ANALYSIS AND VISUALIZATION OF SMART HOUSE DATA SET IN PYTHON PROGRAMMING LANGUAGE

Hana Stefanović[1],
[0000-0003-0890-4410]
Ana Đokić[2]*
[0009-0002-3469-586X]

[1]The Academy of Applied Technical Studies, Belgrade, Serbia

[2]Information Technology School, Belgrade, Serbia

Abstract:

Before making any predictions, it is important to have insight into the data from a given data set. Finding links between variables that may be both intriguing and helpful, as well as identifying glaring errors and better understand patterns within the data are all made possible by having insight into the data. Data scientists examine, study, and enumerate the key features of data sets using exploratory data analysis. Research is where data visualization techniques are most frequently used.

In this paper, data preparation from a given data set and data visualization using the Python programming language are presented. Machine learning and other advanced analysis and data modelling can be performed on the prepared data.

Keywords:

Data Analysis, Data Visualization, Python Programming Language, Smart House.

INTRODUCTION

To determine the best way to alter data sets to extract pertinent and useful information, it is essential to first prepare and visualize the data for analysis.

Through data preparation and visualization, researchers can find trends, identify anomalies, test theories, or validate presumptions. Exploratory data analysis offers a better knowledge of data set variables and the interactions between them and is mostly used to examine what data might be disclosed beyond the formal modelling or hypothesis testing activity. [1] It can also assist in determining the suitability of the statistical methods which are planned for use in data analysis.

Python programming language is the most used tool for data exploration and analysis. [2], [3]Python is a general-purpose, interpreted, open-source, object-oriented programming language with dynamic semantics. Its high-level built-in data structures make it highly attractive for rapid application development, as well as for use in solving tasks in data science—from data manipulation and automation to business analysis and big data research. [4], [5] Python and exploratory analysis can be used together to identify missing values in a dataset, which is important, among other things, when selecting a machine learning model.

Correspondence:
Ana Đokić

e-mail:
ana.djokic@its.edu.rs

DOI: 10.15308/Sinteza-2024-128-134

Data science tasks are supported by several Python libraries, such as the following:

- Numpy [6] for managing arrays with many dimensions (numerical modelling analysis)
- Pandas [7] for analysis and data manipulation (cleaning, filtering, sorting, exploring and displaying data in just a few seconds)
- Matplotlib [8] to create data visualizations (producing basic graphics and charts)
- Seaborn [9] (creating more eye-catching and educational statistic illustrations)

Furthermore, Python is especially well-suited for large-scale machine learning deployments. [10] Python's popularity as a programming language these days can be attributed in great part to its application in the cutting-edge fields of artificial intelligence and machine learning. [1] With the use of tools like Scikit-learn, Keras, and TensorFlow from its portfolio of specialized deep learning and machine learning libraries, data scientists can create complex data models that can be integrated straight into a production system. [11]

2. ANALYSIS AND VISUALIZATION OF SMART HOME DATA SET

2.1. ABOUT DATA SET

This CSV file contains smart meter readings for home appliances in kW at a time span of 1 minute over 350 days, together with weather information particular to that location. The data are in the data set. [12]

Original column names in the data set are described in Table 1 .

Table 1. Descriptions of column names from the data set.

The name of column	Description
time	Time
use [kW]	Total energy consumption
gen [kW]	Total energy generated by means of solar or other power generation resources
House overall [kW]	Overall house energy consumption
Dishwasher [kW]	Energy consumed by specific appliance
Furnace 1 [kW]	Energy consumed by specific appliance
Furnace 2 [kW]	Energy consumed by specific appliance
Home office [kW]	Energy consumed by specific appliance
Fridge [kW]	Energy consumed by specific appliance
Wine cellar [kW]	Energy consumed by specific appliance
Garage door [kW]	Energy consumed by specific appliance
Kitchen 12 [kW]	Energy consumption in kitchen 1
Kitchen 14 [kW]	Energy consumption in kitchen 2
Kitchen 38 [kW]	Energy consumption in kitchen 3
Barn [kW]	Energy consumed by specific appliance
Well [kW]	Energy consumed by specific appliance
Microwave [kW]	Energy consumed by specific appliance
Living room [kW]	Energy consumption in Living room
Solar [kW]	Solar power generation
temperature	Temperature
icon	Overall weather condition (clear-night:39%; clear-day:36%; Other:25%)
humidity	Humidity
visibility	Visibility

