Household Size Analysis using GLM

Group1

2025-03-23

1. Data Exploration & Preprocessing

Load and Inspect Data

```
# Read the dataset
df <- read.csv("dataset01.csv")

# Display structure
str(df)
```

```
'data.frame':   1725 obs. of  11 variables:
 $ Total.Household.Income        : int  480332 198235 82785 107589 189322 152883 198621 13490
 $ Region                        : chr  "CAR" "CAR" "CAR" "CAR" ...
 $ Total.Food.Expenditure        : int  117848 67766 61609 78189 94625 73326 104644 95644 673
 $ Household.Head.Sex            : chr  "Female" "Male" "Male" "Male" ...
 $ Household.Head.Age            : int  49 40 39 52 65 46 45 33 17 53 ...
 $ Type.of.Household             : chr  "Extended Family" "Single Family" "Single Family" "Si
 $ Total.Number.of.Family.members: int  4 3 6 3 4 4 5 5 2 6 ...
 $ House.Floor.Area              : int  80 42 35 30 54 40 35 35 35 70 ...
 $ House.Age                     : int  75 15 12 15 16 7 18 48 8 12 ...
 $ Number.of.bedrooms            : int  3 2 1 1 3 2 1 2 1 3 ...
 $ Electricity                   : int  1 1 0 1 1 1 1 1 1 1 ...
```

```
# Show first few rows
head(df)
```

```
  Total.Household.Income Region Total.Food.Expenditure Household.Head.Sex
1                 480332    CAR                 117848             Female
```

```
2                  198235    CAR                   67766               Male
3                   82785    CAR                   61609               Male
4                  107589    CAR                   78189               Male
5                  189322    CAR                   94625               Male
6                  152883    CAR                   73326               Male
  Household.Head.Age Type.of.Household Total.Number.of.Family.members
1                 49   Extended Family                              4
2                 40     Single Family                              3
3                 39     Single Family                              6
4                 52     Single Family                              3
5                 65     Single Family                              4
6                 46     Single Family                              4
  House.Floor.Area House.Age Number.of.bedrooms Electricity
1               80        75                  3           1
2               42        15                  2           1
3               35        12                  1           0
4               30        15                  1           1
5               54        16                  3           1
6               40         7                  2           1
```

Convert Engel's Coefficient:

```
attach(df)
cor(Total.Food.Expenditure,Total.Household.Income)
```

```
[1] 0.6114945
```

Since the "Total.Food.Expenditure" and "Total.Household.Income" has strong linear relationship, so we use Engel's Coefficient(Total.Food.Expenditure/Total.Household.Income) to summarise the two variables and avoid multicollinearity

```
df$engel <- df$Total.Food.Expenditure / df$Total.Household.Income
```

Convert Binary Variables:

We converted **binary categorical variables** into numerical format to ensure compatibility with GLM, which requires numeric input for continuous predictors. Encoding **Household.Head.Sex** as 0 and 1 allows the model to interpret its effect, while **Electricity** is kept as a factor to treat it as a categorical variable with distinct levels.

```r
# Convert Household.Head.Sex to binary: Male = 1, Female = 0
df$Household.Head.Sex <- ifelse(df$Household.Head.Sex == "Male", 1, 0)

# Ensure Electricity is treated as a factor
df$Electricity <- as.factor(df$Electricity)

# Display summary of modified dataset
summary(df)
```

```
 Total.Household.Income    Region           Total.Food.Expenditure
 Min.   :  11988        Length:1725        Min.   :  6781        
 1st Qu.: 118565        Class :character   1st Qu.: 51922        
 Median : 188580        Mode  :character   Median : 73578        
 Mean   : 269540                           Mean   : 80353        
 3rd Qu.: 328335                           3rd Qu.: 98493        
 Max.   :6042860                           Max.   :327724        
 Household.Head.Sex Household.Head.Age Type.of.Household 
 Min.   :0.0000     Min.   :17.00      Length:1725       
 1st Qu.:1.0000     1st Qu.:41.00      Class :character  
 Median :1.0000     Median :52.00      Mode  :character  
 Mean   :0.7861     Mean   :52.23                        
 3rd Qu.:1.0000     3rd Qu.:63.00                        
 Max.   :1.0000     Max.   :99.00                        
 Total.Number.of.Family.members House.Floor.Area   House.Age     
 Min.   : 1.000                 Min.   :  5.00   Min.   :  0.00  
 1st Qu.: 3.000                 1st Qu.: 32.00   1st Qu.: 12.00  
 Median : 4.000                 Median : 54.00   Median : 20.00  
 Mean   : 4.669                 Mean   : 90.92   Mean   : 22.98  
 3rd Qu.: 6.000                 3rd Qu.:102.00   3rd Qu.: 31.00  
 Max.   :15.000                 Max.   :900.00   Max.   :100.00  
 Number.of.bedrooms Electricity     engel        
 Min.   :0.000      0: 129      Min.   :0.02482  
 1st Qu.:1.000      1:1596      1st Qu.:0.25656  
 Median :2.000                  Median :0.36782  
 Mean   :2.259                  Mean   :0.39916  
 3rd Qu.:3.000                  3rd Qu.:0.51517  
 Max.   :9.000                  Max.   :1.19637  
```

