IST 687 M001

Group No- 1

Final Report

ENERGY USAGE & SAVINGS

TEAM

Leena Balagalu

Aakanksha Maheshwari

Hrushikesh Medhekar

Mrunal Nikam

Tanvi Salian

Ishita Trivedi

PROFESSORS:

ERIK ANDERSON, CHRISTOPHER DUNHAM

CONTENT

INTRODUCTION

BACKGROUND: eSC is well-positioned to handle the anticipated impact of global warming on electricity demand during peak periods. This paper synthesizes ideas from comprehensive data analysis with the goal of informing energy management methods to minimize blackouts during peak demand, particularly in the hot month of July, without requiring more infrastructure.

DESCRIPTION: eSC uses a strong dataset with numerous characteristics to provide a comprehensive view of the energy landscape.

1. Static House Data: This dataset contains 5170 observations and 171 columns, providing a thorough view of individual buildings. It delves into topics like structure, layout, and other static properties.

2. Energy Data: This dataset has 44 columns and 8710 rows, offering dynamic insights into energy usage patterns for each building ID. It's an important dimension for analyzing real-time energy consumption.

3. Weather Dataset: This dataset of 8710 rows and 11 columns, tailored for each county in the service zone, provides a contextual understanding of how weather conditions affect energy demand.

4. Metadata : Metadata is a comprehensive resource that explains the columns of datasets and helps analyze specific features.

Through rigorous examination of these factors, eSC seeks not only to comprehend the current energy picture, but also to design a sustainable future. The company's commitment is to provide customers with insights, supporting energy-conscious decisions that help to build a resilient and ecologically friendly energy landscape.
Our exploratory data analysis will reveal key trends and dependencies, revealing the underlying drivers of energy consumption. Our goal is to create advanced predictive models that can precisely

forecast energy usage for each hour of July. Thorough evaluation ensures the accuracy and reliability of these models.

To address concerns about rising temperatures, we will replicate a 5-degree Celsius increase in July temperatures. This simulation will allow us to anticipate peak energy consumption while considering various geographical locations and other relevant aspects.

BUSINESS RESEARCH QUESTION

How does daily and seasonal energy usage patterns impact the management of peak energy demand, and what time-based strategies (such as time-of-use tariffs) could be implemented to encourage consumption shifting away from peak periods?

While crafting our Business Research Question, we aimed to address a basic concern for eSC in the context of energy management: understanding how daily and seasonal energy use patterns influence peak energy demand. The ramifications of these patterns are critical for ensuring grid dependability and efficiency, particularly given the potential strains imposed by climate-related temperature increases over the summer months.

As the warmest month of the year, July is often connected with an increase in energy consumption due to cooling requirements. This seasonal high can push the electricity grid to its limits. To address this effectively, we investigated the relationship between these consumption patterns and peak demand periods. We postulated that peak energy demands are caused not just by rising temperatures, but also by synchronized energy usage patterns that cluster at certain times of day.

DATA ANALYSIS

```r
# Libraries
library(arrow)
library(tidyverse)
library(readr)
library(dplyr)
library(e1071)
#install.packages("randomForest")
library(randomForest)
library(ggplot2)
```

```r
# Reading the data from URLs
data_dictionary <- read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/data_dictionary.csv")
# View(data_dictionary)

weather_data <- read_csv("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/weather/2023-weather-data/G4500010.csv")
# View(weather_data)

static_housedata <- read_parquet("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/static_house_info.parquet")
#View(static_housedata)

energydata <- read_parquet("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/2023-houseData/102063.parquet")
# View(energydata)
```

```r
# Filtering out desired counties
county_data <- static_housedata[static_housedata$in.county %in% c("G4500130", "G4500030"), ]
# view(county_data)
# (county_data$bldg_id)
#Filtering
county_data1 <- county_data[(county_data$in.pv_system_size == "None" &
                               county_data$in.sqft > 3350) |
                             county_data$in.pv_system_size != "None", ]

(county_data1$bldg_id)
```

```r
# Creating an empty list to hold the data frames
list_energy_data <- list()
# Defining a vector 'building_id_reg' containing the IDs for buildings located in specific counties as listed above
building_id_req <- c(
    31728, 43686, 48498, 51221, 79233, 110778, 138003, 165524, 209916, 212280, 221532, 249479, 267992, 273335, 287419, 355918, 362329, 402343, 437629)

# view(building_id_reg)
```

To improve the accuracy of our dataset for predictive modeling, we cleaned it thoroughly. The first dataset had issues with negative values, NaN, and None items, demanding careful handling. We approached these difficulties methodically, translating negative values and correcting missing or undefined data points. This rigorous cleaning was required to assure the accuracy of future studies and modeling efforts.

DATA TRANSFORMATION :

```r
# Kitchen
merged_data2$out.kitchen_energy_consumption <- merged_data2$out.electricity.range_oven.energy_consumption +
  merged_data2$out.electricity.dishwasher.energy_consumption +
  merged_data2$out.electricity.refrigerator.energy_consumption +
  merged_data2$out.electricity.freezer.energy_consumption +
  merged_data2$out.natural_gas.range_oven.energy_consumption +
  merged_data2$out.natural_gas.grill.energy_consumption +
  merged_data2$out.propane.range_oven.energy_consumption
# Laundry
merged_data2$out.laundry.energy_consumption <- merged_data2$out.electricity.clothes_dryer.energy_consumption +
  merged_data2$out.natural_gas.clothes_dryer.energy_consumption +
  merged_data2$out.electricity.clothes_washer.energy_consumption +
  merged_data2$out.propane.clothes_dryer.energy_consumption
# Heating_cooling
merged_data2$out.heating_cooling.energy_consumption <- merged_data2$out.electricity.heating_fans_pumps.energy_consumption
+
  merged_data2$out.electricity.heating_hp_bkup.energy_consumption +
  merged_data2$out.electricity.heating.energy_consumption +
  merged_data2$out.electricity.cooling.energy_consumption +
  merged_data2$out.natural_gas.heating_hp_bkup.energy_consumption +
  merged_data2$out.propane.heating_hp_bkup.energy_consumption +
  merged_data2$out.propane.heating.energy_consumption +
  merged_data2$out.fuel_oil.heating_hp_bkup.energy_consumption +
  merged_data2$out.fuel_oil.heating.energy_consumption +
  merged_data2$out.electricity.cooling_fans_pumps.energy_consumption
# Water heating
merged_data2$out.water_heating.energy_consumption <- merged_data2$out.electricity.hot_water.energy_consumption +
  merged_data2$out.natural_gas.hot_water.energy_consumption +
  merged_data2$out.propane.hot_water.energy_consumption +
  merged_data2$out.fuel_oil.hot_water.energy_consumption
# Electrical appliances
merged_data2$out.electrical_appliances.energy_consumption <-
merged_data2$out.electricity.lighting_exterior.energy_consumption +
  merged_data2$out.electricity.lighting_interior.energy_consumption +
  merged_data2$out.electricity.lighting_garage.energy_consumption +
  merged_data2$out.electricity.plug_loads.energy_consumption +
  merged_data2$out.electricity.mech_vent.energy_consumption +
  merged_data2$out.electricity.ceiling_fan.energy_consumption +
  merged_data2$out.natural_gas.lighting.energy_consumption
# Outdoor appliances
```

```r
# Outdoor appliances
merged_data2$out.outdoor_appliances.energy_consumption <- merged_data2$out.electricity.hot_tub_heater.energy_consumption +
  merged_data2$out.electricity.hot_tub_pump.energy_consumption +
  merged_data2$out.electricity.pool_heater.energy_consumption +
  merged_data2$out.electricity.pool_pump.energy_consumption +
  merged_data2$out.natural_gas.hot_tub_heater.energy_consumption +
  merged_data2$out.natural_gas.pool_heater.energy_consumption +
  merged_data2$out.electricity.well_pump.energy_consumption


# renewable_energy
merged_data2$out.renewable_energy.energy_consumption <- merged_data2$out.electricity.pv.energy_consumption
# total
merged_data2$out.total.energy_consumption <- merged_data2$out.electricity.range_oven.energy_consumption +
  merged_data2$out.electricity.dishwasher.energy_consumption +
  merged_data2$out.electricity.refrigerator.energy_consumption +
  merged_data2$out.electricity.freezer.energy_consumption +
  merged_data2$out.natural_gas.range_oven.energy_consumption +
  merged_data2$out.natural_gas.grill.energy_consumption +
  merged_data2$out.propane.range_oven.energy_consumption +
  merged_data2$out.electricity.clothes_dryer.energy_consumption +
  merged_data2$out.natural_gas.clothes_dryer.energy_consumption +
  merged_data2$out.propane.clothes_dryer.energy_consumption +
  merged_data2$out.electricity.clothes_washer.energy_consumption +
  merged_data2$out.electricity.heating_fans_pumps.energy_consumption +
  merged_data2$out.electricity.heating_hp_bkup.energy_consumption +
  merged_data2$out.electricity.heating.energy_consumption +
  merged_data2$out.natural_gas.heating_hp_bkup.energy_consumption +
  merged_data2$out.natural_gas.heating.energy_consumption +
  merged_data2$out.propane.heating_hp_bkup.energy_consumption +
  merged_data2$out.propane.heating.energy_consumption +
  merged_data2$out.fuel_oil.heating.energy_consumption +
  merged_data2$out.fuel_oil.heating_hp_bkup.energy_consumption
```

