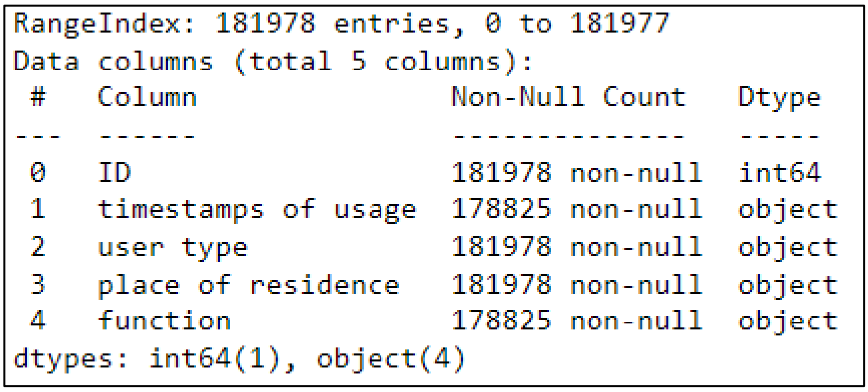
Software Development Company System User Behavior Analysis Report

In this report, we will analyze the system usage records of the company. Through the study of user behavior, our aim is to gain deeper insights into the relationships between different user types, places of residence, and functional usage patterns. By analyzing data such as ID, timestamps of usage, user type, place of residence, and function, we will explore user behavior patterns, uncover trends and patterns in system usage, and provide valuable insights to assist the company in making informed decisions.

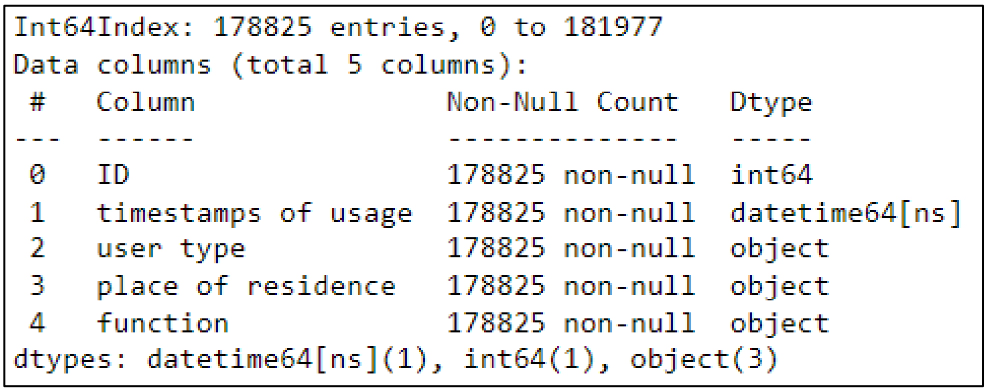
1. Data Preprocess

First, we did data preprocessing. We read in the data and checked its information (Figure 1.1), revealing a total of 5 columns and 181,978 data. Next, we observed that the timestamps of usage column are in datetime format, so we converted their data type accordingly. Further we noticed that there are missing values in both timestamps of usage and function columns by Non-Null column. Therefore, we decided to remove the entire rows containing these missing values. The processed data, as shown in Figure 1.2, now consists of 178,825 data.



```
RangeIndex: 181978 entries, 0 to 181977
Data columns (total 5 columns):
 #   Column               Non-Null Count   Dtype 
---  ------               --------------   ----- 
 0   ID                   181978 non-null  int64 
 1   timestamps of usage  178825 non-null  object
 2   user type            181978 non-null  object
 3   place of residence   181978 non-null  object
 4   function             178825 non-null  object
dtypes: int64(1), object(4)
```

Figure 1.1



```
Int64Index: 178825 entries, 0 to 181977
Data columns (total 5 columns):
 #   Column               Non-Null Count   Dtype         
---  ------               --------------   -----         
 0   ID                   178825 non-null  int64         
 1   timestamps of usage  178825 non-null  datetime64[ns]
 2   user type            178825 non-null  object        
 3   place of residence   178825 non-null  object        
 4   function             178825 non-null  object        
dtypes: datetime64[ns](1), int64(1), object(3)
```

Figure 1.2

2. Employees Analysis

First, we begin by observing the distribution of employees. We noticed that there are 222 unique values in the ID column, indicating that there is a total of 222 employees using this system within the company. Furthermore, there are 3 categories of user types, namely 'A', 'S', and 'L', which may represent different job titles or departments. Additionally, there are 4 categories of places of residence, namely 'P01', 'P02', 'P03', and 'P05', which may represent the residential areas of the employees.

Next, we observed the distribution of user type and places of residence (Figures 2.1 and 2.3). It can be observed that the majority of users are user type of 'A', and most reside in the 'P03' area. Through Figures 2.2 and 2.4, we can gain insights into the proportional distribution of user type and place of residence.

