Statistical Analysis & Modeling

A2 – MULTIPLE REGRESSION ANALYSIS FOR VARIABLES IN UTTAR PRADESH STATE CONSUMPTION & RELATION OF PLAYER PERFORMANCE AND SALARYIN IPL

R - PROGRAMMING

MUGUNTHVENKAT KARTHIKEYAN
V01068767

SUBMISSION DATE: 06 June 2023

TABLE OF CONTENTS

INTRODUCTION

This analysis explores the relationship between variables related to Uttar Pradesh state consumption. Multiple regression analysis is employed to examine how various factors impact consumption patterns in the state. By analyzing these relationships, we can gain insights into the key determinants of consumption behavior in Uttar Pradesh and their potential implications for the economy and policy-making. The multiple regression analysis for variables in Uttar Pradesh state consumption aims to identify the significant factors that influence consumption patterns. By examining a range of variables such as income levels, population demographics, and economic indicators, we can assess their individual and collective impact on consumption. This analysis provides a comprehensive understanding of the drivers behind consumer spending in Uttar Pradesh and offers valuable insights for businesses, policymakers, and researchers. The results of this analysis can inform strategies for targeted marketing, resource allocation, and policy interventions to optimize economic growth and improve the standard of living in the state.

The correlation of player performance and salary in the Indian Premier League (IPL) examines the relationship between the on-field performance of players and their corresponding salaries. This analysis aims to determine whether there is a significant correlation between a player's performance metrics, such as batting average, bowling economy, or wicket count, and their salary. By exploring this correlation, we gain insights into the factors that drive salary decisions in the IPL and shed light on how player performance is valued by team owners and management. Understanding the correlation between performance and salary can help teams make informed decisions regarding player contracts, optimize team composition, and allocate resources effectively. Additionally, this analysis provides valuable insights for players, agents, and stakeholders in the cricket industry, aiding in negotiations and contract discussions.

OBJECTIVES

- ❖ To perform Multiple regression analysis and carry out the regression diagnostics and explain the findings and explain the significant differences
- ❖ Using the IPL data establish the relationship between the performance of the player and payment he receives and discuss the findings.

BUSINESS SIGNIFICANCE

1. Multiple Regression Analysis for Variables in Uttar Pradesh State Consumption:

- Understanding the relationship between various variables in Uttar Pradesh state consumption can provide insights into the factors driving consumption patterns in the region.
- This analysis can help policymakers and businesses identify key factors influencing consumer behavior and make informed decisions regarding resource allocation and policy development.
- By analyzing multiple variables, such as income, population, education levels, and urbanization, we can gain a comprehensive understanding of the complex dynamics affecting consumption patterns in Uttar Pradesh.
- The findings from this analysis can guide businesses in developing targeted marketing strategies and product offerings tailored to the specific needs and preferences of consumers in Uttar Pradesh.
- Policymakers can utilize the results to formulate evidence-based policies aimed at promoting economic growth, addressing social inequalities, and improving the overall well-being of the population in Uttar Pradesh.

2. Correlation of Player Performance and Salary in IPL:

- Understanding the correlation between player performance and salary in the IPL is crucial for team management and player recruitment strategies.
- By identifying the relationship between player performance metrics, such as batting average, bowling economy, or wicket count, and their corresponding salaries, teams can make informed decisions regarding player contracts and salary negotiations.
- This analysis provides valuable insights into the factors considered by team owners and management when determining player salaries, helping teams allocate their resources effectively and optimize team composition.
- Players and agents can utilize the findings to negotiate contracts, establish fair salary expectations, and assess their market value based on their performance metrics.
- Sponsors and advertisers can leverage the correlation between player performance and salary to make informed decisions regarding endorsements and brand partnerships, aligning their investments with high-performing players who command higher salaries.
- The correlation analysis also provides fans and cricket enthusiasts with an understanding of how player performance is valued in the IPL.