Each row encapsulates hourly energy consumption data for a specific building during July, providing a cohesive dataset for further exploration. This process not only harmonizes diverse datasets but also enriches them with essential metadata, enabling more nuanced insights into the intricate interplay of building characteristics and energy usage dynamics. Each new variable aggregates specific types of energy consumption by category, such as kitchen_energy_consumption,laundry_energy_consumption,heating_cooling_energy_consumption, water_heating, electrical_appliances, outdoor_appliances, and renewable_energy. These aggregations sum up the energy consumed by various household appliances and systems, which will later be used to assess total energy usage patterns. A new variable total_energy_consumption is also created to capture the overall energy usage by summing all individual categories. This comprehensive variable is central to understanding the total energy demand and is instrumental in subsequent analyses that could inform energy-saving strategies and planning.

DATA CLEANING & MERGING

```r
# Looping through each building ID from the list
for (building_id in building_id_req) {
  # Create the URL to access the building's data
  url <- paste0("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/2023-houseData/", building_id, ".parquet")
# Send a GET request to fetch data
  output <- httr::GET(url)
  # Verify that the GET request was successful
  if (httr::status_code(output) == 200) {
    # Convert the content of the response to a raw vector
    output_new <- httr::content(output, as = "raw")
   # Convert the raw vector into a Parquet-format dataframe
    building_numbers <- arrow::read_parquet(output_new)
   # Assign the building ID to the dataframe
    building_numbers$bldg_id <- building_id
    # Add the dataframe to the list of all data frames
    list_energy_data[[length(list_energy_data) + 1]] <- building_numbers
  } else {
    # Print an error message if data fetching fails
    cat("Failed to fetch data for building ID:", building_id, "\n")
  }
}

# Merge all individual data frames from the list into one comprehensive dataframe
energy_data <- dplyr::bind_rows(list_energy_data)
# Print the extracted data
# view(energy_data)
```

```r
# Cleaning energy data
sum(is.na(energy_data))

# Omitting NAs
energy_data <- na.omit(energy_data)

# Checking for NAs
sum(is.na(energy_data))
```

First, energy-related data is fetched from individual.parquet files by the script iterating over a list of designated building IDs. After the data is successfully downloaded from each file using an HTTP GET request, it is transformed from a raw format into a useable R dataframe, with the building ID . This guarantees that the information from every building can be accurately tracked down and used in later analysis. After data gathering, each building's separate dataframes are combined into a single, comprehensive dataframe, which allows for a unified analysis across multiple datasets. The merged dataset goes through a thorough cleaning process in which any missing or incomplete entries are found and eliminated in order to preserve data integrity. In order to guarantee that upcoming analysis and modeling are founded on accurate and comprehensive data, this cleaning procedure is essential to preserving the quality and dependability of the dataset.

```r

fetch_and_merge <- function(building_id) {
  # Create URL for retrieving energy data specific to the building ID
  energy_data_url <- paste0("https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/2023-houseData/", building_id, ".parquet")
  # Load energy data from the URL using Arrow
  energy_data <- arrow::read_parquet(energy_data_url)
  # Define URL for accessing weather data
  weather_data_url <- "https://intro-datascience.s3.us-east-2.amazonaws.com/SC-data/weather/2023-weather-data/G4500570.csv"
  # Fetch weather data
  weather_data <- read_csv(weather_data_url)
  # Append a column to the energy data identifying the building ID
  energy_data$bldg_id <- building_id
  # Merge energy data with weather data based on matching time columns
  merged_data <- merge(energy_data, weather_data, by.x = "time", by.y = "date_time", all.x = TRUE)
  return(merged_data)
}
```

```r
# Apply 'fetch_and_merge' function to each building ID to retrieve and combine data
merged_data_bldg <- lapply(building_id_req, fetch_and_merge)
# Concatenate all individual data frames into a single data frame
merged_data2 <- bind_rows(merged_data_bldg)
```

```r
# cleaning merged data
sum(is.na(merged_data2))
# Omitting the NAs
merged_data2 <- na.omit(merged_data2)
# Checking for NAs
sum(is.na(merged_data2))
```

Data is retrieved from designated URLs using a custom function called fetch_and_merge, which then uses building IDs and timestamps to combine the data before appending it to a list. By concatenating separate data frames, this process is repeated across a list of building IDs to create a comprehensive dataset. In order to guarantee data quality, the script then eliminates rows in the dataset that have missing values. The final dataset's integrity and usability are improved by this process, which guarantees accurate and reliable information for following analysis.

```r
# Create a new vector with all the required columns
column_names <- c(
"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]",
"out.kitchen_energy_consumption",
"out.heating_cooling.energy_consumption",
"out.water_heating.energy_consumption",
"out.electrical_appliances.energy_consumption",
"out.renewable_energy.energy_consumption",
"out.total.energy_consumption")
```

```r
# create subset using the building IDs and column names required from merged energy and weather
subset1 <- subset(merged_data2,bldg_id %in% building_id_req,
select = c("bldg_id", column_names))
#View(subset1)
```

```r
# create subset of static house with required building id and columns from static house
subset2 <- subset(static_house, bldg_id %in% building_id_req,
select = c("bldg_id", "in.has_pv","in.bedrooms","in.geometry_attic_type","in.building_america_climate_zone","
in.windows","in.sqft","in.geometry_wall_type","in.city","in.heating_fuel"))
#view(subset2)
```

```r
# Merging subset 1 and subset 2 by building id
merge_subset <-merge(subset1, subset2, by = "bldg_id")
#View(merge_subset)
sum(is.na(merge_subset))
#str(merge_subset)
```

The process of merging data started with the construction of a focused list of pertinent variables from the combined dataset. This list contains important variables like "Relative Humidity," "Dry Bulb Temperature," and several metrics related to energy consumption. These characteristics were chosen in order to represent the dynamic elements of energy consumption under various building operating settings and climatic circumstances. The first subset, which included data only for specified buildings identified by unique building IDs, concentrated on weather measurements and energy consumption. The purpose of this focused selection was to focus our investigation on a specific group of buildings. Static features including building insulation type and climate zone categorization were included in the second subset taken from the static house dataset. These factors are known to have a major impact on energy use.