The name of column	Description
summary	Summarise weather (Clear:75%; Partly Cloudy:12%; Other:13%)
apparentTemperature	Apparent temperature
pressure	Pressure
windspeed	Wind speed
cloudCover	Cloud cover (0 :14%; 0.31 :10%; Other :77%)
windBearing	Wind bearing
precipIntensity	Precipitation Intensity
dewpoint	Dew point
precipProbability	Precipitation probability

2.2. DATA ANALYSIS AND PROCESSING

For the processing and analysis of data, it is necessary to install helpful analytics libraries, what is shown in the Listing 1 .

A program was written to load data set, display the shape of data set, display the first 5 rows from the data set, prints the columns and their types, check if there are entries with null values, drop the line with missing values.

Data set information from a given data set such as column names, number of rows, number of columns, memory space and data type are shown in Listing 2.

The obtained result indicates that within the data set, there exist columns of both object and float data types. Also, it is evident that the data set includes columns pertaining to energy generation, energy consumption per room/appliance, and weather data in addition to a 'time' column.

Below, the program checks whether there are entries with null values, drops the line with missing values and checks the unique values in object columns 'icon', 'summary' and 'cloudCover' in the given data set. Although 'cloudCover' is one of the unique entries, it looks that it should be a float. Entry 'cloudCover' appears in the first 57 minutes of the first hour and seems to be an error. These values have been replaced with the first valid entry, under the assumption that the cloud cover will not change dramatically in the first hour. The column 'cloudCover' with replaced values is shown in Listing 3.

From the correlation matrix of the data set, it can be seen three pairs of variables are highly correlated:

- 'use [kW]' and 'House overall [kW]'
- 'gen [kW]' and 'Solar [kW]'
- 'Temperature' and 'apparentTemperature'

To facilitate analysis, variables 'House overall [kW]', 'Solar [kW]' and 'apparentTemperature' have been dropped. Since the column 'time' is now the data frame's index, it has been dropped. Object columns 'icon' and 'summary' are not interesting for further analysis, so they can be dropped too. Also, there are two variables related to the 'Furnace' and three variables related to the 'kitchen'. These variables can be combined by adding corresponding variables to new columns and dropping individual columns. Columns have been renamed to remove spaces and the [kW] unit. Data set information after the changes have been made is shown in Listing 4 .

```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import time
```

Listing 1. Importing libraries.

```
Dataset information:
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 503911 entries, 0 to 503910
Data columns (total 32 columns):
 #   Column               Non-Null Count   Dtype
---  ------               --------------   -----
 0   time                 503911 non-null  object
 1   use [kW]             503910 non-null  float64
 2   gen [kW]             503910 non-null  float64
 3   House overall [kW]   503910 non-null  float64
 4   Dishwasher [kW]      503910 non-null  float64
 5   Furnace 1 [kW]       503910 non-null  float64
 6   Furnace 2 [kW]       503910 non-null  float64
 7   Home office [kW]     503910 non-null  float64
 8   Fridge [kW]          503910 non-null  float64
 9   Wine cellar [kW]     503910 non-null  float64
 10  Garage door [kW]     503910 non-null  float64
 11  Kitchen 12 [kW]      503910 non-null  float64
 12  Kitchen 14 [kW]      503910 non-null  float64
 13  Kitchen 38 [kW]      503910 non-null  float64
 14  Barn [kW]            503910 non-null  float64
 15  Well [kW]            503910 non-null  float64
 16  Microwave [kW]       503910 non-null  float64
 17  Living room [kW]     503910 non-null  float64
 18  Solar [kW]           503910 non-null  float64
 19  temperature          503910 non-null  float64
 20  icon                 503910 non-null  object
 21  humidity             503910 non-null  float64
 22  visibility           503910 non-null  float64
 23  summary              503910 non-null  object
 24  apparentTemperature  503910 non-null  float64
 25  pressure             503910 non-null  float64
 26  windSpeed            503910 non-null  float64
 27  cloudCover           503910 non-null  object
 28  windBearing          503910 non-null  float64
 29  precipIntensity      503910 non-null  float64
 30  dewPoint             503910 non-null  float64
 31  precipProbability    503910 non-null  float64
dtypes: float64(28), object(4)
memory usage: 123.0+ MB
```

