Introduction to Data Science – IST 687 (M003)

Final Project Report

(Group-13)

Submitted by
Rushikesh Shinde
Srushti Shobhane
Khushi Shah
Siddhi Kale

Under the guidance of
Prof. A. Kumar
Prof. C. Dunham

Spring 2024

Table of Contents

1. Introduction

1.1 Goal

- Predict energy usage if the summer was 5 degrees warmer.
- To identify three major factors causing the most energy consumption for different counties in a region based on the sqft range between 1690 to 3301.

1.2 Description

eSC, an energy company serving residential properties primarily in South Carolina with a smaller portion in North Carolina, is concerned about the impact of global warming on the electricity demand. With the looming threat of blackouts during peak summer months, particularly in July, eSC aims to mitigate this risk by understanding the key drivers of energy usage and promoting energy-saving practices among its customers. Rather than investing in additional energy production facilities, eSC seeks to reduce energy consumption during periods of high demand, thereby ensuring grid stability and environmental sustainability. By incentivizing customers to save energy, eSC aims to meet demand without compromising service reliability or exacerbating environmental impacts.

2. Data Overview

2.1 Static House Data:

a) Contains basic information about approximately 5,000 single-family houses served by eSC.
b) Information includes unique building IDs, house attributes such as sqft, and other static details.
c) Stored in parquet format, optimized for storage.

2.2 Energy Usage Data (individual files for each house):

a) Hour-by-hour energy usage data collected for each house, with calibrated and validated load profiles.
b) Each file corresponds to a specific house identified by its building ID.
c) Describes energy usage from various sources like air conditioning systems, dryers, etc.
d) Stored in parquet format, with approximately 5,000 individual files in the dataset.

2.3 Meta Data:

a) Describes attributes present in the static house and energy usage data.
b) Offers human-readable information about the fields used across different housing data files.

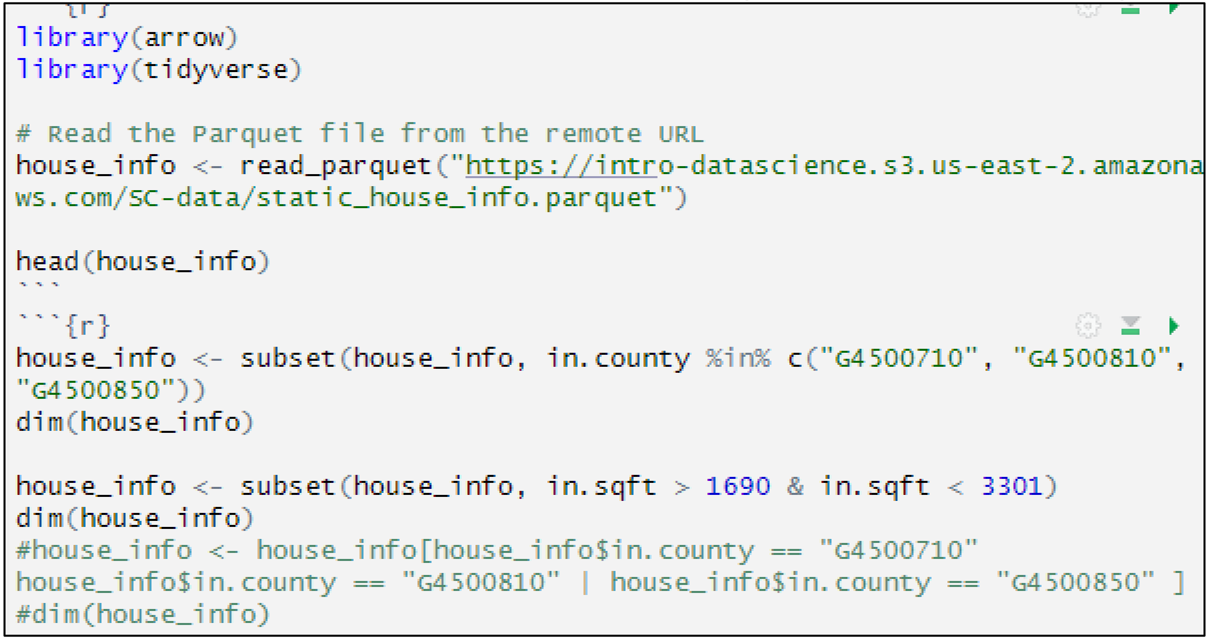
2.4 Weather Data (hourly weather information for each county):

a) Hour-by-hour weather data collected for each county served by eSC, with one file per geographic area.
b) Weather information includes parameters like temperature, humidity, etc.
c) Each file corresponds to a county identified by its county code.
d) Stored in CSV format, with approximately 50 files for the 50 counties in the dataset.

3. Detailed Tasks Overview

3.1 Data Preparation:

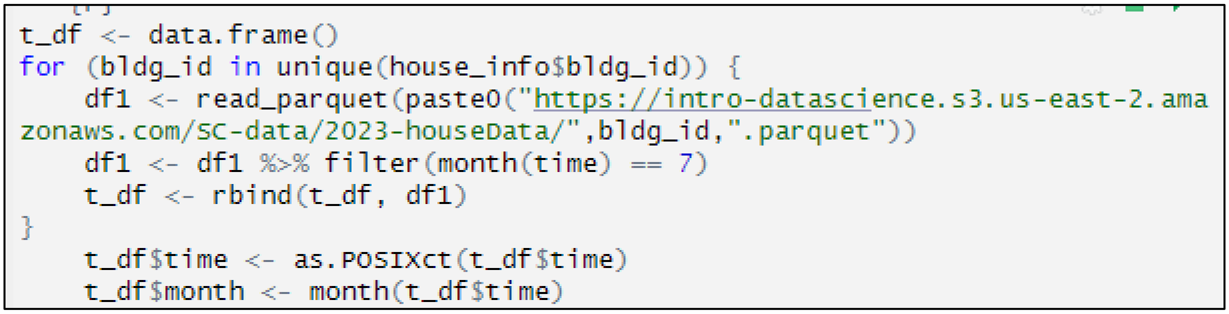
a) Read static house data and subset the data according to counties and sqft range.



```
library(arrow)
library(tidyverse)

# Read the Parquet file from the remote URL
house_info <- read_parquet("https://intro-datascience.s3.us-east-2.amazona
ws.com/SC-data/static_house_info.parquet")

head(house_info)
```
```{r}
house_info <- subset(house_info, in.county %in% c("G4500710", "G4500810",
"G4500850"))
dim(house_info)

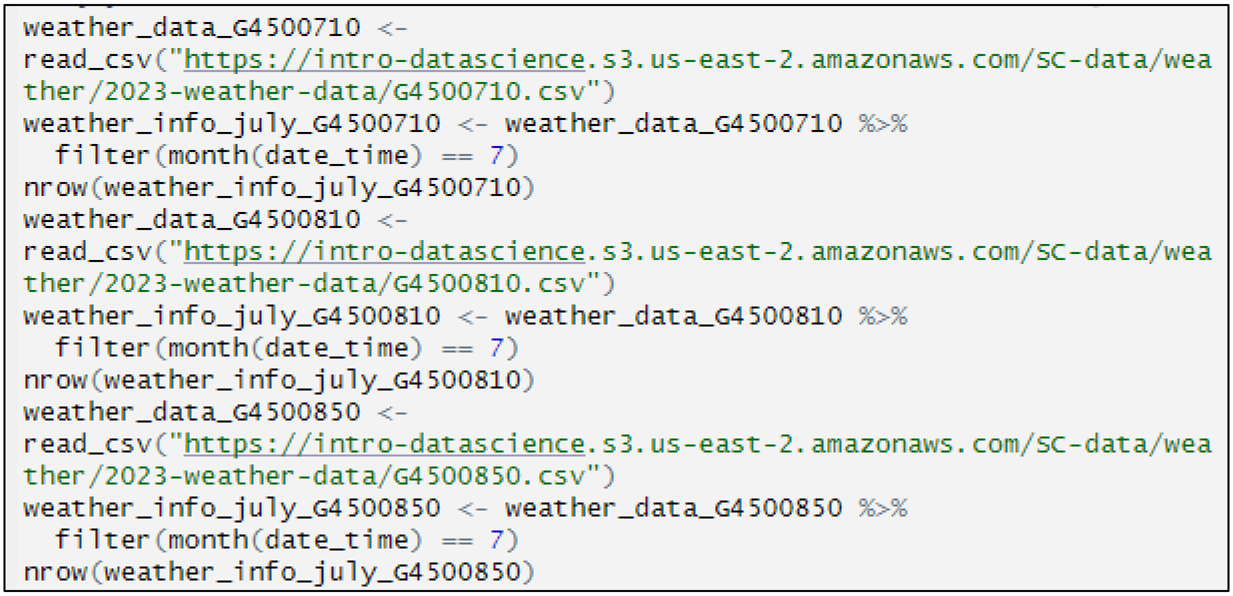
house_info <- subset(house_info, in.sqft > 1690 & in.sqft < 3301)
dim(house_info)
#house_info <- house_info[house_info$in.county == "G4500710"
house_info$in.county == "G4500810" | house_info$in.county == "G4500850" ]
#dim(house_info)
```

b) Read energy usage data for each building ID and subset it for July.