RESULTS & INTERPRETATION

A. Perform Multiple regression analysis and carry out the regression diagnostics and explain your findings. Correct them and revisit your results and explain the significant differences:

```
group_by(Season, batter) %>%
+   summarize(Total_Balls = n(),
+ group_by(Season, batter) %>%
+   summarize(Total_Balls = n(),
+             Total_Runs = sum(total_run))
+ top_run_getters=summary_data %>%
```

From the above results, we group the batters by season. We also summarize the total ball, runs and the player who has secured the highest runs.

```
x1<-up_df$ricetotal_q
> x2<-up_df$wheattotal_q
> x3<-up_df$potato_q
> x4<-up_df$milk_q
> y<-up_df$tot_con
```

From the total dataset, the above variables are taken for the creation of regression model. This is because the major food consumed in the state of UP is rice, wheat, milk and potato which are most prevalent there.

```
x1          x2         x3         x4          y
1    4.1666667   5.833333  2.8333333 10.4000000  23.233333
2    5.0000000   5.000000  2.0000000  3.9000000  15.900000
3    6.0000000   6.300000  3.0000000  3.1200000  18.420000
4    5.0000000   5.000000  2.5000000 15.6000000  28.100000
5    6.2500000   6.500000  2.5000000  1.3000000  16.550000
6    5.0000000   6.000000  2.2000000  3.1200000  16.320000
7    2.5000000   4.500000  2.5000000  1.9500000  11.450000
8    6.4000000   6.000000  3.0000000  6.2400000  21.640000
9    4.2500000   8.500000  1.2500000 15.6000000  29.600000
10   4.1666667   3.333333  1.6666667  5.2000000  14.366667
11   5.0000000   6.250000  3.7500000  6.2400000  21.240000
12   4.1666667   8.333333  6.6666667  6.9333333  26.100000
13   6.4285714   8.571429  2.1428571  4.4571429  21.600000
14  11.2500000   3.750000  1.8750000  0.9750000  17.850000
15   0.5000000   4.333333  1.2500000  1.3000000   7.383333
16  10.0000000   6.666667  5.0000000  2.7733333  24.440000
17  12.0000000   7.000000  1.0000000 31.2000000  51.200000
18   5.2000000   5.600000  0.4000000  0.0000000  11.200000
19   5.2500000   5.000000  2.5000000  3.9000000  16.650000
20   3.0000000   4.000000  2.0000000  0.0000000   9.000000
21   6.2500000   3.250000  0.5000000  0.0000000  10.000000
22   5.5555556   3.333333  2.7777778  6.9333333  18.600000
23   6.7777778   3.666667  4.0000000  9.2444444  23.688889
24   4.0000000   6.000000  2.0000000  6.2400000  18.240000
```