Listing 2. Data set information of given data set.

```
The column cloudCover with replaced values:
58            0.75
59            0.75
60            0.75
...            ...
503907        0.31
503908        0.31
503909        0.31
503908        0.31
503909        0.31
Name: cloudCover, Length: 503910, dtype: float64
```

Listing 3. The column 'cloudCover' with replaced values.

```
Changed dataset:
<class 'pandas.core.frame.DataFrame'>
Int64Index: 466308 entries, 116 to 503909
Data columns (total 23 columns):
 #   Column              Non-Null Count   Dtype
---  ------              --------------   -----
 0   use                 466308 non-null  float64
 1   gen                 466308 non-null  float64
 2   Dishwasher          466308 non-null  float64
 3   Home_office        466308 non-null  float64
 4   Fridge              466308 non-null  float64
 5   Wine_cellar         466308 non-null  float64
 6   Garage_door         466308 non-null  float64
 7   Barn                466308 non-null  float64
 8   Well                466308 non-null  float64
 9   Microwave           466308 non-null  float64
 10  Living_room         466308 non-null  float64
 11  temperature         466308 non-null  float64
 12  humidity            466308 non-null  float64
 13  visibility          466308 non-null  float64
 14  pressure            466308 non-null  float64
 15  windSpeed           466308 non-null  float64
 16  cloudCover          466308 non-null  float64
 17  windBearing         466308 non-null  float64
 18  precipIntensity     466308 non-null  float64
 19  dewPoint            466308 non-null  float64
 20  precipProbability   466308 non-null  float64
 21  kitchen             466308 non-null  float64
 22  Furnace             466308 non-null  float64
dtypes: float64(23)
memory usage: 85.4 MB
```

Listing 4. Data set information of changed data set.

After data set preprocessing, it can be concluded that converting the data, while keeping the relevant information, in a lower-dimensional space, results in a reduction of memory usage.

2.3. DATA VISUALIZATION AND DISCUSSION OF THE RESULTS

To exhibit data, data visualization is both essential and highly helpful. Data analysis is the step that comes before choosing the type of graph, and it helps determine which kind of visualization is best.

A program code was built to construct a pie chart [13] that shows data in columns containing variables related to energy consumption in the premises and for devices. This is depicted in Figure 1.

Considering only those variables related to energy consumption in rooms, it can be concluded that energy consumption is highest in the home office and lowest in the kitchen, for given data set. Approximately the same energy consumption is recorded in wine cellar and living room.

Also, considering only those variables related to energy consumption for devices, it can be concluded that energy consumption is highest for furnace and lowest for microwave and garage door. Approximately the same energy consumption is recorded for the fridge and for the barn. The highest value for furnace's consumer is expected due to it generates heat through the use of electric resistance coils, which is subsequently dispersed throughout the house. This heating system is renowned for operating safely, cleanly, and without creating any hazardous byproducts.

A line graph illustrating the dependency of various energy consumption per months is presented in Figure 2 with months on x axis, and a numerical variable Electricity Consumption [kW] on y axis.

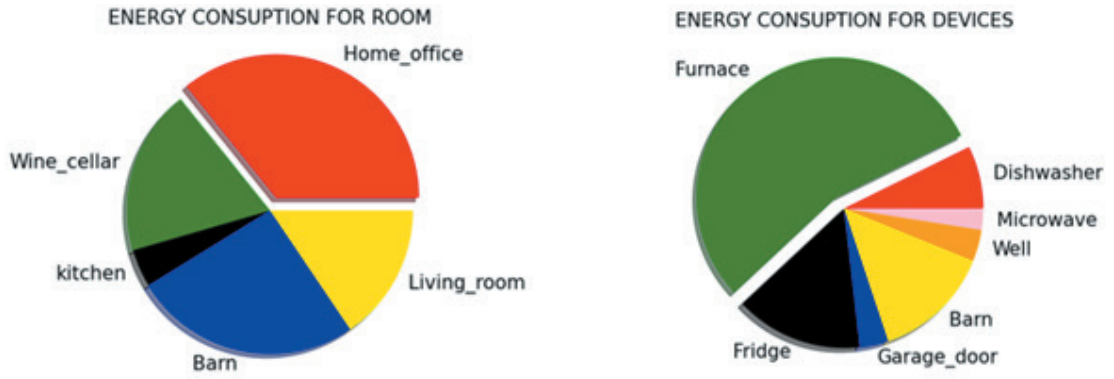

Figure 1. Energy consumption in the rooms and for devices.