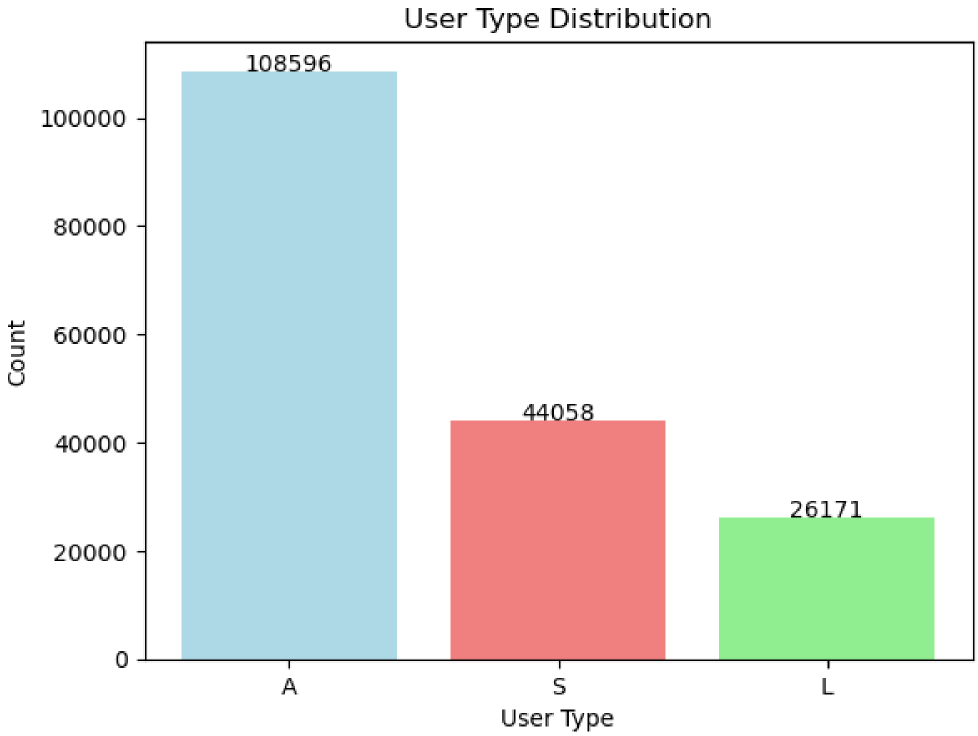


Figure 2.1

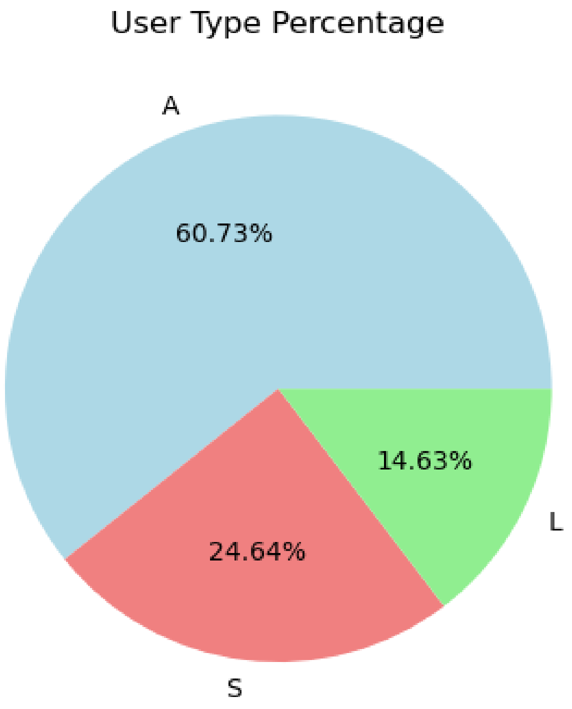


Figure 2.2

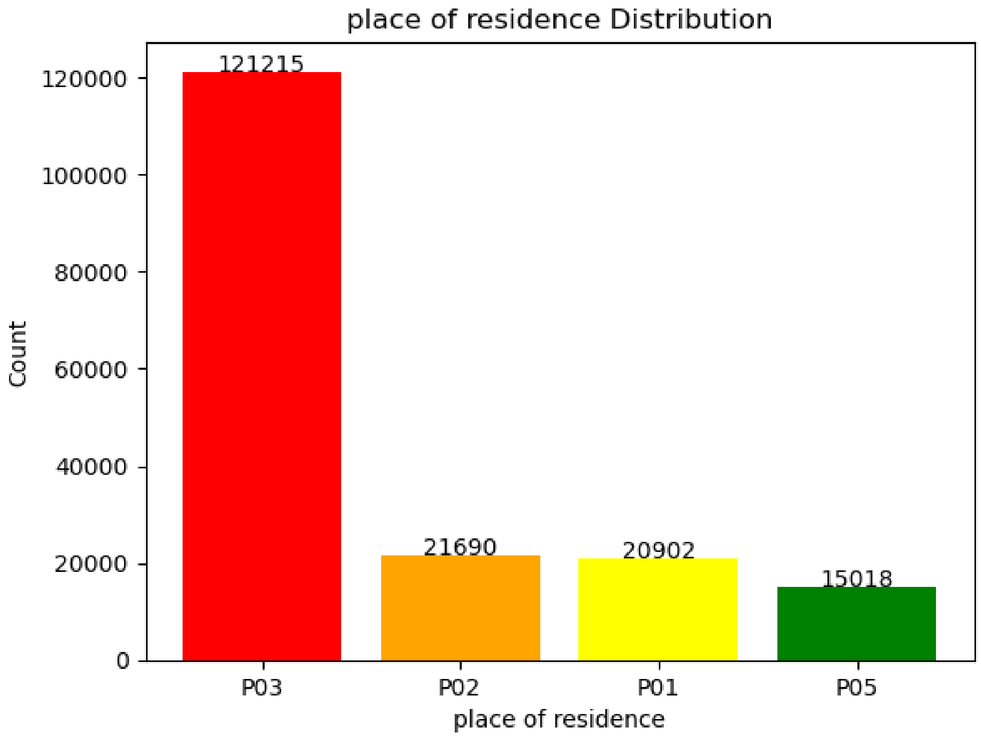


Figure 2.3

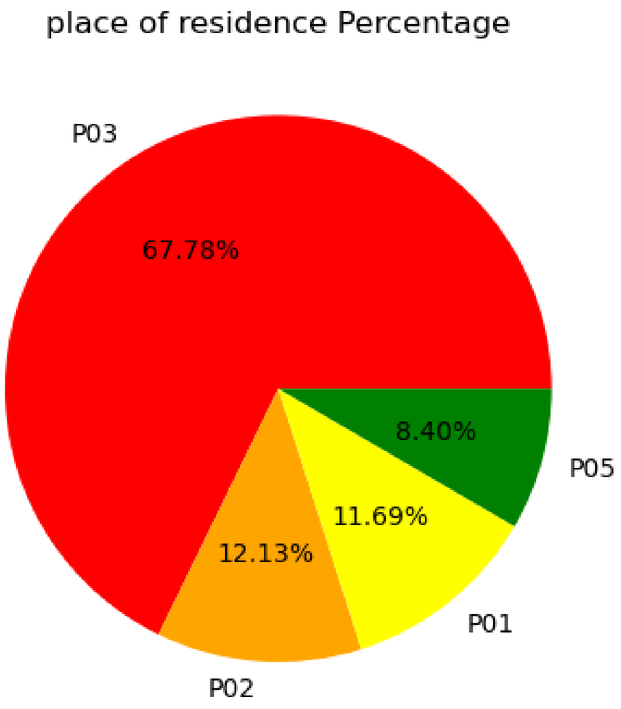


Figure 2.4

3. User Behavior Analysis

In this chapter, we will analyze the system's functionalities and user usage patterns. Firstly, we notice that there is a total of 33 functionalities in the system, labeled as 'F01', 'F02', ..., 'F33', each representing different system functions. As for the timestamps of