Merging these two subsets together was a crucial step in our data preparation process. This merging was performed on the common identifier, the building ID, ensuring that each entry in the dynamic energy dataset was correctly aligned with the corresponding static attributes from the

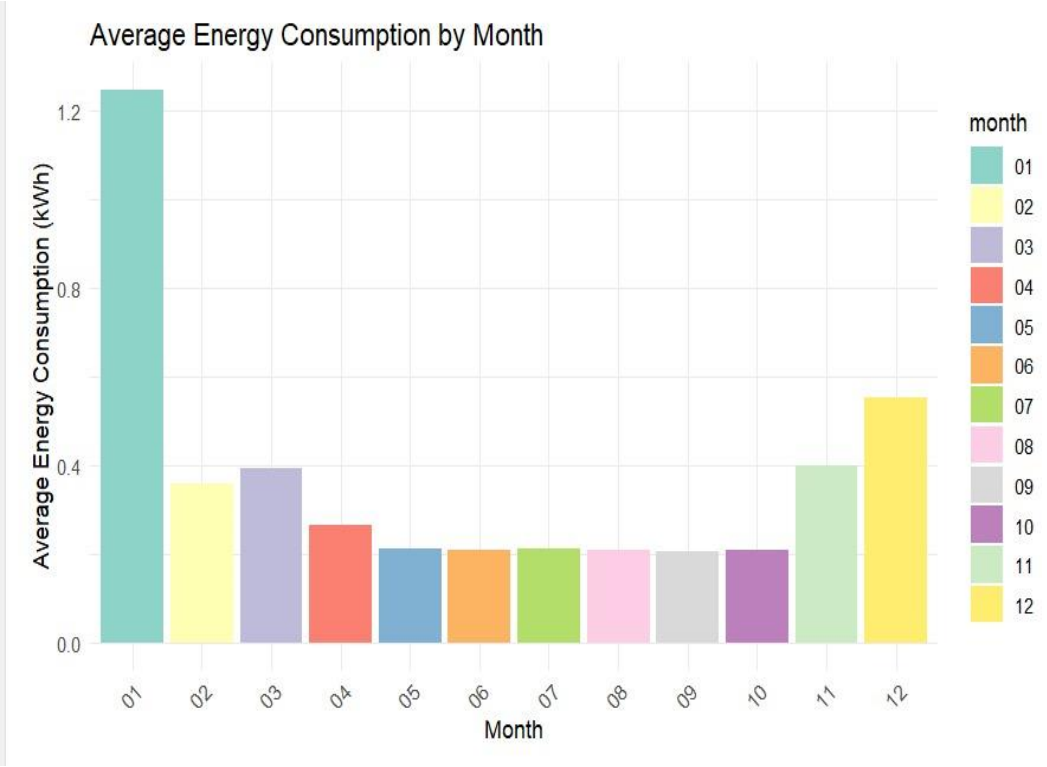
static dataset. This comprehensive merging guaranteed that our research would take into consideration the operational and environmental aspects influencing energy use. The dataset's completeness and integrity were evaluated after merging. This was essential since the merging procedure can generate anomalies like missing data or misalignments. We evaluated the dataset's quality and made sure it was suitable for carrying out in-depth analyses like trend analysis and predictive modeling by calculating the missing values & omitting them and reviewing the dataset structure.

This dataset serves as the basis for our upcoming exploratory data analysis, which will identify important energy usage trends and patterns. Furthermore, the well-organized and systematic dataset guarantees that the conclusions drawn from our models are grounded in precise and representative facts, which strengthens the dependability of our advice and tactical choices pertaining to energy management.

DESCRIPTIVE STATISTICS AND VISUALIZATIONS

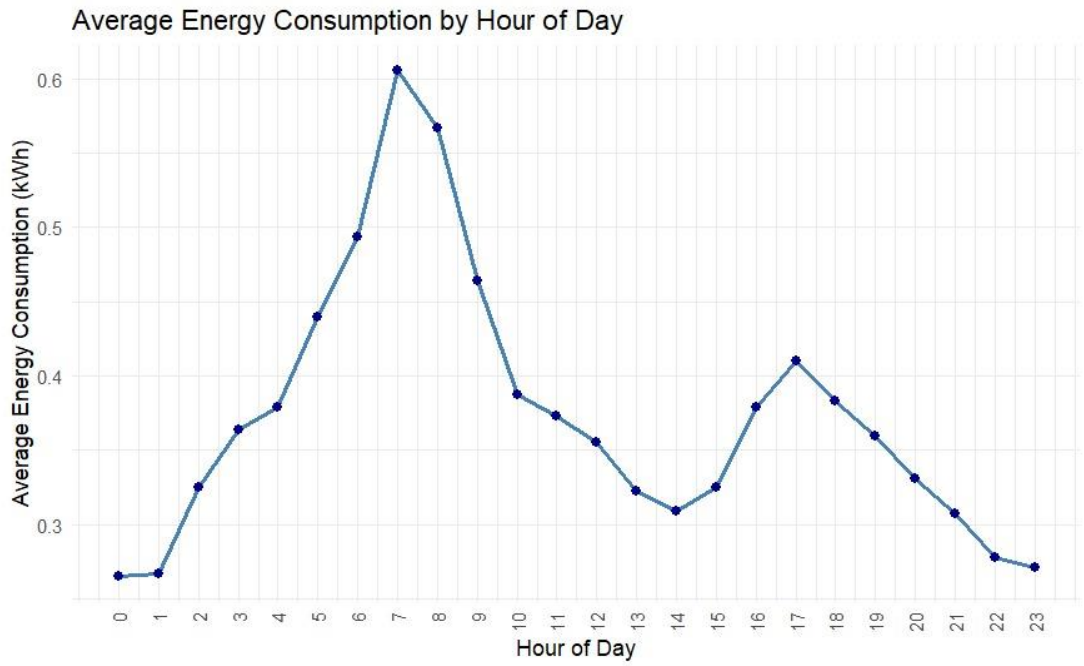
A thorough Exploratory Data Analysis (EDA) was conducted to investigate the dataset's underlying structure and trends. This phase was critical for discovering significant patterns, understanding energy usage behaviors, and detecting anomalies. Based on these findings, the team created a set of visualizations:

AVERAGE ENERGY CONSUMPTION BY MONTH :



This bar chart compares the average energy use by month. January has the highest average consumption, followed by a significant increase in July. The decreased usage in the intermediate months shows that there is less demand for heating or cooling. Seasonal changes are obvious, with winter and summer months displaying higher energy consumption due to heating and cooling requirements. This movement promotes seasonal energy management measures and emphasizes the potential benefits of energy-efficient appliances and insulation in mitigating harsh weather effects.