Check for Missing Values & Handle Missing Data

Checking for missing values is essential to ensure **data quality and model reliability**. Missing data can **bias results**, reduce **statistical power**, or cause models like GLM to **fail or produce inaccurate estimates**. Proper handling (e.g., removal, imputation) ensures the dataset is **clean, complete, and ready for analysis**, allowing for more **trustworthy and interpretable** conclusions.

In this case, no missing values were found, so the dataset was ready for modeling without further preprocessing.

```
# Check for missing values in the dataset
colSums(is.na(df))
```

```
        Total.Household.Income                         Region
                             0                              0
        Total.Food.Expenditure             Household.Head.Sex
                             0                              0
            Household.Head.Age              Type.of.Household
                             0                              0
Total.Number.of.Family.members               House.Floor.Area
                             0                              0
                     House.Age             Number.of.bedrooms
                             0                              0
                   Electricity                          engel
                             0                              0
```

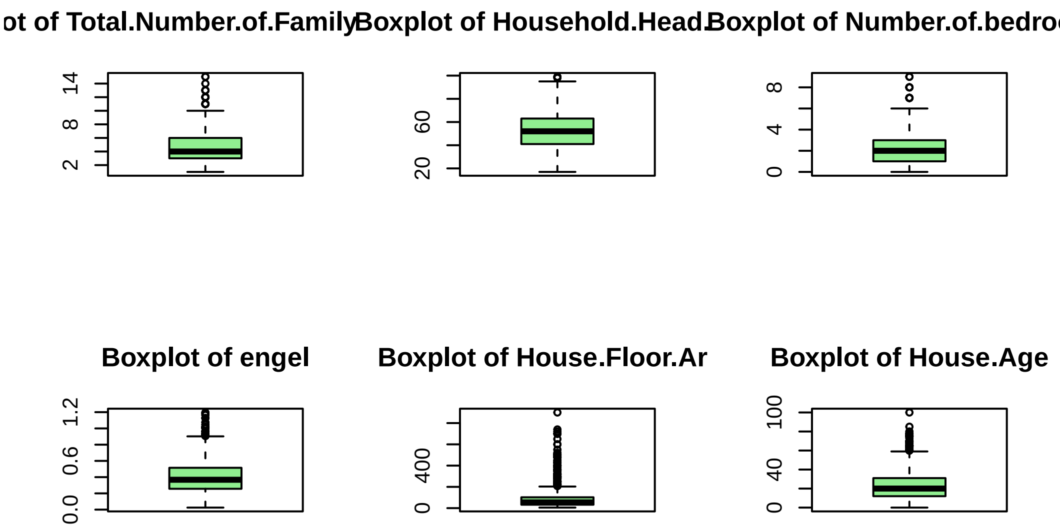
```
# Impute missing values using median (for numerical variables)
df[is.na(df)] <- lapply(df, function(x) ifelse(is.numeric(x), median(x, na.rm = TRUE), x))
```

Data Visualization

Histograms & Boxplots

Histograms help visualize the **distribution and skewness** of numerical variables, while **boxplots** reveal **outliers, spread, and central tendency**. Together, they provide a quick, intuitive understanding of the data's shape, variability, and potential issues before modeling.