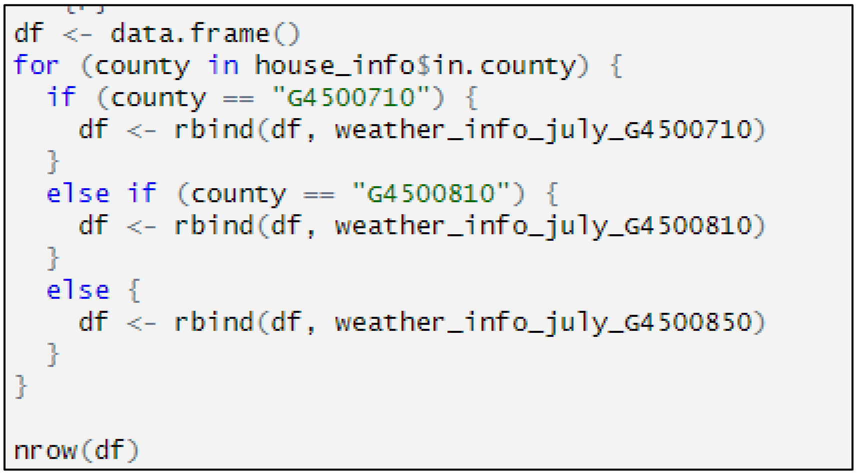
```
t_df <- data.frame()
for (bldg_id in unique(house_info$bldg_id)) {
    df1 <- read_parquet(paste0("https://intro-datascience.s3.us-east-2.ama
zonaws.com/SC-data/2023-houseData/",bldg_id,".parquet"))
    df1 <- df1 %>% filter(month(time) == 7)
    t_df <- rbind(t_df, df1)
}
    t_df$time <- as.POSIXct(t_df$time)
    t_df$month <- month(t_df$time)
```

c) Read weather data for each county and subset it for July.



```
weather_data_G4500710 <-
read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/wea
ther/2023-weather-data/G4500710.csv")
weather_info_july_G4500710 <- weather_data_G4500710 %>%
  filter(month(date_time) == 7)
nrow(weather_info_july_G4500710)
weather_data_G4500810 <-
read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/wea
ther/2023-weather-data/G4500810.csv")
weather_info_july_G4500810 <- weather_data_G4500810 %>%
  filter(month(date_time) == 7)
nrow(weather_info_july_G4500810)
weather_data_G4500850 <-
read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/wea
ther/2023-weather-data/G4500850.csv")
weather_info_july_G4500850 <- weather_data_G4500850 %>%
  filter(month(date_time) == 7)
nrow(weather_info_july_G4500850)
```

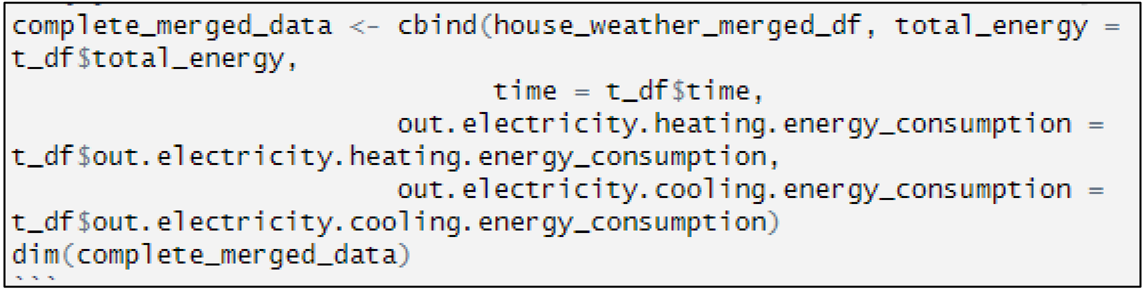
d) Combining the rows of weather data.



```
df <- data.frame()
for (county in house_info$in.county) {
  if (county == "G4500710") {
    df <- rbind(df, weather_info_july_G4500710)
  }
  else if (county == "G4500810") {
    df <- rbind(df, weather_info_july_G4500810)
  }
  else {
    df <- rbind(df, weather_info_july_G4500850)
  }
}

nrow(df)
```

e) Merging the weather and energy usage data.

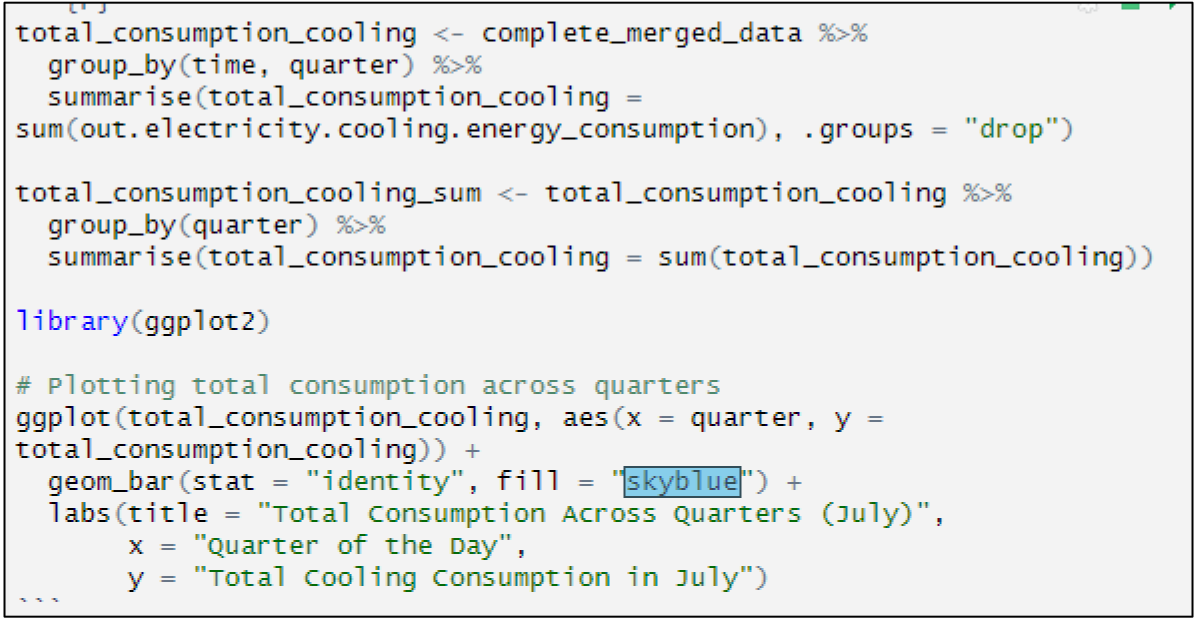


```
complete_merged_data <- cbind(house_weather_merged_df, total_energy =
t_df$total_energy,
                              time = t_df$time,
                        out.electricity.heating.energy_consumption =
t_df$out.electricity.heating.energy_consumption,
                        out.electricity.cooling.energy_consumption =
t_df$out.electricity.cooling.energy_consumption)
dim(complete_merged_data)
```

3.2 Exploratory Data Analysis (EDA):

a) Analyze the distribution of key variables like energy usage, house size, and weather parameters.

b) Identify correlations between variables and plot a chart for energy consumption.

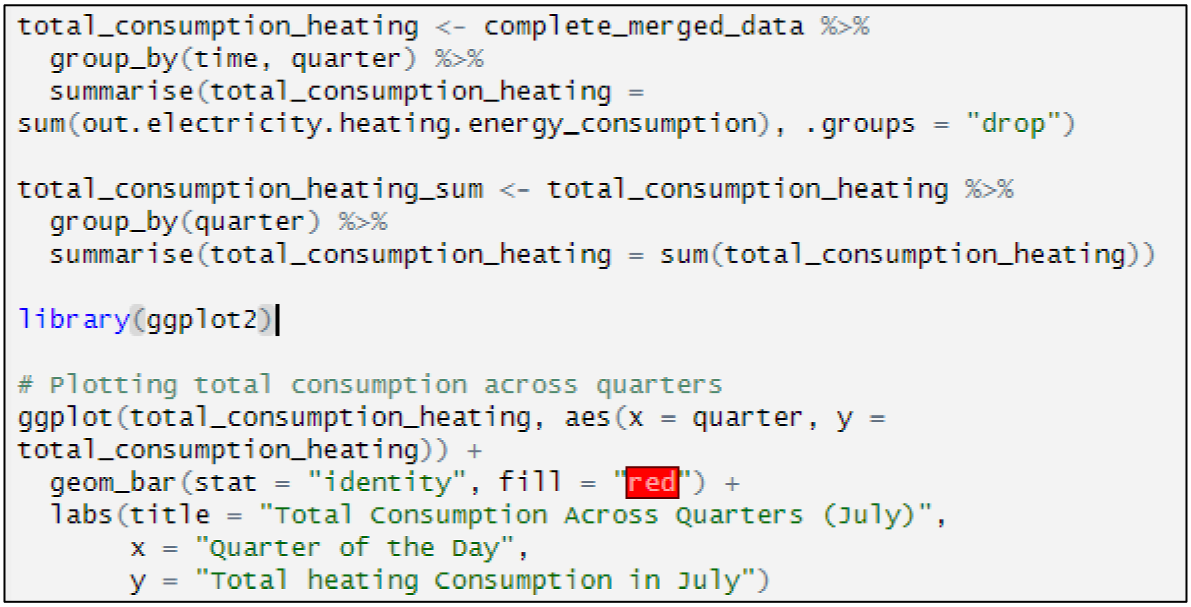


```
total_consumption_cooling <- complete_merged_data %>%
  group_by(time, quarter) %>%
  summarise(total_consumption_cooling =
sum(out.electricity.cooling.energy_consumption), .groups = "drop")

total_consumption_cooling_sum <- total_consumption_cooling %>%
  group_by(quarter) %>%
  summarise(total_consumption_cooling = sum(total_consumption_cooling))

library(ggplot2)

# Plotting total consumption across quarters
ggplot(total_consumption_cooling, aes(x = quarter, y =
total_consumption_cooling)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  labs(title = "Total Consumption Across Quarters (July)",
       x = "Quarter of the Day",
       y = "Total Cooling Consumption in July")
```