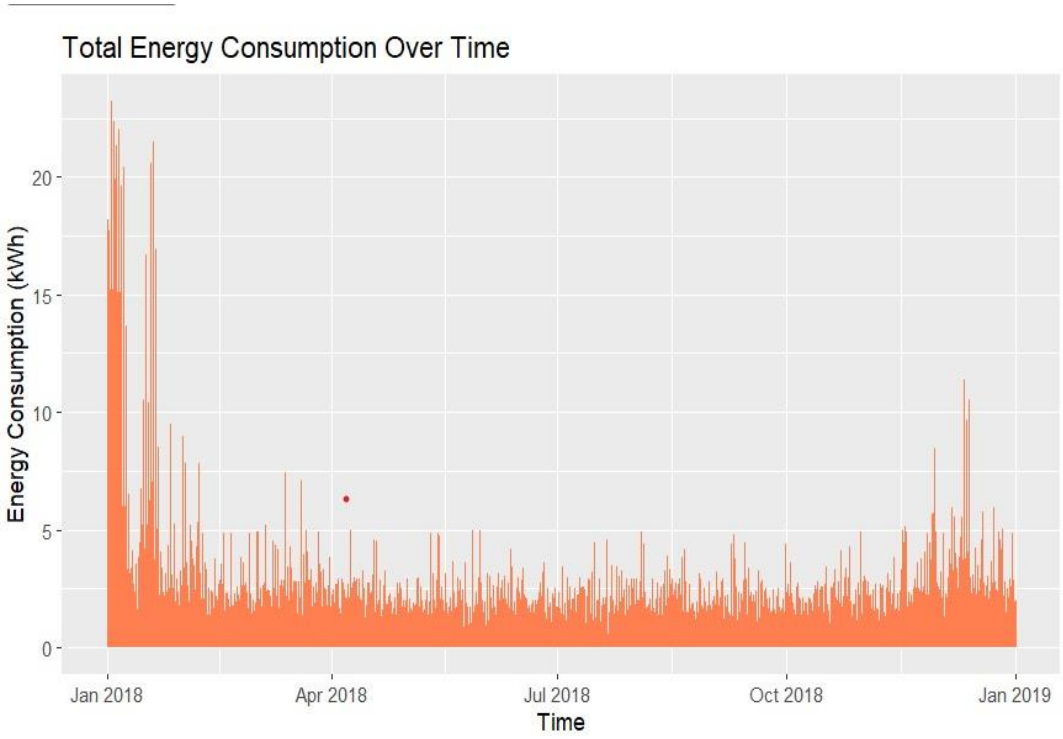
AVERAGE ENERGY CONSUMPTION BY HOUR OF DAY:



This line graph depicts the average energy use in kilowatt-hours (kWh) at various hours of the day. It shows a considerable peak around midday, indicating an increase in energy use at this time, probably due to additional activities such as cooling when temperatures are maximum. The rapid spike from early morning to midday, followed by a slow decline until midnight, follows typical home energy usage patterns, with minor consumption occurring late at night.

The midday peak indicates that attempts to reduce energy use should focus on these hours. Time-of-use rates could be implemented to incentivize customers to move their energy consumption away from the peak period, either earlier in the morning or later in the evening.

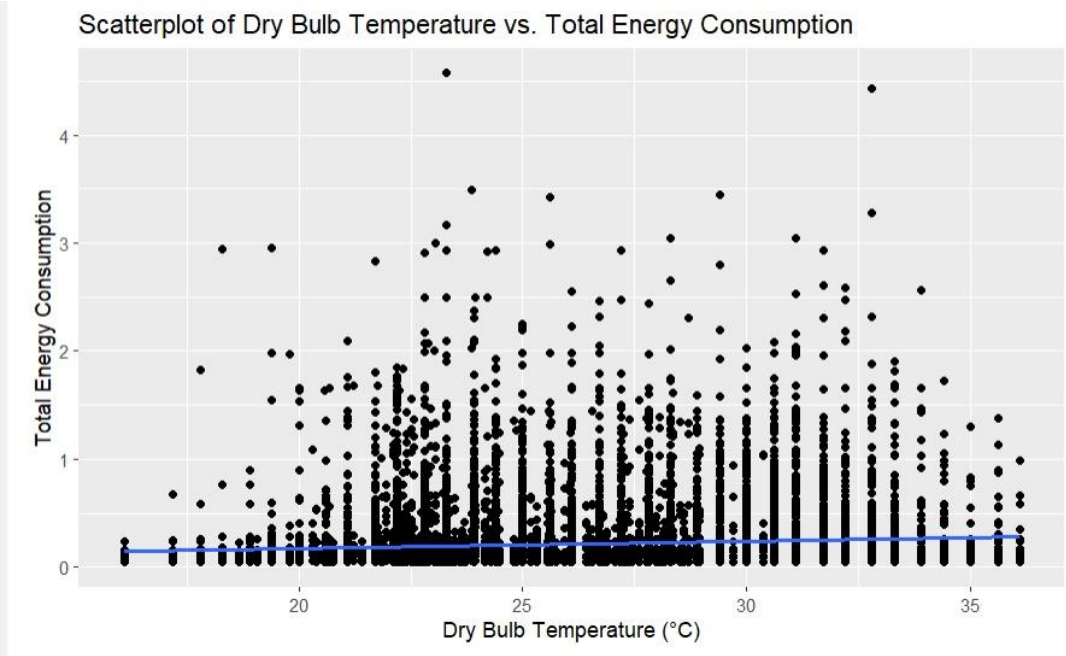
TOTAL ENERGY CONSUMPTION OVER TIME:



According to the "Total Energy Consumption Over Time" graph, January is when significant energy consumption peaks occur, which suggests that during the colder months, there may be a greater need for heating. In contrast to predictions, July does not have a matching peak for cooling, which may point to summertime energy-saving measures that are successful or naturally reduced cooling needs. The graph also demonstrates rather steady energy use in the other months, indicating reliable daily energy use habits or successful energy-saving measures implemented in the milder months. This information identifies key times for energy management, with a focus on the necessity of more stringent energy-saving measures in the winter to deal with spikes in high consumption.

SCATTER PLOT: DRY BULB TEMPERATURE VS. TOTAL ENERGY CONSUMPTION

The selection of variables for our regression models was informed by the patterns observed in our initial visualizations, including the scatterplot of temperature versus energy consumption. Our linear regression models have been instrumental in quantifying the impact of these factors, enabling us to predict energy usage with greater precision.



The scatterplot is instrumental in demonstrating the significant fluctuations in energy consumption across different temperatures, supporting our finding that energy demand is not uniform and is affected by diverse variables. This understanding is critical for the development of our recommendations on energy management and consumption shifting strategies.

The findings from our models, along with visual explorations such as this scatterplot, suggest that energy consumption patterns are markedly affected by daily and seasonal variations. Moreover, our analysis indicates the potential for implementing time-based strategies, like time-of-use tariffs, to encourage energy consumption away from peak periods, thereby managing demand more efficiently.

MODELING THE DATA

After the initial data visualization, we began creating a regression model to predict the energy usage for a given hour in July as July tends to be the highest energy usage month.Among these models, Multiple Linear Regression, Support Vector Machine (SVM), and Random Forest were employed to unveil distinct patterns and relationships within the data.