usage, they record the period of system usage log between April 1, 2023, 00:00:00, and June 30, 2023, 23:58:00.

Begin by checking the distribution of function usage. Figure 3.1, it can be observed that 'F13' is the most frequently used function, with a usage count to 19,281 times during the period.

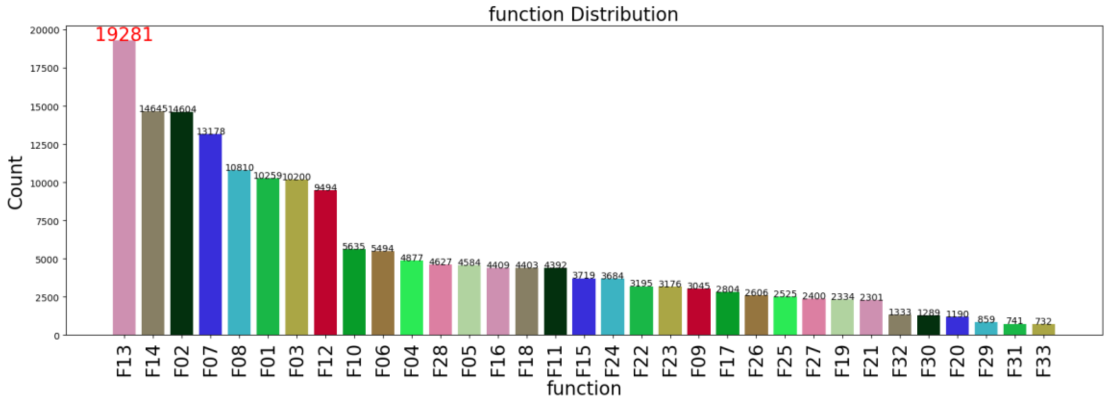


Figure 3.1

Next, we plotted the usage count of system logs according to the timestamps of usage, as shown in Figure 3.2. We will reveal more detailed analysis of this data.