25	4.0000000	6.000000	1.6000000	3.1200000	14.720000
26	1.6666667	1.666667	0.6666667	0.8666667	4.866667
27	4.4000000	6.600000	4.0000000	6.2400000	21.240000
28	4.2857143	7.142857	2.2857143	4.4571429	18.171429
29	6.4285714	3.571429	2.1428571	0.1485714	12.291429
30	5.0000000	5.000000	2.5714286	4.4571429	17.028571
31	2.5000000	4.750000	1.5000000	2.0800000	10.830000
32	10.0000000	10.000000	5.0000000	0.0000000	25.000000
33	2.5000000	7.500000	1.0000000	0.0000000	11.000000
34	4.0000000	5.000000	1.2000000	3.1200000	13.320000
35	6.0000000	9.000000	3.2000000	0.0000000	18.200000
36	6.6666667	7.500000	1.6666667	0.3466667	16.180000
37	4.4444444	5.000000	0.7777778	0.8666667	11.088889
38	4.3750000	5.000000	0.7500000	7.8000000	17.925000
39	5.0000000	5.625000	1.2500000	0.9750000	12.850000
40	10.0000000	7.500000	4.0000000	5.4600000	26.960000
41	5.0000000	10.000000	3.0000000	7.8000000	25.800000
42	5.0000000	5.000000	2.6666667	1.0400000	13.706667
43	6.0000000	6.500000	3.7500000	1.3000000	17.550000
44	5.5555556	5.777778	2.2222222	0.3466667	13.902222
45	5.0000000	5.000000	2.5000000	0.7800000	13.280000
46	5.7142857	5.714286	2.1428571	0.7428571	14.314286
47	4.2857143	7.142857	2.8571429	0.7428571	15.028571
48	5.0000000	10.000000	5.0000000	7.8000000	27.800000
49	25.0000000	7.000000	2.0000000	20.8000000	54.800000
50	6.0000000	9.333333	2.5000000	5.2000000	23.033333
51	6.0000000	5.000000	0.6000000	0.0000000	11.600000
52	5.3333333	6.666667	1.5000000	5.2000000	18.700000
53	5.0000000	4.285714	0.5000000	4.4571429	14.242857
54	5.6250000	4.500000	0.4375000	1.9500000	12.512500
55	4.0000000	5.200000	1.6000000	9.3600000	20.160000
56	3.8888889	3.888889	2.8888889	13.8666667	24.533333
57	1.7500000	4.500000	2.0000000	19.5000000	27.750000
58	2.5000000	4.166667	2.6666667	10.4000000	19.733333
59	4.0000000	8.000000	1.6000000	1.5600000	15.160000
60	0.0000000	2.500000	0.2000000	7.8000000	10.500000
61	40.0000000	100.000000	16.6666667	52.0000000	208.666667
62	1.8750000	5.625000	3.2500000	7.8000000	18.550000
63	4.0000000	6.000000	2.4000000	4.6800000	17.080000
64	7.5000000	5.000000	3.7500000	7.8000000	24.050000
65	8.3333333	3.333333	3.3333333	10.4000000	25.400000
66	2.0000000	13.200000	2.0000000	3.1200000	20.320000
67	6.0000000	6.400000	2.0000000	0.0000000	14.400000
68	3.5714286	4.285714	1.1428571	2.2285714	11.228571
69	5.0000000	5.000000	3.7500000	0.0000000	13.750000
70	6.6666667	10.000000	6.0000000	0.0000000	22.666667
71	5.0000000	5.000000	3.3333333	1.3000000	14.633333
72	5.0000000	3.333333	2.0000000	20.8000000	31.133333
73	2.5000000	3.750000	1.2500000	11.7000000	19.200000
74	4.0000000	7.000000	1.0000000	2.6000000	14.600000
75	4.1666667	7.500000	1.0000000	15.6000000	28.266667
76	2.6666667	4.000000	1.3333333	15.6000000	23.600000
77	1.3333333	1.666667	1.3333333	9.7066667	14.040000
78	12.0000000	18.000000	8.0000000	15.6000000	53.600000
79	1.2500000	9.666667	2.5000000	10.7466667	24.163333
80	1.5714286	8.857143	2.5714286	6.6857143	19.685714
81	2.0000000	11.000000	2.4000000	3.5360000	18.936000
82	1.2500000	9.375000	1.2500000	2.2100000	14.085000
83	2.1250000	12.000000	3.0000000	16.9000000	34.025000
84	1.2142857	11.428571	1.0714286	2.9714286	16.685714
85	1.8333333	9.666667	3.0000000	5.8933333	20.393333
86	1.6000000	11.200000	3.4000000	3.3280000	19.528000
87	1.0000000	10.000000	2.0000000	23.4000000	36.400000
88	0.7142857	11.428571	1.4285714	8.9142857	22.485714
89	0.6000000	8.000000	2.2000000	6.2400000	17.040000
90	0.5000000	10.000000	3.7500000	7.8000000	22.050000
91	0.2857143	7.142857	2.8571429	8.9142857	19.200000
92	0.2500000	7.187500	2.5000000	7.8000000	17.737500
93	0.1818182	11.545455	0.9090909	2.8363636	15.472727
94	0.2857143	8.214286	2.5714286	4.4571429	15.528571
95	0.5000000	7.000000	2.4000000	12.4800000	22.380000