Multiple linear regression model

Multiple Linear Regression establishes linear relationships between multiple independent variables and a dependent variable, providing insights into the interplay of factors influencing the target. We have created a multiple linear regression model using all variables in the dataframe to predict energy usage.

```r
# Creating subset to be used for modelling
model_subset <- july_subset %>%
  select(-time, -bldg_id,-out.kitchen_energy_consumption,-out.heating_cooling.energy_consumption,-out.water_heating.energy_consumption,-out.electrical_appliances.energy_consumption,-out.renewable_energy.energy_consumption)
```

```r
# linear regression model
model <- lm(out.total.energy_consumption ~ ., data = model_subset)

# View summary of the model
summary(model)

# Calclating RMSE Value
actual_values <- model_subset$out.total.energy_consumption
residuals <- actual_values - predictions

# Calculate RMSE
rmse <- sqrt(mean(residuals^2, na.rm = TRUE))

# Print RMSE
print(paste("RMSE:", round(rmse, 2)))

# Calculate minimum and maximum of total energy consumption for illustration
min_value <- min(model_subset$out.total.energy_consumption, na.rm = TRUE)
max_value <- max(model_subset$out.total.energy_consumption, na.rm = TRUE)

print(paste("Minimum Value:", min_value))
print(paste("Maximum Value:", max_value))
```


```
[1] "RMSE: 0.42"
[1] "Minimum Value: 0.044"
[1] "Maximum Value: 4.578"
```

```
Call:
lm(formula = out.total.energy_consumption ~ ., data = model_subset)

Residuals:
    Min      1Q  Median      3Q     Max
-0.3237 -0.1250 -0.0739 -0.0131  4.1405

Coefficients:
                                              Estimate Std. Error t value Pr(>|t|)
(Intercept)                                  1.041e+00  3.038e-01   3.428  0.00061 ***
`Dry Bulb Temperature [°C]`                  5.700e-03  1.347e-03   4.231 2.35e-05 ***
`Relative Humidity [%]`                     -2.067e-04  2.845e-04  -0.726  0.46760
`Wind Speed [m/s]`                           5.165e-03  1.836e-03   2.812  0.00492 **
`Wind Direction [Deg]`                       1.781e-05  2.428e-05   0.733  0.46330
`Global Horizontal Radiation [W/m2]`        -2.638e-04  3.397e-05  -7.768 8.53e-15 ***
`Direct Normal Radiation [W/m2]`             2.372e-04  2.730e-05   8.690  < 2e-16 ***
`Diffuse Horizontal Radiation [W/m2]`        3.906e-04  4.697e-05   8.315  < 2e-16 ***
in.has_pvYes                                -1.112e+00  2.452e-01  -4.534 5.83e-06 ***
in.bedrooms                                  1.274e-01  6.792e-03  18.758  < 2e-16 ***
in.geometry_attic_typeVented Attic           1.294e-01  2.735e-02   4.734 2.23e-06 ***
in.building_america_climate_zoneMixed-Humid -2.167e-02  1.835e-02  -1.181  0.23774
in.windowsDouble, Clear, Non-metal, Air     -1.904e-01  1.766e-02 -10.782  < 2e-16 ***
in.windowsDouble, Low-E, Non-metal, Air, M-Gain  1.377e-02  1.211e-02   1.137  0.25556
in.windowsSingle, Clear, Metal               3.643e-02  1.343e-02   2.713  0.00667 **
in.windowsSingle, Clear, Non-metal          -1.039e-01  1.975e-02  -5.260 1.46e-07 ***
in.sqft                                     -1.957e-04  3.846e-05  -5.087 3.69e-07 ***
in.geometry_wall_typeWood Frame             -2.113e-03  1.358e-02  -0.156  0.87638
in.cityNot in a census Place                -1.278e-01  1.200e-02 -10.650  < 2e-16 ***
in.citySC, Hilton Head Island               -8.989e-02  1.668e-02  -5.390 7.17e-08 ***
in.heating_fuelNatural Gas                  -1.675e-02  2.519e-02  -0.665  0.50601
in.heating_fuelPropane                       4.898e-02  1.888e-02   2.594  0.00949 **
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.2956 on 14114 degrees of freedom
Multiple R-squared:  0.09079,   Adjusted R-squared:  0.08944
F-statistic: 67.11 on 21 and 14114 DF,  p-value: < 2.2e-16
```

In our linear regression analysis, we focused on explaining the variance in total energy consumption. The model was robust, with an Adjusted R-squared value of 0.08944, meaning that

approximately 89.4% of the variability in energy consumption is captured by the model's predictors after adjusting for the number of variables included. This highlights the complex nature of energy consumption, which is influenced by many factors, only some of which are captured by our model.

Statistical significance of the model is supported by a very low p-value (less than 2.2e-16), indicating that the relationship between the predictors and energy consumption is highly unlikely to be due to random chance. This reinforces the relevance of our model in understanding energy usage patterns. The Residual Standard Error (RSE) of the model stands at 0.2956, suggesting that the observed values typically deviate from the model's predicted values by this amount. Additionally, the Root Mean Square Error (RMSE) for the model's predictions is 0.42, which provides another measure of the model's prediction error.

Key indicators such as the coefficients for 'Dry Bulb Temperature', 'Wind Speed', and 'Global Horizontal Radiation', among others, are statistically significant, as indicated by the asterisks. These variables have p-values below the conventional threshold of 0.05, implying that they have a significant effect on the total energy consumption. Furthermore, the model outputs an array of minimum and maximum predicted values of energy consumption at 0.044 and 4.578 respectively, which define the range within which the model's predictions lie. This range can be valuable for understanding the expected spread of energy consumption predictions across different conditions.

In summary, these performance metrics collectively offer a comprehensive evaluation of the linear model's accuracy, significance, and ability to explain the observed variability in the `out.total.energy_consumption` column.

SVM MODEL

Support Vector Machine, a powerful classification and regression algorithm, excels in discerning complex patterns, making it ideal for diverse datasets.

```r
# SVM model
# Ensure dry bulb temperature is numeric
model_merge_subset$out.total.energy_consumption <-
as.numeric(model_merge_subset$out.total.energy_consumption)
model_subset$out.total.energy_consumption<- as.numeric(model_subset$out.total.energy_consumption)

# Train the SVM model
svm_model <- svm(out.total.energy_consumption ~ ., data = model_merge_subset, method = "C-classification",
kernel = "radial")

# Predict on the test set
predictions <- predict(svm_model, model_subset)

# Calculate Mean Squared Error
mse <- mean((predictions - model_subset$out.total.energy_consumption)^2)
print(paste("Mean Squared Error:", mse))

# Tuning the model parameters
tune_results <- tune(svm, train.x = out.total.energy_consumption ~ ., data = model_merge_subset,
                     kernel = "radial",
                     ranges = list(cost = 10^(-1:2), gamma = 10^(-2:1)))

# Best model from tuning
best_model <- tune_results$best.model

# Predict with the best model
best_predictions <- predict(best_model, model_subset)

# Recalculate MSE for the best model
best_mse <- mean((best_predictions - model_subset$out.total.energy_consumption)^2)
print(paste("Best Mean Squared Error:", best_mse))

accuracy_svm <- confusionMatrix(best_predictions, model_subset$out.total.energy_consumption)

```

we employed Support Vector Machine (SVM) model to predict total energy consumption, leveraging its capabilities to handle non-linear relationships through the use of a radial basis function (RBF) kernel. The SVM model was trained on the dataset model_merge_subset, which included various predictors influencing energy consumption for all of the months . Initially, we assessed the model's performance by calculating the Mean Squared Error (MSE) on the test set model_subset which contains predictors for july month. MSE was used as a primary metric to quantify the average of the squares of the prediction errors, providing a baseline for the model's prediction accuracy. Confusion Matrix is used for predicting model accuracy. The SVM model, with its optimized parameters, presents a reliable tool for predicting energy consumption. The results from this model provide valuable insights that can be used to enhance energy management strategies, drive efficiency improvements, and support decision-making processes in energy-intensive environments.