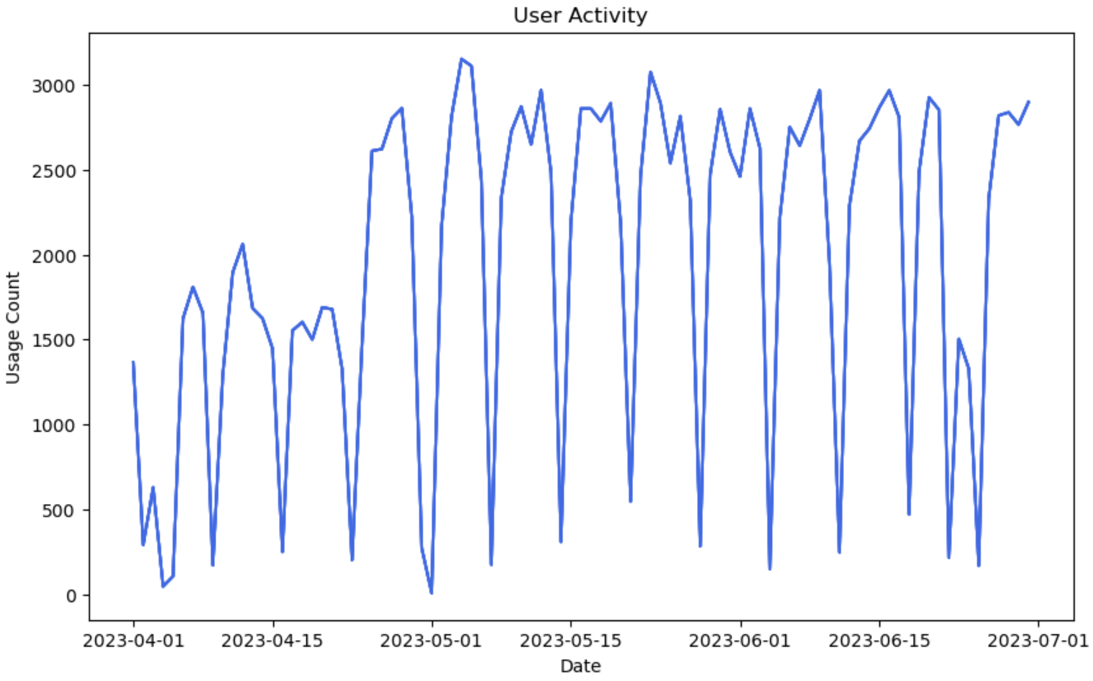


Figure 3.2

(1) Analysis By 24 Hours of The Day

In this section, we extracted the hour from the timestamps of usage and calculated the usage count of system functions for each of the 24 hours in a day. As shown in Figure 3.3, we observed a rapid increase in system usage starting around 5 A.M. each day, reaching its peak at 8 A.M. Subsequently, the usage gradually decreased until around 5 P.M., followed by another rapid increase, peaking again at 8 P.M. After that, the usage declined once again. From this pattern, we can inform that the system is likely related to clock-in/clock-out activities or other functions associated with the beginning and end of the workday, as it higher usage during the mornings around 6 to 10 A.M. and evenings around 6 to 10 P.M.

Next, we cluster the data, grouping the logs by user type and plotting the distribution of system usage by hour, as shown in Figure 3.4. Upon observation, the distribution after clustering remains similar to the original unclustered data, still indicating higher usage around 6 to 10 A.M. and 6 to 10 P.M. daily.

We also grouped the data according to the place of residence and analyzed the usage, as shown in Figure 3.5. The results observed were similar to before, still indicating higher usage around 6 to 10 A.M. and 6 to 10 P.M.