96	3.0000000	6.000000	2.5000000	15.6000000	27.100000
97	0.4285714	5.428571	2.2857143	6.6857143	14.828571
98	0.8333333	4.166667	1.6666667	7.8000000	14.466667
99	1.0000000	8.000000	6.0000000	3.1200000	18.120000
100	1.0000000	5.000000	3.3333333	5.2000000	14.533333
101	5.0000000	3.333333	2.3333333	2.6000000	13.266667
102	0.5000000	5.000000	1.5000000	5.2000000	12.200000
103	2.5000000	7.500000	2.0000000	15.6000000	27.600000
104	2.5000000	6.250000	2.5000000	15.6000000	26.850000
105	1.5000000	9.500000	3.0000000	7.8000000	21.800000
106	1.7500000	7.500000	1.5000000	3.9000000	14.650000
107	2.5000000	10.500000	4.1250000	15.6000000	32.725000
108	1.2142857	8.285714	2.1428571	6.6857143	18.328571
109	2.6000000	8.000000	2.0000000	6.2400000	18.840000
110	1.0000000	8.000000	2.0000000	3.1200000	14.120000
111	1.2500000	6.250000	0.5000000	15.6000000	23.600000
112	1.2500000	7.500000	1.0000000	15.6000000	25.350000
113	0.6250000	8.750000	4.0000000	7.8000000	21.175000
114	1.2500000	5.000000	0.7500000	11.7000000	18.700000
115	0.6666667	12.222222	2.6666667	6.9333333	22.488889
116	0.6250000	11.250000	4.2500000	7.8000000	23.925000
117	0.6666667	8.333333	4.0000000	10.4000000	23.400000
118	0.0000000	9.090909	0.9090909	1.4181818	11.418182
119	0.6666667	5.000000	3.3333333	10.4000000	19.400000
120	0.4000000	9.000000	3.0000000	6.2400000	18.640000
121	0.5000000	6.250000	3.5000000	7.8000000	18.050000
122	0.0000000	7.500000	3.3333333	5.2000000	16.033333
123	0.4500000	6.500000	1.8000000	6.2400000	14.990000
124	0.0000000	8.000000	3.0000000	6.2400000	17.240000
125	0.5000000	6.250000	3.0000000	7.8000000	17.550000
126	0.0000000	5.000000	3.7500000	5.2000000	13.950000
127	1.0000000	6.666667	5.0000000	10.4000000	23.066667
128	1.2500000	7.500000	5.0000000	15.6000000	29.350000
129	0.8750000	5.000000	2.5000000	11.7000000	20.075000
130	0.6250000	5.625000	3.1250000	5.8500000	15.225000
131	0.5555556	6.666667	3.3333333	6.9333333	17.488889
132	0.7500000	5.500000	2.2500000	11.7000000	20.200000
133	0.6250000	6.000000	2.0000000	7.8000000	16.425000
134	0.6000000	6.000000	2.8000000	6.2400000	15.640000
135	0.0000000	0.000000	0.0000000	0.0000000	0.000000
136	0.6250000	10.000000	6.2500000	23.4000000	40.275000
137	1.0000000	10.000000	6.6666667	15.6000000	33.266667
138	0.6666667	11.666667	10.0000000	20.8000000	43.133333
139	0.6666667	10.000000	8.3333333	5.2000000	24.200000
140	0.7500000	7.500000	5.0000000	15.6000000	28.850000
141	0.3888889	8.333333	6.6666667	3.4666667	18.855556
142	0.4285714	7.142857	5.0000000	4.4571429	17.028571
143	0.0000000	0.000000	0.0000000	31.2000000	31.200000
144	1.0000000	9.285714	1.7142857	6.6857143	18.685714
145	2.0000000	8.750000	5.0000000	7.8000000	23.550000
146	2.0000000	8.000000	4.0000000	15.6000000	29.600000
147	1.1428571	5.000000	2.8571429	4.4571429	13.457143
148	1.2500000	5.250000	3.5000000	7.8000000	17.800000
149	0.8571429	5.000000	4.2857143	2.2285714	12.371429
150	0.6666667	5.000000	2.5000000	2.6000000	10.766667
151	1.0000000	5.000000	3.0000000	15.6000000	24.600000
152	0.5000000	5.000000	3.0000000	13.0000000	21.500000
153	1.0000000	4.000000	3.0000000	7.8000000	15.800000
154	0.6666667	6.666667	2.3333333	10.4000000	20.066667
155	2.0000000	8.000000	2.4000000	9.3600000	21.760000
156	1.0000000	5.000000	3.0000000	7.8000000	16.800000
157	0.7500000	3.500000	1.5000000	7.8000000	13.550000
158	0.7500000	8.750000	2.5000000	6.5000000	18.500000
159	0.6666667	4.666667	2.3333333	10.4000000	18.066667
160	0.8333333	12.500000	2.5000000	5.2000000	21.033333
161	0.6000000	4.000000	4.0000000	6.2400000	14.840000
162	0.6000000	5.000000	1.6000000	6.2400000	13.440000
163	0.4500000	6.000000	1.4000000	6.2400000	14.090000
164	0.6666667	6.666667	5.0000000	10.4000000	22.733333
165	0.3333333	5.000000	1.5000000	5.2000000	12.033333
166	0.5000000	5.000000	3.7500000	5.2000000	14.450000

