Introduction to Data Science
IST687

Final Project Report

Submitted by

Mrunmai Musale
Net ID: mmusale
Greeshma Pothineni
Net ID: gpothine
Sree Chandan Kamireddi
Net ID: skamired
Sai Sinduri Vangala
Net ID: ssvangal

Under the guidance of

Prof. Akit Kumar, Prof. C. Dunham
Professor

Spring 2024

Contents

1 Project Overview

1.1 Goal

Build a model that predicts the energy usage, for the month of July if the temperature was 5 degrees warmer and propose actionable insights to reduce the energy consumption.

1.2 Description

Energy supply is offered to South Carolina residential premises (and a small part of North Carolina) by eSC Energy Company. In light of the effects of global warming on energy demand and the potential for blackouts brought on by excessive energy demand, it is important to suggest a business strategy that might raise public awareness of energy use and conservation and, as a result, lower overall resident energy consumption. In order to fulfill the growing demand for energy, eSC seeks to prevent the construction of new power plants and instead encourages customers to cut back on their energy use.

The goal is to use less energy in July because it's becoming hotter outside and there's a rapidly growing trend in energy consumption.

2 Requirements

2.1 Business Requirements

1. **Reduce energy usage**: Provide strategies to assist eSC in raising South Carolina residential owners' understanding of energy use.
2. **Predict future energy usage**: To assist eSC in getting prepared to accommodate an increase in energy supply, forecast the region's future energy use. This will help eSC prepare ahead of time and stop blackouts.
3. **Identify the city with peak energy usage**: Determine which area uses the most energy. Determine the causes of the high energy consumption.

2.2 Technical Requirements

1. **Ensure energy supply availability:** Analyze consumption trends of energy in order to suggest the expected energy supply.
2. **Predict energy demand:** Estimate the energy consumption for the upcoming summer (July) to assist eSC in making strategic decisions and preparations.
3. **Reduce overhead cost:** To help eSC avoid the expense of constructing a new facility to satisfy the energy demands, suggest ways to reduce energy usage.

3 General Overview

3.1 Data Provided

1. **Static House Data:**
   (a) Description: Includes data regarding over 5,000 family residences that use energy from eSC.
   (b) Attributes: Comprises of house attributes like the building ID, house size and static information. The file is saved in the ’parquet’ format, which is optimal for storage.

2. **Energy Usage Data:**
   (a) Description: Provides information on each home's hourly energy use in the Static residence data.
   (b) Dataset Structure: One dataset file with validated and calibrated energy consumption and one-hour load profiles is available for each residence.
   (c) Data Variety: Details the amount of energy used by each home for each hour from a variety of sources (such as the dryer and air conditioner).

3. **Meta Data:**
   (a) Description: A data description file that describes the fields used in the various housing data files.
   (b) Format: A simple, human-readable CSV file with attribute descriptions.

4. **Weather Data:**
   (a) Description: Hourly weather data, with one file for each geographic area (county).
   (b) Data: Based on a county code, time-series weather data is collected and stored.
   (c) County Code Reference: The 'in.county' column in the Static House Data indicates the county code for each house.
   (d) Format: The file is stored in a CSV format which is easy to use.

3.2 Tasks and Deliverables

1. Data Preparation: Establish a methodology for accessing and consolidating the dataset.

2. Exploratory Analysis: Obtain insights from the consolidated data, refine it for efficacy and usability.

3. Model Development: Construct a predictive model to estimate energy consumption in 5-bedroom residences situated in the western region, considering 6 hourly variations per day.

4. Demand Analysis: Incorporate factors like high temperatures, daily usage patterns, and seasonal temperature fluctuations to forecast future energy demand.

5. Demand Mitigation Strategies: Recommend tactics to facilitate demand reduction based on the analytical findings, aiming to decrease overhead expenses for eSC.

6. Shiny Applications: Create an interactive application featuring graphical representations for presentation to the CEO of eSC.

4 Detailed Overview

4.1 Data Preparation

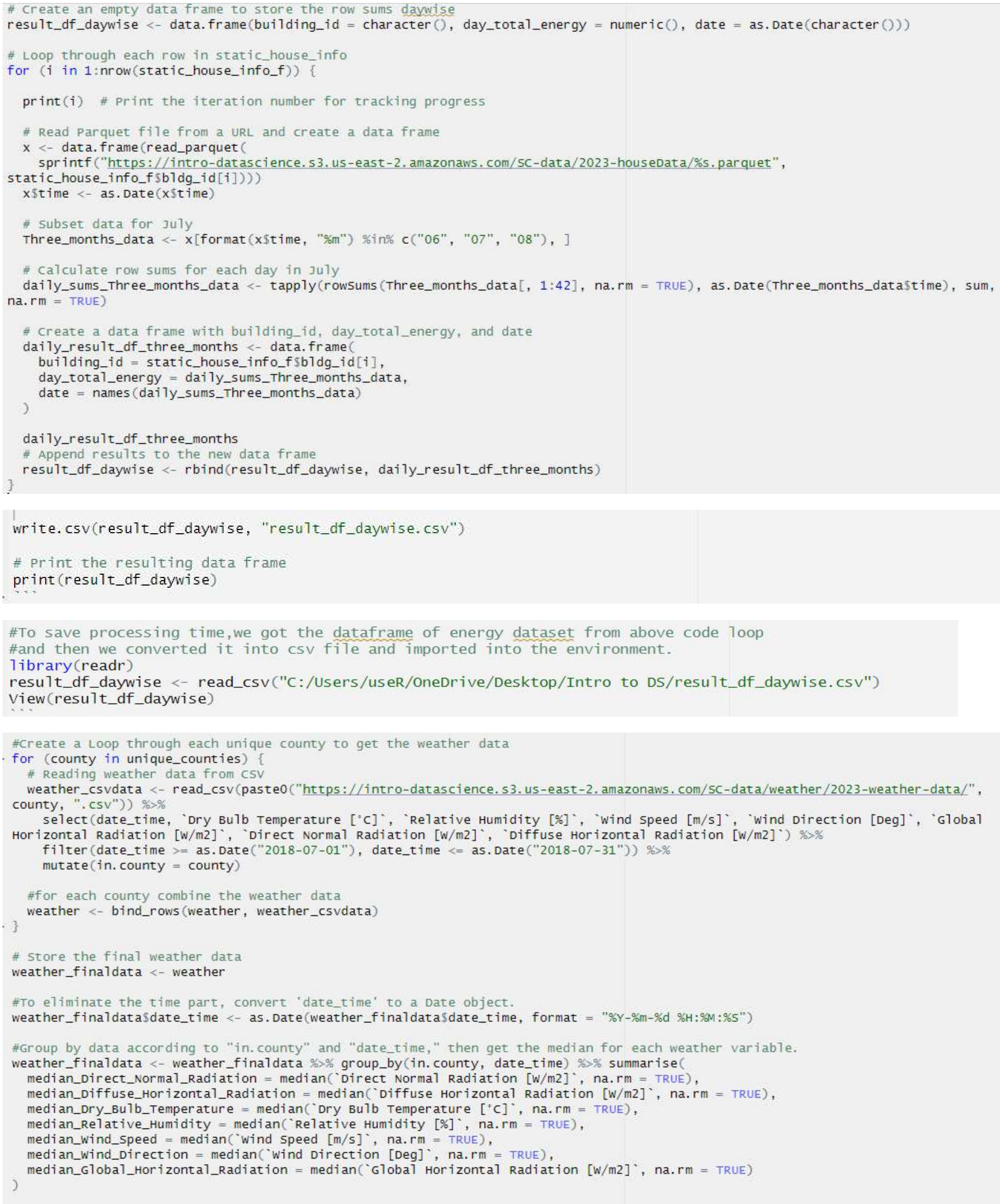
1. Data Cleaning for Static House Info:
   (a) The code reads a Parquet file containing static house information
   (b) It removes specific columns defined in columns_to_remove.
   (c) The resulting dataset is stored in static_house_info.