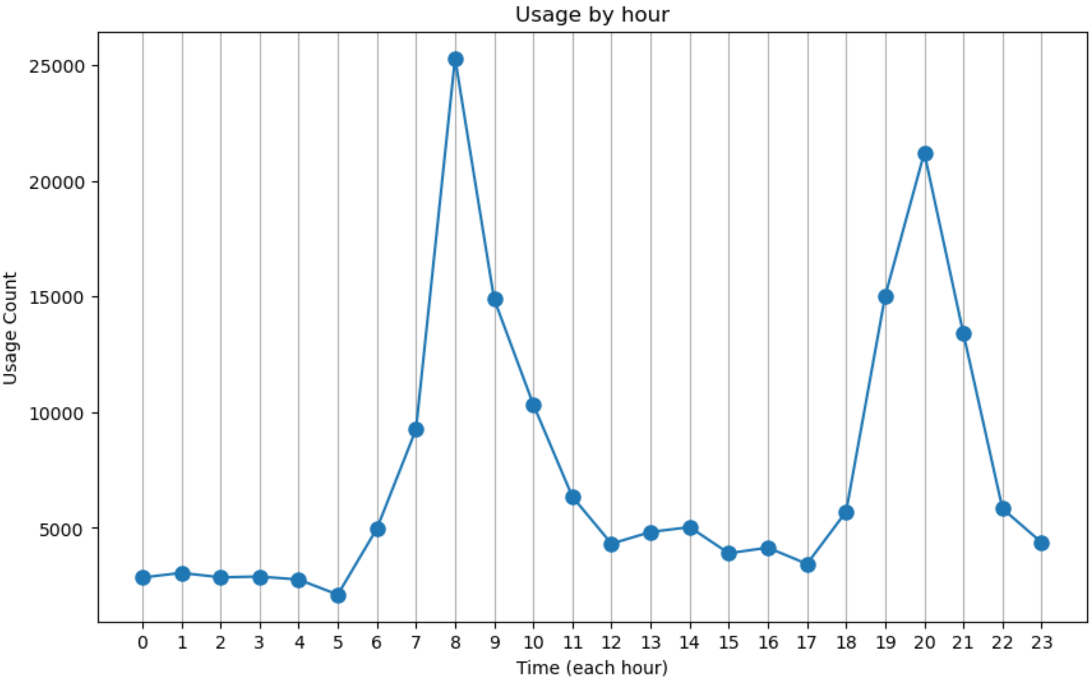


Figure 3.3

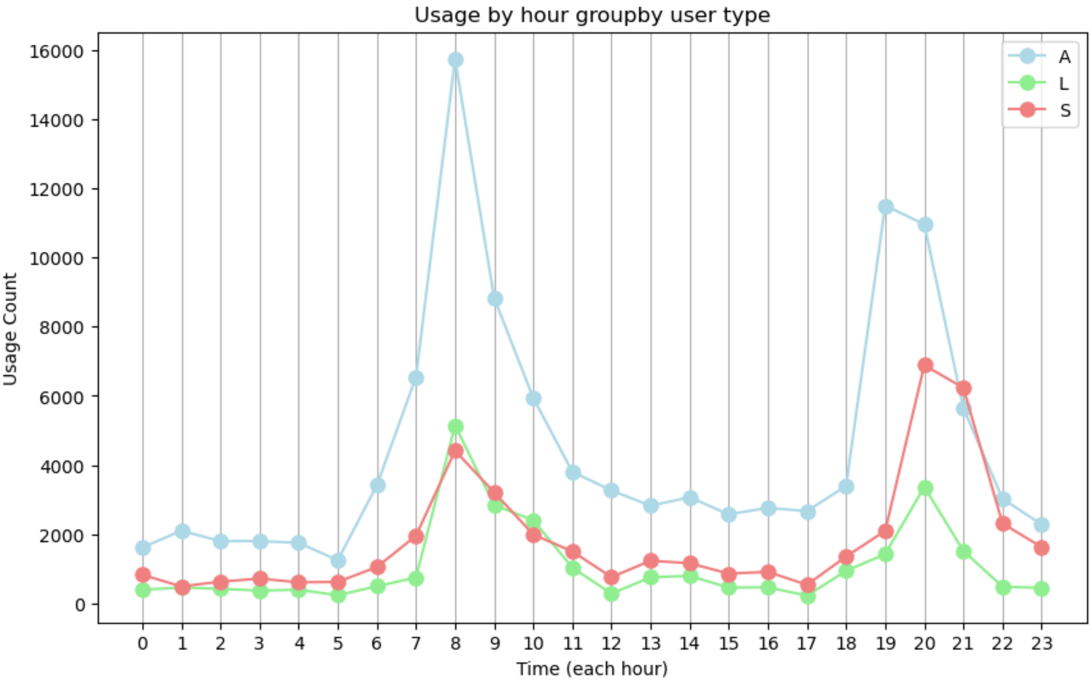


Figure 3.4

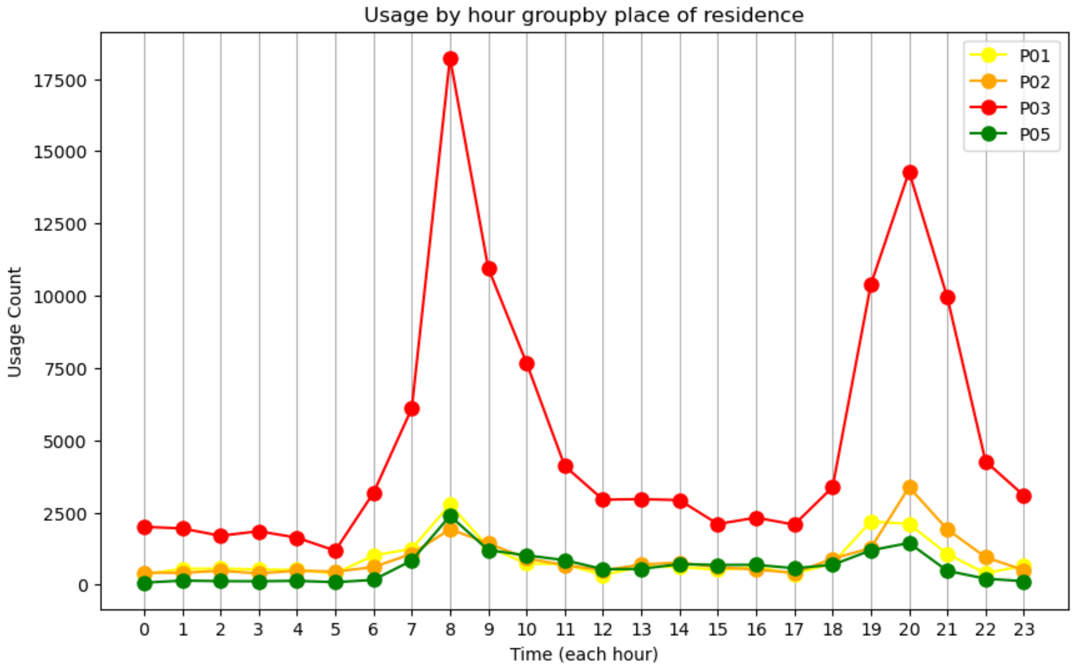


Figure 3.5

We also analyzed the usage of each function every hour, observing the most frequently used function and its usage count per hour, as shown