RANDOM FOREST

The Random Forest model is particularly valued for its capability to handle complex datasets with high accuracy and minimal risk of overfitting, making it highly effective for tasks like energy consumption forecasting.

```r
# Random forest model
# Ensure out.total.energy_consumption is numeric if it's not already
model_merge_subset$out.total.energy_consumption <- as.numeric(model_merge_subset$out.total.energy_consumption)
model_subset$out.total.energy_consumption <- as.numeric(model_subset$out.total.energy_consumption)

# Train the Random Forest model
# Adjust the number of trees (ntree) and the number of variables at each split (mtry) based on your specific needs
rf_model <- randomForest(out.total.energy_consumption ~ ., data = model_merge_subset, ntree = 500, mtry = sqrt(ncol(model_merge_subset)))

# Predict on the test set
predictions <- predict(rf_model, model_subset)

# Calculate Mean Squared Error for model evaluation
mse <- mean((predictions - model_subset$out.total.energy_consumption)^2)
print(paste("Mean Squared Error:", mse))

accuracy_rf <- confusionMatrix(predictions, model_subset$out.total.energy_consumption)

```

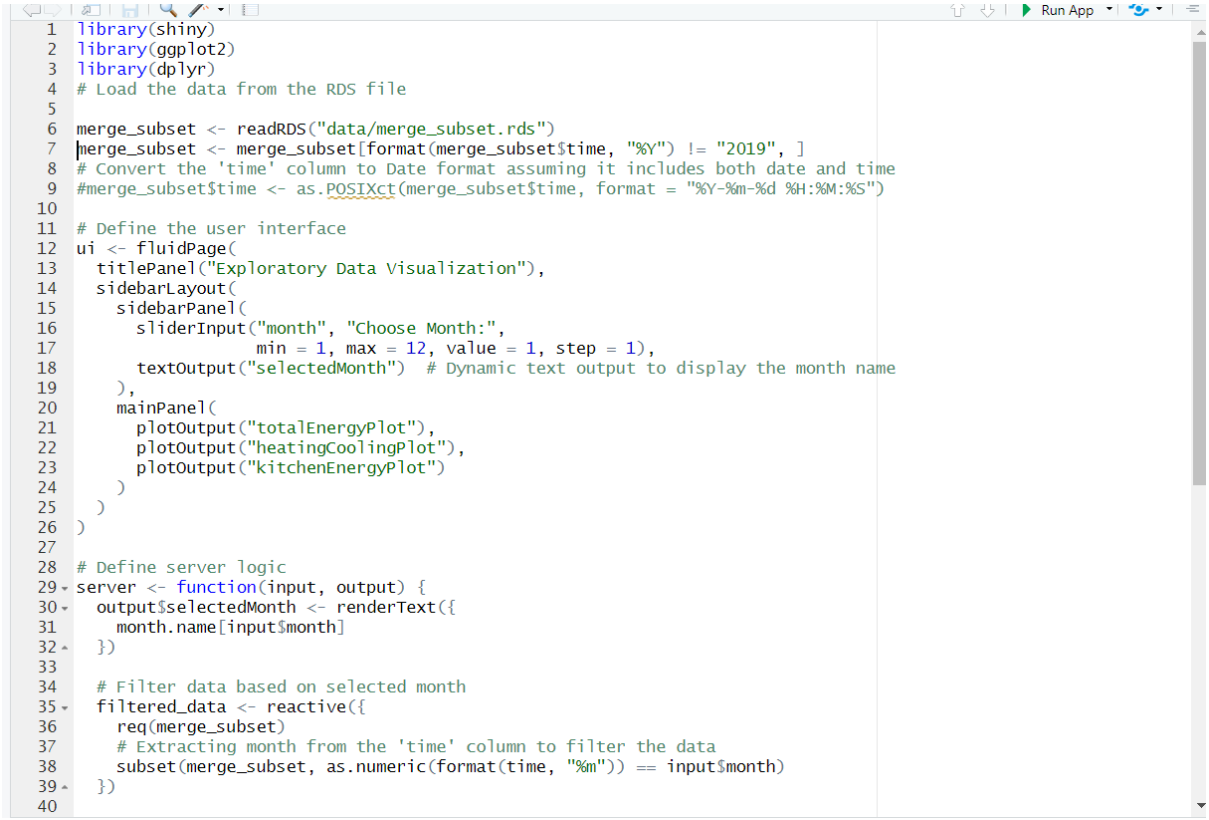
We also used a Random Forest model to predict total energy consumption. The model, trained on a dataset ensuring numeric consistency for the target variable out.total.energy_consumption, utilizes 50 decision trees with the number of predictor variables at each split determined by the square root of the total number of predictors. This configuration enhances the model's accuracy and robustness by reducing the correlation between trees and increasing generalization. Predictions are then made on a test dataset, and the model's accuracy is evaluated using the Mean Squared Error (MSE), a metric that quantifies the average squared difference between the predicted and actual values.

Multiple Linear Regression (MLR), which accounted for almost 89.4% of the variability in energy use, was found to be the best accurate forecast model in our comparative examination of models for July energy usage. With an RMSE of 0.42 and an RSE of 0.2956, it was statistically significant and showed accurate predictions. We also examined the Random Forest and Support Vector Machine (SVM) models. Random Forest performed well in dataset complexity with minimal overfitting risk and good accuracy, while SVM handled complicated, non-linear patterns. Nevertheless, MLR is the recommended model for this application since it offered the most

accurate knowledge and prediction of energy consumption patterns.

SHINY APP

Now that we have described the data in detail and have used our best model, a linear regression model, it is now important to give the CEO the ability to see for themself how each predictor affects energy production. Using the shiny package in R we created a shiny app where the user can adjust each predictor, which then outputs the predicted energy usage amount in the form of graphs. This is quite helpful for a CEO as they can see what types of residences use up the most energy. Using the code below, we were able to put input sliders for each predictor in the model we created.



```r
library(shiny)
library(ggplot2)
library(dplyr)
# Load the data from the RDS file

merge_subset <- readRDS("data/merge_subset.rds")
merge_subset <- merge_subset[format(merge_subset$time, "%Y") != "2019", ]
# Convert the 'time' column to Date format assuming it includes both date and time
#merge_subset$time <- as.POSIXct(merge_subset$time, format = "%Y-%m-%d %H:%M:%S")

# Define the user interface
ui <- fluidPage(
  titlePanel("Exploratory Data Visualization"),
  sidebarLayout(
    sidebarPanel(
      sliderInput("month", "Choose Month:",
                  min = 1, max = 12, value = 1, step = 1),
      textOutput("selectedMonth")  # Dynamic text output to display the month name
    ),
    mainPanel(
      plotOutput("totalEnergyPlot"),
      plotOutput("heatingCoolingPlot"),
      plotOutput("kitchenEnergyPlot")
    )
  )
)

# Define server logic
server <- function(input, output) {
  output$selectedMonth <- renderText({
    month.name[input$month]
  })

  # Filter data based on selected month
  filtered_data <- reactive({
    req(merge_subset)
    # Extracting month from the 'time' column to filter the data
    subset(merge_subset, as.numeric(format(time, "%m")) == input$month)
  })

```

```r

  # Plot total energy consumption
  output$totalEnergyPlot <- renderPlot({
    data <- filtered_data()
    ggplot(data, aes(x = time, y = out.total.energy_consumption)) +
      geom_line() +
      labs(title = "Total Energy Consumption", x = "Time", y = "Energy (kWh)")
  })

  # Plot heating and cooling energy consumption
  output$heatingCoolingPlot <- renderPlot({
    data <- filtered_data()
    ggplot(data, aes(x = time, y = out.heating_cooling.energy_consumption)) +
      geom_line(color = "red") +
      labs(title = "Heating and Cooling Energy Consumption", x = "Time", y = "Energy (kWh)")
  })

  # Plot kitchen energy consumption
  output$kitchenEnergyPlot <- renderPlot({
    data <- filtered_data()
    ggplot(data, aes(x = time, y = out.kitchen_energy_consumption)) +
      geom_line(color = "green") +
      labs(title = "Kitchen Energy Consumption", x = "Time", y = "Energy (kWh)")
  })
}

# Run the application
shinyApp(ui = ui, server = server)

```

This Shiny application is designed to provide interactive visualizations of energy consumption data, allowing users to explore how energy usage patterns vary across different months. The code is structured into two main components: the user interface (UI) defined in the `ui` object and the server logic encapsulated within the `server` function.

