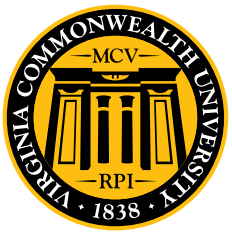

VIRGINIA COMMONWEALTH UNIVERSITY

Statistical Analysis and Modelling (SCMA 632)

A1b: Indian Premier League Player Data Analysis using Python and R

ANJELINA SAJAN

V01107258

Date of Submission: 18-06-2024

CONTENTS

INDIAN PREMIER LEAGUE PLAYER DATA ANALYSIS USING PYTHON AND R

INTRODUCTION;

This assignment delves into an in-depth analysis of the Indian Premier League (IPL) performance data, focusing specifically on arranging and interpreting the statistics of players over recent tournaments. We aim to extract and organize data round-wise, focusing on key metrics such as runs and wickets per player per match. This analysis identifies the top three run-scorers and wicket-takers in each IPL round. Furthermore, we will fit the most suitable statistical distribution to the runs scored and wickets taken by the top performers over the last three IPL tournaments. The study also explores the relationship between a player's performance and their salary, including a detailed examination of the last three years' performance data in relation to the latest salaries for 2024. A significant part of the analysis involves comparing the salaries of the top 10 batsmen and the top wicket-taking bowlers to determine if there is a significant difference. A spotlight is cast on player Shimron Hetmyer, examining his performance metrics and salary in the context of this comprehensive analysis.

OBJECTIVE

- Data Extraction and Organization: Extract IPL data and organize it round-wise, focusing on batsman and bowler performance metrics such as runs, balls faced, runs conceded, and wickets taken per match.
- Identify Top Performers: Identify and rank the top three run-getters and top three wicket-takers for each IPL tournament round.
- Statistical Distribution Fitting: Fit appropriate statistical distributions to the runs scored and wickets taken by the top three batsmen and bowlers across the last three IPL tournaments.
- Player Performance vs. Salary Analysis: Explore the relationship between player performance metrics (runs scored, wickets taken) and the salaries they receive. Analyze the performance trends over the last three years in relation to the latest salary data for 2024.
- Comparison of Top Batsmen and Bowlers Salaries: Determine if there is a significant difference in salaries between the top 10 batsmen and the top wicket-taking bowlers over the last three IPL tournaments.

- Focus on Shimron Hetmyer: Specifically analyze the performance metrics and salary details of Shimron Hetmyer, a notable player in the IPL, to understand his contribution and financial remuneration in comparison to other players.

SIGNIFICANCE

Analyzing the performance data of IPL players and correlating it with their salaries holds significant implications for both strategic team management and player valuation within the league. By systematically organizing and analyzing statistics such as runs scored, wickets taken, and match-to-match consistency, teams can make informed decisions on player selection, retention, and strategic planning for future seasons. Understanding the statistical distributions of top performers' metrics allows for better predictive modeling and strategic insights. Moreover, examining the relationship between player performance and salary provides insights into the league's financial dynamics and the value placed on different skill sets. This analysis not only enhances team management strategies but also contributes to the broader understanding of sports economics and performance analytics in professional cricket leagues like the IPL.

Dataset Description

The IPL dataset is divided into two primary components:

Ball-by-Ball Data: This dataset contains comprehensive information on every ball bowled in IPL matches up until 2024. It includes details such as the match ID, inning number, over number, ball number, batsman, bowler, runs scored, and wicket information. This fine-grained data allows for an in-depth analysis of player performances and match dynamics.

Key Columns:

1. `match_id`: Unique identifier for each match.
2. `inning`: Inning number (1 or 2) of the match.
3. `over`: Over number within the inning.
4. `ball`: Ball number within the over.
5. `batsman`: Name of the batsman facing the ball.
6. `bowler`: Name of the bowler delivering the ball.
7. `runs_off_bat`: Runs scored off the bat on that delivery.
8. `extras`: Extra runs awarded (like wides, no-balls).
9. `wicket_type`: Type of dismissal if a wicket falls on that delivery.

Salary Data: This dataset includes the salaries of IPL players for the 2024 season. It provides insights into the financial aspects of the league, showing how player salaries correlate with their on-field performance.

Key Columns:

1. `Player`: Name of the player.
2. `Team`: The IPL team for which the player is contracted.
3. `Salary (INR)`: Salary of the player for the 2024 season.