in Figure 3.6. It can be seen that at 8 A.M., function F13 had the highest usage count, reaching 2,704 times.

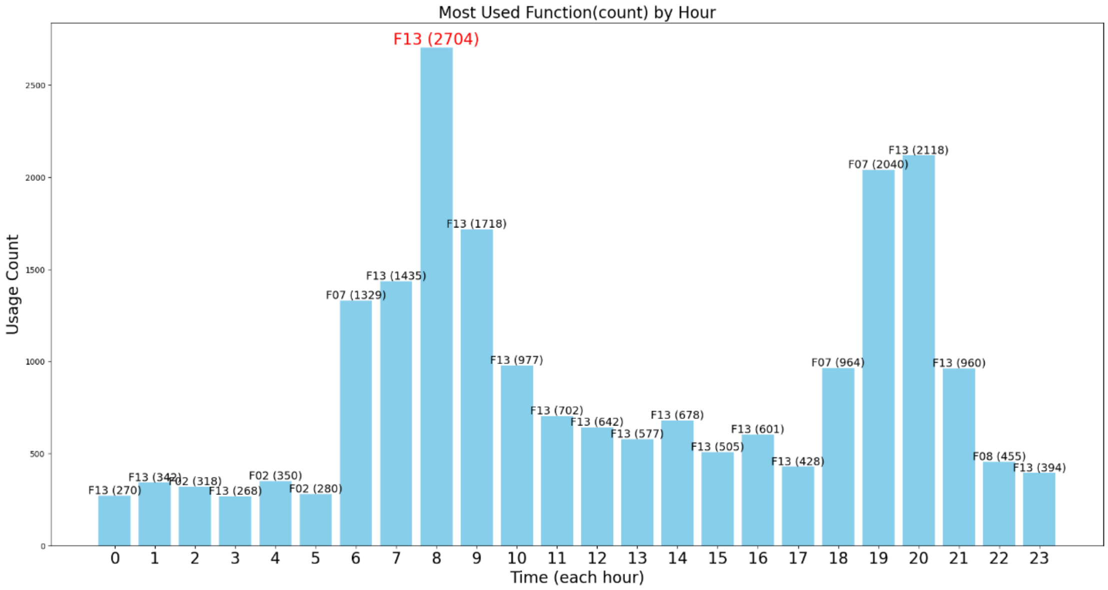


Figure 3.6

Finally, we picked the most frequently used function, F13, and observed its usage count throughout the day, as shown in Figure 3.7.