2. Calculating Total Energy for Each Building:
   (a) The code initializes an empty data frame (result_df_daywise) to store building-wise total energy.
   (b) It iterates over each building, reads its Parquet file, and calculates total energy for June, July and August.
   (c) The results are appended to result_df_daywise.

3. Processing Weather Data for June, July and August:
   (a) The code reads weather data for different counties in June, July and August.
   (b) It calculates median values for various weather variables.
   (c) It updates corresponding columns in static_house_info with the calculated median values.

4. Creating the Final Output Dataframe:
   (a) A final output dataframe (merge static house info df4) is created by selecting specific columns from static house info.

5. Merging Dataframes:
   (a) The code attempts to merge
      i. static_house_info (original dataset)
      ii. result_df_datewise (energy dataset daywise)
      iii. weather_final (weather dataset including date and county)
      iv. static_house_info df1 (merging of static house info and result df datewise)
      v. merge_static_house_info_df (merging of static house info df1 and weather final including date and county)
      vi. merge static house info df4 (dataset after cleaning the original one)
   (b) A left join is performed using the merge function and later using the left join function from the dplyr package.
   (c) Columns not necessary for further analysis are removed from the merged dataframe.



```r
# Create an empty data frame to store the row sums daywise
result_df_daywise <- data.frame(building_id = character(), day_total_energy = numeric(), date = as.Date(character()))

# Loop through each row in static_house_info
for (i in 1:nrow(static_house_info_f)) {

  print(i)  # Print the iteration number for tracking progress

  # Read Parquet file from a URL and create a data frame
  x <- data.frame(read_parquet(
    sprintf("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/2023-houseData/%s.parquet",
static_house_info_f$bldg_id[i])))
  x$time <- as.Date(x$time)

  # Subset data for July
  Three_months_data <- x[format(x$time, "%m") %in% c("06", "07", "08"), ]

  # Calculate row sums for each day in July
  daily_sums_Three_months_data <- tapply(rowSums(Three_months_data[, 1:42], na.rm = TRUE), as.Date(Three_months_data$time), sum,
na.rm = TRUE)

  # Create a data frame with building_id, day_total_energy, and date
  daily_result_df_three_months <- data.frame(
    building_id = static_house_info_f$bldg_id[i],
    day_total_energy = daily_sums_Three_months_data,
    date = names(daily_sums_Three_months_data)
  )

  daily_result_df_three_months
  # Append results to the new data frame
  result_df_daywise <- rbind(result_df_daywise, daily_result_df_three_months)
}
```

```r
write.csv(result_df_daywise, "result_df_daywise.csv")

# Print the resulting data frame
print(result_df_daywise)
```

```r
#To save processing time,we got the dataframe of energy dataset from above code loop
#and then we converted it into csv file and imported into the environment.
library(readr)
result_df_daywise <- read_csv("C:/Users/useR/OneDrive/Desktop/Intro to DS/result_df_daywise.csv")
View(result_df_daywise)
```

```r
#Create a Loop through each unique county to get the weather data
for (county in unique_counties) {
  # Reading weather data from CSV
  weather_csvdata <- read_csv(paste0("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/weather/2023-weather-data/",
county, ".csv")) %>%
    select(date_time, `Dry Bulb Temperature [°C]`, `Relative Humidity [%]`, `Wind Speed [m/s]`, `Wind Direction [Deg]`, `Global
Horizontal Radiation [W/m2]`, `Direct Normal Radiation [W/m2]`, `Diffuse Horizontal Radiation [W/m2]`) %>%
    filter(date_time >= as.Date("2018-07-01"), date_time <= as.Date("2018-07-31")) %>%
    mutate(in.county = county)

  #for each county combine the weather data
  weather <- bind_rows(weather, weather_csvdata)
}

# Store the final weather data
weather_finaldata <- weather

#To eliminate the time part, convert 'date_time' to a Date object.
weather_finaldata$date_time <- as.Date(weather_finaldata$date_time, format = "%Y-%m-%d %H:%M:%S")

#Group by data according to "in.county" and "date_time," then get the median for each weather variable.
weather_finaldata <- weather_finaldata %>% group_by(in.county, date_time) %>% summarise(
  median_Direct_Normal_Radiation = median(`Direct Normal Radiation [W/m2]`, na.rm = TRUE),
  median_Diffuse_Horizontal_Radiation = median(`Diffuse Horizontal Radiation [W/m2]`, na.rm = TRUE),
  median_Dry_Bulb_Temperature = median(`Dry Bulb Temperature [°C]`, na.rm = TRUE),
  median_Relative_Humidity = median(`Relative Humidity [%]`, na.rm = TRUE),
  median_Wind_Speed = median(`Wind Speed [m/s]`, na.rm = TRUE),
  median_Wind_Direction = median(`Wind Direction [Deg]`, na.rm = TRUE),
  median_Global_Horizontal_Radiation = median(`Global Horizontal Radiation [W/m2]`, na.rm = TRUE)
)
```

Figure 1 : Code for Data cleaning and Preparation

4.2 Exploratory Analysis

1. Understand the Data:
   (a) Utilize functions such as str(), head(), summary(), and dim() to inspect the structure of the datasets.
   (b) Identify the types of variables—numeric, categorical, date/time, etc and identify any missing data, employing techniques like interpolation to fill in missing values.
2. Descriptive Statistics:
   (a) Utilize summary() to calculate basic descriptive statistics for numeric variables.
   (b) Employ density plots, box plots, or histograms to assess the distribution of numeric variables.
3. Categorical data:
   (a) Utilize frequency tables, bar charts, or pie charts to explore the distribution of categorical data.
   (b) Identify any uncommon or distinctive categories.
4. Correlation Analysis:
   (a) Utilize scatter plots or correlation matrices to investigate relationships between numeric variables.
   (b) Boxplots serve as a valuable tool for visualizing correlations.
5. Data Cleaning:
   (a) Address missing values through imputation or removal.
   (b) Examine data for abnormalities and outliers using domain knowledge, determining whether to retain or discard them.

4.3 Model Building

1. Data Splitting:
   (a) We utilized the createDataPartition function to split the dataset into training (80%) and testing (20%) subsets.
2. Managing Categorical Variables:
   (a) Initially, we identified which values in the character column are unique.
   (b) Then, only the test data rows with values that relate to the distinct values in the training data are kept.
3. Finding Constant Variables:
   (a) In both the training and testing datasets, we detected and removed the variables characterized by a single level,referred to as constant variables.
4. Linear Regression Model:
   (a) We build a linear regression model suitable for the training set of data using the lm function.

5. Evaluation of the Model:

(a) Provided the detailed summary of the linear regression model.

(b) We generated predictions and displayed statistics such as the median, minimum, and maximum values of the target variable using the test data.

(c) By calculating and displaying the Mean Absolute Percentage Error (MAPE), the model's accuracy is assessed.