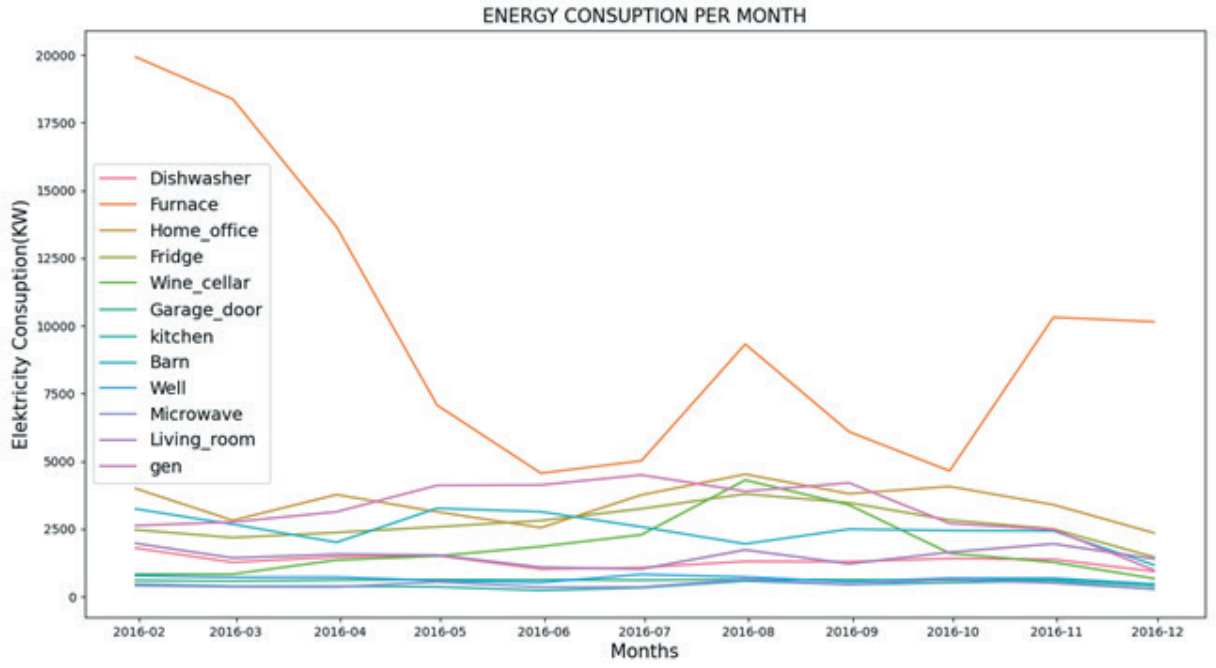

Figure 2. Energy consumption per months.

The obtained results show that the highest values are recorded in February, July and November. These results are consistent with expectations, considering the increased consumption during winter months due to heating and during summer months due to cooling.

The production of energy from solar panels is significantly reduced during the winter months.

For devices such as microwave, fridge and garage door, there is no significant difference in consumption throughout the year.

A line graph illustrating home activity in one day, during 10 hours is presented in Figure 3 .

The obtained results show that the consumption is highest around 3:00 PM and around 10:00 PM in the kitchen. The results are consistent with expectations considering the typical daily activities and consumer habits.

The graph shows an increase in consumption in the living room during the afternoon hours. Consumption in the home office remains low throughout the day, indicating that consumers are not working from home. For the garage doors, consumption is registered in time intervals that follow consumers' obligations and activities, with peaks around 5:00 PM and around 9:00 PM.

