VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modelling (SCMA 632)

A2b: Regression - Predictive Analytics

JYOTHIS KANIYAMPARAMBIL THANKACHAN

V01110144

Date of Submission: 23-06-2024

CONTENTS

Introduction

An internationally recognised cricket competition, the Indian Premier competition (IPL) is renowned for its exciting matches and elite players. In order to determine the correlation between player performance metrics and pay, this research will examine player data from the Indian Premier League for the last three years. The impact of performance criteria on player compensation is statistically measured in this study through the use of rigorous regression analysis. The results will support well-informed decisions about player acquisition, contract negotiations, and team strategies made by stakeholders, cricket analysts, and team management. It is anticipated that the findings would shed light on the IPL's market dynamics and deepen our comprehension of the factors that affect player salaries. This paper's main goal is to add to the ongoing conversation about player worth and performance evaluation in professional cricket by offering useful information that may enhance team performance and foster league-wide strategic decision-making.

Objectives

- Investigates correlation between performance metrics and salary.
- Identifies key performance indicators (KPIs) impacting player compensation.
- Performs regression analysis to identify strong statistical correlations between performance indicators and salaries.
- Analyzes time trends in player remuneration and performance indicators over the past three years.
- Provides insights for stakeholders to enhance player recruitment, retention, and contract negotiations.
- Participates in cricket analytics and player valuation to enhance understanding of player assessment in professional cricket leagues.
- Suggests future research and analysis to enhance comprehension of player performance correlation and data collection and analysis methods.

Business Significance

There are major business ramifications for all parties involved in cricket, including club owners, coaches, players, sponsors, and spectators, due to the link between IPL player wages and success. IPL club owners may enhance their ability to attract and retain players, optimise squad configuration, and negotiate contracts by finding performance metrics that are associated with greater compensation. In order to ensure fair compensation based on contributions made on the field, it might be helpful to understand the particular parameters that affect salary increases during contract talks.

Understanding the connection between player performance and compensation is becoming more and more interesting to fans and sponsors since more transparency in player value may boost fan engagement and give advertisers insightful information. By examining the relationship between pay and performance, data-driven decision-making is promoted while minimising biases and subjective player evaluations.

Teams who are able to find discounted players and those who are performing extraordinarily well in comparison to their pay through the effective application of data analytics to evaluate player performance and compensation dynamics will have a competitive edge. By enabling further research into player value evaluation, modelling, and the development of performance metrics, this information advances sports analytics in professional sports and cricket.

In short, optimising team performance, raising stakeholder engagement, and enhancing the general competitiveness and long-term sustainability of IPL franchises all depend on a knowledge of the relationship between IPL player performance and compensation

Results and Interpretation using R

```
# Convert Date column to datetime format
>       match_details$Date <- as.Date(match_details$Date, format = "%d-%m-%Y")

# Filter last one year of data
>       last_three_years <- match_details %>%
+       filter(Date >= "2024-01-01")

# Filter last two years of data
>       last_three_years <- match_details %>%
+       filter(Date >= "2023-01-01")

# Filter last three years of data
>       last_three_years <- match_details %>%
+       filter(Date >= "2022-01-01")

# Calculate performance metrics
>       performance <- last_three_years %>%
+       group_by(Striker) %>%
+       summarise(Total_Runs = sum(runs_scored), Balls_Faced = n())

# Clean and transform salary data
>       player_salary$Salary <- as.character(player_salary$Salary) # ensure Sala ry is a
character vector
>       player_salary$Salary <- gsub("s", "", player_salary$Salary)
>       player_salary$Salary <- gsub(",", "", player_salary$Salary)
>
>       player_salary$Salary <- ifelse(grepl("lakh", tolower(player_salary$Salar y)),
+       as.integer(as.numeric(gsub(" lakh", "", p layer_salary$Salary)) * 100000),
+       ifelse(grepl("crore", tolower(player_sala ry$Salary)),
+       as.integer(as.numeric(gsub(" crore
", "", player_salary$Salary)) * 10000000),
+       as.integer(as.numeric(player_salar
y$Salary))))

# Replace NAs with 0
>       player_salary$Salary[is.na(player_salary$Salary)] <- 0
>       names(performance)
[1] "Striker"    "Total_Runs" "Balls_Faced"

>       names(player_salary)
[1] "Player"     "Salary"          "Rs"     "international" "iconi c"

>       performance <- performance %>% rename(Player = Striker)

>       merged_data <- inner_join(performance, player_salary, by = "Player")

>       merged_data <- inner_join(performance, player_salary, by = c("Player" = "Player"))
```

```
>       common_columns <- intersect(names(performance), names(player_salary))
>       common_columns
[1] "Player"

>       merged_data <- inner_join(performance, player_salary, by = "Player")
>       common_column <- common_columns[1]
>       merged_data <- inner_join(performance, player_salary, by = common_column
)
>       common_cols <- intersect(names(performance), names(player_salary))
>       merged_data <- performance %>%
+       inner_join(player_salary, by = setNames(common_columns, common_columns
))
```

