00000000	0.03891471	0.53148992	-0.48509425
Bounce_Back	0.1710503	0.1892404	-0.05888865	0.09449977	0.03891471	1.00000000	0.02940786	-0.05113548
Scrambling	0.1113780	-0.5306839	0.27419362	-0.25548831	0.53148992	0.02940786	1.00000000	-0.64964592
PPR	-0.1112738	0.3821219	0.11522450	0.73365074	-0.48509425	-0.05113548	-0.64964592	1.00000000

So, the insights from the summary statistics are the correlation value of the GIR and PPR variables are greater than 0.7 so, there are chances of multicollinearity in the model.

<u>Data Visualization using Scatter Plot:</u>

Using Scatter plot, we can observe that relationship between response variable which is Scoring Average, and each explanatory variable is not linear.



4. **Multiple Linear Regression: Initial Model**

In the initial model, we have used all the explanatory variables except Rank and Player Name as these are not considered for prediction and ran the multiple linear regression model as shown below:

```
# Fit regression model
PGATour_Model_Allvariables <- lm(Scoring.Average ~ ., data = PGATour[,c(-1,-2)])
summary(PGATour_Model_Allvariables)
get_regression_table(PGATour_Model_Allvariables)
```

Output of the initial model considering all the explanatory variables:

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.972e+01  2.033e+00  34.292  < 2e-16 ***
Money       -7.144e-08  1.745e-08  -4.095 7.86e-05 ***
DrDist      -4.464e-03  3.082e-03  -1.448    0.150    
DrAccu      -2.958e-03  4.984e-03  -0.593    0.554    
GIR         -1.641e-01  1.082e-02 -15.158  < 2e-16 ***
Sand.Saves  -3.640e-03  3.575e-03  -1.018    0.311    
PPR          5.588e-01  6.580e-02   8.493 7.78e-14 ***
Scrambling  -4.293e-02  9.596e-03  -4.473 1.81e-05 ***
Bounce_Back -6.236e-03  6.891e-03  -0.905    0.367    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1745 on 116 degrees of freedom
Multiple R-squared:  0.8401,	Adjusted R-squared:  0.829 
F-statistic: 76.16 on 8 and 116 DF,  p-value: < 2.2e-16

> get_regression_table(PGATour_Model_Allvariables)
# A tibble: 9 x 7
  term        estimate std_error statistic p_value lower_ci upper_ci
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept     69.7       2.03      34.3    0        65.7     73.7  
2 Money          0         0         -4.10   0         0        0    
3 DrDist        -0.004     0.003     -1.45   0.15     -0.011    0.002
4 DrAccu        -0.003     0.005     -0.593  0.554    -0.013    0.007
5 GIR           -0.164     0.011    -15.2    0        -0.186   -0.143
6 Sand.Saves    -0.004     0.004     -1.02   0.311    -0.011    0.003
7 PPR            0.559     0.066      8.49   0         0.429    0.689
8 Scrambling    -0.043     0.01      -4.47   0        -0.062   -0.024
9 Bounce_Back   -0.006     0.007     -0.905  0.367    -0.02     0.007
```

The variability of our model is about 82.9% so it is a good model. But, there are several variables whose p-value is greater than 0.05. The variables are DrDist, DrAccu, Sand.Saves and Bounce_Back. Hence, these variables are insignificant for our model.

Regression model considering only the significant variables:

Only the significant variables which are Money, GIR, PPR and Scrambling are used to develop the model.

```
# From the linear regression model we can see that DrDist, DrAccu, Sand.Saves and Bounce.back are the insignificant varibales so we
#Model by using significant explanatory variables
PGATour_sig <- lm(Scoring.Average ~ . - DrDist - DrAccu - Sand.Saves - Bounce_Back, data = PGATour[,c(-1,-2)])
summary(PGATour_sig)
get_regression_table(PGATour_sig)
```

Output:

```
Coefficients:
               Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.768e+01  1.785e+00  37.923  < 2e-16 ***
Money       -8.127e-08  1.607e-08  -5.058 1.54e-06 ***
GIR         -1.692e-01  1.012e-02 -16.712  < 2e-16 ***
PPR          5.742e-01  6.354e-02   9.037 3.36e-15 ***
Scrambling  -4.002e-02  8.377e-03  -4.777 5.09e-06 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.175 on 120 degrees of freedom
Multiple R-squared:  0.8336,	Adjusted R-squared:  0.8281 
F-statistic: 150.3 on 4 and 120 DF,  p-value: < 2.2e-16

> get_regression_table(PGATour_sig)
# A tibble: 5 × 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept    67.7       1.78      37.9        0   64.1     71.2  
2 Money         0         0         -5.06       0    0        0    
3 GIR          -0.169     0.01     -16.7        0   -0.189   -0.149
4 PPR           0.574     0.064      9.04       0    0.448    0.7  
5 Scrambling   -0.04      0.008     -4.78       0   -0.057   -0.023
```

The variability observed from the above model is about 82.81% which is almost the same as the model which we ran before using all the variables.

5. Converting our model into test and training model

Model 1:

We split our entire dataset into 20% as a test data and 80% as the training data to evaluate our model accuracy. Again, ran the multiple linear regression model considering all the variables.

```
# Creating Test Model
num_rows <- nrow(PGATour)
num_cols <- ncol(PGATour)
set.seed(123)
?sample
train.index <- sample(row.names(PGATour), floor(0.8*num_rows))
test.index <- setdiff(row.names(PGATour), train.index)
train.df <- PGATour[train.index, -c(1,2)]
test.df <- PGATour[test.index, -c(1,2)]
PGATour_mod1 <- lm(Scoring.Average ~ ., data = train.df)
summary(PGATour_mod1)
preds.PGATour_mod1 <- predict(PGATour_mod1, newdata = test.df)
MSE1 <- mean((preds.PGATour_mod1 - test.df$Scoring.Average)^2)
RMSE1 <- sqrt(MSE1)
print(RMSE1)
```

Output:

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  6.893e+01  2.339e+00  29.470  < 2e-16 ***
Money       -8.443e-08  2.275e-08  -3.711 0.000356 ***
DrDist      -6.977e-04  3.500e-03  -0.199 0.842442
DrAccu       1.287e-03  5.513e-03   0.233 0.815998
GIR         -1.620e-01  1.210e-02 -13.385  < 2e-16 ***
Sand.Saves  -6.127e-03  4.132e-03  -1.483 0.141601
PPR          5.309e-01  7.527e-02   7.054 3.31e-10 ***
Scrambling  -3.726e-02  1.062e-02  -3.509 0.000701 ***
Bounce_Back -9.751e-03  7.665e-03  -1.272 0.206598
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1752 on 91 degrees of freedom
Multiple R-squared:  0.8243,    Adjusted R-squared:  0.8089
F-statistic: 53.37 on 8 and 91 DF,  p-value: < 2.2e-16

> preds.PGATour_mod1 <- predict(PGATour_mod1, newdata = test.df)
> MSE1 <- mean((preds.PGATour_mod1 - test.df$Scoring.Average)^2)
> RMSE1 <- sqrt(MSE1)
> print(RMSE1)
[1] 0.1845073
```

As a result, while considering all the variables the RMSE (Root Mean Square Error) is about 0.1845 for Model 1 and the R-squared value is 0.8089.

Model 2: Considering only the significant variables for developing our regression model.

```
# Another model without the insignificant variables
PGATour_mod2 <- lm(Scoring.Average ~ . - DrDist - DrAccu - Bounce_Back - Sand.Saves , data = train.df)
summary(PGATour_mod2)
get_regression_table(PGATour_mod2)
preds.PGATour_mod2 <- predict(PGATour_mod2, newdata = test.df)
MSE2 <- mean((preds.PGATour_mod2 - test.df$Scoring.Average)^2)
RMSE2 <- sqrt(MSE2)
print(RMSE2)
```

Output:

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)  6.773e+01  2.055e+00  32.954  < 2e-16 ***
Money       -1.018e-07  2.062e-08  -4.936 3.39e-06 ***
GIR         -1.629e-01  1.142e-02 -14.261  < 2e-16 ***
PPR          5.555e-01  7.369e-02   7.539 2.79e-11 ***
Scrambling  -3.795e-02  9.140e-03  -4.152 7.20e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1752 on 95 degrees of freedom
Multiple R-squared:  0.8167,    Adjusted R-squared:  0.8089 
F-statistic: 105.8 on 4 and 95 DF,  p-value: < 2.2e-16

> get_regression_table(PGATour_mod2)
# A tibble: 5 × 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept    67.7       2.06      33.0        0   63.6     71.8 
2 Money         0         0         -4.94       0    0        0   
3 GIR          -0.163     0.011    -14.3        0   -0.186   -0.14 
4 PPR           0.556     0.074      7.54       0    0.409    0.702
5 Scrambling   -0.038     0.009     -4.15       0   -0.056   -0.02 
> preds.PGATour_mod2 <- predict(PGATour_mod2, newdata = test.df)
> MSE2 <- mean((preds.PGATour_mod2 - test.df$Scoring.Average)^2)
> RMSE2 <- sqrt(MSE2)
> print(RMSE2)
[1] 0.1815581
```

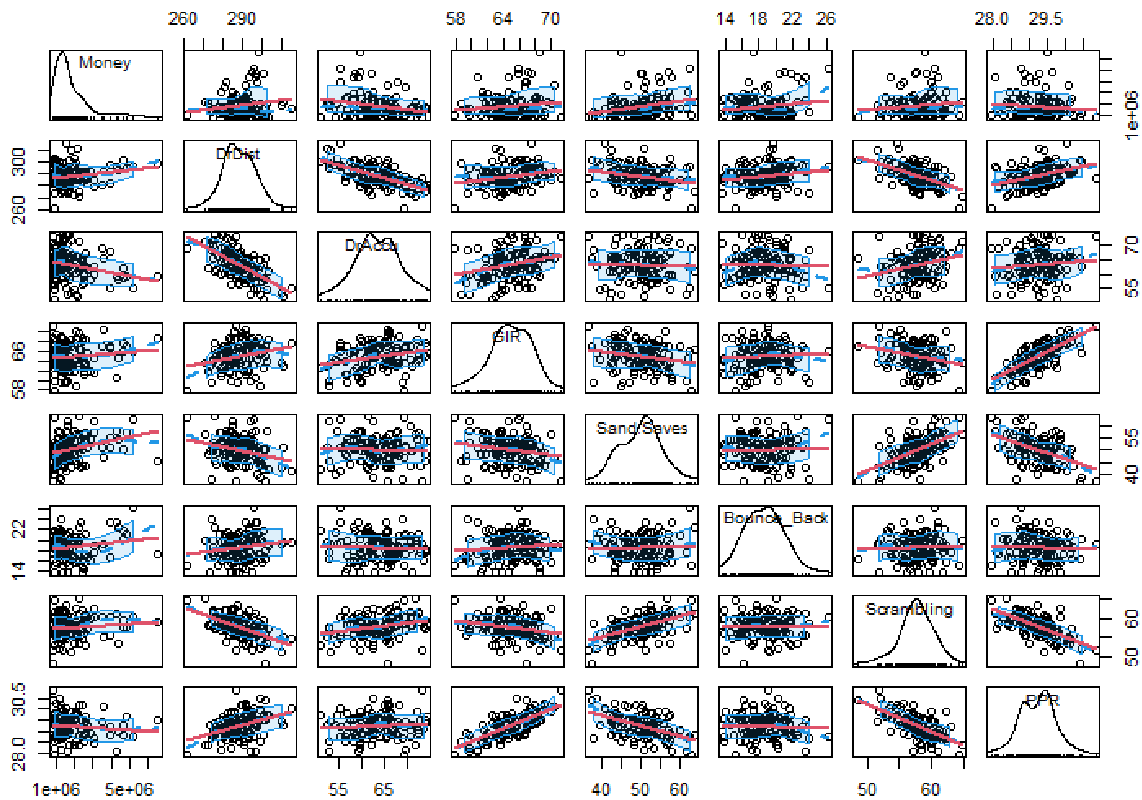
So, here if we could observe the RMSE value improved a little bit compared to the above model. The RMSE value is 0.1815 and R-squared value is 0.8089 which is good. From the above two models the prediction accuracy is not that good we need to come up with some enhancement in our model to further improve our model accuracy.

Enhancement in the model:

To decide which transformation to implement for our model. Let's again perform exploratory data analysis on our model.

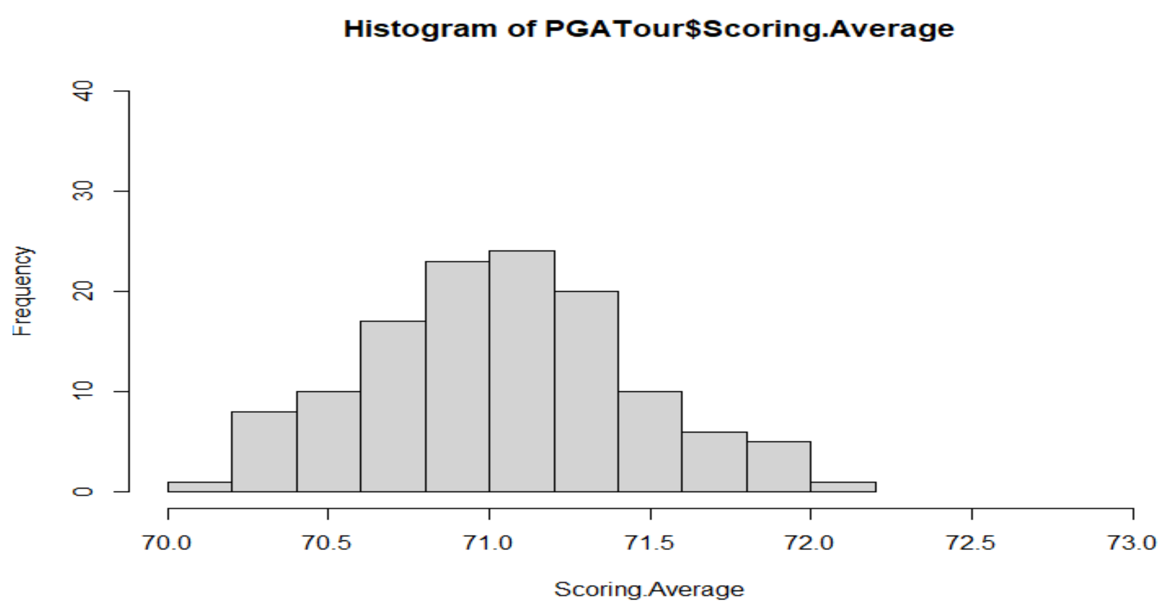
Data visualization: Scatter plot matrix for each explanatory variable

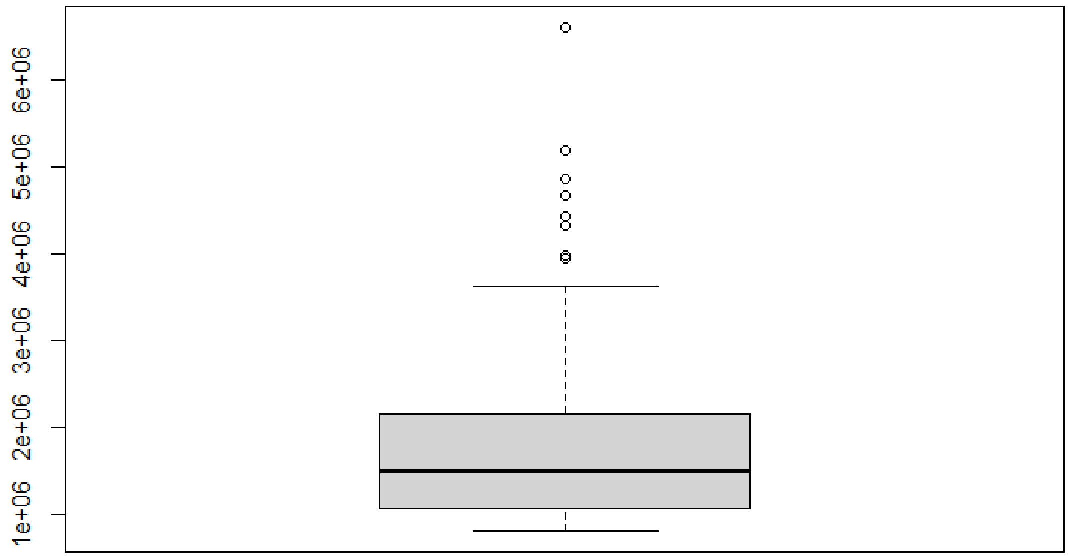
```
options(warn=-1)