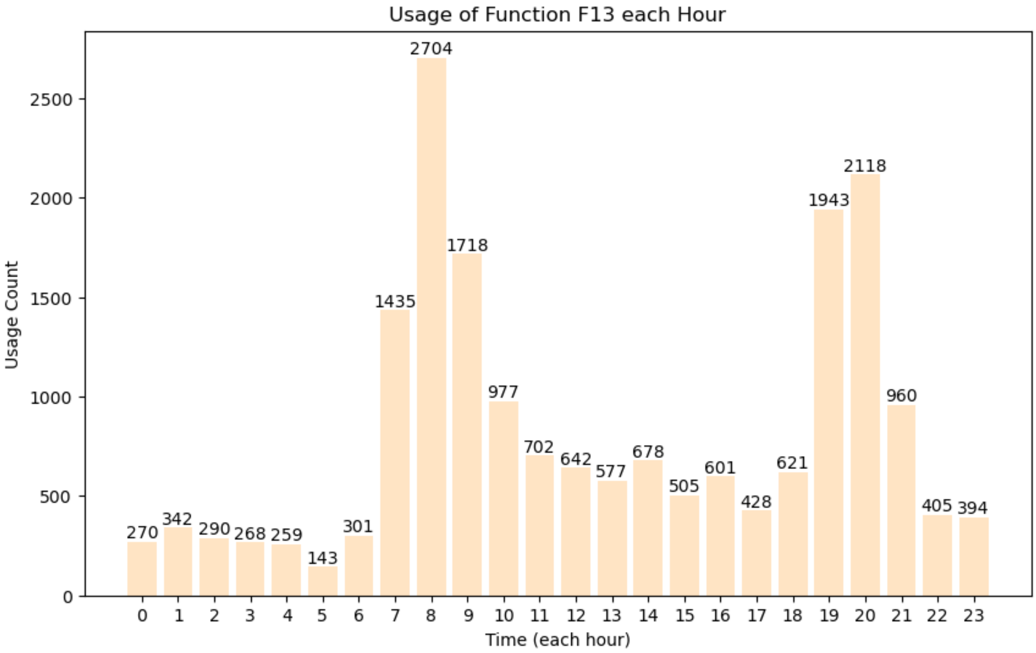


Figure 3.7

(2) Analysis By 7 Days of The Week

In this section, we will group the system usage patterns into 7 days of the week. We first convert the timestamps of usage into corresponding days of the week and then analyze the usage counts for each of the 7 days. As shown in Figure 3.8, we observe that the usage counts from Monday to Saturday are consistently high, with at least 25,000 usages or more. However, on Sunday, the usage count drops to below 5,000. This seems that most employees are likely on a break or off duty on Sunday, resulting in lower system usage during this time.

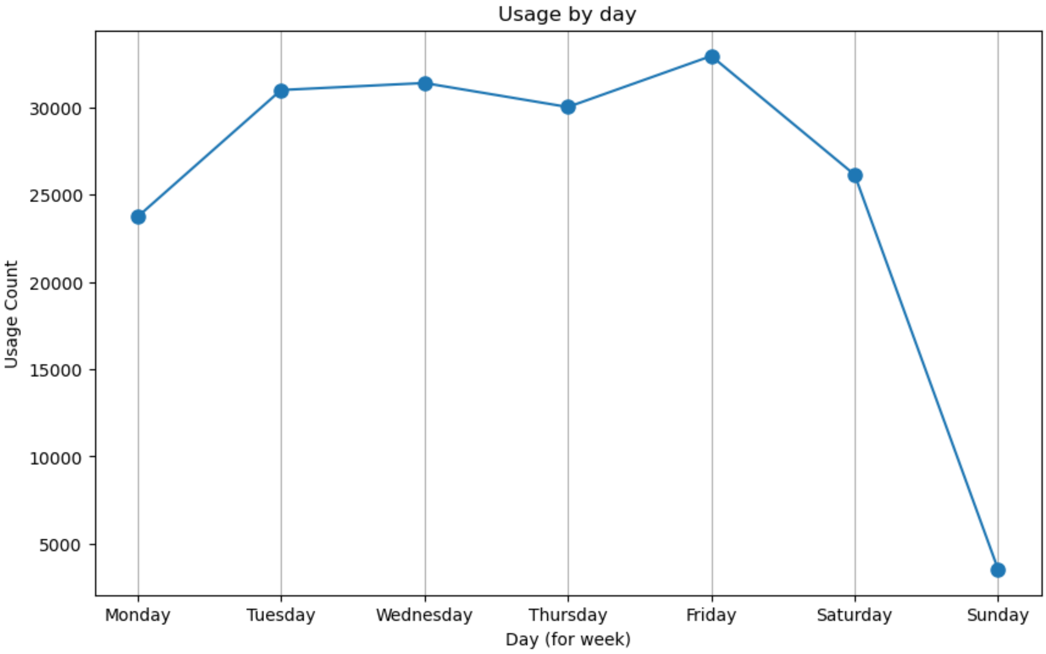


Figure 3.8

(3) User Behavior Prediction by ML

In this section, I utilized machine learning models to predict the system usage patterns, focusing on the prediction of user type and place of residence. I used four different classification models for the prediction: KNN, XGBoost, Decision Tree, and Random Forest. Finally, I calculated the accuracy of the predictions compared to the actual results.

I. In the first part, I predict on user type. Firstly, I preprocessed the data by converting the 'timestamps of usage' column into three separate columns: 'timestamps_date', 'timestamps_day', and 'timestamps_hour'. Then, I removed the original 'timestamps of usage' and 'ID' columns. Subsequently, I performed label encoding on the user type, extracting them as labels (y), while the remaining columns, including 'place of residence' and 'function', were encoded by using one-hot encoding and extracted as features (x). Following this, I split the data into training and testing sets in a 9:1 ratio and dealed with training and prediction of the models. Finally, I predicted the test set using four different models and calculated the accuracy. The results are shown in Figure 3.9, where the XGBoost model performed the best with accuracy 67.73%.