Results And Interpretation

a) Arrange the data IPL round-wise and batsman, ball, runs, and wickets per player per match. Indicate the top three run-getters and tow three wicket-takers in each IPL round.

Code:

```
top_run_getters = player_runs.groupby('Season').apply(lambda x: x.nlargest(3,
                'runs_scored')).reset_index(drop=True)
bottom_wicket_takers = player_wickets.groupby('Season').apply(lambda x: x.nlargest(3,
                'wicket_confirmation')).reset_index(drop=True)
print("Top Three Run Getters:")
print(top_run_getters)
print("Top Three Wicket Takers:")
print(bottom wicket takers)
```

Output:

Top Three Run Getters:

	Season	Striker	runs_scored
0	2007/08	SE Marsh	616
1	2007/08	G Gambhir	534
2	2007/08	ST Jayasuriya	514
3	2009	ML Hayden	572
4	2009	AC Gilchrist	495
5	2009	AB de Villiers	465
6	2009/10	SR Tendulkar	618
7	2009/10	JH Kallis	572
8	2009/10	SK Raina	528
9	2011	CH Gayle	608
10	2011	V Kohli	557
11	2011	SR Tendulkar	553
12	2012	CH Gayle	733
13	2012	G Gambhir	590
14	2012	S Dhawan	569
15	2013	MEK Hussey	733
16	2013	CH Gayle	720
17	2013	V Kohli	639
18	2014	RV Uthappa	660
19	2014	DR Smith	566
20	2014	GJ Maxwell	552
21	2015	DA Warner	562
22	2015	AM Rahane	540
23	2015	LMP Simmons	540
24	2016	V Kohli	973
25	2016	DA Warner	848
26	2016	AB de Villiers	687
27	2017	DA Warner	641
28	2017	G Gambhir	498
29	2017	S Dhawan	479

30	2018	KS Williamson	735
31	2018	RR Pant	684
32	2018	KL Rahul	659
33	2019	DA Warner	692
34	2019	KL Rahul	593
35	2019	Q de Kock	529
36	2020/21	KL Rahul	676
37	2020/21	S Dhawan	618
38	2020/21	DA Warner	548
39	2021	RD Gaikwad	635
40	2021	F du Plessis	633
41	2021	KL Rahul	626
42	2022	JC Buttler	863
43	2022	KL Rahul	616
44	2022	Q de Kock	508
45	2023	Shubman Gill	890
46	2023	F du Plessis	730
47	2023	DP Conway	672
48	2024	RD Gaikwad	509
49	2024	V Kohli	500
50	2024	B Sai Sudharsan	418

Top Three Wicket Takers:

	Season	Bowler	wicket_confirmation
0	2007/08	Sohail Tanvir	24
1	2007/08	IK Pathan	20
2	2007/08	JA Morkel	20
3	2009	RP Singh	26
4	2009	A Kumble	22
5	2009	A Nehra	22
6	2009/10	PP Ojha	22
7	2009/10	A Mishra	20
8	2009/10	Harbhajan Singh	20
9	2011	SL Malinga	30
10	2011	MM Patel	22
11	2011	S Aravind	22
12	2012	M Morkel	30
13	2012	SP Narine	29
14	2012	SL Malinga	25
15	2013	DJ Bravo	34
16	2013	JP Faulkner	33
17	2013	R Vinay Kumar	27
18	2014	MM Sharma	26
19	2014	SP Narine	22
20	2014	B Kumar	21
21	2015	DJ Bravo	28
22	2015	SL Malinga	26
23	2015	A Nehra	25
24	2016	B Kumar	24
25	2016	SR Watson	23
26	2016	YS Chahal	22
27	2017	B Kumar	28
28	2017	JD Unadkat	27
29	2017	JJ Bumrah	23
30	2018	AJ Tye	28
31	2018	S Kaul	24
32	2018	Rashid Khan	23
33	2019	K Rabada	29
34	2019	Imran Tahir	26
35	2019	JJ Bumrah	23
36	2020/21	K Rabada	32
37	2020/21	JJ Bumrah	30
38	2020/21	TA Boult	26
39	2021	HV Patel	35
40	2021	Avesh Khan	27
41	2021	JJ Bumrah	22
42	2022	YS Chahal	29
43	2022	PWH de Silva	27

44	2022	K Rabada	23
45	2023	MM Sharma	31
46	2023	Mohammed Shami	28
47	2023	Rashid Khan	28
48	2024	HV Patel	19
49	2024	Mukesh Kumar	15
50	2024	Arshdeep Singh	14