scatterplotMatrix(~ Money + DrDist + DrAccu + GIR + Sand.Saves +  Bounce_Back + Scrambling + PPR , regLine = list(col = 2),
                  col = 1, smooth = list(col.smooth = 4, col.spread = 4), data = PGATour)
```

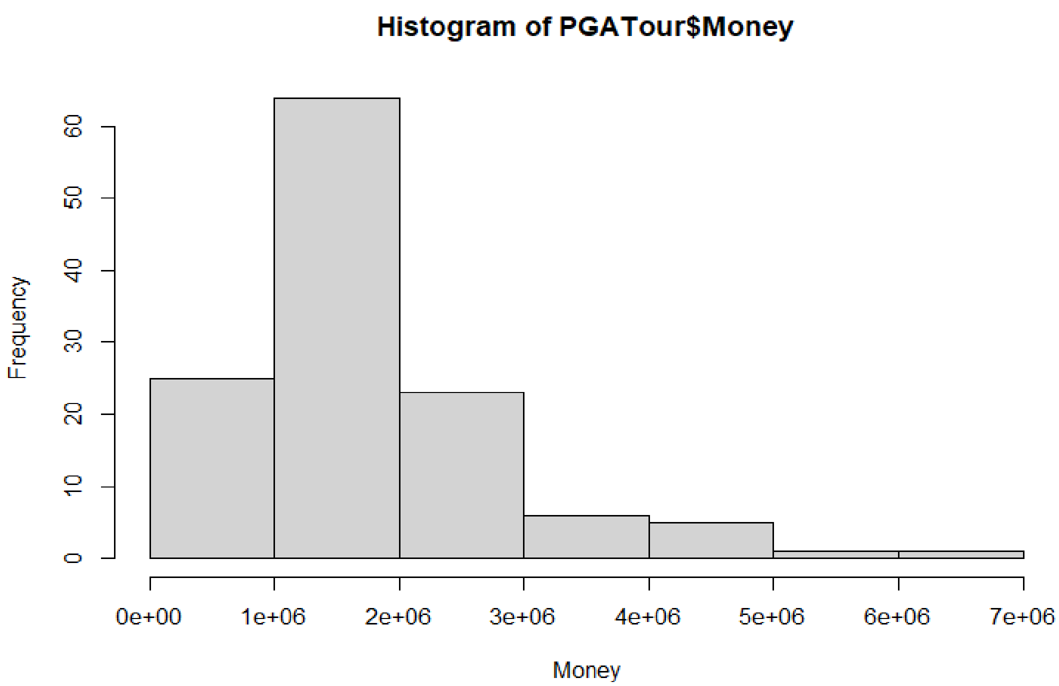


So, from the above matrix it is visible that all the variables have uniform distribution except money.

Used boxplot and histogram of the Money variable to visualize the variable for the skewedness and outliers. Also, used histogram to visualize the Scoring Average variable.







So, from the above two visualization Money variable has outliers, and it is right skewed.

We have tried various model transformation to improve our model accuracy. Created new variables as shown below and ran the model using polynomial transformation.

Model 3: Used polynomial transformation on the Money, GIR, PPR and Scrambling variables considering all the variables in the model.