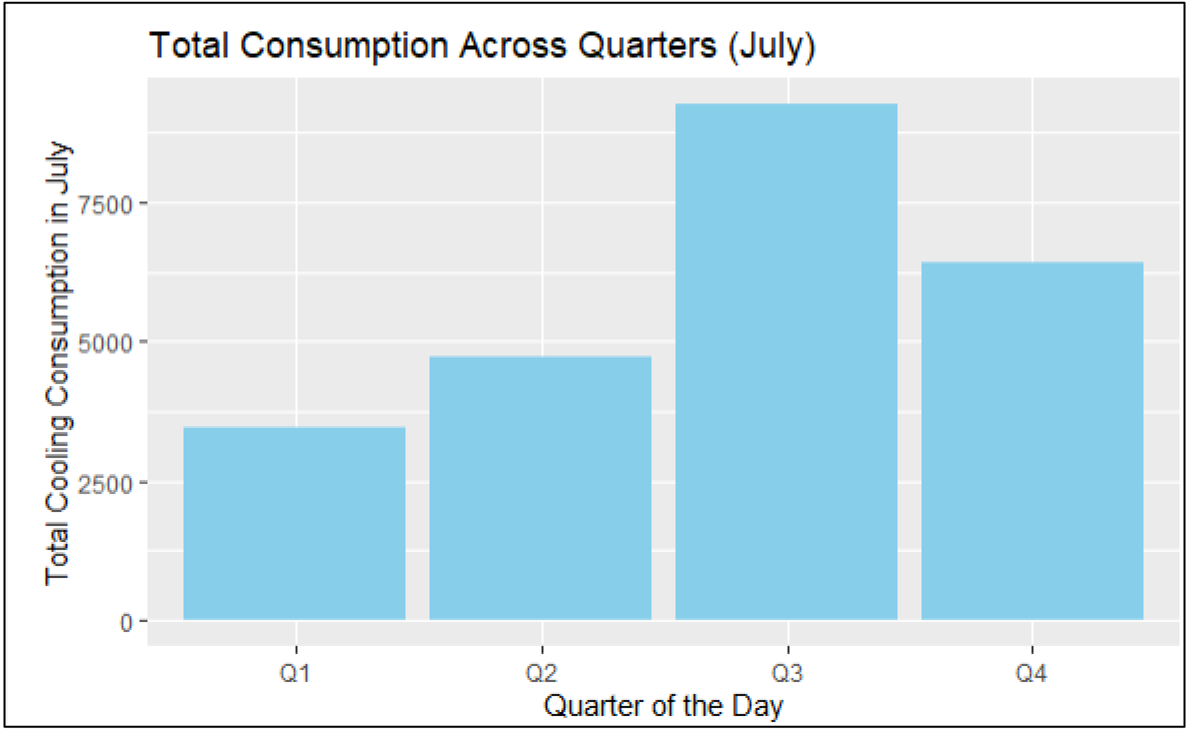
```
total_consumption_heating <- complete_merged_data %>%
  group_by(time, quarter) %>%
  summarise(total_consumption_heating =
sum(out.electricity.heating.energy_consumption), .groups = "drop")

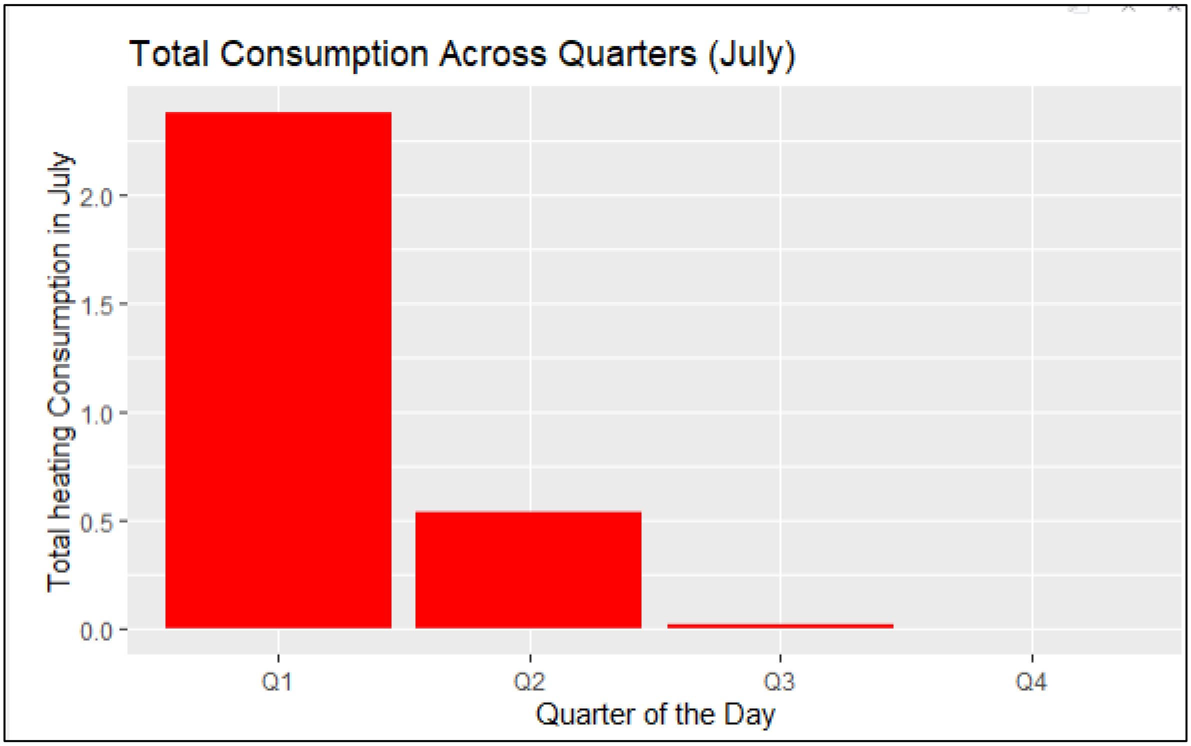
total_consumption_heating_sum <- total_consumption_heating %>%
  group_by(quarter) %>%
  summarise(total_consumption_heating = sum(total_consumption_heating))

library(ggplot2)

# Plotting total consumption across quarters
ggplot(total_consumption_heating, aes(x = quarter, y =
total_consumption_heating)) +
  geom_bar(stat = "identity", fill = "red") +
  labs(title = "Total Consumption Across Quarters (July)",
       x = "Quarter of the Day",
       y = "Total heating Consumption in July")
```

c) Visualize trends and patterns using plots and charts.

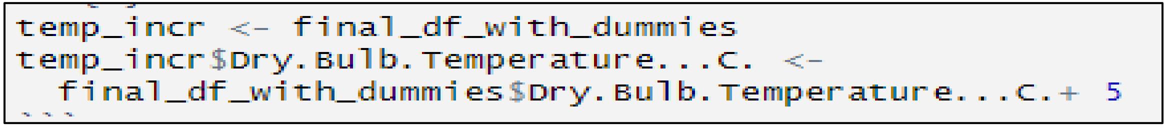




d) Extract insights regarding energy consumption patterns and potential drivers.

3.3 Weather Data Manipulation:

a) Create a new weather dataset by increasing July temperatures by 5 degrees.

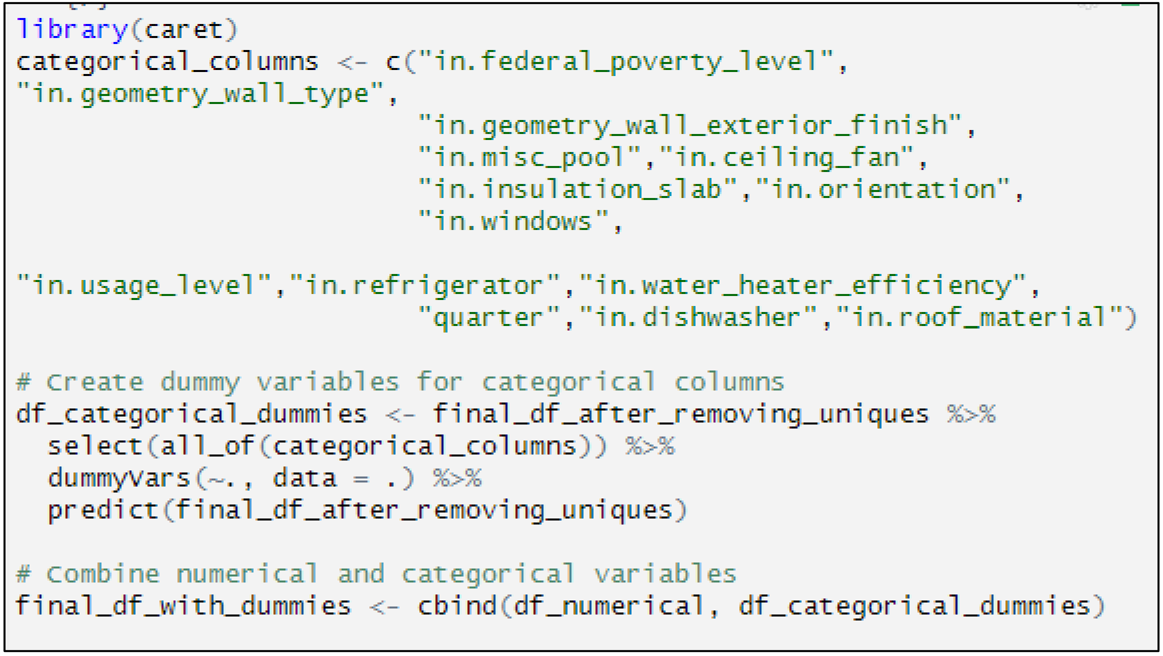


```
temp_incr <- final_df_with_dummies
temp_incr$Dry.Bulb.Temperature...C. <-
  final_df_with_dummies$Dry.Bulb.Temperature...C.+ 5
```

b) Ensure consistency in date and time indices to align with existing datasets.

3.4 Model Building for Energy Usage Prediction:

a) Store the factors affecting the energy consumption to use for a linear model.