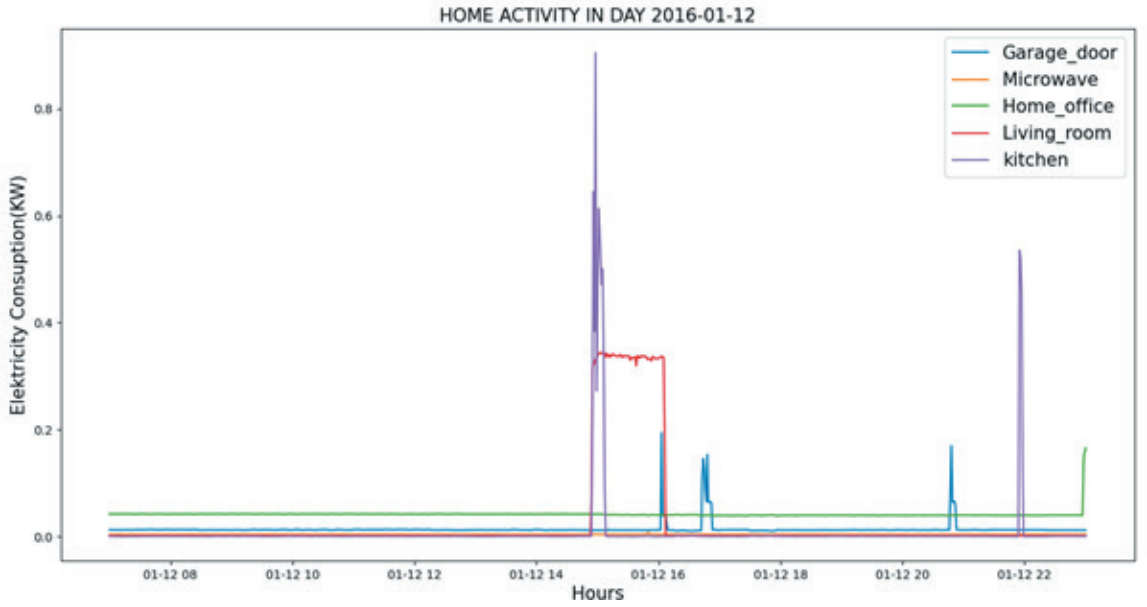

Figure 3. Home activity in one day.

3. CONCLUSION

One of the fundamental and important phases of a data science project is an exploratory data analysis. Almost 70% of a data scientist's work is spent completing an exploratory data analysis on their data set.

Techniques like dimension reduction and grouping aid in producing graphical representations of high-dimensional data with several variables in the given data set are presented in this paper. To adopt the skills and techniques of data visualization and processing, this paper shows the outcomes of data processing, analysis, and visualization on the given data set.

Exploratory data analysis is typically done as a first step before more formal statistical studies or modelling is done. It also aims to demonstrate how data analysis approaches issue solutions. As this paper also demonstrates, selecting a programming language is a crucial step in data analysis.

In this paper, energy consumption collected from given data set has been analysed and pre-processed. So, in future research, they can be used for training and testing of the predictive models.

4. REFERENCES

[1] Z. Nagy, Artifical Intelligence and Machine Learning, Birmingham, UK: Packt Publishing, 2018.

[2] [Online]. Available: https://www.python.org/. [Accessed 20 March 2024].

[3] [Online]. Available: https://anaconda.org/. [Accessed 20 March 2024].

[4] S. Rascha, V. Mirjalili, Y. Liu, Machine Learning with PyTorch and Scikit-Learn, Birmingham, UK: Packt Publishing, 2022.

[5] C. Albon, Machine Learning with Python Cookbook, O'Reilly Media, 2018.

[6] [Online]. Available: https://numpy.org/. [Accessed 5 April 2024].

[7] [Online]. Available: https://pandas.org/. [Accessed 5 April 2024].

[8] [Online]. Available: https://matplotlib.org/. [Accessed 5 April 2024].

[9] [Online]. Available: https://seaborn.org/. [Accessed 5 April 2024].

[10] K. Sahoo, A. K. Samal, J. Pramanik, S. K. Pani, "Exploratory Data Analysis using Python," *International Journal of Innovative Technology and Exploring Engineering (IJITE)*, vol. 8, no. 12, pp. 4727-4735, 2019.

[11] S. Russell, P. Norvig, Artifical Inteligence: A Modern Aproach, Pearson Education Limited, 2016.

[12] [Online]. Available: https://www.kaggle.com/datasets/taranvee/smart-home-dataset-with-weather-information/. [Accessed 15 February 2024].

[13] [Online]. Available: https://plotly.com/. [Accessed 5 April 2024].