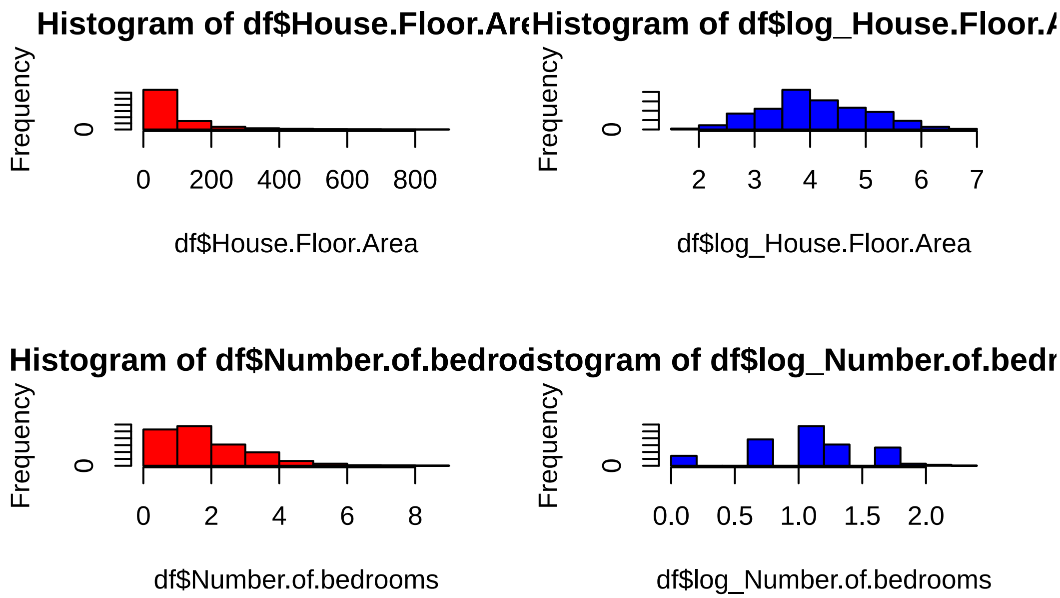
```
# Boxplots for numerical variables
par(mfrow = c(2, 3))
numeric_cols <- c("Total.Number.of.Family.members","Household.Head.Age","Number.of.bedrooms"
                  "House.Floor.Area", "House.Age")

for (col in numeric_cols) {
  boxplot(df[[col]], main = paste("Boxplot of", col), col = "lightgreen", border = "black")
}
```




```
# Histogram for numerical variables
par(mfrow=c(2,3))

hist(df$Total.Number.of.Family.members, col="lightblue", main="Family Members", xlab="Count")
hist(df$Household.Head.Age, col="lightgreen", main="Household Head Age", xlab="Age")
hist(df$Number.of.bedrooms, col="lightcoral", main="Bedrooms", xlab="Number of Bedrooms")
hist(df$House.Floor.Area, col="gold", main="Floor Area", xlab="Square Meters")
hist(df$House.Age, col="purple", main="House Age", xlab="Years")
hist(df$engel, col="pink", main="Engel's Coefficient", xlab="Engel's Ratio")
```