6. Multiple R-squared (R^2):

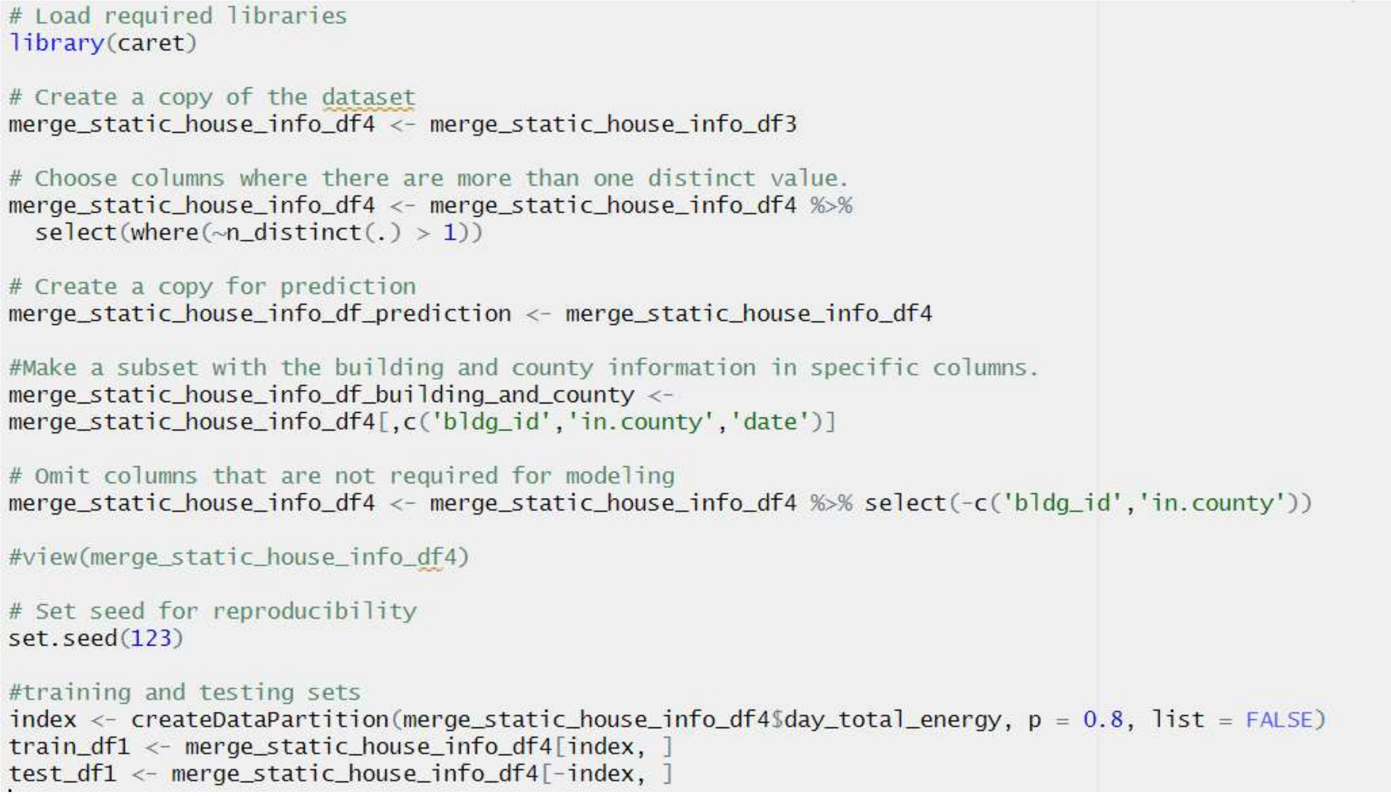
(a) This measure indicates how much of the response variable's variance (total_energy) can be explained by the predictors.

(b) In this case, the model shows the R-squared value is approximately 89.85%.

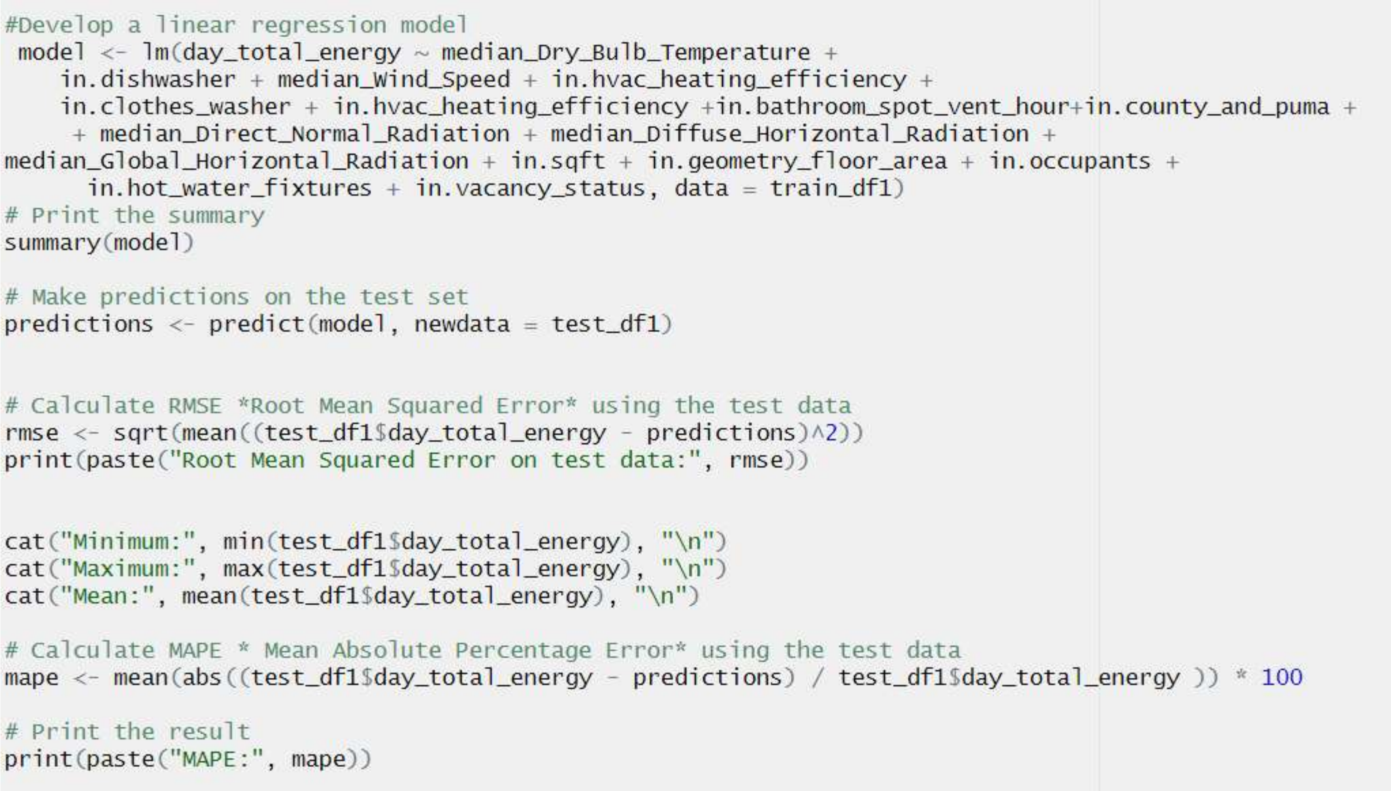
(c) Adjusted R-squared: This measure of model fit is more precise since it considers the number of predictors included in the model.

7. P-value:

(a) The near-zero p-value associated to the F-statistic (less than 2.2e-16) shows the statistical significance of the model, denying the null hypothesis that all coefficients are zero.