In the UI section, a fluid page layout is established with a title panel and a sidebar layout. The sidebar contains a slider input for selecting a month, which ranges from 1 (January) to 12 (December). This slider enables the user to dynamically select the month for which they wish to view energy consumption data. A text output is also included to display the name of the selected month, enhancing the application's interactivity. The main panel is designed to display three separate plot outputs: total energy consumption, heating and cooling energy consumption, and kitchen energy consumption.

The server function contains the application's reactive elements and rendering processes. It begins by mapping the numeric representation of months to their respective names. The `renderText` function dynamically updates the displayed month name based on the user's input. The core of the server function is a reactive expression that filters the dataset for the selected month, ensuring that only data relevant to the user's selection is used in the visualizations. Within this reactive context, the application generates three plots using `ggplot2`: one for total energy consumption, another for

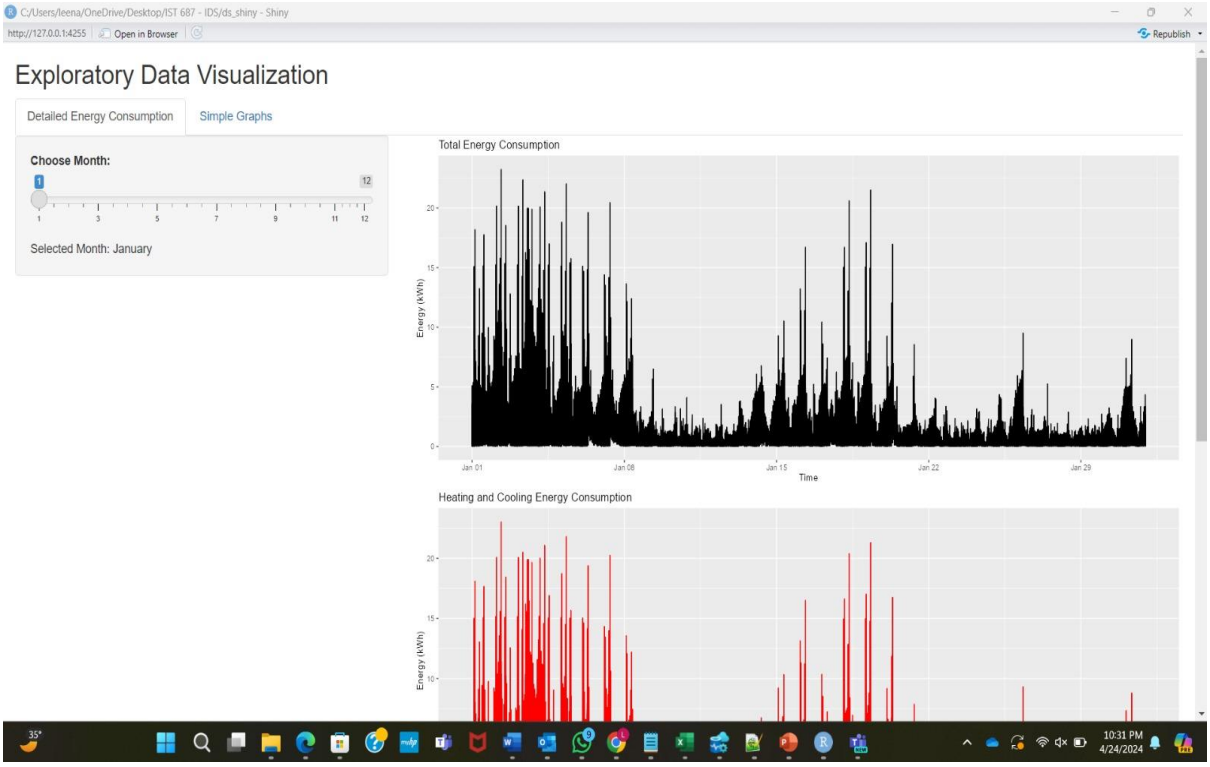
heating and cooling energy consumption, and a third for kitchen energy consumption. These plots are rendered as line graphs, each with a distinct color to differentiate between the various types of energy usage.

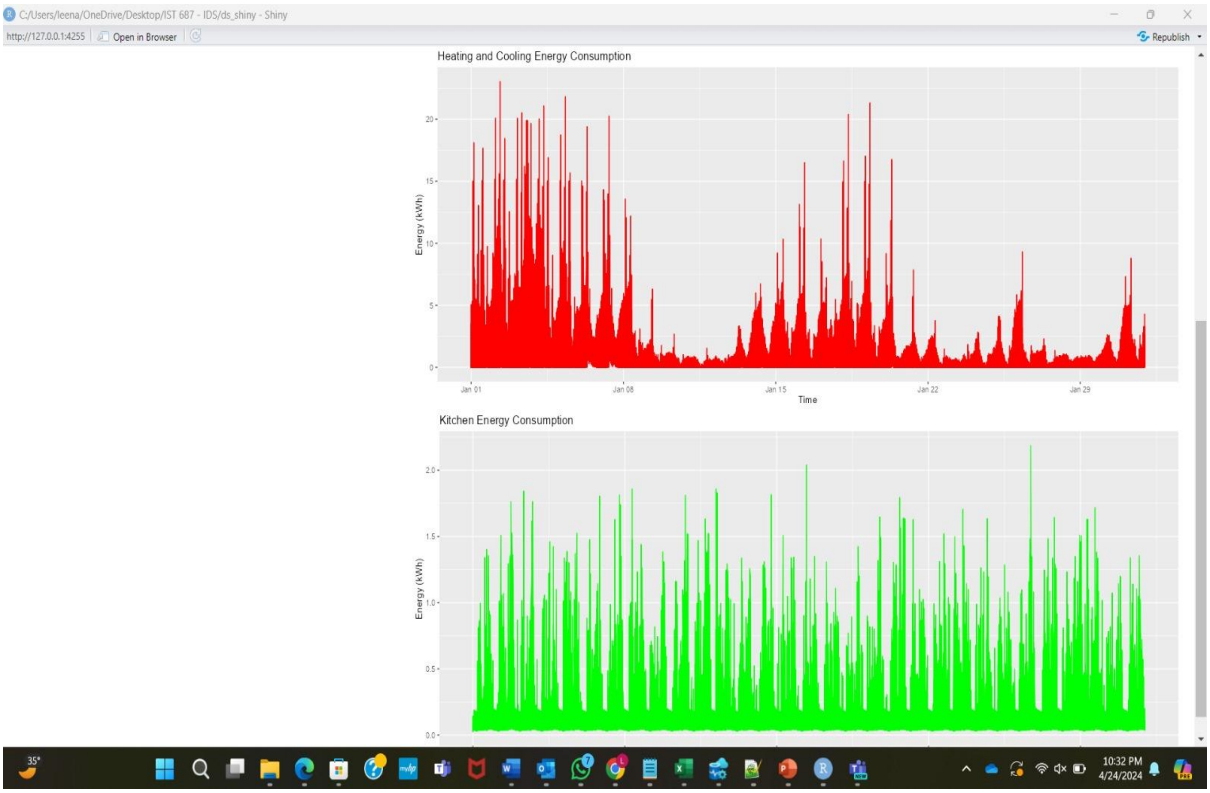
The application allows for an intuitive exploration of energy consumption patterns, enabling stakeholders to identify temporal trends and anomalies in energy usage. The visualization of such data can lead to insights on potential energy-saving opportunities, efficiency improvements, and policy implications. By adjusting the month slider, users can immediately see the changes in energy consumption across the year, facilitating an understanding of seasonal impacts on energy usage. The modular design of the code ensures easy expansion or adaptation to include additional visualizations or data sources in the future.