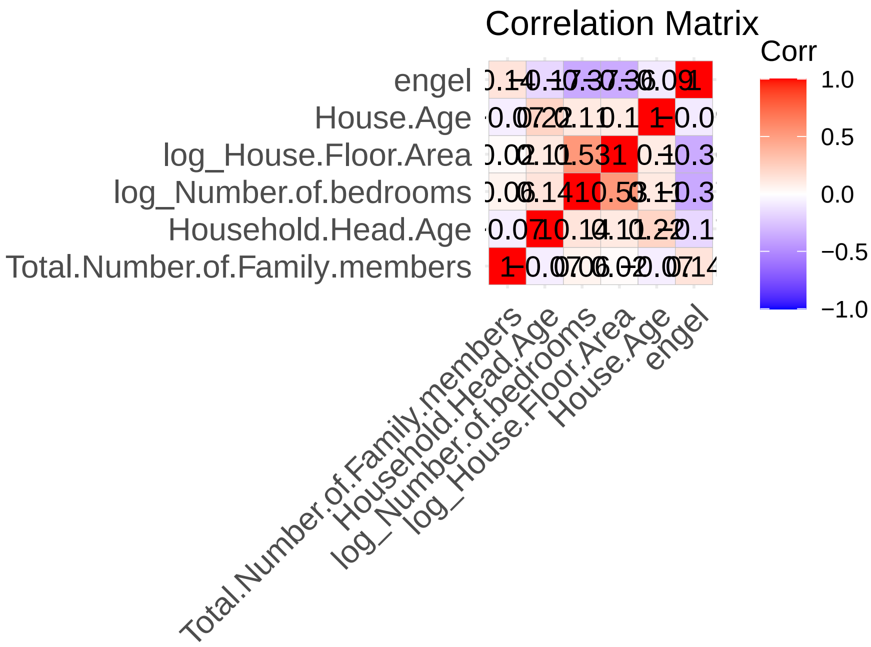
From the boxplots and histogram we can see "Number.of.bedrooms" and "House.Floor.Area" have many outliers and their skewness are quite big. So, we use log transformations to "Number.of.bedrooms" and "House.Floor.Area" to reduce skewness, normalizes distributions, and minimizes outliers' influence.

```
df$log_House.Floor.Area<- log(df$House.Floor.Area)
df$log_Number.of.bedrooms<- log(df$Number.of.bedrooms+1)

par(mfrow=c(2,2))

hist(df$House.Floor.Area, col="red")
hist(df$log_House.Floor.Area, col="blue")
hist(df$Number.of.bedrooms, col="red")
hist(df$log_Number.of.bedrooms, col="blue")
```



Correlation Matrix Using ggplot

```
num_vars <- df[, c("Total.Number.of.Family.members","Household.Head.Age",
                   "log_Number.of.bedrooms","log_House.Floor.Area",
                   "House.Age", "engel")]


cor_matrix <- cor(num_vars, use="complete.obs", method="pearson")

ggcorrplot(cor_matrix,
           lab=TRUE,
           colors = c("blue", "white", "red"),
           title = "Correlation Matrix")
```



Key Interpretation of (Total.Number.of.Family.members) Relationships

- A **weak positive correlation with (Engle's coefficient) (0.14)** suggests that higher ratio of food expenditure is associated with larger households.
- Other variables'impacts are minimal.

2.Household Size and Its Determinants: A GLM Approach

Model 1: Poisson Regression

```
# Fit Poisson GLM using log-transformed predictors
poisson_model <- glm(Total.Number.of.Family.members ~ Household.Head.Age+log_Number.of.bedro
                    House.Age+engel+Electricity,
                    family = poisson(link = "log"),
                    data = df)

# View model summary
summary(poisson_model)
```

```
Call:
```

```
glm(formula = Total.Number.of.Family.members ~ Household.Head.Age +
    log_Number.of.bedrooms + log_House.Floor.Area + House.Age +
    engel + Electricity, family = poisson(link = "log"), data = df)

Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
(Intercept)             1.0800356  0.0914195  11.814  < 2e-16 ***
Household.Head.Age     -0.0011920  0.0008056  -1.480 0.138961
log_Number.of.bedrooms  0.1105076  0.0296748   3.724 0.000196 ***
log_House.Floor.Area    0.0184744  0.0152487   1.212 0.225688
House.Age              -0.0022819  0.0007681  -2.971 0.002968 **
engel                   0.4876682  0.0631490   7.723 1.14e-14 ***
Electricity1            0.1944628  0.0474577   4.098 4.17e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for poisson family taken to be 1)

    Null deviance: 2024.4  on 1724  degrees of freedom
Residual deviance: 1925.5  on 1718  degrees of freedom
AIC: 7602

Number of Fisher Scoring iterations: 5
```