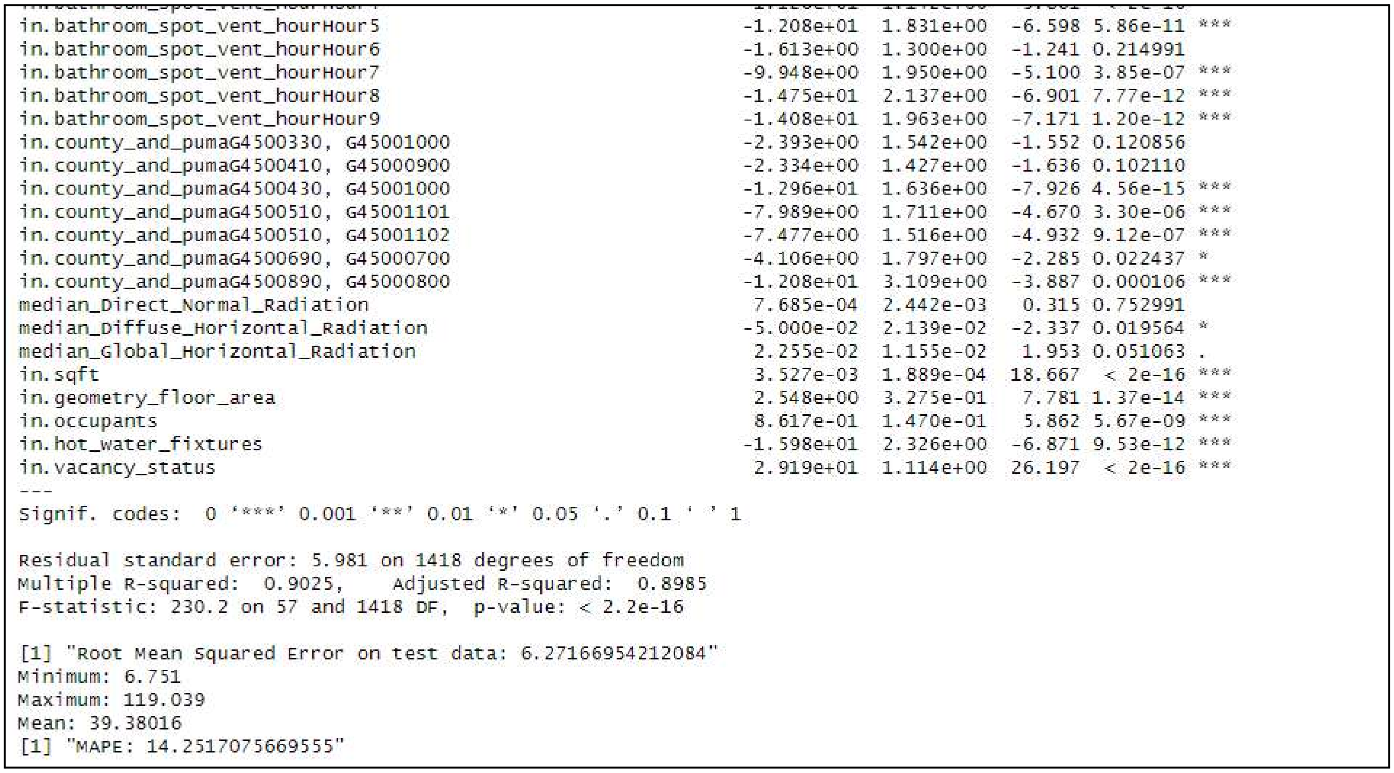


```
# Load required libraries
library(caret)

# Create a copy of the dataset
merge_static_house_info_df4 <- merge_static_house_info_df3

# Choose columns where there are more than one distinct value.
merge_static_house_info_df4 <- merge_static_house_info_df4 %>%
  select(where(~n_distinct(.) > 1))

# Create a copy for prediction
merge_static_house_info_df_prediction <- merge_static_house_info_df4

#Make a subset with the building and county information in specific columns.
merge_static_house_info_df_building_and_county <-
merge_static_house_info_df4[,c('bldg_id','in.county','date')]

# Omit columns that are not required for modeling
merge_static_house_info_df4 <- merge_static_house_info_df4 %>% select(-c('bldg_id','in.county'))

#view(merge_static_house_info_df4)

# Set seed for reproducibility
set.seed(123)

#training and testing sets
index <- createDataPartition(merge_static_house_info_df4$day_total_energy, p = 0.8, list = FALSE)
train_df1 <- merge_static_house_info_df4[index, ]
test_df1 <- merge_static_house_info_df4[-index, ]
```



```
#Develop a linear regression model
 model <- lm(day_total_energy ~ median_Dry_Bulb_Temperature +
    in.dishwasher + median_Wind_Speed + in.hvac_heating_efficiency +
    in.clothes_washer + in.hvac_heating_efficiency +in.bathroom_spot_vent_hour+in.county_and_puma +
     + median_Direct_Normal_Radiation + median_Diffuse_Horizontal_Radiation +
median_Global_Horizontal_Radiation + in.sqft + in.geometry_floor_area + in.occupants +
      in.hot_water_fixtures + in.vacancy_status, data = train_df1)
# Print the summary
summary(model)

# Make predictions on the test set
predictions <- predict(model, newdata = test_df1)


# Calculate RMSE *Root Mean Squared Error* using the test data
rmse <- sqrt(mean((test_df1$day_total_energy - predictions)^2))
print(paste("Root Mean Squared Error on test data:", rmse))


cat("Minimum:", min(test_df1$day_total_energy), "\n")
cat("Maximum:", max(test_df1$day_total_energy), "\n")
cat("Mean:", mean(test_df1$day_total_energy), "\n")

# Calculate MAPE * Mean Absolute Percentage Error* using the test data
mape <- mean(abs((test_df1$day_total_energy - predictions) / test_df1$day_total_energy )) * 100

# Print the result
print(paste("MAPE:", mape))
```

Figure 2 : Code for the Prediction Model



```
in.bathroom_spot_vent_hourHour5                              -1.208e+01  1.831e+00  -6.598 5.86e-11 ***
in.bathroom_spot_vent_hourHour6                              -1.613e+00  1.300e+00  -1.241 0.214991    
in.bathroom_spot_vent_hourHour7                              -9.948e+00  1.950e+00  -5.100 3.85e-07 ***
in.bathroom_spot_vent_hourHour8                              -1.475e+01  2.137e+00  -6.901 7.77e-12 ***
in.bathroom_spot_vent_hourHour9                              -1.408e+01  1.963e+00  -7.171 1.20e-12 ***
in.county_and_pumaG4500330, G45001000                        -2.393e+00  1.542e+00  -1.552 0.120856    
in.county_and_pumaG4500410, G45000900                        -2.334e+00  1.427e+00  -1.636 0.102110    
in.county_and_pumaG4500430, G45001000                        -1.296e+01  1.636e+00  -7.926 4.56e-15 ***
in.county_and_pumaG4500510, G45001101                        -7.989e+00  1.711e+00  -4.670 3.30e-06 ***
in.county_and_pumaG4500510, G45001102                        -7.477e+00  1.516e+00  -4.932 9.12e-07 ***
in.county_and_pumaG4500690, G45000700                        -4.106e+00  1.797e+00  -2.285 0.022437 *  
in.county_and_pumaG4500890, G45000800                        -1.208e+01  3.109e+00  -3.887 0.000106 ***
median_Direct_Normal_Radiation                                7.685e-04  2.442e-03   0.315 0.752991    
median_Diffuse_Horizontal_Radiation                          -5.000e-02  2.139e-02  -2.337 0.019564 *  
median_Global_Horizontal_Radiation                            2.255e-02  1.155e-02   1.953 0.051063 .  
in.sqft                                                       3.527e-03  1.889e-04  18.667  < 2e-16 ***
in.geometry_floor_area                                        2.548e+00  3.275e-01   7.781 1.37e-14 ***
in.occupants                                                  8.617e-01  1.470e-01   5.862 5.67e-09 ***
in.hot_water_fixtures                                        -1.598e+01  2.326e+00  -6.871 9.53e-12 ***
in.vacancy_status                                             2.919e+01  1.114e+00  26.197  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 5.981 on 1418 degrees of freedom
Multiple R-squared:  0.9025,	Adjusted R-squared:  0.8985 
F-statistic: 230.2 on 57 and 1418 DF,  p-value: < 2.2e-16

[1] "Root Mean Squared Error on test data: 6.27166954212084"
Minimum: 6.751 
Maximum: 119.039 
Mean: 39.38016 
[1] "MAPE: 14.2517075669555"
```