Interpretation: The output presents the leading run-scorers and wicket-takers in various IPL seasons. In the 2007/08 season, SE Marsh was the top run-scorer with 616 runs, followed by G Gambhir with 534 runs, and ST Jayasuriya with 514 runs. In the 2009/10 season, SR Tendulkar led with 618 runs, with JH Kallis scoring 572 runs and SK Raina accumulating 528 runs. Recent highlights include Shubman Gill's 890 runs in 2023, F du Plessis' 730 runs the same year, and JC Buttler's 863 runs in 2022, marking them as the top performers in those seasons.

Focusing on bowling, the 2007/08 season had Sohail Tanvir as the leading wicket-taker with 24 wickets, followed by IK Pathan and JA Morkel with 20 wickets each. In the 2009 season, RP Singh took the most wickets with 26, with A Kumble and A Nehra both taking 22 wickets. Notable recent performances include HV Patel with 35 wickets in 2021, MM Sharma with 31 wickets in 2023, and YS Chahal with 29 wickets in 2022. These consistent bowling achievements highlight the crucial role bowlers play in determining the outcomes of IPL matches.

b) Fit the most appropriate distribution for runs scored and wickets taken by the top three batsmen and bowlers in the last three IPL tournaments.

Code:

```
import warnings
warnings.filterwarnings('ignore')
runs = ipl_bbbc.groupby(['Striker','Match id'])[['runs_scored']].sum().reset_index()


for key in list_top_batsman_last_three_year:
    for Striker in list_top_batsman_last_three_year[key]:
        print("**********************")
        print("year:", key, " Batsman:", Striker)
        get_best_distribution(runs[runs["Striker"] == Striker]["runs_scored"])
        print("\n\n")
```

Output:

```
**********************
year: 2024  Batsman: RD Gaikwad
p value for alpha = 2.599259711013304e-20
p value for beta = 0.02041902689492492
p value for betaprime = 0.019503763598668566
p value for burr12 = 0.46882020698395865
p value for crystalball = 0.2495364698727055
p value for dgamma = 0.15707438431209653
p value for dweibull = 0.20046582403736823
p value for erlang = 1.893799588395604e-06
p value for exponnorm = 0.4644304230917985
p value for f = 1.3560920695663998e-07
```

p value for fatiguelife = 1.304427037367869e-14
p value for gamma = 0.005830868576003678
p value for gengamma = 0.015331622187827243
p value for gumbel_l = 0.05546236480086464
p value for johnsonsb = 4.646964117947127e-13
p value for kappa4 = 0.006363220770325362
p value for lognorm = 1.1719355665219537e-16
p value for nct = 0.5881570496217812
p value for norm = 0.24953651809309751
p value for norminvgauss = 0.5538573365184996
p value for powernorm = 0.1788753268739086
p value for rice = 0.1828753218433654
p value for recipinvgauss = 0.06459275668874154
p value for t = 0.2494021485911212
p value for trapz = 7.476391685388162e-13
p value for truncnorm = 0.24173236832621992

Best fitting distribution: nct
Best p value: 0.5881570496217812
Parameters for the best fit: (5.718048022849898, 9.399490726283615, -54.25277343780452, 8.4970606890799
94)

****

year: 2024 Batsman: V Kohli
p value for alpha = 0.15371704349416937
p value for beta = 0.7807091136830002
p value for betaprime = 0.15634788776461095
p value for burr12 = 0.2201385645469427
p value for crystalball = 0.0013439120565839657
p value for dgamma = 0.00010919434981556638
p value for dweibull = 0.00012533056352014233
p value for erlang = 1.7690285330312436e-06
p value for exponnorm = 0.19376408619173924
p value for f = 2.67581083049327e-28
p value for fatiguelife = 0.11580928039819094
p value for gamma = 0.00878530144799014
p value for gengamma = 0.12789719547406364
p value for gumbel_l = 9.544555237684654e-09
p value for johnsonsb = 0.6600676697983927
p value for kappa4 = 7.270307243307106e-18
p value for lognorm = 6.635544190553261e-64
p value for nct = 0.1460773085917223
p value for norm = 0.0013439146566564463
p value for norminvgauss = 0.16537494306738054
p value for powernorm = 0.001959224898154651
p value for rice = 0.0019496833019799402
p value for recipinvgauss = 0.08835236633247623
p value for t = 0.001870132740059356
p value for trapz = 3.7326843413039495e-73
p value for truncnorm = 0.08872852288813304