Model 2: Negative Binomial Regression

```
neg_bin_model <- glm.nb(Total.Number.of.Family.members ~ Household.Head.Age+log_Number.of.bed
                   House.Age+engel+Electricity,
                   data = df)
  summary(neg_bin_model)
```

```
Call:
glm.nb(formula = Total.Number.of.Family.members ~ Household.Head.Age +
    log_Number.of.bedrooms + log_House.Floor.Area + House.Age +
    engel + Electricity, data = df, init.theta = 44.86099906,
    link = log)

Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
(Intercept)             1.0787239  0.0958501  11.254  < 2e-16 ***
Household.Head.Age     -0.0012207  0.0008463  -1.442 0.149188
```

```
log_Number.of.bedrooms  0.1103329  0.0311407   3.543 0.000396 ***
log_House.Floor.Area    0.0188573  0.0160210   1.177 0.239180
House.Age              -0.0022904  0.0008059  -2.842 0.004484 **
engel                   0.4893206  0.0665531   7.352 1.95e-13 ***
Electricity1            0.1954781  0.0495833   3.942 8.07e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(44.861) family taken to be 1)

    Null deviance: 1836.8  on 1724  degrees of freedom
Residual deviance: 1746.8  on 1718  degrees of freedom
AIC: 7594.8

Number of Fisher Scoring iterations: 1

              Theta:  44.9
          Std. Err.:  15.9

 2 x log-likelihood:  -7578.82
```

3.Comparative Analysis of GLM Models

```r
# Create AIC comparison table (excluding Quasi-Poisson)
aic_values <- data.frame(
  Model = c("Poisson", "Negative Binomial"),
  AIC = c(AIC(poisson_model), AIC(neg_bin_model))
)

# View AIC values
aic_values
```

```
              Model      AIC
1           Poisson 7602.011
2 Negative Binomial 7594.820
```

Since the Negative Binomial model has the smaller AIC, we choose it as our final model.

Formal analysis

```r
summary(neg_bin_model)
```

```
Call:
glm.nb(formula = Total.Number.of.Family.members ~ Household.Head.Age +
    log_Number.of.bedrooms + log_House.Floor.Area + House.Age +
    engel + Electricity, data = df, init.theta = 44.86099906,
    link = log)

Coefficients:
                         Estimate Std. Error z value Pr(>|z|)
(Intercept)             1.0787239  0.0958501  11.254  < 2e-16 ***
Household.Head.Age     -0.0012207  0.0008463  -1.442 0.149188
log_Number.of.bedrooms  0.1103329  0.0311407   3.543 0.000396 ***
log_House.Floor.Area    0.0188573  0.0160210   1.177 0.239180
House.Age              -0.0022904  0.0008059  -2.842 0.004484 **
engel                   0.4893206  0.0665531   7.352 1.95e-13 ***
Electricity1            0.1954781  0.0495833   3.942 8.07e-05 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for Negative Binomial(44.861) family taken to be 1)

    Null deviance: 1836.8  on 1724  degrees of freedom
Residual deviance: 1746.8  on 1718  degrees of freedom
AIC: 7594.8

Number of Fisher Scoring iterations: 1

              Theta:  44.9
          Std. Err.:  15.9

 2 x log-likelihood:  -7578.82
```

Model Formula

We model the expected number of household members (E(Y)) using a **Negative Binomial Generalized Linear Model (GLM)** with a log link function:

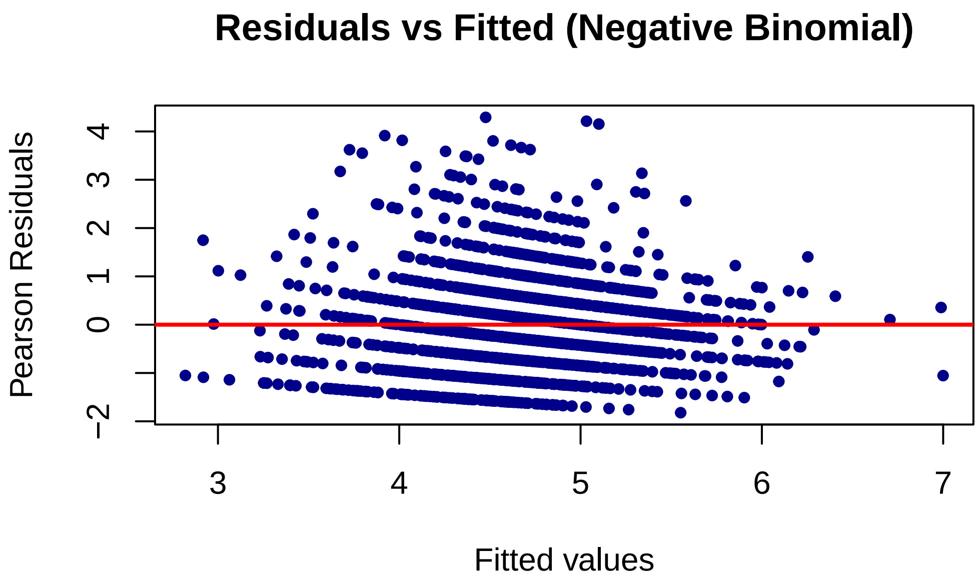
$\log(E(Y)) = \beta_0 + \beta_1 \cdot \text{Household.Head.Age} + \beta_2 \cdot \log(\text{Number.of.bedrooms}) + \beta_3 \cdot \log(\text{House.Floor.Area}) + \beta_4 \cdot \text{House.A}$