```r
# Consider this model as our final model
# Developed model using log transformation on Scoring.Average & Money variables
PGATour$GIR_new = poly(PGATour$GIR, 2)
PGATour$PPR_new = poly(PGATour$PPR, 2)
PGATour$Money_new = log(PGATour$Money)
PGATour$Money_new2 = poly(PGATour$Money, 2)
PGATour$Scrambling_new = poly(PGATour$Scrambling, 2)
PGATour$Scoring_Avg_new = log(PGATour$Scoring.Average)

# Considering test and training data with the new variables
num_rows <- nrow(PGATour)
num_cols <- ncol(PGATour)
set.seed(123)
train.index <- sample(row.names(PGATour), floor(0.8*num_rows))
test.index <- setdiff(row.names(PGATour), train.index)
train.df <- PGATour[train.index, -c(1,2)]
test.df <- PGATour[test.index, -c(1,2)]


# Considered polynomial transformation of the GIR, Money, PPR and Scrambling variables with all explanatory variables
PGA_Tour_mod3 <- lm(Scoring.Average ~ Money_new2 + DrAccu + DrDist + GIR_new + Scrambling_new + Bounce_Back + Sand.Saves + PPR_new - Money_new, data = train.df
summary(PGA_Tour_mod3)
get_regression_table(PGA_Tour_mod3)
preds.PGA_Tour_mod3 <- predict(PGA_Tour_mod3, newdata = test.df)
MSE3 <- mean((preds.PGA_Tour_mod3 - test.df$Scoring.Average)^2)
RMSE3 <- sqrt(MSE3)
print(RMSE3)
# 0.185539  ...R-square : 0.807
```

Model 4: Used polynomial transformation on the significant variables without considering insignificant variables.

```r
# Considered polynomial transformation of the GIR, Money and Scrambling variables with only significant variables
PGA_Tour_mod4 <- lm(Scoring.Average ~ Money_new2 + GIR + Scrambling + PPR - GIR_new - Scrambling_new - PPR_new , data = train.df)
summary(PGA_Tour_mod4)
get_regression_table(PGA_Tour_mod4)
preds.PGA_Tour_mod4 <- predict(PGA_Tour_mod4, newdata = test.df)
MSE4 <- mean((preds.PGA_Tour_mod4 - test.df$Scoring.Average)^2)
RMSE4 <- sqrt(MSE4)
print(RMSE4)

# 0.1858984 ...R-square : 0.8067
```

But from the above two models which is Model 3 and Model 4 the R- squared value for each model is about 0.807 but if we consider the RMSE value it is about 0.1855 for Model 3 and 0.1858 for Model 4. It concludes that polynomial transformation is not a correct model to use for prediction.

Model 5: Log transformation on Money variable without considering insignificant variables.

```
# In this model used log transformation on Money variable without considering the insignificant variables to improve model's RMSE value
PGATour_mod5 <- lm(Scoring.Average ~ + Money_new + GIR + Scrambling + PPR - DrDist - DrAccu - Bounce_Back - Sand.Saves - Money_new2 - GIR_new - Scrambling_new - PPR_new , data = train.df
summary(PGATour_mod5)
preds.PGATour_mod5 <- predict(PGATour_mod5, newdata = test.df)
MSE5 <- mean((preds.PGATour_mod5 - test.df$Scoring.Average)^2)
RMSE5 <- sqrt(MSE5)
print(RMSE5)
```

Output:

```
Call:
lm(formula = Scoring.Average ~ +Money_new + GIR + Scrambling +
    PPR - DrDist - DrAccu - Bounce_Back - Sand.Saves - Money_new2 -
    GIR_new - Scrambling_new - PPR_new, data = train.df)

Residuals:
     Min       1Q   Median       3Q      Max
-0.62274 -0.10166  0.00652  0.12094  0.38137

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept) 69.89262    2.29002  30.521  < 2e-16 ***
Money_new   -0.19700    0.04228  -4.659 1.03e-05 ***
GIR         -0.16657    0.01137 -14.643  < 2e-16 ***
Scrambling  -0.03585    0.00919  -3.900 0.000179 ***
PPR          0.57551    0.07349   7.831 6.82e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.1772 on 95 degrees of freedom
Multiple R-squared:  0.8125,    Adjusted R-squared:  0.8046
F-statistic: 102.9 on 4 and 95 DF,  p-value: < 2.2e-16