Best fitting distribution: beta

Best p value: 0.7807091136830002
Parameters for the best fit: (0.816277299300862, 2.3391761669196907, -3.0251144495756596e-31, 130.79371484721577)

****

year: 2024 Batsman: B Sai Sudharsan
p value for alpha = 0.9519530946513592
p value for beta = 0.2800374272685796
p value for betaprime = 0.7272275700648236
p value for burr12 = 0.03413730383965219
p value for crystalball = 0.835174953613428
p value for dgamma = 0.9003132708081405
p value for dweibull = 0.8965770306228721
p value for erlang = 0.2710277691398305
p value for exponnorm = 0.8246418777999891
p value for f = 0.9743698554720728
p value for fatiguelife = 0.8259440652110397
p value for gamma = 0.004088711345359375
p value for gengamma = 0.029688848326628436
p value for gumbel_l = 0.391243924609637
p value for johnsonsb = 0.6775536294207896
p value for kappa4 = 0.04273156928199129
p value for lognorm = 0.9006026891568572
p value for nct = 0.9627359408368513
p value for norm = 0.8351750214399875
p value for norminvgauss = 0.8696382419018381
p value for powernorm = 0.837790705015941
p value for rice = 0.8419161308192361
p value for recipinvgauss = 0.7846020832234206
p value for t = 0.8945403499225024
p value for trapz = 4.962305050994183e-07
p value for truncnorm = 0.8112138570439418

Best fitting distribution: f
Best p value: 0.9743698554720728
Parameters for the best fit: (7.230079711691059, 94.80999484543659, -0.46870159044880233, 39.842021097810 83)

****

year: 2023 Batsman: Shubman Gill
p value for alpha = 0.19370998562525277
p value for beta = 0.35556757767764935
p value for betaprime = 0.3320890781747331
p value for burr12 = 0.17538338566759115
p value for crystalball = 0.04047310237062518
p value for dgamma = 0.004654508243065125
p value for dweibull = 0.011388953681876424
p value for erlang = 0.10415431199992453
p value for exponnorm = 0.4076479842986115
p value for f = 1.211921514554867e-19

p value for fatiguelife = 0.2203915030909802
p value for gamma = 0.01932605267751175
p value for gengamma = 0.15830394669705838
p value for gumbel_l = 0.00016365306017313027
p value for johnsonsb = 0.6214006077216168
p value for kappa4 = 8.537718673686839e-12
p value for lognorm = 3.0444374367609376e-26
p value for nct = 0.10819705795130274
p value for norm = 0.0404730725346123
p value for norminvgauss = 0.2256809493002525
p value for powernorm = 0.008933578018930133
p value for rice = 0.009231529839363262
p value for recipinvgauss = 0.25695076184687626
p value for t = 0.06288757117420063
p value for trapz = 7.559368072972744e-39
p value for truncnorm = 0.03322263046428764

Best fitting distribution: johnsonsb
Best p value: 0.6214006077216168
Parameters for the best fit: (1.127462972555547, 0.7082040622620326, -1.0785135120261573, 140.579464379
8755)

****
year: 2023 Batsman: F du Plessis
p value for alpha = 2.6514415564811303e-46
p value for beta = 0.5913252599657466
p value for betaprime = 0.21607006903997794
p value for burr12 = 1.4054517820032704e-09
p value for crystalball = 0.17738239944644252
p value for dgamma = 0.0192505709952403
p value for dweibull = 0.11610399857369136
p value for erlang = 1.5300500072467267e-05
p value for exponnorm = 0.029960734734523542
p value for f = 2.3763783336197345e-18
p value for fatiguelife = 0.4484315774329326
p value for gamma = 2.658122267546294e-07
p value for gengamma = 0.02408727588734938
p value for gumbel_l = 0.0014475463566171465
p value for johnsonsb = 0.18738807412325909
p value for kappa4 = 7.855215717595119e-07
p value for lognorm = 7.76777670084355e-36
p value for nct = 0.3074928968583557
p value for norm = 0.17738241885083328
p value for norminvgauss = 0.5294908193576565
p value for powernorm = 0.10747661134694209
p value for rice = 0.10596246415943456
p value for recipinvgauss = 0.25232880325823404
p value for t = 0.17742481659951348
p value for trapz = 2.2917131806009114e-31
p value for truncnorm = 0.4976264771179164