Figure 3 : Per hour energy usage prediction

4.4 Demand Analysis

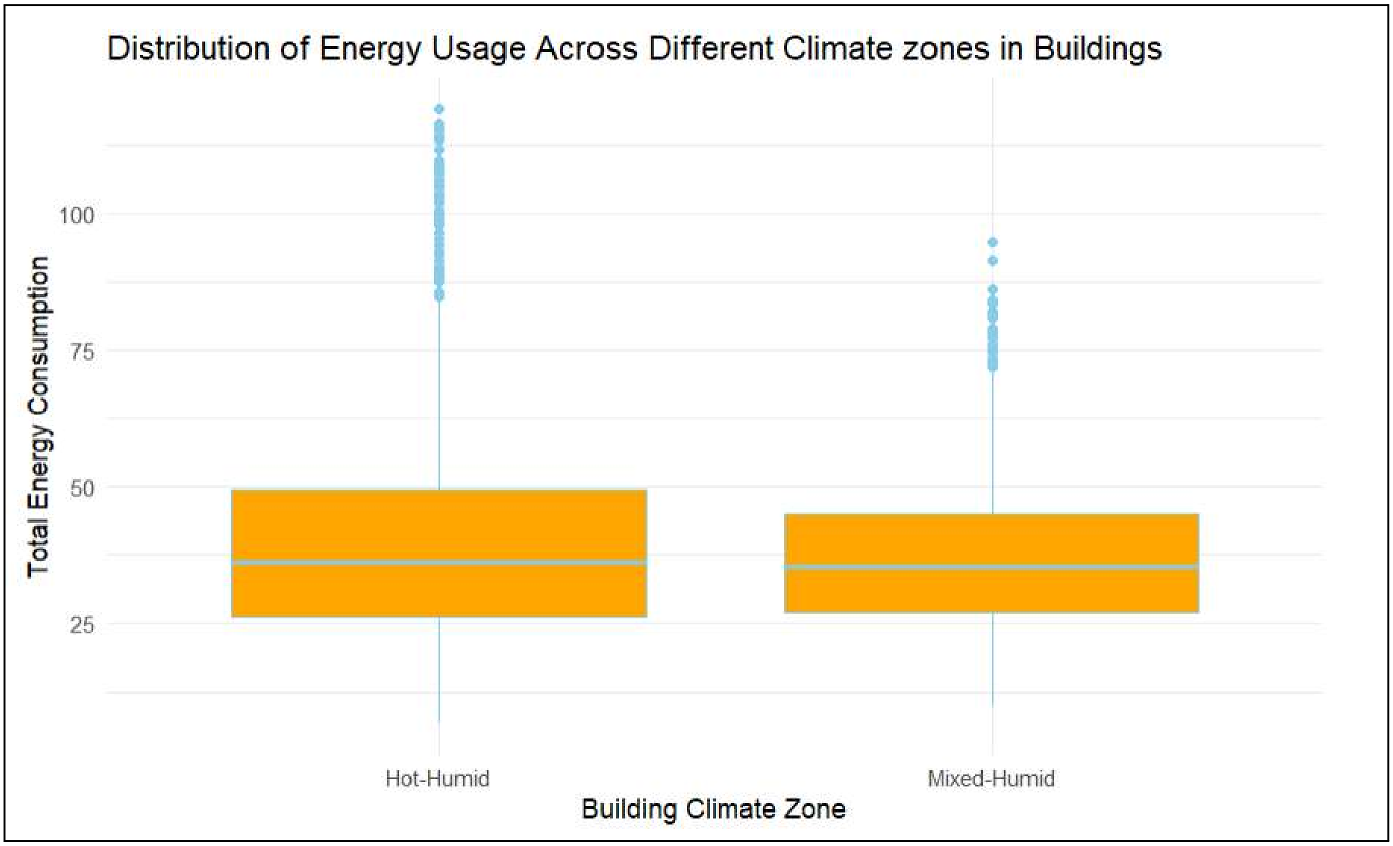


Figure 4 : Distribution of Total Energy Consumption

Box plot analysis shows how building energy use varies depending on the climate zone. Here's the breakdown:

Hot-Humid: Buildings in this zone use energy pretty consistently. This suggests similar building types or energy-saving practices are common.

Mixed-Humid: Energy use is much more spread out here. This points to a wider variety of buildings, or maybe differences in how people use energy. Notice those outliers – those buildings are using way more energy than the rest, making them prime targets for efficiency upgrades.

Overall, the data underscores the importance of considering climate zone influences when building energy management strategies.

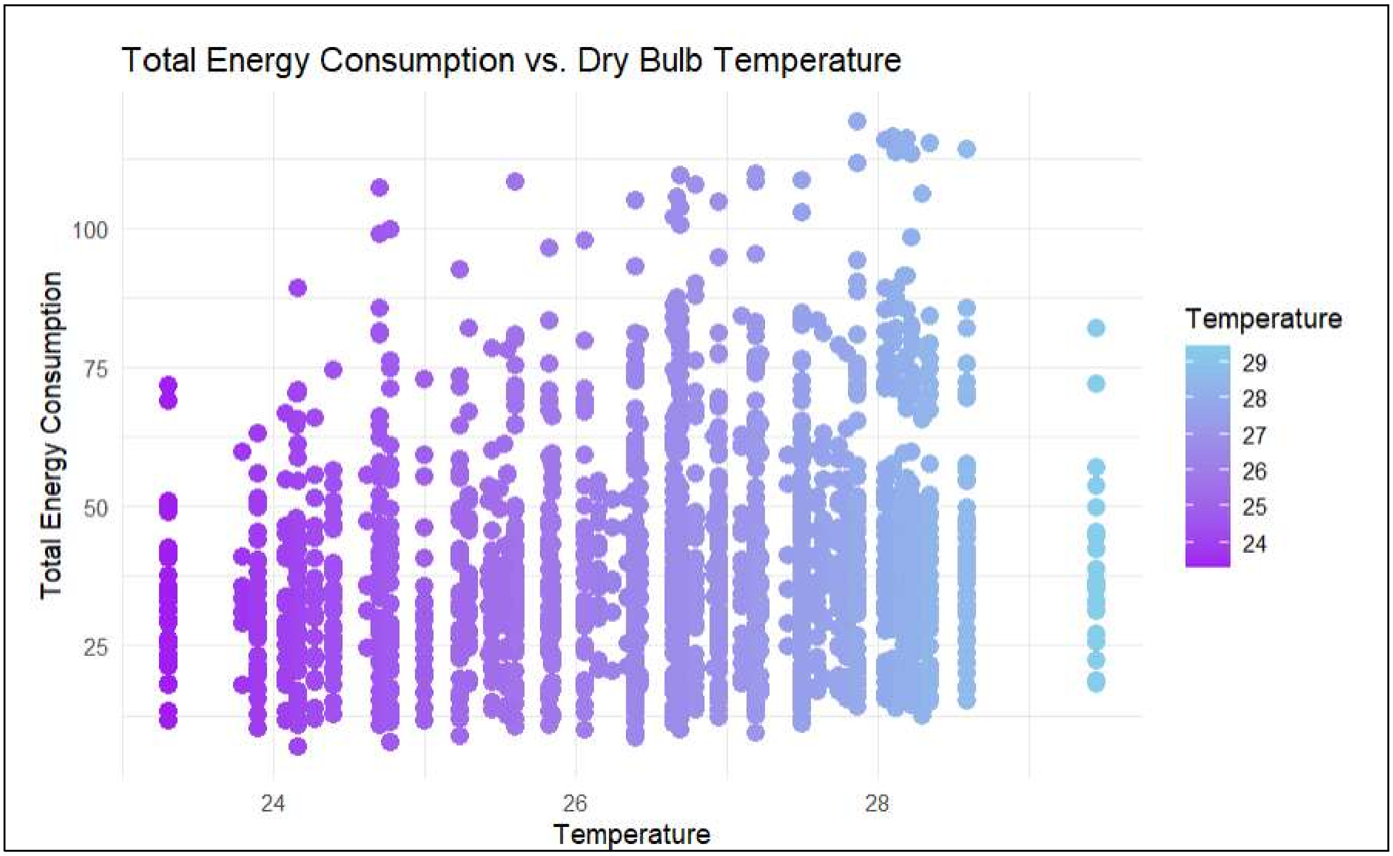


Figure 5 : Total Energy Consumption

The graph indicates that the temperature of the dry bulb and overall energy consumption are positively correlated. This implies that the overall energy usage rises in parallel with the temperature of the dry bulb. This scatterplot reveals a clear connection between temperature and building energy use. Notice how the color changes from purple (coolest) to sky blue (warmest). As temperatures rise, the dots cluster downwards, showing lower energy consumption.In simple terms, buildings use more energy to stay warm than cool. This isn't surprising, but it emphasizes two things:

1. **Climate matters:** Energy-saving plans must consider the local climate and how much heating is needed.