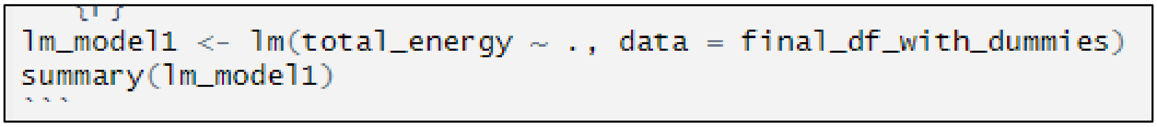
```
library(caret)
categorical_columns <- c("in.federal_poverty_level",
"in.geometry_wall_type",
                          "in.geometry_wall_exterior_finish",
                          "in.misc_pool","in.ceiling_fan",
                          "in.insulation_slab","in.orientation",
                          "in.windows",

"in.usage_level","in.refrigerator","in.water_heater_efficiency",
                          "quarter","in.dishwasher","in.roof_material")

# Create dummy variables for categorical columns
df_categorical_dummies <- final_df_after_removing_uniques %>%
  select(all_of(categorical_columns)) %>%
  dummyVars(~., data = .) %>%
  predict(final_df_after_removing_uniques)

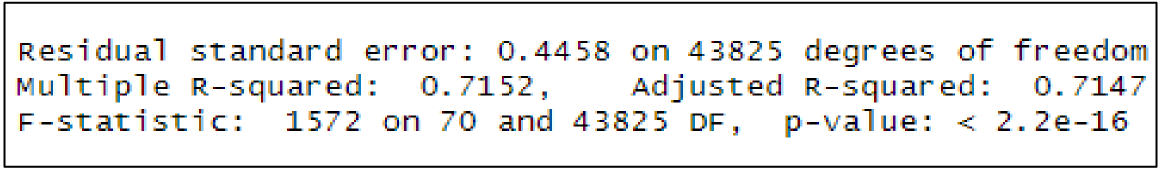
# Combine numerical and categorical variables
final_df_with_dummies <- cbind(df_numerical, df_categorical_dummies)
```

b) Try various machine learning models such as linear regression and decision trees.



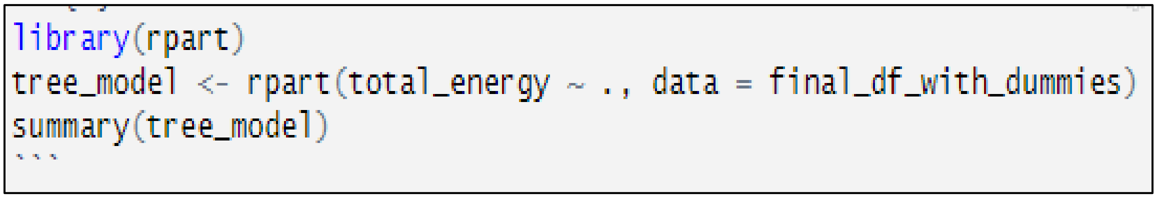
```
lm_model1 <- lm(total_energy ~ ., data = final_df_with_dummies)
summary(lm_model1)
```

Output:



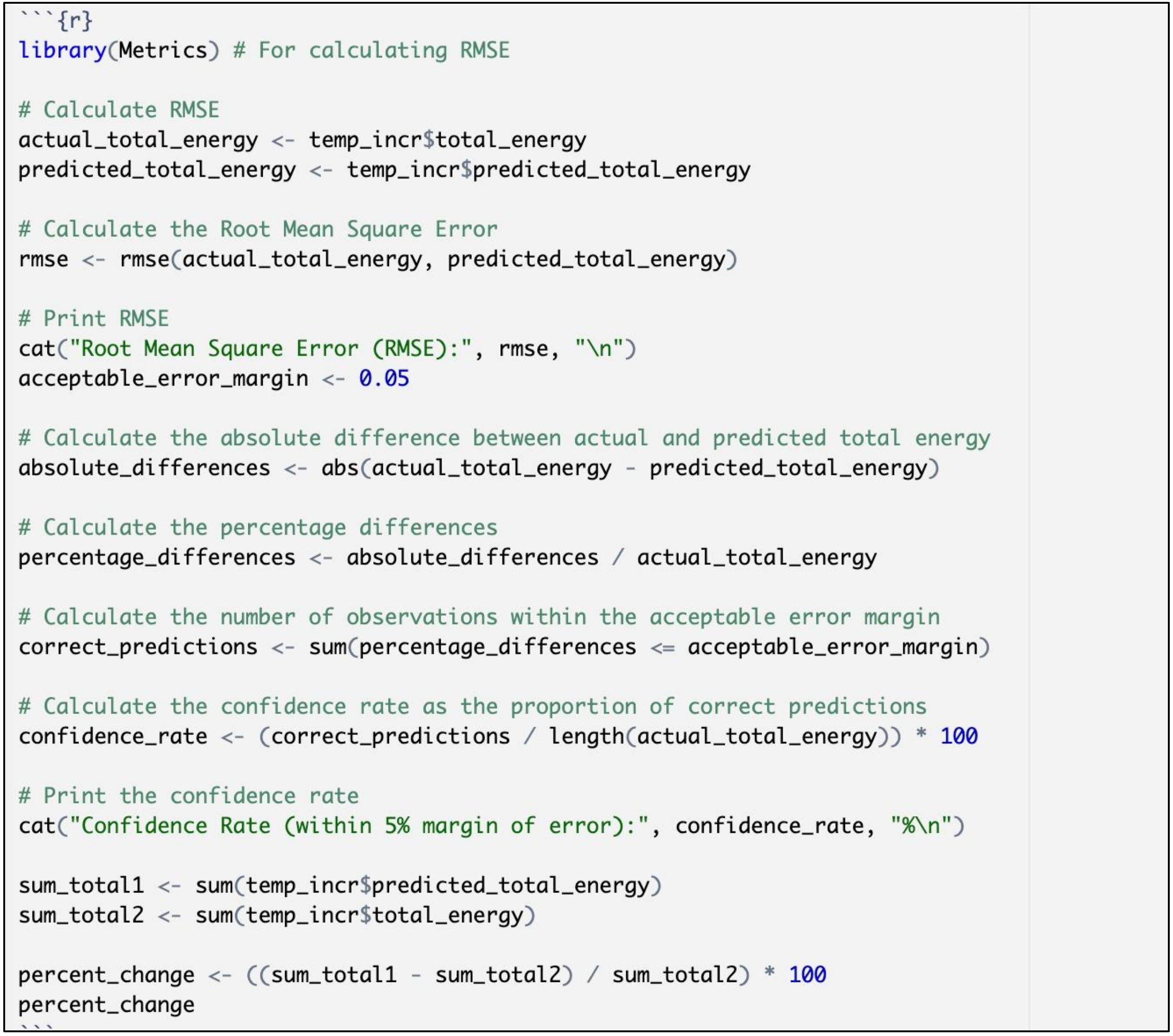
```
Residual standard error: 0.4458 on 43825 degrees of freedom
Multiple R-squared:  0.7152,	Adjusted R-squared:  0.7147
F-statistic:  1572 on 70 and 43825 DF,  p-value: < 2.2e-16
```

Tree model:



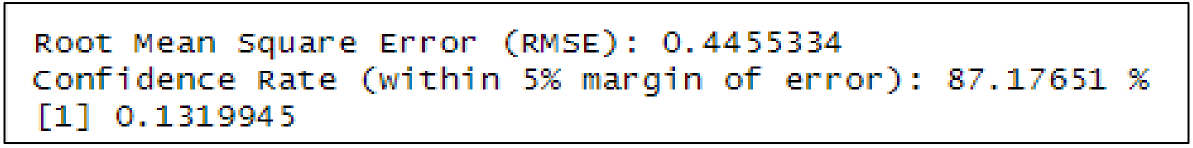
~~~~
library(rpart)
tree_model <- rpart(total_energy ~ ., data = final_df_with_dummies)
summary(tree_model)
```
~~~~

c) Evaluate models using metrics like RMSE (Root Mean Squared Error).



~~~~
```{r}
library(Metrics) # For calculating RMSE

# Calculate RMSE
actual_total_energy <- temp_incr$total_energy
predicted_total_energy <- temp_incr$predicted_total_energy

# Calculate the Root Mean Square Error
rmse <- rmse(actual_total_energy, predicted_total_energy)

# Print RMSE
cat("Root Mean Square Error (RMSE):", rmse, "\n")
acceptable_error_margin <- 0.05

# Calculate the absolute difference between actual and predicted total energy
absolute_differences <- abs(actual_total_energy - predicted_total_energy)

# Calculate the percentage differences
percentage_differences <- absolute_differences / actual_total_energy

# Calculate the number of observations within the acceptable error margin
correct_predictions <- sum(percentage_differences <= acceptable_error_margin)

# Calculate the confidence rate as the proportion of correct predictions
confidence_rate <- (correct_predictions / length(actual_total_energy)) * 100

# Print the confidence rate
cat("Confidence Rate (within 5% margin of error):", confidence_rate, "%\n")

sum_total1 <- sum(temp_incr$predicted_total_energy)
sum_total2 <- sum(temp_incr$total_energy)

percent_change <- ((sum_total1 - sum_total2) / sum_total2) * 100
percent_change
```
~~~~

Output:



```
Root Mean Square Error (RMSE): 0.4455334
Confidence Rate (within 5% margin of error): 87.17651 %
[1] 0.1319945
```

d) Select the best-performing model based on evaluation results.
   <u>Selected lm model based on the results between two models.</u>

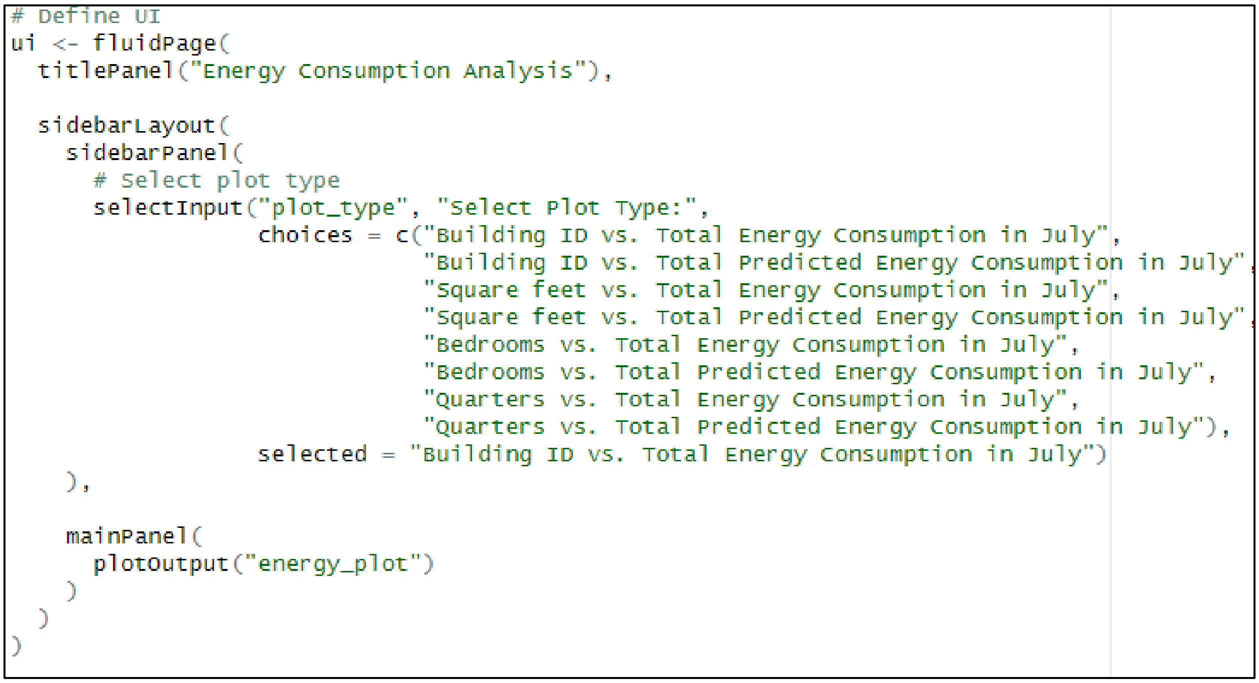
3.5 Model Accuracy Explanation:

a) Explain how accuracy metrics like RMSE or MAE quantify the difference between predicted and actual energy usage.

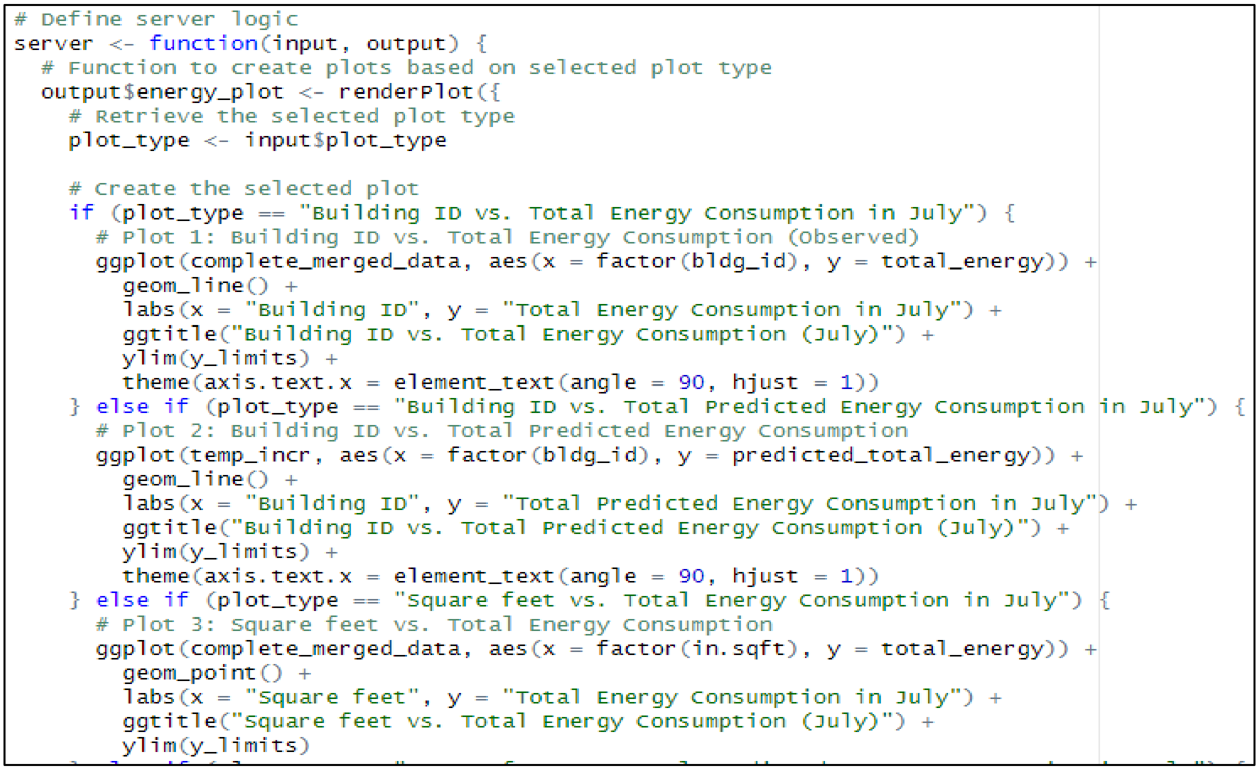
b) Provide insights into factors influencing model accuracy, such as data quality, feature selection, and model complexity.

3.6 Shiny Application Development:

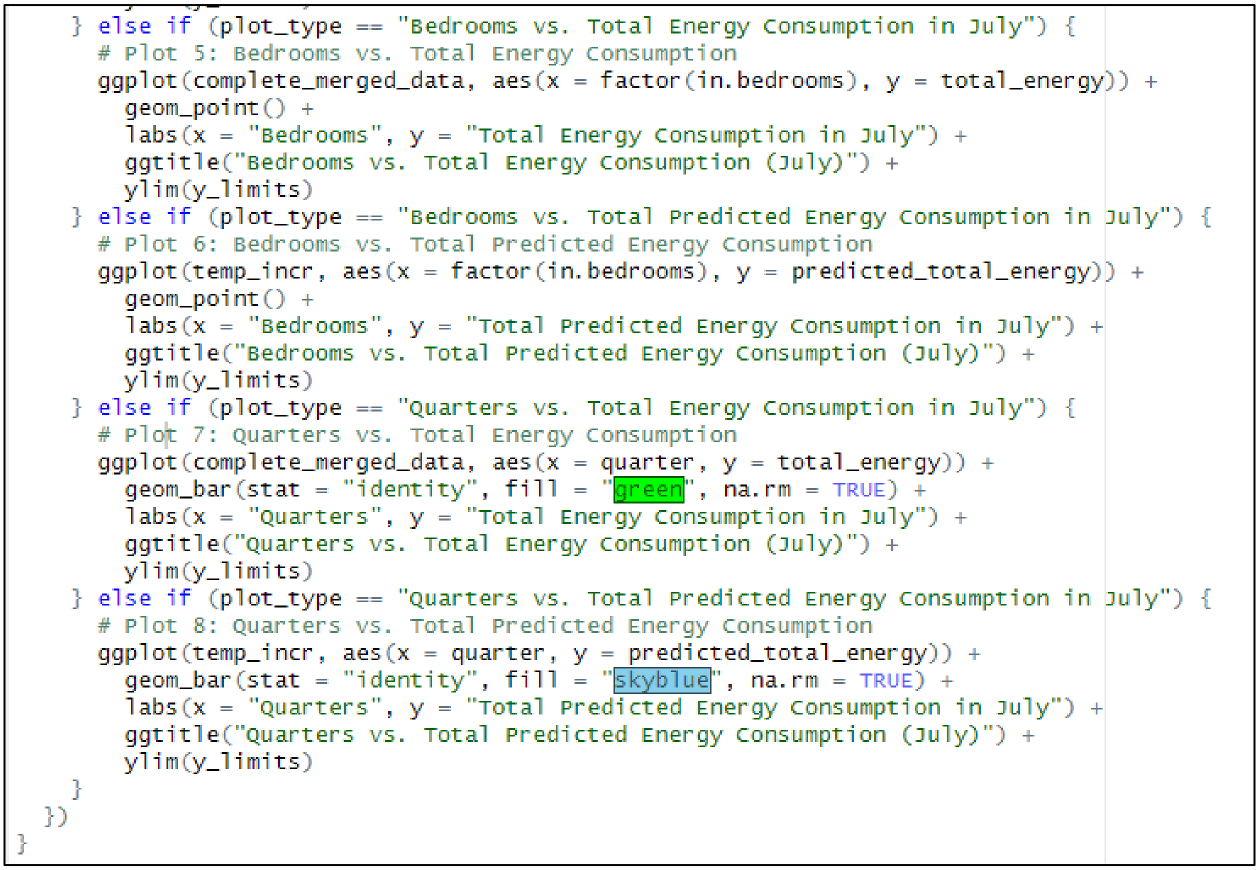
a) Develop a Shiny application to interactively explore energy prediction and future demand.