Where:

- Y : Total number of family members
- Household.Head.Age: Age of the head of household (in years)
- Number.of.bedrooms: Log-transformed number of bedrooms in the house
- House.Floor.Area: Log-transformed total floor area of the house
- House.Age: Age of the house (in years)
- Engel: Engel ratio (food expenditure divided by income)
- Electricity: Binary variable (1 = electricity available, 0 = no electricity)

Residuals vs fitted

```
# Generate residuals vs fitted plot for Negative Binomial model
library(MASS)

# Assuming your model is named neg_bin_model
residuals_nb <- resid(neg_bin_model, type = "pearson")
fitted_nb <- fitted(neg_bin_model)

# Plot
plot(fitted_nb, residuals_nb,
     xlab = "Fitted values",
     ylab = "Pearson Residuals",
     main = "Residuals vs Fitted (Negative Binomial)",
     pch = 20, col = "darkblue")
abline(h = 0, col = "red", lwd = 2)
```



Coefficients Interpretation

The number of bedrooms, Engel coefficient, house age, and electricity access significantly influence family size. A 1% increase in the number of bedrooms is associated with a 0.11% increase in expected family members. Higher Engel coefficients are linked to larger families, indicating that households spending more on food relative to income tend to be bigger. Homes with electricity have, on average, 21.5% more family members than those without. In contrast, older houses are associated with slightly smaller families, with each additional year reducing expected family size by about 0.23%. The age of the household head and house floor area have no significant effect on family size in this model.

4.Model Assumptions

1.Overdispersion

```
mean_y <- mean(df$Total.Number.of.Family.members)
var_y <- var(df$Total.Number.of.Family.members)
print(paste("Mean:", mean_y, "Variance:", var_y))
```

```
[1] "Mean: 4.66898550724638 Variance: 5.44315007229564"
```

The variance of Total.Number.of.Family.members is bigger than the mean of Total.Number.of.Family.members. So it makes sense to fit the Negative Binomial model instead of poisson model to avoid overdispersion.

2.Independence of Errors

```
dwtest(neg_bin_model)
```

```

	Durbin-Watson test

data:  neg_bin_model
DW = 1.844, p-value = 0.0005338
alternative hypothesis: true autocorrelation is greater than 0
```

The value of Durbin-Watson Test is 1.844(betwwen 0 and 2), so autocorrelation is not a major issue.

3.Multicollinearity

```
print(vif(neg_bin_model))
```

```
    Household.Head.Age log_Number.of.bedrooms   log_House.Floor.Area
              1.087765               1.524012               1.461157
             House.Age                  engel            Electricity
              1.066380               1.262763               1.067128
```

No variables' variance inflation factor is higher than 5. So the model does not have the problem of multicollinearity.

5. Conclusion.

Our investigation sought to determine which household-related variables significantly influence the number of people living in a household. Using a Generalized Linear Model (GLM), specifically a Negative Binomial regression (selected due to overdispersion in the count data), we identified several key factors.

The analysis revealed that:

Number of bedrooms, Engel's coefficient (food expenditure relative to income), age of the house, and electricity access are statistically significant predictors of household size. Households with more bedrooms and greater food expenditure relative to income tend to have more members. Access to electricity is associated with larger household sizes, suggesting links to infrastructure or socioeconomic status. Conversely, older homes tend to house fewer people. The age of the household head and floor area of the house were not found to be significant predictors in this model. These findings provide valuable insights for policymakers. Investments in housing infrastructure, improving household utilities, and understanding economic pressures on food spending may all play a role in addressing housing needs and demographic planning.