VIRGINIA COMMONWEALTH UNIVERSITY

Statistical analysis and modelling (SCMA 632)

A2a: Multiple regression analysis and diagnostics of data

PUSHPAK DEVAKI

V01108254

Date of Submission: 23-06-2024

CONTENTS

Sl. No.	**Title**	**Page No.**
1.	Introduction	**1**
2.	Results	**1**
3.	Interpretations	**1**

INTRODUCTION

This analysis focuses on analyzing IPL cricket data to extract valuable insights into player performances and their financial rewards. Using R/Python, powerful statistical programming languages, the dataset from IPL organizers will be cleaned and organized round-wise to include detailed statistics such as batsman, ball, runs, and wickets per player per match. The analysis aims to identify the top three run-getters and top three wicket-takers in each IPL round. By fitting the most appropriate statistical distributions for the runs scored and wickets taken by these top performers over the last three IPL tournaments, we will gain a deeper understanding of performance patterns. Additionally, the project will investigate the relationship between players' on-field performance and their salaries, exploring how remuneration correlates with cricket contributions.

OBJECTIVES

a) To perform the multiple regression analysis and carry out the regression diagnostics.
b) To find the appropriate results and explain.

RESULTS & INTERPRETATION

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

<u>Code (R):</u>

```
# Group data by season, innings, striker, and bowler

grouped_data <- df_ipl %>%

  group_by(Season, Innings_No, Striker, Bowler) %>%

  summarise (runs_scored = sum(runs_scored), wicket_confirmation =
sum(wicket_confirmation))

# Fit linear regression model

model <- lm(Rs ~ runs_scored, data = df_merged[train_index, ])

summary(model)

# Repeat the process for wickets

df_salary$Matched_Player <- sapply(df_salary$Player, function(x) match_names(x,
total_wicket_each_year$Bowler))
```

```
df_merged <- merge(df_salary, total_wicket_each_year, by.x = "Matched_Player", by.y = "Bowler")

df_merged <- df_merged %>% filter(Season %in% c("2022"))

set.seed(42)

train_index <- createDataPartition(df_merged$Rs, p = 0.8, list = FALSE)

X_train <- df_merged[train_index, "wicket_confirmation"]

y_train <- df_merged[train_index, "Rs"]

X_test <- df_merged[-train_index, "wicket_confirmation"]

y_test <- df_merged[-train_index, "Rs"]

model <- lm(Rs ~ wicket_confirmation, data = df_merged[train_index, ])

summary(model)
```

Code (Python):

```
# Unique states

print(data['state_1'].unique())


# Impute missing values with mean

subset_data['Education'].fillna(subset_data['Education'].mean(), inplace=True)

print(subset_data['Education'].isna().sum())

# Fit the regression model

model = ols('foodtotal_q ~ MPCE_MRP + MPCE_URP + Age + Meals_At_Home + Possess_ration_card + Education', data=subset_data).fit()

# Print the regression results

print(model.summary())
```

Result:

```
 OLS Regression Results
==============================================================================
Dep. Variable:          foodtotal_q   R-squared:                    0.286
Model:                         OLS   Adj. R-squared:              0.277
Method:              Least Squares   F-statistic:                  29.31
Date:             Sun, 23 Jun 2024   Prob (F-statistic):         1.60e-29
Time:                     20:52:19   Log-Likelihood:             -1396.0
No. Observations:              445   AIC:                         2806.
Df Residuals:                  438   BIC:                         2835.
Df Model:                        6
Covariance Type:         nonrobust
```

```
==============================================================================
======
                 coef    std err          t      P>|t|      [0.025      0.975]
------------------------------------------------------------------------------
Intercept              9.6802      3.891      2.488      0.013       2.033      17.328
MPCE_MRP               0.0019      0.000      7.703      0.000       0.001       0.002
MPCE_URP              -0.0001      0.000     -0.725      0.469      -0.000       0.000
Age                    0.0038      0.023      0.163      0.871      -0.042       0.050
Meals_At_Home          0.1296      0.051      2.520      0.012       0.029       0.231
Possess_ration_card   -2.2873      1.379     -1.658      0.098      -4.998       0.423
Education              0.2469      0.093      2.649      0.008       0.064       0.430
==============================================================================
Omnibus:                       42.314   Durbin-Watson:                   1.685
Prob(Omnibus):                  0.000   Jarque-Bera (JB):              122.715
Skew:                           0.423   Prob(JB):                     2.25e-27
Kurtosis:                       5.429   Cond. No.                     7.72e+04
==============================================================================

Notes:
[1] Standard Errors assume that the covariance matrix of the errors is correctly specified.
[2] The condition number is large, 7.72e+04. This might indicate that there are
strong multicollinearity or other numerical problems.
```

Interpretation:

The OLS regression model indicates that "MPCE_MRP," "Meals_At_Home," and "Education" are significant predictors of "foodtotal_q" with p-values less than 0.05. The positive coefficients suggest that increases in these variables are associated with increases in "foodtotal_q." The R-squared value of 0.286 indicates that the model explains 28.6% of the variability in "foodtotal_q." Significant issues include potential multicollinearity (condition number = 7.72e+04) and non-normal residuals (highly significant Omnibus and Jarque-Bera tests). The Durbin-Watson statistic (1.685) suggests mild autocorrelation. Overall, the model has moderate explanatory power but may need adjustments to address multicollinearity and residual normality issues.