Interpretation:

Using the `Date` column as a criteria, the `match_details` dataset was filtered as part of the data analysis process to extract data from the preceding three years. For each player in the {Striker} position, performance measures such as {Total_Runs} and {Balls_Faced} were computed. To standardise the `Salary} column, the `player_salary} dataset underwent many cleaning and transformation processes. The `inner_join} function from the `dplyr} package was used to comb through the cleaned `performance} and `player_salary} datasets. Now included in the merged dataset, {merged_d ata}, are player-level performance metrics together with the corresponding compensation data, `compensation}. The integration of this data enables additional study to examine the relationship between player performance and compensation in the Indian Premier League throughout the previous three seasons. Regression analysis and other statistical methods may be applied to the combined data to quantify and comprehend the impact of player performance indicators on compensation. This dataset can also help IPL teams make strategic decisions about player recruitment, contract negotiations, and squad composition by providing useful information on the financial assessments of players based on their performance on the pitch.

Code:

```
>    # Correlation analysis
>    corr_matrix <- cor(merged_data[, c("Total_Runs",
"Balls_Faced", "Salary"
)])
```

```
>    print(corr_matrix)


     Total_Runs    Balls_Faced   Salary
Total_Runs    1.00000000.99646580.5801451
Balls_Faced   0.99646581.00000000.5881400
Salary    0.58014510.58814001.0000000
```

Interpretation:

The variables "Total_Runs}, "Balls_Faced}, and "Salary}" have substantial relationships, according to the correlation matrix from the dataset "merged_data}". A player's total runs and the number of batting opportunities are directly correlated, as seen by the significant correlation between total runs and number of balls faced. Since the ability to score runs is highly prized in player contracts and negotiations, players with greater total runs scored typically get larger salary. The somewhat positive correlation shown between total runs and wage implies that players who are more valued are those who spend more time at the crease and make a big contribution to the success of their club. IPL team management and other stakeholders may use the study's numerical insights on the relationship between player performance metrics and compensation to inform their strategic decisions about player contracts and team composition.

Code:

```
>       # Regression analysis
>       df <- merged_data
>       # Define X and y
>       X <- df[, c("Balls_Faced", "Total_Runs")]
>       y <- df$Salary
>       # Fit the linear regression model
>       X <- as.matrix(df[, c("Balls_Faced", "Total_Runs")])
>       model <- lm(y ~ X)
>       model <- lm(y ~ Balls_Faced + Total_Runs, data = df)
>       # Print the summary of the model
>       summary(model)

Call:
lm(formula = y ~ Balls_Faced + Total_Runs, data = df)

Residuals:
Min      1Q  Median      3Q     Max
-68980662 -18766746 -15277907 18056217 123708037
```

Coefficients:
Estimate Std. Error t value Pr(>|t|)
(Intercept) 17156955 8490768 2.021 0.052 .
Balls_Faced 220781 267155 0.826 0.415
Total_Runs -93832 192778 -0.487 0.630

Signif. codes: 0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 40140000 on 31 degrees of freedom Multiple R-squared: 0.3509, Adjusted R-squared: 0.309
F-statistic: 8.378 on 2 and 31 DF, p-value: 0.001234

Interpretation:

The predictive relationship between `Balls_Faced} and `Total_Runs} on {Salary} f or IPL players is shown by the regression analysis of the model {lm(y ~ Balls_Faced + Total_Runs, data = df)}. The intercept coefficient, which is 17156955, indicates that the expected salary is about 17,156,955 when both variables are zero. At the 0.05 significance level, it appears to be marginally significant, based on the p-value (0.052) associated with the intercept. There appears to be no statistical significance between the variable and the prediction of {Salary}, as indicated by the regression coefficient of 220781 for {Balls_Faced} and the standard error of 267155. The fact that the total model fit is statistically significant raises the possibility that other factors exist to explain variations in p layer wages. Further exploration of new factors or various modeling approaches may be essential to better the accuracy of projecting salary determination in the IPL based on pl ayer performance indicators.

