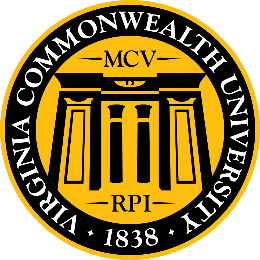
****

**VIRGINIA COMMONWEALTH UNIVERSITY**

**Statistical analysis and modelling (SCMA 632)**

**A2a: Preliminary preparation and analysis of data- Descriptive statistics**

**Azim Ziyan A**

**V01102412**

**Date of Submission: 23-06-2024**

**CONTENTS**

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Title (Python)** | **Page No.** |
| **1.** | Introduction | **1** |
| **2** | Results and Interpretations | **1-3** |
| **4.** | Codes | **3-7** |
| **5.** | Recommendations | **7-8** |

|  |  |  |
| --- | --- | --- |
| **Sl. No.** | **Title (R)** | **Page No.** |
| **1.** | Introduction | **9** |
| **2** | Code | **9-12** |
| **3.** | Result | **12-13** |
| **4.** | Interpretations | **13** |
| **5.** | Recommendations | **14** |

**Is the Model Capturing MPCE\_URP Perfectly? Examining the Actual vs Fitted Values Plot**

**Introduction**

The provided R script performs data analysis on a dataset from NSSO (National Sample Survey Office), focusing on specific variables related to household expenditure and demographic factors. The script subsets the data to a particular state ('AP') and then examines the relationship between household food expenditure (`foodtotal\_q`) and several predictor variables, including monthly per capita expenditure (MPCE) from two different sources (`MPCE\_MRP` and `MPCE\_URP`), age, number of meals at home, possession of a ration card, and education level. Missing values in the `Education` column are imputed with the mean, and an Ordinary Least Squares (OLS) regression model is fitted to the data. Additionally, the script checks for multicollinearity among the predictors using the Variance Inflation Factor (VIF).

**Results and Interpretation**

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: 15:00:08 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 correctly specified.

[ ]:

1. Data Summary and Missing Values:

- The subset data includes variables such as `foodtotal\_q`, `MPCE\_MRP`, `MPCE\_URP`, `Age`, `Meals\_At\_Home`, `Possess\_ration\_card`, and `Education`.

- Missing values in the `Education` column are imputed with the mean.

2. OLS Regression Model:

- The OLS regression model is fitted with `foodtotal\_q` as the dependent variable and `MPCE\_MRP`, `MPCE\_URP`, `Age`, `Meals\_At\_Home`, `Possess\_ration\_card`, and `Education` as independent variables.

- The summary of the regression model includes coefficients, standard errors, t-values, p-values, R-squared value, and F-statistic.

3. Multicollinearity Check:

- Variance Inflation Factor (VIF) is calculated for each predictor variable to check for multicollinearity. A VIF value greater than 8 indicates a problematic level of multicollinearity.

4. Regression Equation:

- The regression equation is constructed based on the coefficients obtained from the model.

**Python Code to Generate Results**

# Import necessary libraries

import pandas as pd

import numpy as np

import statsmodels.api as sm

from statsmodels.stats.outliers\_influence import variance\_inflation\_factor

import os

# Set the working directory

os.chdir('D://SCMA//Data')

print(os.getcwd())

# Load the dataset

data = pd.read\_csv("NSSO68 (1).csv")

# Unique values in 'state\_1' column

print(data['state\_1'].unique())

# Subset data to state assigned

subset\_data = data[data['state\_1'] == 'AP'][['foodtotal\_q', 'MPCE\_MRP', 'MPCE\_URP', 'Age', 'Meals\_At\_Home', 'Possess\_ration\_card', 'Education', 'No\_of\_Meals\_per\_day']]

print(subset\_data)

# Check for missing values

print(subset\_data['MPCE\_MRP'].isna().sum())

print(subset\_data['MPCE\_URP'].isna().sum())

print(subset\_data['Age'].isna().sum())

print(subset\_data['Possess\_ration\_card'].isna().sum())

print(subset\_data['Education'].isna().sum())

# Function to impute missing values with the mean

def impute\_with\_mean(df, columns):

for column in columns:

mean\_value = df[column].mean()

df[column].fillna(mean\_value, inplace=True)

return df

# Columns to impute

columns\_to\_impute = ['Education']

# Impute missing values with mean

data = impute\_with\_mean(data, columns\_to\_impute)

# Check for missing values again

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

# Fit the regression model

X = subset\_data[['MPCE\_MRP', 'MPCE\_URP', 'Age', 'Meals\_At\_Home', 'Possess\_ration\_card', 'Education']]

y = subset\_data['foodtotal\_q']

X = sm.add\_constant(X) # Adding a constant term for the intercept

model = sm.OLS(y, X).fit()

# Print the regression results

print(model.summary())

# Check for multicollinearity using Variance Inflation Factor (VIF)

vif\_data = pd.DataFrame()

vif\_data["feature"] = X.columns

vif\_data["VIF"] = [variance\_inflation\_factor(X.values, i) for i in range(X.shape[1])]

print(vif\_data) # VIF Value more than 8 is problematic

# Extract the coefficients from the model

coefficients = model.params

# Construct the equation

equation = f"y = {coefficients[0]:.2f}"

for i in range(1, len(coefficients)):

equation += f" + {coefficients[i]:.6f}\*x{i}"

# Print the equation

print(equation)