> preds.PGATour_mod5 <- predict(PGATour_mod5, newdata = test.df)
> MSE5 <- mean((preds.PGATour_mod5 - test.df$Scoring.Average)^2)
> RMSE5 <- sqrt(MSE5)
> print(RMSE5)
[1] 0.1750503
```

From the above model the RMSE value improved which is 0.1750 so it shows a good improvement in the prediction accuracy and the R-squared value is also about 0.8046 which is like the above models.

Model 6: Log transformation on Scoring Average with all the variables

```
#In this model we have taken log of Scoring Average variable and considered all the variables
PGA_Tour_mod6 <- lm(Scoring_Avg_new ~ + GIR + PPR + Money + Scrambling + Bounce_Back + DrDist + DrAccu + Sand.Saves - Money_new -
                    Money_new2 - Scrambling_new - GIR_new - PPR_new, data = train.df)
summary(PGA_Tour_mod6)
get_regression_table(PGA_Tour_mod6)
preds.PGA_Tour_mod6 <- predict(PGA_Tour_mod6, newdata = test.df)
MSE6 <- mean((preds.PGA_Tour_mod6 - test.df$Scoring_Avg_new)^2)
RMSE6 <- sqrt(MSE6)
print(RMSE6)

# 0.002599 and R-squared : 0.8087 ...
```

Output:

```
Residual standard error: 0.002467 on 91 degrees of freedom
Multiple R-squared:  0.8242,	Adjusted R-squared:  0.8087
F-statistic: 53.31 on 8 and 91 DF,  p-value: < 2.2e-16

> get_regression_table(PGA_Tour_mod6)
# A tibble: 9 x 7
  term        estimate std_error statistic p_value lower_ci upper_ci
  <chr>          <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept      4.23      0.033   129.      0        4.17     4.30
2 GIR           -0.002     0       -13.4     0       -0.003   -0.002
3 PPR            0.007     0.001     7.04    0        0.005    0.01
4 Money          0         0        -3.74    0        0        0
5 Scrambling    -0.001     0        -3.51    0.001   -0.001    0
6 Bounce_Back    0         0        -1.27    0.207    0        0
7 DrDist         0         0        -0.205   0.838    0        0
8 DrAccu         0         0         0.23    0.818    0        0
9 Sand.Saves     0         0        -1.47    0.144    0        0
> preds.PGA_Tour_mod6 <- predict(PGA_Tour_mod6, newdata = test.df)
> MSE6 <- mean((preds.PGA_Tour_mod6 - test.df$Scoring_Avg_new)^2)
> RMSE6 <- sqrt(MSE6)
> print(RMSE6)
[1] 0.002599486
```

From the Model 6 the RMSE value further improved which is 0.002599 so it shows a decrease in RMSE value and increase in the prediction accuracy of our model and the R-squared value is also about 0.8087. Hence, for improving the model accuracy we developed the below test models and came up with the final test model which is Model 8.

Model	R-squared	RMSE
Mod 1 - All variables	0.8089	0.1845
Mod 2 - Significant variables	0.8089	0.1815
Mod 3 - Polynomial transformation on GIR, Money, PPR and Scrambling with all variables	0.807	0.1855
Mod 4 - Polynomial transformation on GIR, Money, PPR and Scrambling	0.8067	0.18589
Mod 5 - Log transformation on Money with all significant variables	0.8046	0.17505
Mod 6 - Log transformation on Scoring Avg with all variables	0.8087	0.002599
Mod 7 - Log transformation on Scoring Avg and Money with all variables	0.8061	0.002542
Mod 8 - Log transformation on Scoring Avg and Money with only significant variables	0.8044	0.002467

Final Test Model: Log transformation used on Scoring Average and Money variable and used only significant variables in the model.