```
# Define UI
ui <- fluidPage(
  titlePanel("Energy Consumption Analysis"),

  sidebarLayout(
    sidebarPanel(
      # Select plot type
      selectInput("plot_type", "Select Plot Type:",
                  choices = c("Building ID vs. Total Energy Consumption in July",
                              "Building ID vs. Total Predicted Energy Consumption in July",
                              "Square feet vs. Total Energy Consumption in July",
                              "Square feet vs. Total Predicted Energy Consumption in July",
                              "Bedrooms vs. Total Energy Consumption in July",
                              "Bedrooms vs. Total Predicted Energy Consumption in July",
                              "Quarters vs. Total Energy Consumption in July",
                              "Quarters vs. Total Predicted Energy Consumption in July"),
                  selected = "Building ID vs. Total Energy Consumption in July")
    ),

    mainPanel(
      plotOutput("energy_plot")
    )
  )
)
```

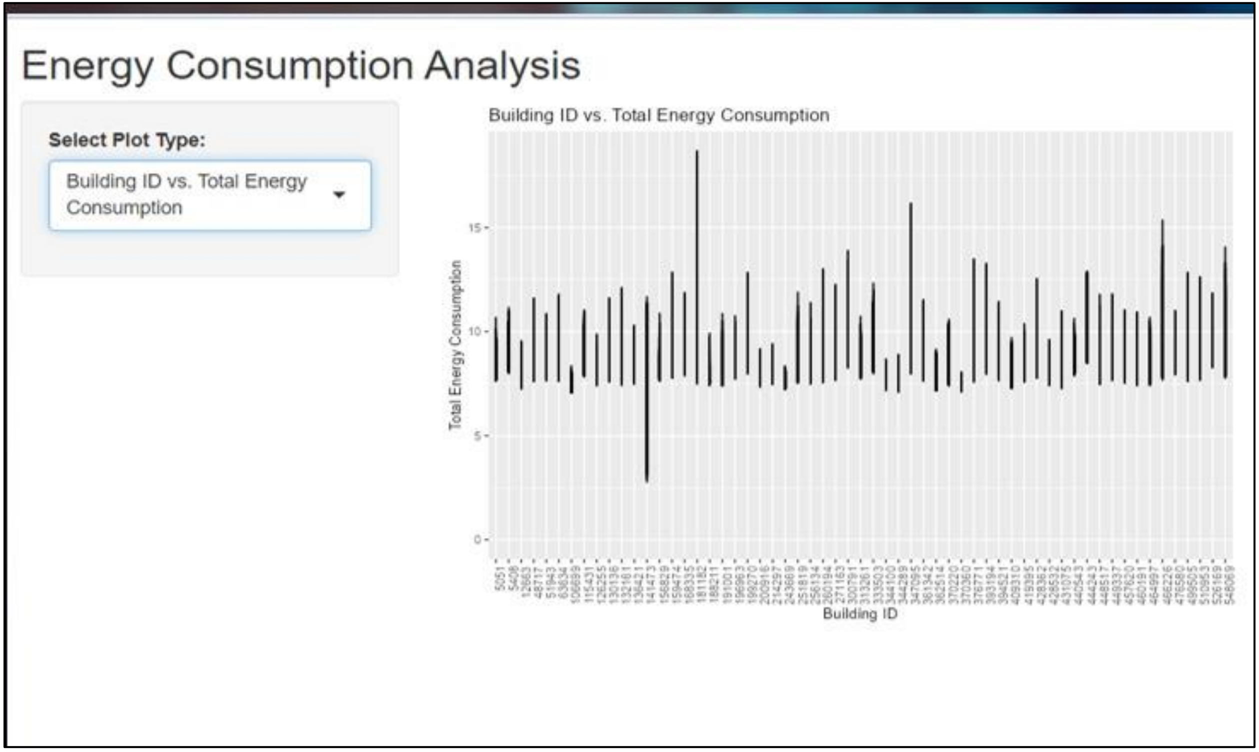


```
# Define server logic
server <- function(input, output) {
  # Function to create plots based on selected plot type
  output$energy_plot <- renderPlot({
    # Retrieve the selected plot type
    plot_type <- input$plot_type

    # Create the selected plot
    if (plot_type == "Building ID vs. Total Energy Consumption in July") {
      # Plot 1: Building ID vs. Total Energy Consumption (Observed)
      ggplot(complete_merged_data, aes(x = factor(bldg_id), y = total_energy)) +
        geom_line() +
        labs(x = "Building ID", y = "Total Energy Consumption in July") +
        ggtitle("Building ID vs. Total Energy Consumption (July)") +
        ylim(y_limits) +
        theme(axis.text.x = element_text(angle = 90, hjust = 1))
    } else if (plot_type == "Building ID vs. Total Predicted Energy Consumption in July") {
      # Plot 2: Building ID vs. Total Predicted Energy Consumption
      ggplot(temp_incr, aes(x = factor(bldg_id), y = predicted_total_energy)) +
        geom_line() +
        labs(x = "Building ID", y = "Total Predicted Energy Consumption in July") +
        ggtitle("Building ID vs. Total Predicted Energy Consumption (July)") +
        ylim(y_limits) +
        theme(axis.text.x = element_text(angle = 90, hjust = 1))
    } else if (plot_type == "Square feet vs. Total Energy Consumption in July") {
      # Plot 3: Square feet vs. Total Energy Consumption
      ggplot(complete_merged_data, aes(x = factor(in.sqft), y = total_energy)) +
        geom_point() +
        labs(x = "Square feet", y = "Total Energy Consumption in July") +
        ggtitle("Square feet vs. Total Energy Consumption (July)") +
        ylim(y_limits)
```