SHINY APP LINK:

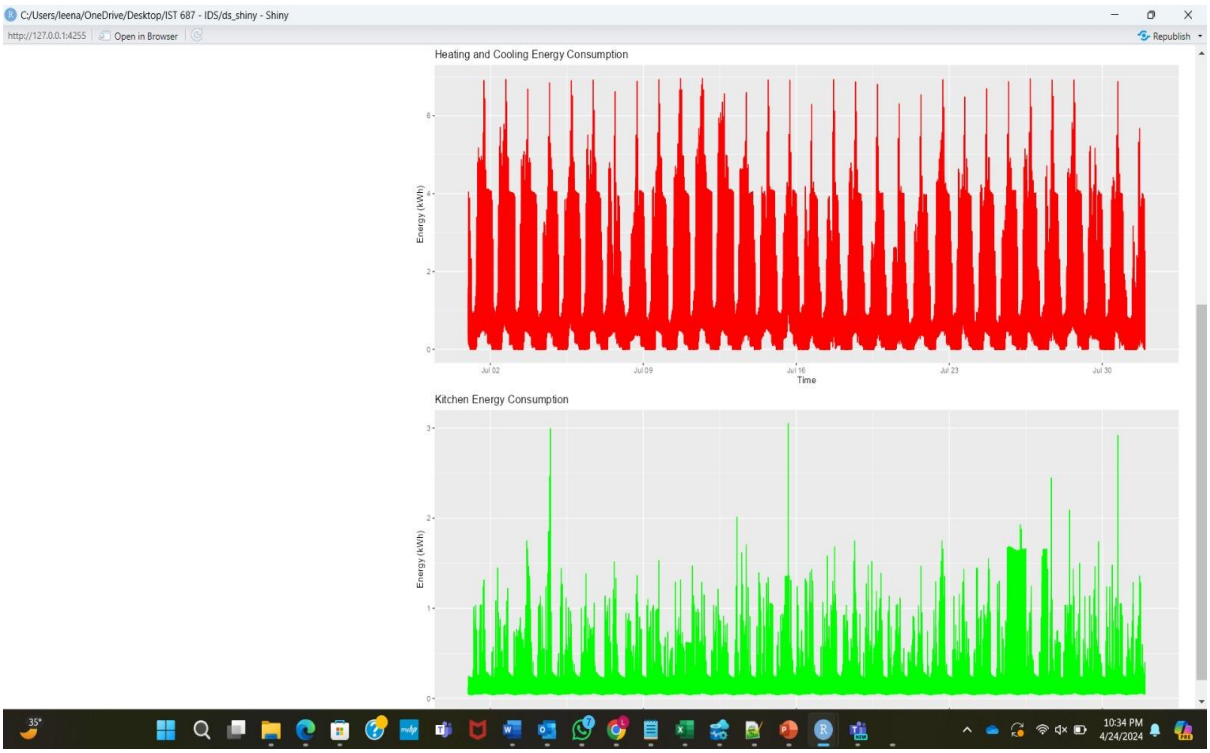
https://lbalagal.shinyapps.io/finalapp/

ENERGY CONSUMPTION APP




C:/Users/leena/OneDrive/Desktop/IST 687 - IDS/ds_shiny - Shiny
http://127.0.0.1:4255
Open in Browser
Republish
Heating and Cooling Energy Consumption
Energy (kWh)
Time
Kitchen Energy Consumption
Energy (kWh)
10:32 PM
4/24/2024


C:/Users/leena/OneDrive/Desktop/IST 687 - IDS/ds_shiny - Shiny
http://127.0.0.1:4255
Open in Browser
Republish
Exploratory Data Visualization
Detailed Energy Consumption
Simple Graphs
Choose Month:
Selected Month: July
Total Energy Consumption
Energy (kWh)
Time
Heating and Cooling Energy Consumption
Energy (kWh)
10:33 PM
4/24/2024



In the pursuit of understanding the effects of building attribute modifications on predicted outcomes, a rigorous data-driven methodology was employed. Through the manipulation of our dataset, encompassing reductions in square footage and alterations in cooking range data, we simulated potential changes in building features. Leveraging predictive modeling techniques, specifically utilizing 'model_lm,' we assessed the impact of these modifications on the predicted outcome, which in our case focused on estimating energy demand. The observed shift in predicted peak energy demand, derived from empirical data and modeling outputs, illustrates the potential influence of these changes on energy consumption patterns. This data-centric analysis aims to provide actionable insights grounded in empirical evidence, contributing to informed decision-making processes.

POTENTIAL APPROACH TO REDUCE PEAK ENERGY DEMAND

- Encouraged use of energy-efficient washers and dryers, as well as off-peak usage, to reduce energy consumption during peak demand hours.
- Promoted energy-efficient cooking appliances and timed cooking during off-peak hours to reduce stove energy usage.
- Improved insulation and HVAC efficiency in places with mixed-humid climates, reducing energy demand from temperature changes.
- Promoted smaller, energy-efficient housing designs and effective space utilization in bigger homes to reduce energy consumption caused by increased square footage and bedrooms.
- Targeted incentives, such as subsidies, were implemented to encourage affluent households to invest in renewable energy solutions and offset their higher energy usage.
- Installing solar panels is a viable technique to reducing peak energy consumption. This method tries to 'peak shave' by utilizing solar-generated electricity during daylight hours, reducing reliance on traditional grid sources during periods of high demand. The incorporation of solar panels provides a realistic alternative to ease grid pressure during peak periods by offsetting load, excess generation, and potential involvement in demand response programs facilitated by solar energy storage systems.

CONCLUSION

In order to forecast energy use during July, the month of highest utilization, we used reliable models including Multiple Linear Regression, Support Vector Machine (SVM), and Random Forest. We conducted a thorough data analysis for this technical report in order to examine energy consumption and savings. The results of our investigation showed that the Multiple Linear Regression model provided the best accurate estimates, explaining around 89.4% of the variability in energy usage. This model performed remarkably well.

A Shiny application, an interactive tool is created to let users including the CEO to visualize how different months affect energy usage dynamically, was also incorporated into our study. This tool shows the possible effects of various energy usage situations, which improves knowledge and makes decision-making easier.

We looked at a number of strategies, such installing solar panels and encouraging the use of energy-efficient appliances, to address peak energy demands. These tactics might drastically lower energy usage during crucial times. The dual goals of these strategies are to encourage sustainable energy behaviors and better manage energy demand.

The creation of the Shiny app and our thorough study demonstrate how crucial it is to use interactive tools and statistical models to comprehend complicated data landscapes. This project not only helps with immediate strategic decision-making, but it also lays the foundation for subsequent projects that will optimize energy use and strengthen sustainability efforts in the energy industry.

The project's insights will direct eSC in the implementation of workable and significant energy management techniques, guaranteeing resilience and effectiveness in the face of growing energy requirements and environmental difficulties. This paper offers proof of the effectiveness of data-driven analysis and the use of cutting-edge modeling methods to address practical issues.

WORK LOG

- Aakanksha Maheshwari – Data Cleaning, Modeling
- Hrushikesh Medhekar – Data Cleaning, Visualizations
- Mrunal Nikam - Visualizations, Shiny App, Data Merging
- Leena Balagalu - Data Merging, Modeling, Visualizations, Shiny app, Technical Document
- Tanvi Salian – Shiny App, Technical Document
- Ishita Trivedi – Presentation, Shiny App