167	1.6666667	6.666667	0.8333333	0.0000000	9.166667
168	1.7500000	10.000000	3.5000000	0.0000000	15.250000
169	0.8000000	10.000000	3.6000000	3.1200000	17.520000
170	1.3333333	8.000000	3.3333333	5.2000000	17.866667
171	1.6666667	10.000000	1.6666667	15.6000000	28.933333
172	0.6250000	6.250000	3.1250000	0.9100000	10.910000
173	1.2142857	8.857143	2.1428571	5.3485714	17.562857
174	2.5000000	10.000000	3.7500000	3.9000000	20.150000
175	1.5000000	12.500000	1.5000000	15.6000000	31.100000
176	1.5000000	9.000000	3.5000000	7.8000000	21.800000
177	1.0000000	10.500000	3.5000000	7.8000000	22.800000
178	1.0000000	9.500000	3.2500000	4.1600000	17.910000
179	1.2500000	12.500000	4.0000000	13.0000000	30.750000
180	1.0000000	9.000000	3.0000000	1.6640000	14.664000
181	1.0000000	7.500000	3.3750000	1.8200000	13.695000
182	3.3333333	10.500000	2.0000000	8.3200000	24.153333
183	0.6666667	6.333333	2.6666667	10.4000000	20.066667
184	0.8333333	6.666667	3.3333333	10.4000000	21.233333
185	1.0000000	6.000000	5.0000000	6.2400000	18.240000
186	0.7500000	7.000000	3.0000000	22.8800000	33.630000
187	0.4285714	6.428571	2.8571429	2.9714286	12.685714
188	0.5000000	5.000000	2.5000000	7.8000000	15.800000
189	0.3750000	6.250000	1.8750000	3.9000000	12.400000
190	0.0000000	4.571429	2.8571429	2.2285714	9.657143
191	0.0000000	7.777778	1.1111111	1.1555556	10.044444
192	0.7142857	7.142857	0.7142857	5.6457143	14.217143
193	1.5000000	7.500000	2.0000000	5.2000000	16.200000
194	0.5000000	7.500000	1.0000000	2.0800000	11.080000
195	2.0000000	8.000000	0.8000000	12.4800000	23.280000
196	1.9230769	7.692308	1.1538462	4.8000000	15.569231
197	2.0000000	8.000000	2.0000000	0.0000000	12.000000
198	0.5000000	8.000000	1.5000000	3.1200000	13.120000
199	5.5000000	5.750000	5.0000000	8.3200000	24.570000
200	1.2142857	8.571429	3.4285714	5.2000000	18.414286

The above values have been obtained after separation of the variables.

```
predictions<-predict(model,newdata = test)
> r_squared<-cor(predictions,test$y)^2
> print(paste("R_squared:",r_squared))
[1] "R_squared: 1"
```

We have obtained the R square value as 1 which means that the model that has been built is a very good model to depict the significance of the selected variable on the total consumption of the state UP.