```
    } else if (plot_type == "Bedrooms vs. Total Energy Consumption in July") {
      # Plot 5: Bedrooms vs. Total Energy Consumption
      ggplot(complete_merged_data, aes(x = factor(in.bedrooms), y = total_energy)) +
        geom_point() +
        labs(x = "Bedrooms", y = "Total Energy Consumption in July") +
        ggtitle("Bedrooms vs. Total Energy Consumption (July)") +
        ylim(y_limits)
    } else if (plot_type == "Bedrooms vs. Total Predicted Energy Consumption in July") {
      # Plot 6: Bedrooms vs. Total Predicted Energy Consumption
      ggplot(temp_incr, aes(x = factor(in.bedrooms), y = predicted_total_energy)) +
        geom_point() +
        labs(x = "Bedrooms", y = "Total Predicted Energy Consumption in July") +
        ggtitle("Bedrooms vs. Total Predicted Energy Consumption (July)") +
        ylim(y_limits)
    } else if (plot_type == "Quarters vs. Total Energy Consumption in July") {
      # Plot 7: Quarters vs. Total Energy Consumption
      ggplot(complete_merged_data, aes(x = quarter, y = total_energy)) +
        geom_bar(stat = "identity", fill = "green", na.rm = TRUE) +
        labs(x = "Quarters", y = "Total Energy Consumption in July") +
        ggtitle("Quarters vs. Total Energy Consumption (July)") +
        ylim(y_limits)
    } else if (plot_type == "Quarters vs. Total Predicted Energy Consumption in July") {
      # Plot 8: Quarters vs. Total Predicted Energy Consumption
      ggplot(temp_incr, aes(x = quarter, y = predicted_total_energy)) +
        geom_bar(stat = "identity", fill = "skyblue", na.rm = TRUE) +
        labs(x = "Quarters", y = "Total Predicted Energy Consumption in July") +
        ggtitle("Quarters vs. Total Predicted Energy Consumption (July)") +
        ylim(y_limits)
    }
  })
}
```

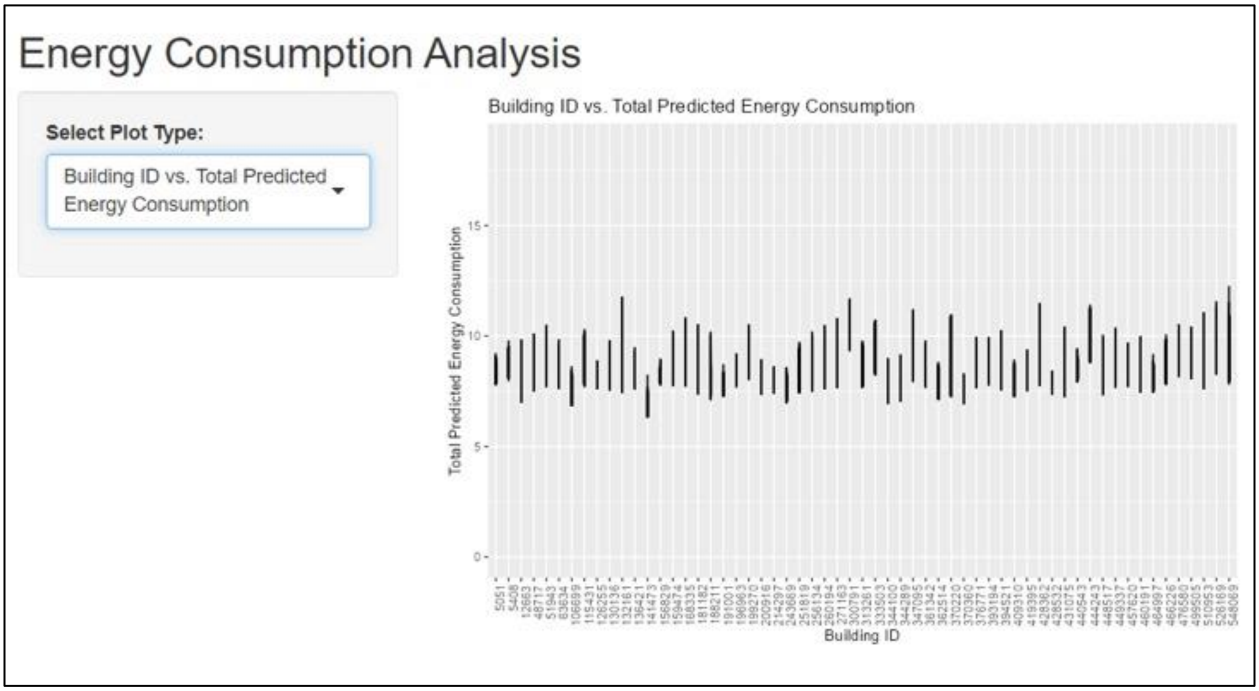
b) Provide visualization tools for a better understanding of model predictions and drivers of energy needs.



Energy Consumption Analysis

Select Plot Type:

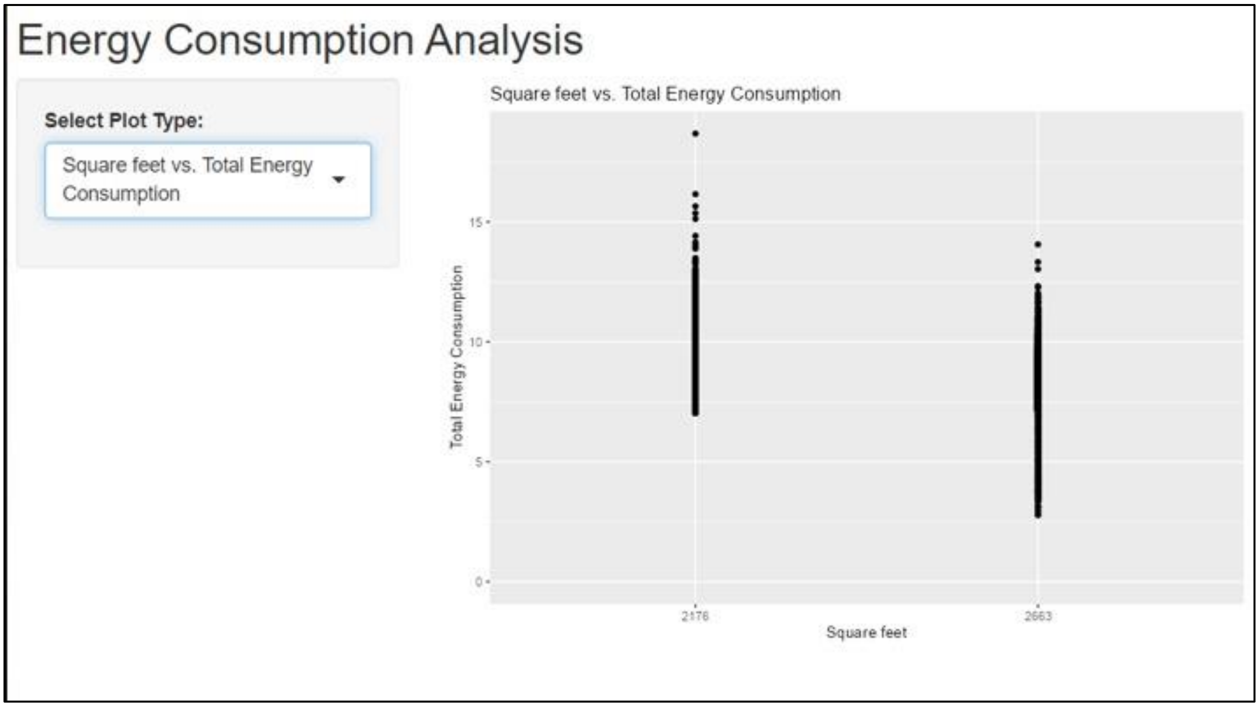
Building ID vs. Total Energy Consumption



Energy Consumption Analysis

Select Plot Type:

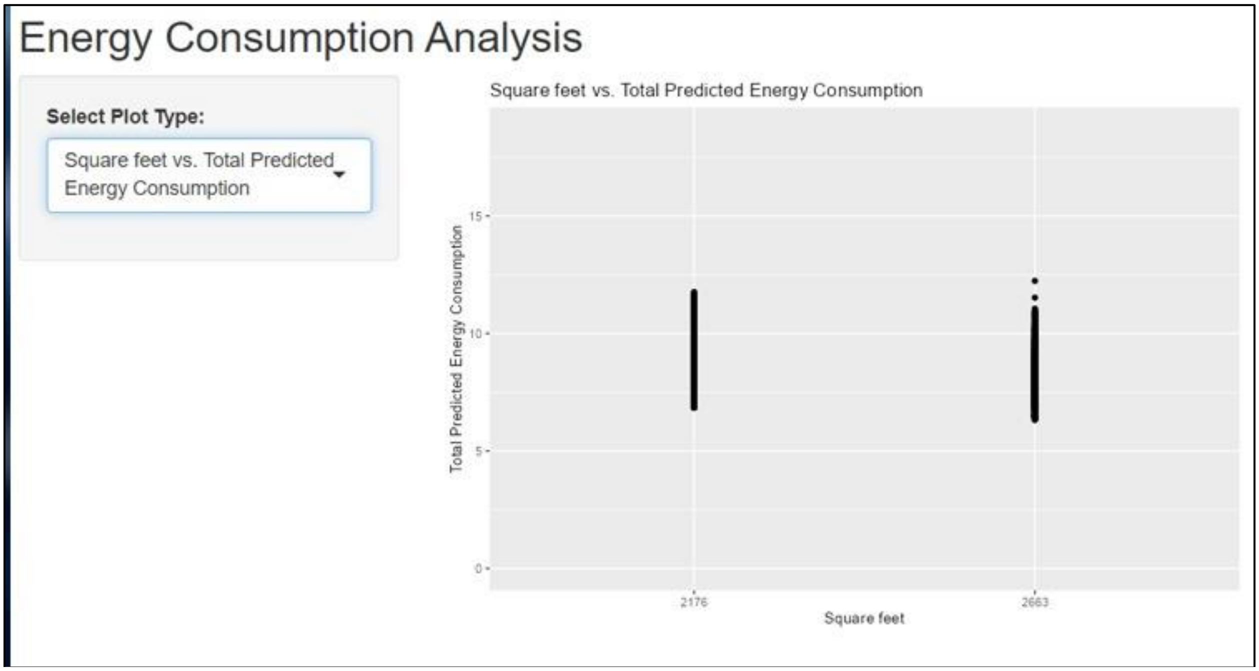
Building ID vs. Total Predicted Energy Consumption