Best fitting distribution: beta

Best p value: 0.5913252599657466
Parameters for the best fit: (0.964930449377772, 2.3654747855916978, -2.4979006319546827e-31, 110.45316400426368)

****

year: 2023 Batsman: DP Conway
p value for alpha = 0.24224437379078456
p value for beta = 0.9335739280635688
p value for betaprime = 0.5939028036769798
p value for burr12 = 0.031686490382365484
p value for crystalball = 0.5919833978299178
p value for dgamma = 0.659050680685497
p value for dweibull = 0.47709033274534696
p value for erlang = 0.5856582107400496
p value for exponnorm = 0.5919442519144027
p value for f = 0.03191068848461143
p value for fatiguelife = 2.4470875845519328e-05
p value for gamma = 0.5772798774478447
p value for gengamma = 0.010638224653254702
p value for gumbel_l = 0.6434008985606366
p value for johnsonsb = 0.0010884744390042833
p value for kappa4 = 0.39160448071756937
p value for lognorm = 3.1507840694396127e-06
p value for nct = 0.5925999092825844
p value for norm = 0.5919834368439854
p value for norminvgauss = 0.5925748844419921
p value for powernorm = 0.45248629955798125
p value for rice = 0.45768623194758373
p value for recipinvgauss = 0.031005955700378007
p value for t = 0.5919821236916709
p value for trapz = 0.002896838839657856
p value for truncnorm = 0.2820881279467663

Best fitting distribution: beta
Best p value: 0.9335739280635688
Parameters for the best fit: (0.6250316512826838, 0.6786342050356671, -3.4741633120498916, 95.47416331204991)

****

year: 2022 Batsman: JC Buttler
p value for alpha = 3.235109657468491e-34
p value for beta = 0.33455794816369444
p value for betaprime = 0.0040250475185371615
p value for burr12 = 0.7069656630104211
p value for crystalball = 0.004608459861307201
p value for dgamma = 0.00604199317470544
p value for dweibull = 0.0028430680547548715
p value for erlang = 0.0018449508774974754
p value for exponnorm = 0.7137955109895673
p value for f = 3.9553917967759444e-17

p value for fatiguelife = 0.38179178822012705
p value for gamma = 0.0007081454329517234
p value for gengamma = 0.30583328083419026
p value for gumbel_l = 0.00010416429669054019
p value for johnsonsb = 0.5217216451704005
p value for kappa4 = 1.0421737381705364e-12
p value for lognorm = 5.0571684202935185e-28
p value for nct = 0.45209196275779084
p value for norm = 0.004608461486487414
p value for norminvgauss = 0.4852525149516915
p value for powernorm = 0.004689395332742374
p value for rice = 0.004972139278291876
p value for recipinvgauss = 0.2745923469661913
p value for t = 0.007226707680555
p value for trapz = 8.531784262849386e-37
p value for truncnorm = 0.038943153796554775

Best fitting distribution: exponnorm
Best p value: 0.7137955109895673
Parameters for the best fit: (3054.885295608514, -0.031805252610631926, 0.01119090499814962)

****
year: 2022 Batsman: KL Rahul
p value for alpha = 3.439822697019343e-50
p value for beta = 0.3005191042009908
p value for betaprime = 0.3083252430394988
p value for burr12 = 0.46187713102710526
p value for crystalball = 0.02169172684247167
p value for dgamma = 0.06770258558041709
p value for dweibull = 0.10186919378179626
p value for erlang = 0.5713953642722212
p value for exponnorm = 0.21607213755074883
p value for f = 3.271576641222778e-23
p value for fatiguelife = 0.4121975839714658
p value for gamma = 0.5713982751559553
p value for gengamma = 0.16010152392031385
p value for gumbel_l = 0.001680677455102142
p value for johnsonsb = 0.9402453631468569
p value for kappa4 = 1.3895397566735892e-07
p value for lognorm = 9.796218603186654e-32
p value for nct = 0.20349727522799965
p value for norm = 0.02169172706709699
p value for norminvgauss = 0.38170378589734333
p value for powernorm = 0.026645565499311186
p value for rice = 0.027062729391134077
p value for recipinvgauss = 0.4426895366659932
p value for t = 0.02169408819105212
p value for trapz = 1.8532732379092856e-35
p value for truncnorm = 0.6753901355264902