2**. Heating tech is key:** Finding efficient ways to heat buildings will have a major impact on overall energy use.

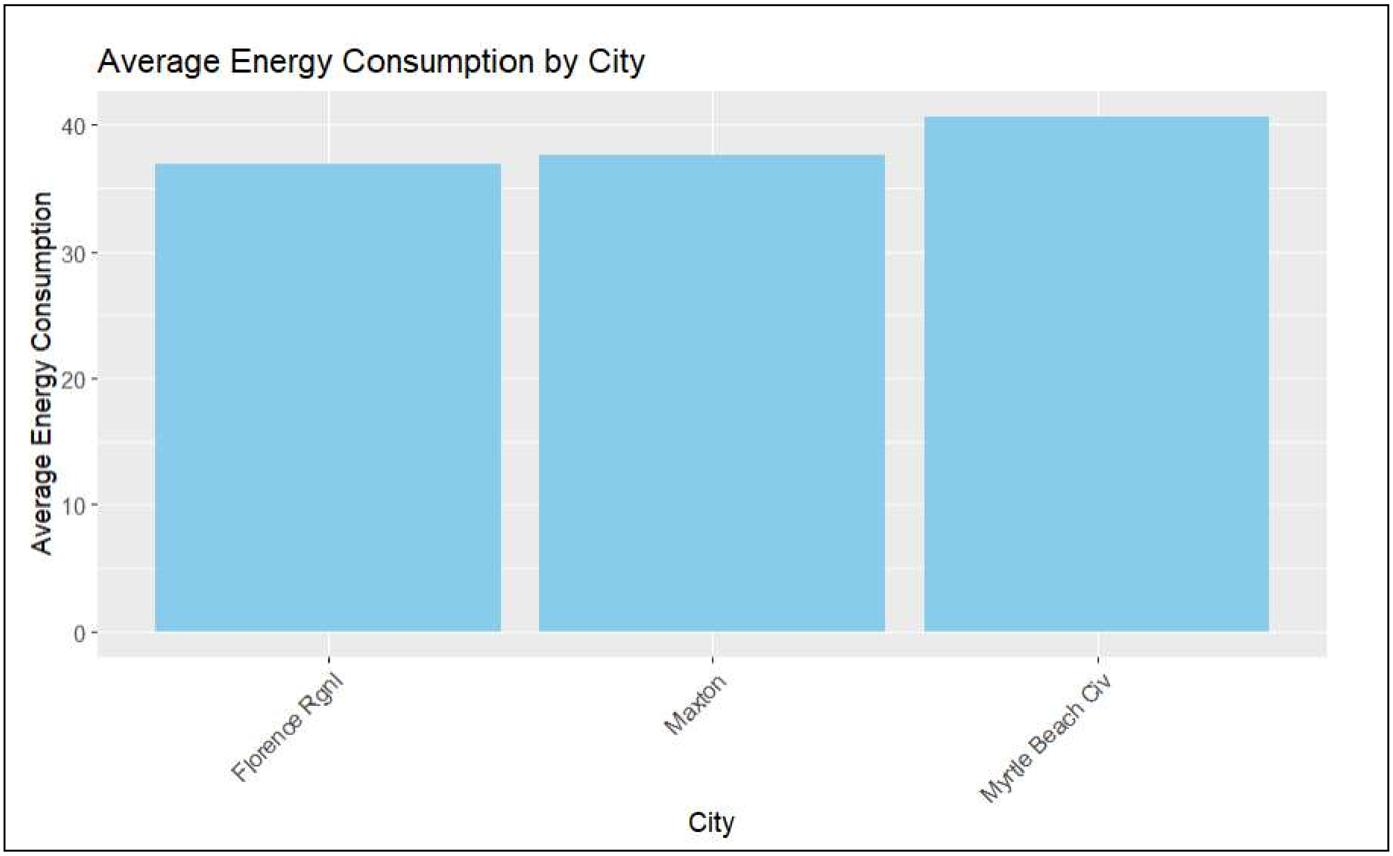


Figure 6 : Average Energy Consumption

The chart illustrates the average energy usage of houses across various cities, with taller bars showing cities where homes consume more energy on average, and shorter bars representing cities with lower average energy consumption. This visual comparison makes it easy to pinpoint which cities have higher or lower energy usage. Differences in these averages could be due to elements like local weather conditions, how densely populated an area is, or the quality of the city's infrastructure. This bar chart reveals something interesting about Florence, Marion, and Myrtle Beach: they all use about the same amount of energy on average, with Myrtle Beach just slightly ahead. What this tells us is, Location alone doesn't seem to be the major reason one of these cities might use more energy than the others.