Energy Consumption Analysis

Select Plot Type:

Square feet vs. Total Energy Consumption

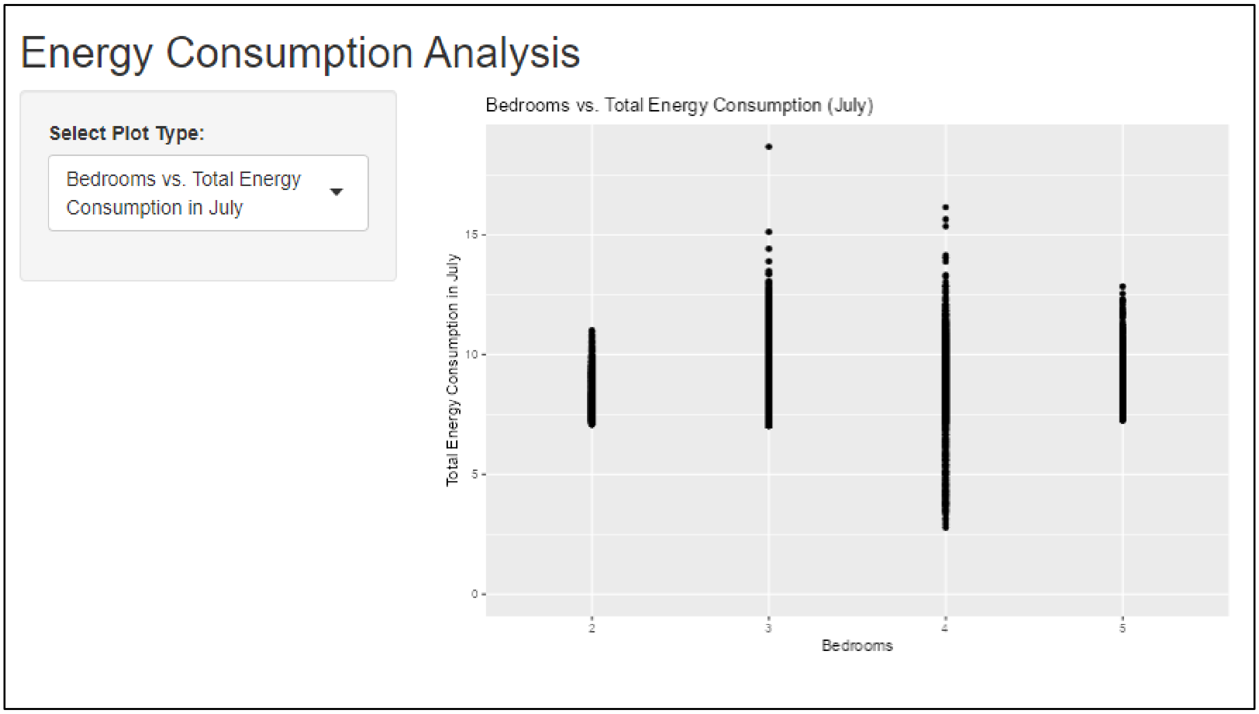


Energy Consumption Analysis

Select Plot Type:

Square feet vs. Total Predicted Energy Consumption

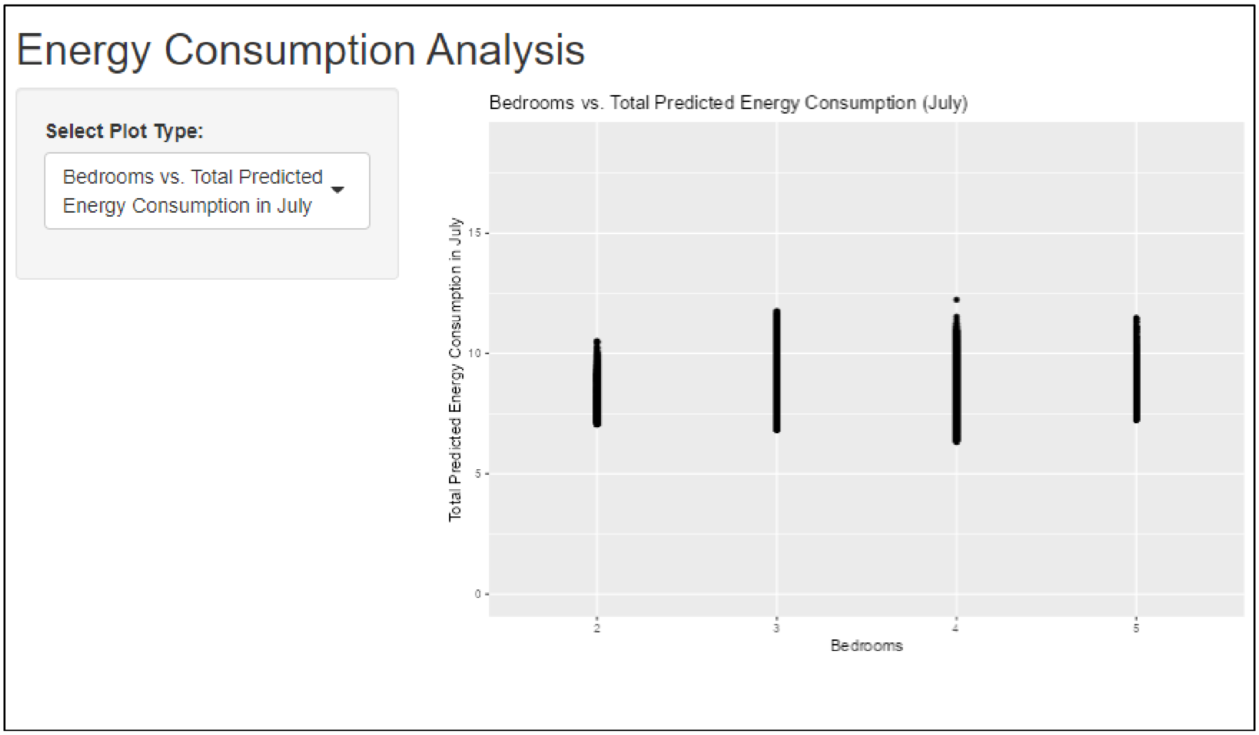
Square feet vs. Total Predicted Energy Consumption
Total Predicted Energy Consumption
15
10
5
0
2176
2663
Square feet



Energy Consumption Analysis

Select Plot Type: Bedrooms vs. Total Energy Consumption in July

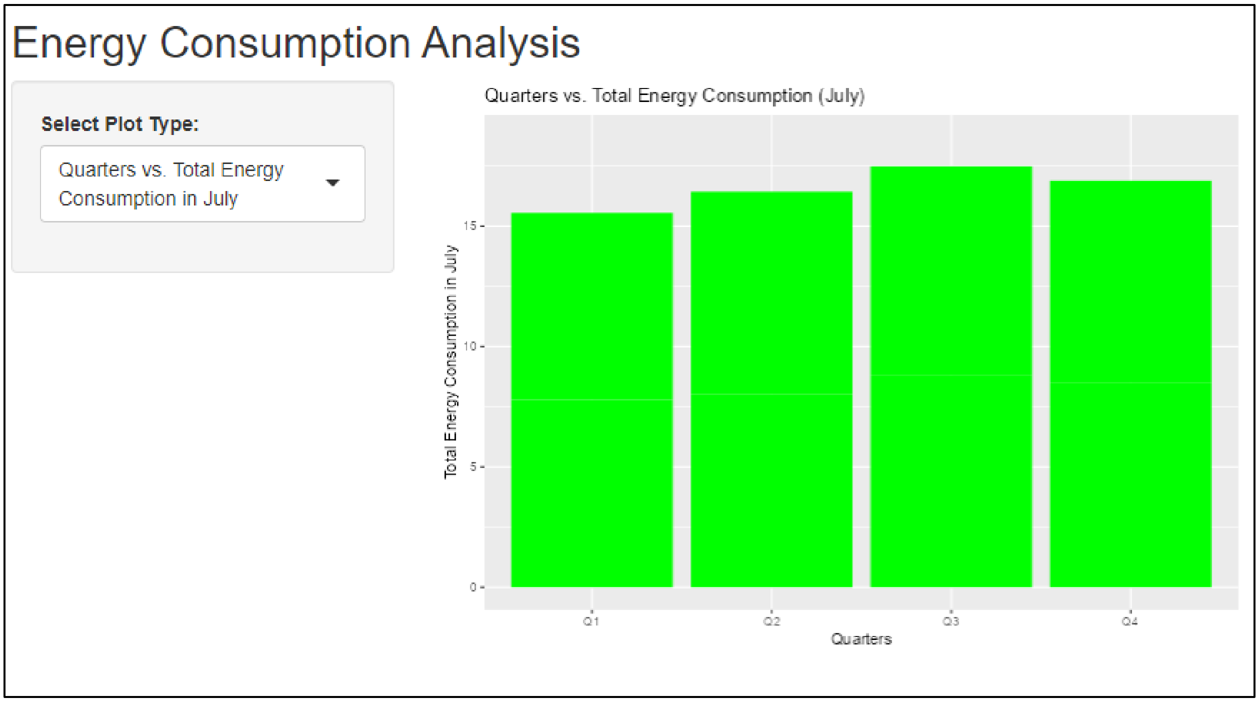
Bedrooms vs. Total Energy Consumption (July)
Total Energy Consumption in July
15
10
5
0
2
3
4
5
Bedrooms



Energy Consumption Analysis

Select Plot Type: Bedrooms vs. Total Predicted Energy Consumption in July

Bedrooms vs. Total Predicted Energy Consumption (July)
Total Predicted Energy Consumption in July
15
10
5
0
2
3
4
5
Bedrooms

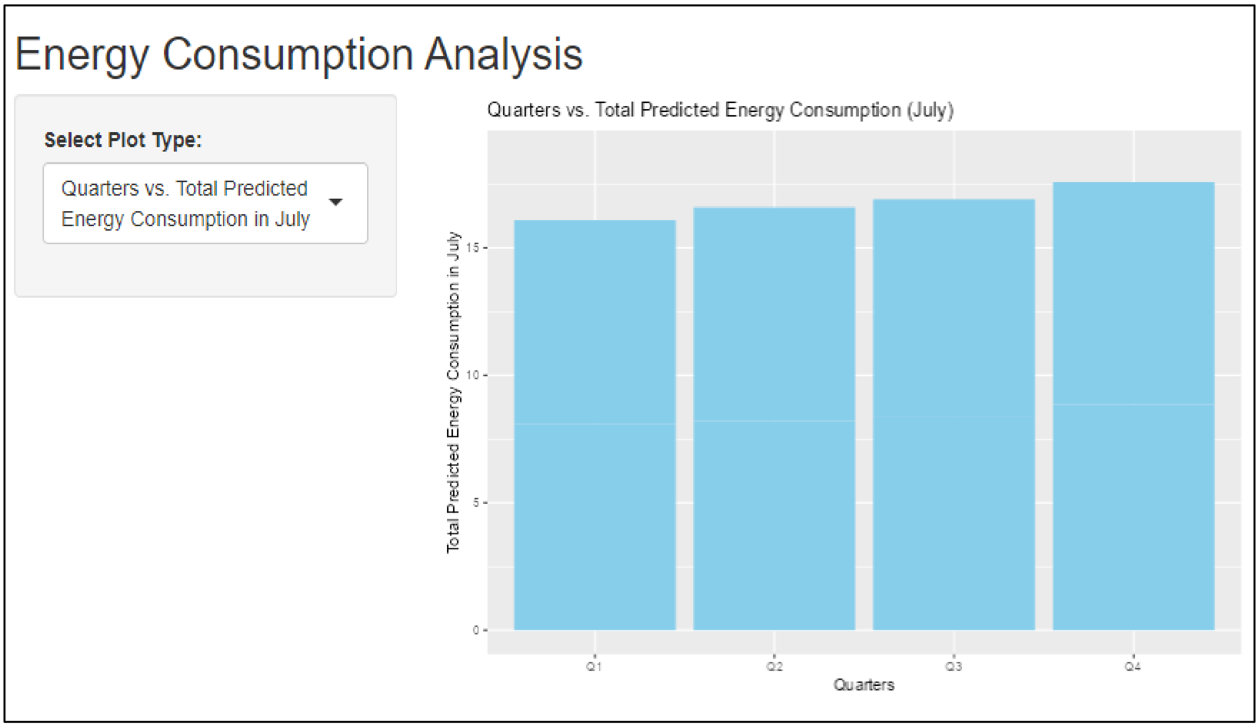


Energy Consumption Analysis

Select Plot Type:

Quarters vs. Total Energy Consumption in July

Quarters vs. Total Energy Consumption (July)
Total Energy Consumption in July
0
5
10
15
Q1
Q2
Q3
Q4
Quarters



Energy Consumption Analysis

Select Plot Type:

Quarters vs. Total Predicted Energy Consumption in July

Quarters vs. Total Predicted Energy Consumption (July)
Total Predicted Energy Consumption in July
0
5
10
15
Q1
Q2
Q3
Q4
Quarters

3.7 Recommendations:

a) Summarize findings and recommendations for eSC, emphasizing the importance of proactive measures to manage peak energy demand.

b) Provide actionable insights based on analysis results, model accuracy, and potential strategies for reducing energy consumption during peak periods.

4. Conclusion:

In summation, our collaborative endeavor has yielded substantive insights into predicting energy consumption for residential properties in South Carolina. Through a judicious blend of data analysis, modeling, and simulation, we have identified pivotal factors influencing energy consumption and devised predictive models to forecast consumption levels. We are confident that our findings and recommendations will empower eSC to make informed decisions, fostering enhanced energy efficiency and service excellence.