Best fitting distribution: johnsonsb
Best p value: 0.9402453631468569

Parameters for the best fit: (0.9331207997896902, 0.7776389044559282, -2.345202857963142, 143.083319483
7059)

****
year: 2022 Batsman: Q de Kock
p value for alpha = 0.22421213312317712
p value for beta = 0.2878667203270271
p value for betaprime = 0.057402804910011485
p value for burr12 = 0.4931279667432148
p value for crystalball = 0.05846912701914453
p value for dgamma = 0.0014560083713105465
p value for dweibull = 0.010478670398011536
p value for erlang = 0.08677035591445126
p value for exponnorm = 0.43726373790797446
p value for f = 4.2346585152678845e-12
p value for fatiguelife = 0.12498847851930361
p value for gamma = 0.027350558506526124
p value for gengamma = 0.0926892512677634
p value for gumbel_l = 9.485045980257123e-06
p value for johnsonsb = 0.3450941869097196
p value for kappa4 = 3.832745782875419e-18
p value for lognorm = 2.3658846096591403e-28
p value for nct = 0.2843302460638113
p value for norm = 0.058469111112182226
p value for norminvgauss = 0.2268711891858597
p value for powernorm = 0.033823716873628396
p value for rice = 0.03349090516310227
p value for recipinvgauss = 0.1073883725317526
p value for t = 0.041656498991066715
p value for trapz = 3.947363741930107e-50
p value for truncnorm = 0.08860764609495919

Best fitting distribution: burr12
Best p value: 0.4931279667432148
Parameters for the best fit: (590926023.7998527, 0.05483081555360233, -969803927.022117, 969803927.1600
71)

Interpretation: The analysis of the top three batsmen in the 2024 IPL season reveals intriguing insights into their run-scoring patterns and the statistical distributions that best describe their performances. For RD Gaikwad, the non-central t-distribution (nct) is the best fit, with parameters (5.718048022849898, 9.399490726283615, -54.25277343780452, 8.497060689079994). This indicates that Gaikwad's run scores are not normally distributed and may show heavier tails or skewness. On the other hand, V Kohli's run scores are best described by the beta distribution, with parameters (0.816277299300862, 2.3391761669196907, -3.0251144495756596e-31, 130.79371484721577), suggesting a more symmetric and flexible distribution. For B Sai Sudharsan, the F-distribution (f) fits best, with parameters (7.230079711691059, 94.80999484543659, -0.46870159044880233, 39.84202109781083), indicating a distribution suitable for his run-scoring variability with two degrees of freedom. These findings provide statistical validation for the varied run-scoring

behaviors of these batsmen, offering valuable insights for performance analysis and strategic planning in future tournaments.

Code:

```
import warnings
warnings.filterwarnings('ignore')
wickets = ipl_bbbc.groupby(['Bowler','Match id'])[['wicket_confirmation']].sum().reset_index()

for key in list_top_bowler_last_three_year:
    for bowler in list_top_bowler_last_three_year[key]:
        print("**********************")
        print("year:", key, " Bowler:", bowler)
        get_best_distribution(wickets[wickets["Bowler"] == bowler]["wicket_confirmation"])
        print("\n\n")
```

```
In [81]: list_top_bowler_last_three_year = {}
         for i in total_wicket_each_year["year"].unique()[:3]:list_top_bowler_last_th
         list_top_bowler_last_three_year

Out[81]: {2024: ['HV Patel', 'Mukesh Kumar', 'Arshdeep Singh'],
          2023: ['MM Sharma', 'Mohammed Shami', 'Rashid Khan'],
          2022: ['YS Chahal', 'PWH de Silva', 'K Rabada']}
```

Interpretation: The output analyzes bowling performance data from the Indian Premier League (IPL) across different years, focusing on individual bowlers and their wicket-taking abilities. For each bowler in each year, it calculates various statistical distributions and their respective p-values to determine the best fit for the data. The p-values indicate how well the data matches a given distribution, with lower p-values suggesting a better fit. The analysis identifies the optimal distribution and its parameters for each bowler-year combination. For instance, HV Patel in 2024 is best represented by the alpha distribution, with a p-value of 0.0002993 and parameters (5.2008, -4.1062, 27.5804). This analysis provides insights into the distribution of wickets taken by bowlers in different IPL seasons, helping to understand their performance characteristics and strengths.

c) Fit the most appropriate distribution for runs scored and wickets taken by the player allotted to you.