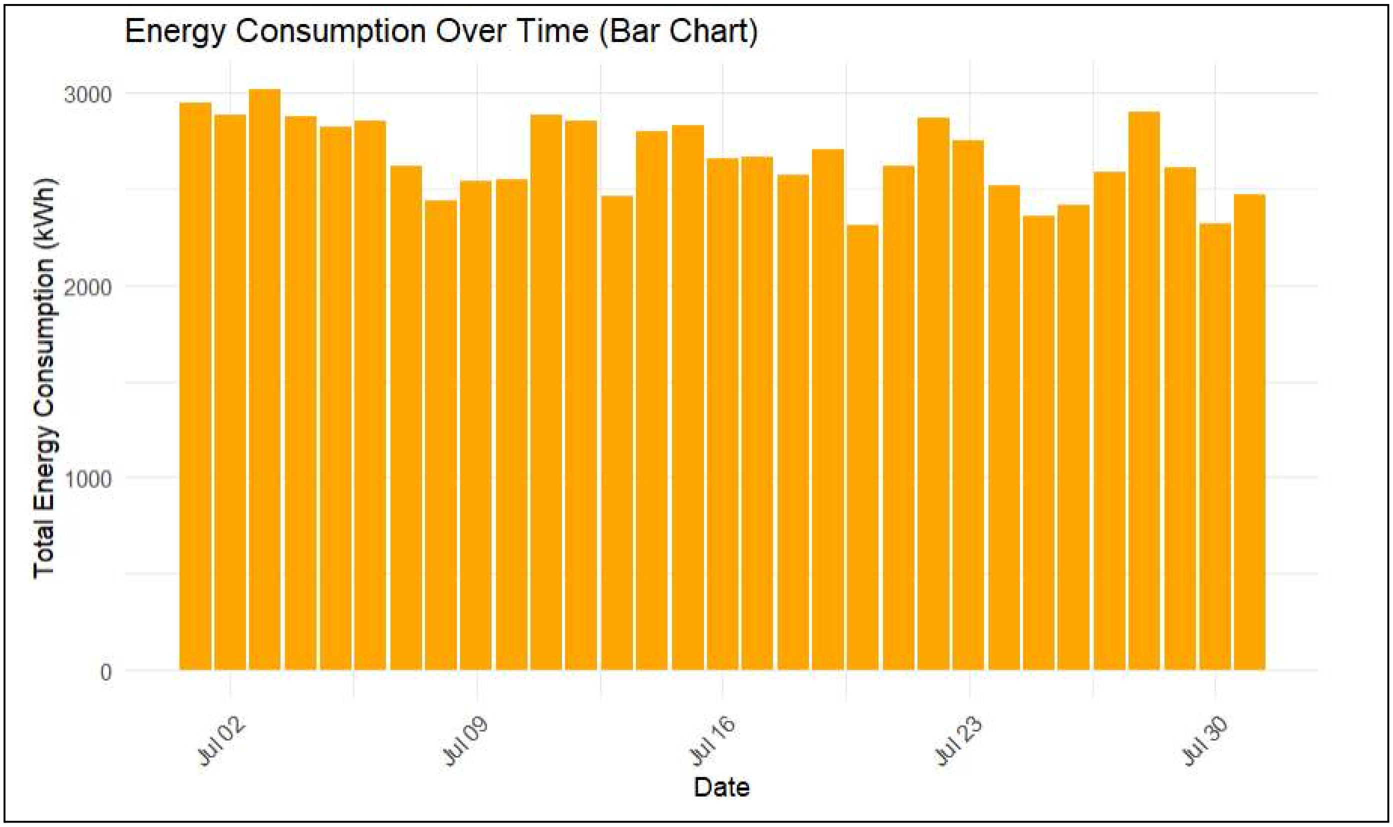


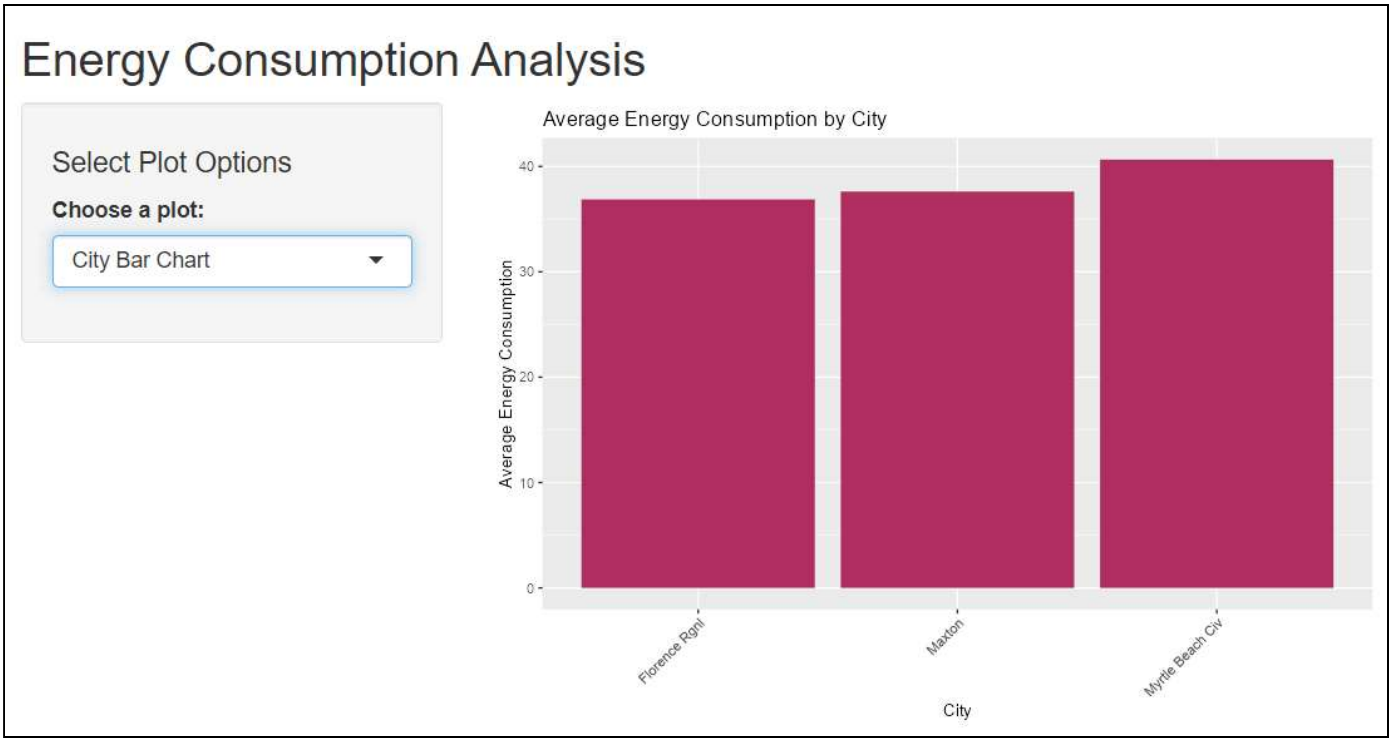
Figure 7 : Energy consumption for the month of July

The bar chart shows that energy use throughout July isn't constant. We see regular spikes in energy use about a week apart. This likely mirrors work schedules or maybe even weekly weather patterns. The biggest spike is at the start of the month (around July 3rd), which might mean a few super hot days drove up AC usage. Interestingly, energy use drops a bit right after each spike, hinting that people might adjust their habits in response. So, if we want to save energy, it's not enough to look at overall usage. We need to target these high-use days with specific strategies.

4.5 Shiny App

Shiny App is valuable for data analysis and visualization because they provide interactive, web-based interfaces that allow users to manipulate and explore data dynamically. In the context of energy consumption analysis, a Shiny app can enable users to select from various graphical representations, such as box plots, bar charts, or scatter plots, through an easy-to-use dropdown menu. This flexibility enhances user engagement by allowing for customized views of the data according to specific needs or interests. By interactively selecting different options, users can uncover unique insights, compare various data points, and make informed decisions based on real-time visual feedback.

URL: https://greeshmashinyapp.shinyapps.io/shiny/



Energy Consumption Analysis

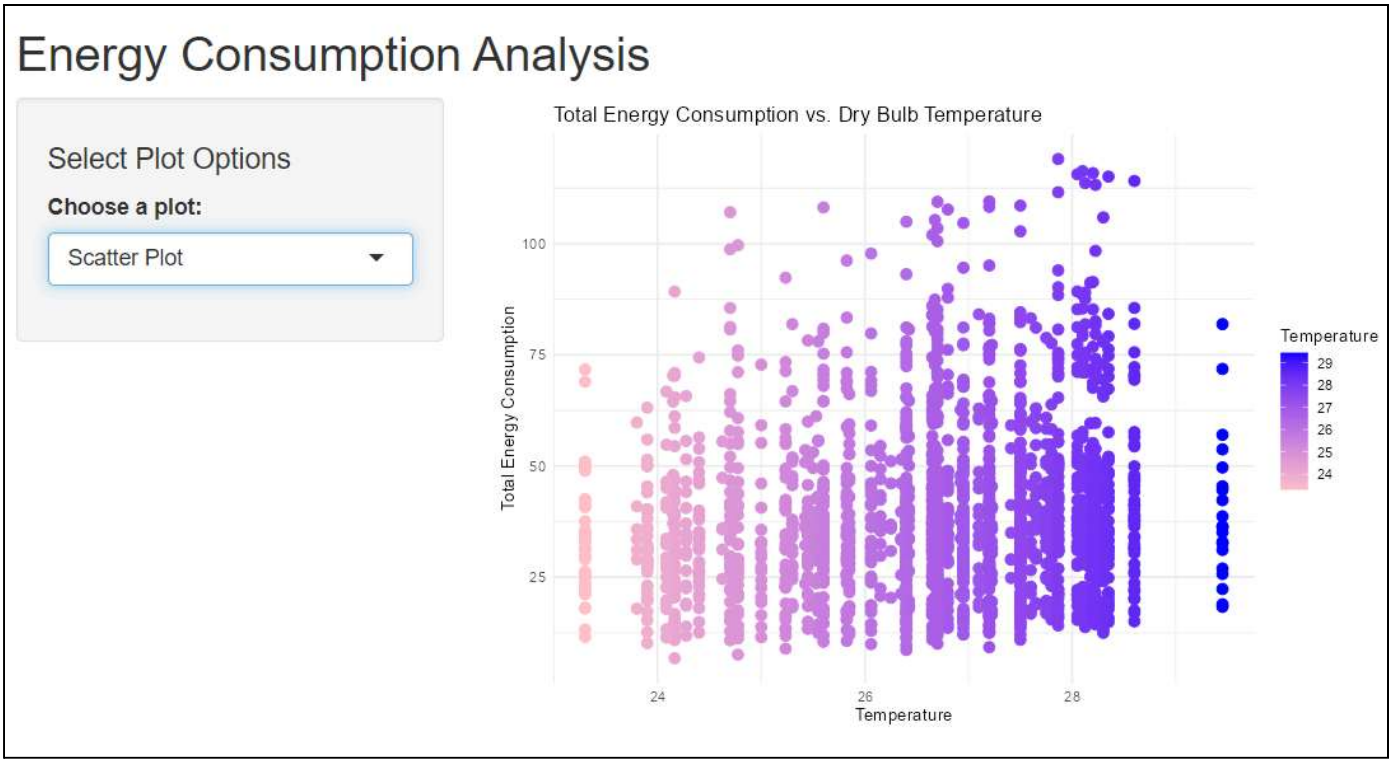
Select Plot Options

Choose a plot:

City Bar Chart ▾

Average Energy Consumption by City
Average Energy Consumption
40
30
20
10
0
Florence Rgnl
Maxton
Myrtle Beach Civ
City

Figure 8 : Shiny App: Energy Consumption Analysis by City

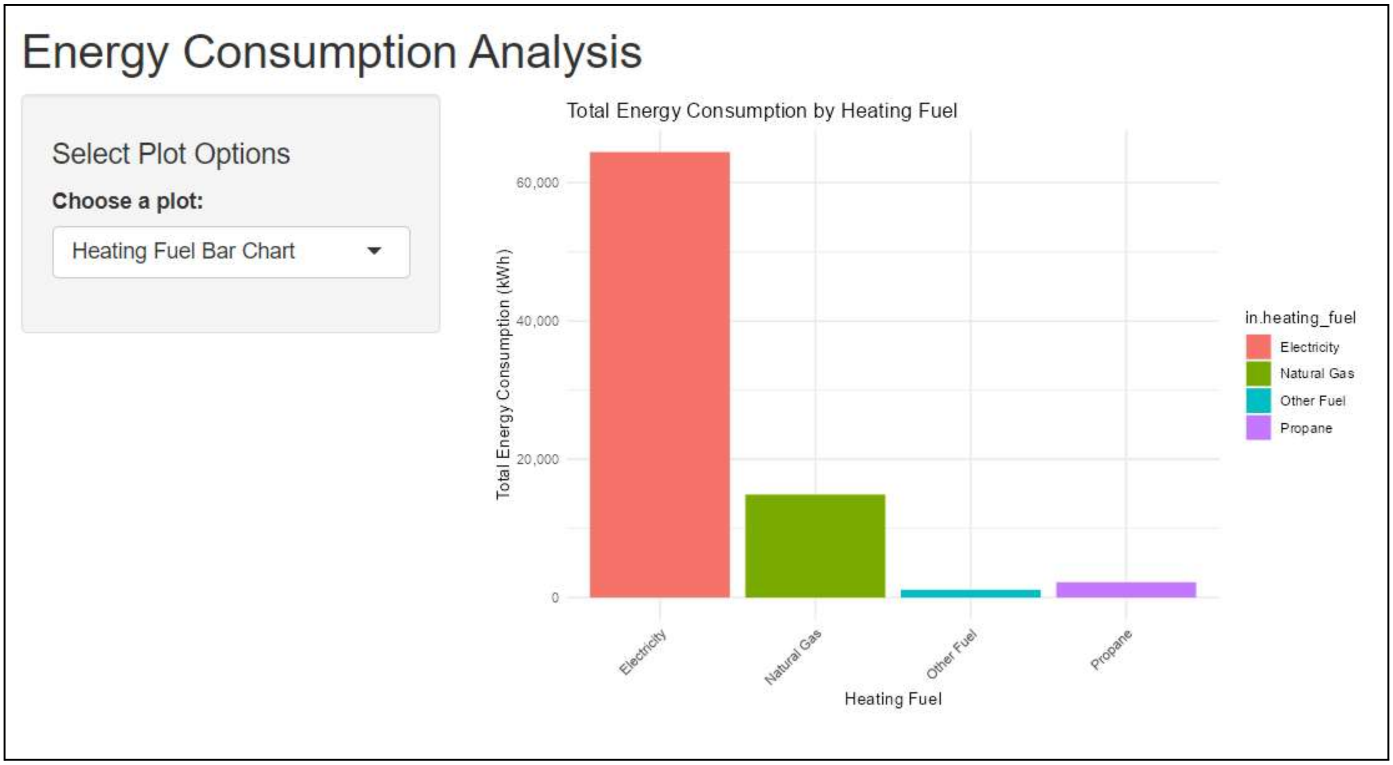


Energy Consumption Analysis

Select Plot Options

Choose a plot: Scatter Plot

Figure 9: Shiny App: Energy Consumption Analysis by Dry Bulb Temperature



Energy Consumption Analysis

Select Plot Options

Choose a plot: Heating Fuel Bar Chart

Figure 10 : Shiny App: Energy Consumption Analysis by Heating Fuel Type

4.6 Demand Reduction Strategy

1. Occupancy Monitoring:
   (a) Utilize data concerning corridor usage and occupancy schedules.
   (b) Implement energy management solutions that utilize real-time occupancy data.
   (c) Adjust lighting, HVAC settings, and other resources based on occupancy patterns, for instance.

2. Incorporating Solar Power:
   (a) Assess the feasibility of incorporating solar power systems.
   (b) Evaluate the building's potential for solar energy generation.
   (c) Install solar panels on rooftops to produce clean energy, for instance.

3. Implementation of Smart Grids:
   (a) Introduce technologies for utility-provider/building communication.
   (b) Enhance energy distribution efficiency via real-time data exchange.
   (c) Deploy smart meters and demand response systems.

4. Monitoring Systems for Energy:
   (a) Deploy continuous energy consumption monitoring systems.
   (b) Identify unusual consumption trends and areas for improvement.
   (c) Utilize smart meters and energy monitoring software.

5. Academic Programs:
   (a) Organize programs to raise awareness about energy-saving practices.
   (b) Promote the adoption of energy-saving behaviors among residents.
   (c) Examples include workshops and educational campaigns.

6. Regular Maintenance:
   (a) Establish maintenance schedules for all systems and equipment.
   (b) Ensure optimal operation of HVAC, lighting, and other systems.
   (c) Conduct regular inspections and replace filters as needed.

7. Evaluation and Testing:
   (a) Conduct pilot initiatives to assess new energy solutions.
   (b) Evaluate effectiveness of each method before widespread implementation.
   (c) Test energy-efficient solutions in specific locations before full deployment.

5 Challenges faced

1. Column Names and Types: Ensuring consistency of column names and data types across datasets to facilitate successful merging.
2. Data Cleaning: Removing null values and handling with imputation methods.
3. Data Matching: Verifying similarity of building IDs used for merging across datasets.
4. Memory Considerations: Depending on dataset sizes, merging large datasets can consume substantial time and memory resources.
5. Model Selection: Choosing and implementing a model to yield precise predictions.

6 Contributions

1. Greeshma: Analyze and merge data, Data Cleaning, Data Modeling, Shiny App
2. Mrunmai: Analyze and merge data, Data Visualization, Report
3. Sinduri: Data Modeling, Data Visualization, Report
4. Sree Chandan: Data Cleaning, Data Visualization, Presentation

7 Conclusion

We express sincere appreciation to Professors Akit Kumar and Christopher Dunham for their invaluable guidance throughout this enriching course. Our involvement in this project presented a blend of gratifying moments and hurdles as we delved into the intricacies of programming and analytical reasoning.

Assessing and contrasting predictions and reports provided valuable insights into the project's effectiveness, shedding light on areas for enhancement and potential analysis bottlenecks. The variety of tasks within the project, coupled with the requirement for ongoing learning, rendered it intellectually stimulating.

In essence, this experience encompassed a mixture of challenges, perpetual learning, and problem-solving, shaping a rewarding journey of growth and development.

References

[1] *ChatGPT*.
[2] *Kanban board and the Data Science book , Professor Akit Kumar had given us access to the tool to work on the project.*