Code:

```
# Get the coefficients
>       coefficients <- tidy(model)

# Print the coefficients
>       print(coefficients) # A tibble: 3 × 5
term      estimate std.error statistic p.value
<chr>   <dbl>   <dbl>   <dbl>   <dbl>
1 (Intercept)     17156955.       8490768.        2.02      0.0520
2 Balls_Faced   220781.267155.0.826    0.415
3 Total_Runs    -93831. 192778.-0.487   0.630

# Get the R-squared value
>       r_squared <- glance(model)
```

```
# Print the R-squared value
>       print(r_squared) # A tibble: 1 × 12
r.squared adj.r.squared   sigma statistic p.value    df logLik          AIC BIC deviance

dbl>     <dbl>
<dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl>   <dbl> <
1       0.351   0.309   40144934.       8.38    0.001232        -642.    1292. 1
298.    5.00e16
# i 2 more variables: df.residual <int>, nobs <int>
```

Interpretation :

The 'tidy()' and 'glance(model)' functions are used to analyse the linear regression model that was used to estimate player salary using the variables 'Balls_Faced' and 'Total_Runs'. While there is no statistically significant connection between the variables according to the intercept coefficient (17156955), the r egression coefficient for 'Balls_Faced' is -93831 with a standard error of 192778. The coefficient of determination (R-squared) is 0.351, indicating that the predictors can explain about 35.1% of the variability in the'Salary' variable. There exists a correlation between 'Salary' and at least one predictor, since the model exhibits statistical significance (p = 0.00123). Nevertheless, the R-squared value of 0.351 suggests that other significant factors are not taken into consideration and that the model only explains a small portion of the variability in "Salary." In order to improve the precision of projecting IPL player wages based on performance measures, the model recommends investigating further variables or different modelling methods.

Code:

```
# Load the ggplot2 library
>       library(ggplot2)

# Create a scatterplot of the data with a regression line
>       ggplot(df, aes(x = Balls_Faced, y = Total_Runs)) +

+       geom_point() +
+       geom_smooth(method = "lm", se = FALSE) +
+       labs(x = "Balls Faced", y = "Total Runs") +
+       theme_classic()
`geom_smooth()` using formula = 'y ~ x'
```

Interpretation:

A scatterplot displaying the correlation between {Balls_Faced} and `Total_Runs} in IPL matches is produced by the `ggplot2} code. The scatterplot points show how well a player performed in terms of total runs scored and balls faced. The linear regression model that best matches the connection between the variables is shown by the regression line, a blue line drawn on top of the scatterplot. The graph indicates that there is a positive connection between the two variables, indicating that players who face more balls are probably going to score more runs. According to the summary of the regression model, {Balls_Faced} does not significantly predict salary, indicating that additional factors need to be taken into account. When analysing and presenting the association between the variables in relation to IPL player performance, the scatterplot and regression line serve as useful visual aids.

Results and Interpretation using Python

```
import pandas as pd
from sklearn.model_selection import train_test_split
importstatsmodels.api as sm

# Assuming df_merged is already defined and contains the necessary columns
X = df_merged[['runs_scored']] # Independent variable(s)
y = df_merged['Rs'] # Dependent variable

# Split the data into training and test sets (80% for training, 20% for
testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Add a constant to the model (intercept)
X_train_sm = sm.add_constant(X_train)

# Create a statsmodels OLS regression model
model = sm.OLS(y_train, X_train_sm).fit()

# Get the summary of the model
summary = model.summary()
print(summary)
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                     Rs   R-squared:                       0.080
Model:                            OLS   Adj. R-squared:                  0.075
Method:                 Least Squares   F-statistic:                     15.83
Date:                Sun, 23 Jun 2024   Prob (F-statistic):           0.000100
Time:                        19:41:05   Log-Likelihood:                -1379.8
No. Observations:                 183   AIC:                             2764.
Df Residuals:                     181   BIC:                             2770.
Df Model:                           1
Covariance Type:            nonrobust
================================================ ==============================
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
-
const        430.8473     46.111      9.344      0.000     339.864     521.831
runs_scored    0.6895      0.173      3.979      0.000       0.348       1.031
==============================================================================
Omnibus:                       15.690   Durbin-Watson:                   2.100
Prob(Omnibus):                  0.000   Jarque-Bera (JB):               18.057
Skew:                           0.764   Prob(JB):                     0.000120
Kurtosis:                       2.823   Cond. No.                         363.
==============================================================================
```

```
Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correct
ly specified.
```

Interpretation:

A statistically significant positive link between the amount of runs scored and the IPL players' compensation in Indian rupees is revealed by the Ordinary Least Squares (OLS) regression analysis. The variable runs_scored appears to be responsible for approximately 8.0% of the variance in salary, as indicated by the coefficient of determination (R-squared) of 0.080. With an adjusted R-squared of 0.075, runs_scored accounts for around 7.5% of the variation in compensation. With an incredibly low probability (Prob) value of 0.000100 and an F-statistic of 15.83, the model is statistically significant overall. The coefficients show the predicted variation in {Rs} when {runs_scored} rises by one unit and the estimated value of {Rs} when the independent variable is equal to zero. The AIC and BIC criteria, together with the Log-Lik elihood, are used to assess the model fit. The study is predicated on the mistakes having no autocorrelation and a constant variance.

```python
import pandas as pd
from sklearn.model_selection import train_test_split
import statsmodels.api as sm

# Assuming df_merged is already defined and contains the necessary columns
X = df_merged[['wicket_confirmation']] # Independent variable(s)
y = df_merged['Rs'] # Dependent variable

# Split the data into training and test sets (80% for training, 20% for
testing)
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)

# Add a constant to the model (intercept)
X_train_sm = sm.add_constant(X_train)

# Create a statsmodels OLS regression model
model = sm.OLS(y_train, X_train_sm).fit()

# Get the summary of the model
summary = model.summary()
print(summary)
```

```
OLS Regression Results
==============================================================================
Dep. Variable:                     Rs   R-squared:                       0.074
Model:                            OLS   Adj. R-squared:                  0.054
Method:                 Least Squares   F-statistic:                     3.688
Date:                Sun, 23 Jun 2024   Prob (F-statistic):             0.0610
Time:                        19:45:07   Log-Likelihood:                -360.96
No. Observations:                  48   AIC:                             725.9
Df Residuals:                      46   BIC:                             729.7
Df Model:                           1
Covariance Type:            nonrobust
==============================================================================
                     coef     std err      t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
const                396.6881   91.270    4.346    0.000    212.971     580.405
wicket_confirmation   17.6635    9.198    1.920    0.061     -0.851      36.179
==============================================================================
Omnibus:                        6.984   Durbin-Watson:                   2.451
Prob(Omnibus):                  0.030   Jarque-Bera (JB):                6.309
Skew:                           0.877   Prob(JB):                       0.0427
Kurtosis:                       3.274   Cond. No.                         13.8
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correct
ly specified.
```

Interpretation:

Using data from the Indian Premier League (IPL), the regression study employed Ordinary Least Squares (OLS) to investigate the link between the independent variable 'wicket_confirmation' and the dependent variable 'Rs' (salary in Indian Rupees). The findings suggested that the variable 'wicket_confirmation' would somewhat help IPL players' salaries. The low 'R-squared' value, however, raises the possibility that it only partially explains the variation in salaries. The 'wicket_confirmation' variable's marginal significance (P>|t| = 0.0610) suggests that further information or new variables would be needed to offer a more thorough explanation of pay fluctuation in the context of the IPL. The model may not be statistically significant at the standard significance level of 0.05, according to the F-statistic of 3.688 and the probability (Prob) of 0.0610.

Recommendations

A study that examined the relationship between IPL players' pay and performance metrics using R and Python has produced a number of recommendations for stakeholders, club management, and cricket experts. In order to calculate player pay, key performance indicators (KPIs) including total runs scored and balls fouled are essential. Higher run scorers typically demand bigger salaries, and there is a strong correlation between income and the quantity of balls faced.

Although {Balls_Faced} and {Total_Runs} have a considerable impact on pay, regression research showed that their combined influence only explains a sizable portion of the difference in wage. This implies that other factors can potentially affect player compensation. The accuracy of compensation calculation algorithms might be raised by adding contextual elements or additional performance indicators.

Performance metrics may be used to find underappreciated players that make a big difference in the team's output relative to their salary and use that information to inform strategic decisions. Data-driven contract negotiations may guarantee equitable remuneration according to on-field performance, promoting trust and satisfaction between players and their representatives.

Additional components, longitudinal study, fan interaction and sponsorship, and ongoing use of analytics are some of the future research objectives. These perceptions can assist the Ams in strengthening their competitive position in the league, enhancing player satisfaction, and optimising squad combinations.

In conclusion, it is critical for IPL team management, stakeholders, and cricket pundits to comprehend the relationship between IPL player performance and salary. Regression analysis and correlation studies provide information that is used to influence data-driven decisions in player acquisition, contract negotiations, and club strategy formulation.

References

1. www.github.com
2. www.geeksforgeeks.com
3. www.datacamp.com