```
# Log transformation on Money and Scoring Average variable considering only significant variables
PGA_Tour_mod8 <- lm(Scoring_Avg_new ~ Money_new + GIR + Scrambling  + PPR, data = train.df)
summary(PGA_Tour_mod8)
get_regression_table(PGA_Tour_mod8)
preds.PGA_Tour_mod8 <- predict(PGA_Tour_mod8, newdata = test.df)
MSE8 <- mean((preds.PGA_Tour_mod8 - test.df$Scoring_Avg_new)^2)
RMSE8 <- sqrt(MSE8)
print(RMSE8)
# 0.002467448...R-square : 0.8044
```

Output:

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.2473644  0.0322454 131.720  < 2e-16 ***
Money_new   -0.0027851  0.0005954  -4.678 9.59e-06 ***
GIR         -0.0023424  0.0001602 -14.625  < 2e-16 ***
Scrambling  -0.0005042  0.0001294  -3.897 0.000182 ***
PPR          0.0080915  0.0010348   7.819 7.22e-12 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.002495 on 95 degrees of freedom
Multiple R-squared:  0.8123,	Adjusted R-squared:  0.8044
F-statistic: 102.8 on 4 and 95 DF,  p-value: < 2.2e-16

> get_regression_table(PGA_Tour_mod8)
# A tibble: 5 × 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept     4.25      0.032    132.         0    4.18     4.31
2 Money_new    -0.003     0.001     -4.68       0   -0.004   -0.002
3 GIR          -0.002     0        -14.6        0   -0.003   -0.002
4 Scrambling   -0.001     0         -3.90       0   -0.001    0
5 PPR           0.008     0.001      7.82       0    0.006    0.01
> preds.PGA_Tour_mod8 <- predict(PGA_Tour_mod8, newdata = test.df)
> MSE8 <- mean((preds.PGA_Tour_mod8 - test.df$Scoring_Avg_new)^2)
> RMSE8 <- sqrt(MSE8)
> print(RMSE8)
[1] 0.002467448
> # 0.002467448...R-square : 0.8044
```

This model is the best test model for our project as it has the least RMSE value compared to all the other models. The RMSE value is 0.002467 and R-squared value is 0.8044. Now, this test model we will use for developing our final model.

Enhanced Model: Combining the test and training data and developing the final model.

```
# Final model
PGATour_Final.df = rbind(train.df, test.df)
PGATour_Final_Model <- lm(Scoring_Avg_new ~ Money_new + GIR + Scrambling  + PPR , data = PGATour_Final.df)
summary(PGATour_Final_Model)
get_regression_table(PGATour_Final_Model)
preds.PGATour_Final_Model <- predict(PGATour_Final_Model, newdata = PGATour_Final.df)
MSE_final <- mean((preds.PGATour_Final_Model - test.df$Scoring_Avg_new)^2)
RMSE_final <- sqrt(MSE_final)
print(RMSE_final)
```

Output:

```
Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  4.2458702  0.0272506 155.808  < 2e-16 ***
Money_new   -0.0024321  0.0004923  -4.940 2.56e-06 ***
GIR         -0.0024191  0.0001413 -17.124  < 2e-16 ***
Scrambling  -0.0005502  0.0001184  -4.647 8.70e-06 ***
PPR          0.0082331  0.0008926   9.224 1.21e-15 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.002477 on 120 degrees of freedom
Multiple R-squared:  0.8317,	Adjusted R-squared:  0.8261 
F-statistic: 148.3 on 4 and 120 DF,  p-value: < 2.2e-16

> get_regression_table(PGATour_Final_Model)
# A tibble: 5 x 7
  term       estimate std_error statistic p_value lower_ci upper_ci
  <chr>         <dbl>     <dbl>     <dbl>   <dbl>    <dbl>    <dbl>
1 intercept     4.25      0.027    156.         0    4.19     4.3  
2 Money_new    -0.002     0         -4.94       0   -0.003   -0.001
3 GIR          -0.002     0        -17.1        0   -0.003   -0.002
4 Scrambling   -0.001     0         -4.65       0   -0.001    0    
5 PPR           0.008     0.001      9.22       0    0.006    0.01 
> preds.PGATour_Final_Model <- predict(PGATour_Final_Model, newdata = PGATour_Final.df)
> MSE_final <- mean((preds.PGATour_Final_Model - test.df$Scoring_Avg_new)^2)
> RMSE_final <- sqrt(MSE_final)
> print(RMSE_final)
[1] 0.007352203
```

Estimated Regression Equation for Scoring Average:

log(Scoring.Average) = 4.25 - 0.002 log(Money) - 0.002 * GIR - 0.001 * Scrambling + 0.008 * PPR