We can conclude that the consumption of the state depends on the above variables.

```
print(Mape_df)
[1] Inf
```

The mape value is not available as this does not have any error.

B. Using he IPL data establish the relationship between the

performance of the player and payment he receives:

```
"R_squared: 0.333813797229825
```

Here the R_square value is considerably low but the variables support the establishment of relation betwe en the payment and player’s performance.

```
4.837156
```

The Mape value is considerably high but the relation can be explained.

RECOMMENDATIONS

Multiple Regression Analysis for Variables in Uttar Pradesh State Consumption:

1. Based on the analysis results, policymakers should focus on addressing factors that contribute to lower consumption levels in certain regions of Uttar Pradesh. This may involve implementing targeted initiatives to improve income levels, education, and infrastructure in those areas.
2. Businesses operating in Uttar Pradesh can utilize the findings to identify market segments with high consumption potential and tailor their products and marketing strategies accordingly.
3. The analysis can help guide resource allocation decisions, allowing policymakers to prioritize investments in sectors that have a significant impact on consumption, such as healthcare, education, and infrastructure development.
4. Policymakers should consider the long-term sustainability of consumption patterns in Uttar Pradesh and develop strategies that promote balanced economic growth and environmental conservation

Correlation of Player Performance and Salary in IPL:

1. IPL team owners and management should consider the performance metrics of players, such as batting average, bowling economy, strike rate, and wickets taken, when determining player salaries. This ensures a fair and merit-based approach to compensation.
2. Analysis of player performance and salary correlation can help identify overpaid or underpaid players. This information can be utilized by team owners to make informed decisions during player auctions and contract negotiations.
3. Teams should focus on investing in players who consistently demonstrate high performance levels relative to their salaries, as they provide the best value for money and contribute significantly to the team's success.
4. The analysis can shed light on the factors that have the most significant impact on player performance. This information can be used to develop training programs, coaching strategies, and talent scouting techniques that improve player performance and maximize the return on investment.

CODES

#Perform Multiple regression analysis and carry out the regression diagnostics and explain your findings. Correct them and revisit your results and explain the significant differences you observe.

```
up_df<-state%>%
select(ricetotal_q,wheattotal_q, potato_q, milk_q)
up_df$tot_con<-up_df$ricetotal_q+up_df$wheattotal_q+up_df$potato_q+up_df$milk_q
x1<-up_df$ricetotal_q
x2<-up_df$wheattotal_q
x3<-up_df$potato_q
x4<-up_df$milk_q
y<-up_df$tot_con
install.packages("caTools")
library(caTools)
library(caret)
data_df<-data.frame(x1,x2,x3,x4,y)
dim(data_df)
show(data_df)
```

```
data_df<-na.omit(data_df)
set.seed(123)
split<-sample.split(data_df$y,SplitRatio =0.8)
train<-subset(data_df,split==TRUE)
test<-subset(data_df,split==FALSE)
model <- lm(y ~ x1 + x2 + x3 + x4, data = train)
predictions<-predict(model,newdata = test)
r_squared<-cor(predictions,test$y)^2
print(paste("R_squared:",r_squared))
##MAPE
MAPE<-function(actual,predicted){
  mean(abs((actual-predicted)/actual))
}
```

##Assigment 2

##b) Using he IPL data establish the relationship between the performance of the player and payment he receives

```
library(caTools)

library(caret)

library(ggplot2)

library(lattice)

data_df<-data.frame(x,y)

dim(data_df)

show(data_df)

data_df<-na.omit(data_df)

set.seed(123)

split<-sample.split(data_df$y,SplitRatio =0.8)

train<-subset(data_df,split==TRUE)
```

```
test<-subset(data_df,split==FALSE)

model <- lm(y ~ x, data = train)

predictions<-predict(model,newdata = test)

r_squared<-cor(predictions,test$y)^2

print(paste("R_squared:",r_squared))

##MAPE

MAPE<-function(actual,predicted){

  mean(abs((actual-predicted)/actual))

}

Mape_df<-MAPE(test$y,predictions)

print(Mape_df)
```