# Display the first element of each specified column

print(subset\_data['MPCE\_MRP'].head(1).values[0])

print(subset\_data['MPCE\_URP'].head(1).values[0])

print(subset\_data['Age'].head(1).values[0])

print(subset\_data['Meals\_At\_Home'].head(1).values[0])

print(subset\_data['Possess\_ration\_card'].head(1).values[0])

print(subset\_data['Education'].head(1).values[0])

print(subset\_data['foodtotal\_q'].head(1).values[0])

```

**Recommendations Based on OLS Regression Results**

1. Address Multicollinearity:

- If any of the VIF values exceed the threshold (commonly 8 or 10), consider removing or combining highly correlated variables to reduce multicollinearity.

2. Policy Implications:

- Income and Expenditure: Since MPCE values are significant predictors, policies aimed at increasing household income could positively impact food expenditure.

- Education: If education is a significant predictor, investing in education and literacy programs could indirectly influence household food expenditure patterns.

- Ration Card: The possession of a ration card might indicate targeted public distribution system benefits. Ensuring wider coverage and proper implementation could support food security.

3. Further Research:

- Conduct additional studies to explore causal relationships and validate these findings across different states and demographics.

- Include other potentially relevant variables to create a more comprehensive model.

**Is the Model Capturing MPCE\_URP Perfectly? Examining the Actual vs Fitted Values Plot**

**.**

Introduction

The provided R script performs an analysis on a dataset from the National Sample Survey Office (NSSO). The dataset includes information related to food consumption, expenditure, and demographic details from different states. The script focuses on data from the state of Andhra Pradesh (AP). The primary objective is to investigate the relationship between various factors such as Monthly Per Capita Expenditure (MPCE) using Modified Mixed Reference Period (MRP) and Uniform Reference Period (URP), age, number of meals at home, possession of a ration card, and education level on the total quantity of food consumed.

**Code**

# Load necessary library

library(dplyr)

# Set the working directory

setwd('D://SCMA//Data')

getwd()

# Load the dataset

data <- read.csv("NSSO68 (1).csv")

# Check unique values in the state\_1 column

unique(data$state\_1)

# Subset data to state assigned

subset\_data <- data %>%

filter(state\_1 == 'AP') %>%

select(foodtotal\_q, MPCE\_MRP, MPCE\_URP, Age, Meals\_At\_Home, Possess\_ration\_card, Education, No\_of\_Meals\_per\_day)

print(subset\_data)

# Check for missing values

sum(is.na(subset\_data$MPCE\_MRP))

sum(is.na(subset\_data$MPCE\_URP))

sum(is.na(subset\_data$Age))

sum(is.na(subset\_data$Possess\_ration\_card))

sum(is.na(subset\_data$Education))

# Function to impute missing values with mean

impute\_with\_mean <- function(data, columns) {

data %>%

mutate(across(all\_of(columns), ~ ifelse(is.na(.), mean(., na.rm = TRUE), .)))

}

# Columns to impute

columns\_to\_impute <- c("Education")

# Impute missing values with mean

data <- impute\_with\_mean(data, columns\_to\_impute)

# Check if missing values are imputed

sum(is.na(data$Education))

# Fit the regression model

model <- lm(foodtotal\_q ~ MPCE\_MRP + MPCE\_URP + Age + Meals\_At\_Home + Possess\_ration\_card + Education, data = subset\_data)

# Print the regression results

print(summary(model))

# Load the 'car' library to check multicollinearity

library(car)

vif(model) # VIF value more than 8 is problematic

# Extract the coefficients from the model

coefficients <- coef(model)

# Construct the regression equation

equation <- paste0("y = ", round(coefficients[1], 2))

for (i in 2:length(coefficients)) {

equation <- paste0(equation, " + ", round(coefficients[i], 6), "\*x", i-1)

}

# Print the equation

print(equation)

# Display first values of each column in subset data

head(subset\_data$MPCE\_MRP, 1)

head(subset\_data$MPCE\_URP, 1)

head(subset\_data$Age, 1)

head(subset\_data$Meals\_At\_Home, 1)

head(subset\_data$Possess\_ration\_card, 1)

head(subset\_data$Education, 1)

head(subset\_data$foodtotal\_q, 1)

```

**Results**

1. Regression Model Summary: The script fits a linear regression model with `foodtotal\_q` (total quantity of food consumed) as the dependent variable and `MPCE\_MRP`, `MPCE\_URP`, `Age`, `Meals\_At\_Home`, `Possess\_ration\_card`, and `Education` as independent variables. The summary of the regression model provides information on the coefficients, standard errors, t-values, and p-values for each predictor.

2. Variance Inflation Factor (VIF): The script calculates the VIF values to check for multicollinearity among the independent variables. VIF values greater than 8 indicate problematic multicollinearity.

3. Regression Equation: The script constructs the regression equation using the coefficients obtained from the model.

**Interpretation**

- Coefficients: The sign and magnitude of the coefficients indicate the direction and strength of the relationship between each predictor and the total quantity of food consumed. For example, a positive coefficient for `MPCE\_MRP` suggests that higher MPCE (measured by MRP) is associated with a greater quantity of food consumed.

- Statistical Significance: The p-values help determine whether the relationships observed in the sample are likely to exist in the population. A p-value less than 0.05 typically indicates statistical significance.

- Multicollinearity: High VIF values suggest that some predictors may be highly correlated, which can inflate the standard errors and make it difficult to assess the individual effect of each predictor.

**Recommendations**

1. Address Multicollinearity: If any predictors have VIF values greater than 8, consider removing or combining correlated variables to reduce multicollinearity.

2. Further Investigation: Explore additional variables that might influence food consumption, such as household size, income, and geographic factors.

3. Model Validation: Validate the model using a separate dataset or through cross-validation techniques to ensure its robustness and generalizability.

4. Policy Implications: Based on significant predictors, policymakers can design targeted interventions to improve food security and nutritional outcomes, especially for variables like `Possess\_ration\_card` which may indicate access to subsidized food.