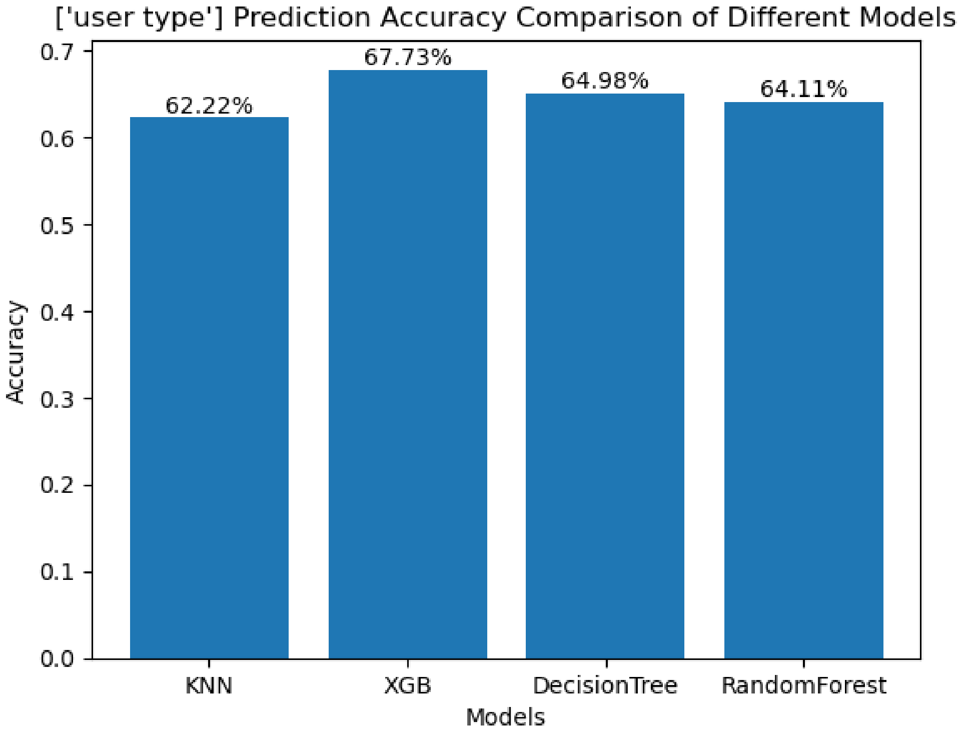


Figure 3.9

II. In the second part, we predict the place of residence. Firstly, the data went preprocessing where the 'timestamps of usage' column was converted into three separate columns: 'timestamps_date', 'timestamps_day', and 'timestamps_hour'. Subsequently, the original 'timestamps of usage' and 'ID' columns were dropped. The 'place of residence' column was then subjected to label encoding to be as the label ('y'), while the remaining columns including 'user type' and 'function' went one-hot encoding and were extracted as features (x). The data was then split into training and testing sets in a 9:1 ratio before training and predicting using the models. Finally, predictions were made using four different models, and the accuracy of the test data was calculated. The results, as in Figure 3.10, once again showed XGBoost as the top model, achieving accuracy to 69.80%.

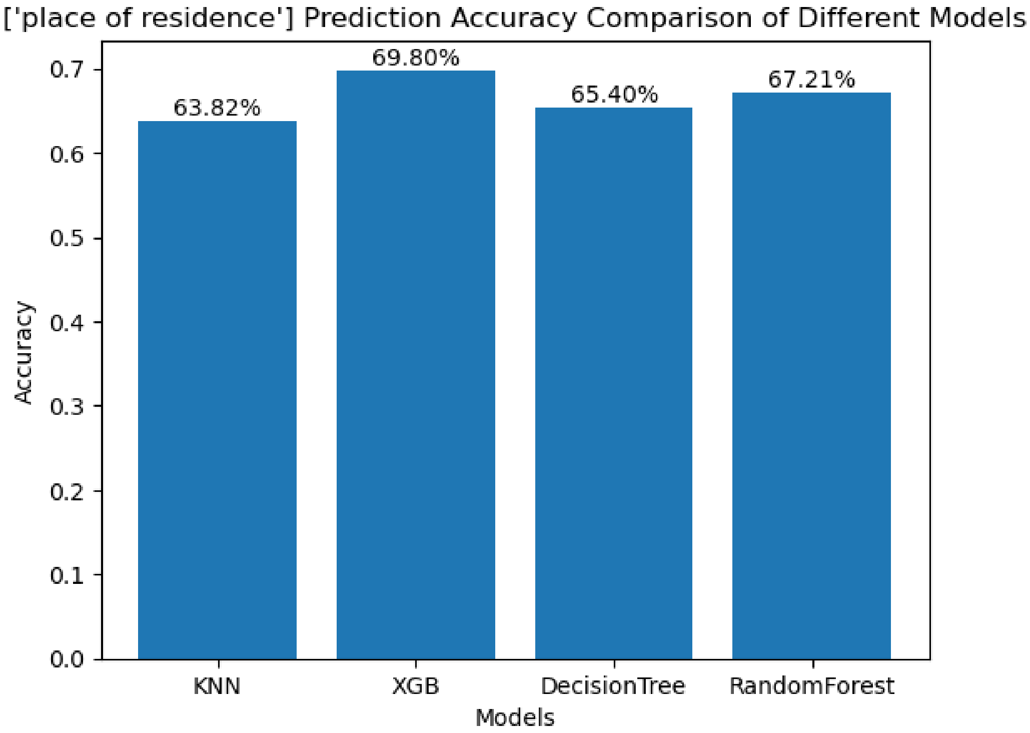


Figure 3.10