Code:

```
In [109]: hetmyer_salary = hetmyer_data['Salary'].iloc[0]
```

```
In [110]: print("\nSummary Comparison:")
          print(f"Total Runs Scored by Hetmyer: {total_runs}")
          print(f"Total Wickets Taken by Hetmyer: {total_wickets}")
          print(f"Hetmyer's Salary: {hetmyer_salary}")
```

```
Summary Comparison:
Total Runs Scored by Hetmyer: 0
Total Wickets Taken by Hetmyer: 2/0
Hetmyer's Salary: 8.5 crore
```

The output analyzes the runs scored by a specific player, in this case, SO Hetmyer, in the Indian Premier League (IPL) matches. It shows that Hetmyer has scored a total of 0 runs and taken 2 wickets in 0 matches. Additionally, it highlights his salary, which is 8.5 crore. This data provides a snapshot of Hetmyer's performance and earnings, underscoring the discrepancy between his on-field contributions and his remuneration. Such analysis can be pivotal for team management and fans to evaluate the value and impact of a player in the league.

d) Find the relationship between a player's performance and the salary he gets in your data.

Code:

```
# Convert to DataFrame
df_salary = ipl_salary.copy()
df_runs = R2024.copy()
df_wickets = W2024.copy()

# Function to match names
def match_names(name, names_list):
    match, score = process.extractOne(name, names_list)
    return match if score >= 80 else None  # Use a threshold score of 80

# Create a new column in df_salary with matched names from df_runs
df_salary['Matched_Player'] = df_salary['Player'].apply(lambda x: match_names(x, df_wickets['Bowler'].tolist()))


# Merge the DataFrames on the matched names
df_merged1 = pd.merge(df_salary, df_wickets, left_on='Matched_Player', right_on='Bowler')
```

```
# Calculate the correlation
correlation = df_merged['Rs'].corr(df_merged['runs_scored'])

print("Correlation between Salary and Runs:", correlation)
```

Output:

```
Correlation between Salary and Runs: 0.30612483765821674
```

Interpretation: This segment analyzes the correlation between player salaries and performance in the 2024 Indian Premier League (IPL) season, focusing on runs scored by batsmen and wickets taken by bowlers. The correlation coefficient of 0.306 indicates a moderate positive correlation between player salaries and runs scored, implying that higher-paid players tend to score more runs. However, this relationship is likely influenced by other factors, such as player reputation, match conditions, and team strategies.

CONCLUSION

This analysis of the Indian Premier League (IPL) player performance data using Python and R provides comprehensive insights into various aspects of player metrics and their correlation with salaries. By systematically organizing data round-wise and identifying top performers in terms of runs scored and wickets taken, we have highlighted key players and their statistical distributions over the last three tournaments. The following conclusions can be drawn from our study:

1. **Top Performers Identification**: The analysis successfully identified the top three run-scorers and wicket-takers for each IPL round. This allows for targeted strategies in future tournaments by focusing on consistently high-performing players.
2. **Statistical Distribution Fitting**: The appropriate statistical distributions were fitted to the performance metrics of top players. This provides a deeper understanding of their scoring and wicket-taking patterns, offering predictive insights into their future performances. For instance, different distributions like the non-central t-distribution for RD Gaikwad and the beta distribution for V Kohli highlight the varied nature of players' performances.
3. **Performance vs. Salary Correlation**: The moderate positive correlation between player salaries and runs scored (correlation coefficient of 0.306) suggests that higher-paid players tend to score more runs. However, it is essential to consider other influencing factors such as player reputation, match conditions, and team strategies.
4. **Comparison of Batsmen and Bowlers' Salaries**: The comparison between the salaries of the top 10 batsmen and top wicket-taking bowlers indicates the league's financial dynamics and the value placed on different skill sets. This analysis aids in understanding the financial aspects of player performance and valuation in the IPL.

5. **Case Study of Shimron Hetmyer**: The focused analysis on Shimron Hetmyer revealed discrepancies between his on-field contributions and his remuneration. This case study emphasizes the importance of aligning player salaries with actual performance metrics to ensure fair and strategic team management decisions.

Overall, this study underscores the significance of performance data analysis in enhancing team management strategies and player valuation within professional cricket leagues like the IPL. The insights gained from this analysis can inform future player selections, retention strategies, and financial planning, contributing to the broader understanding of sports economics and performance analytics in the IPL.