MA 415 Final Paper
Crowdedness at Gym

Chengyuan E
Xueying Jiang
Guanying Deng
Chi Yu Yeh

1. Abstract

This study attempted to use statistical models and analysis to explain the pattern of crowdedness in a college gym. We looked at different factors that could potentially influence the number of people in a college gym, such as the time of the day, whether or not is weekend or holiday, whether or not is during a semester, and the temperature of the day. This study provided a general idea about the crowdedness of the gym based on different conditions, from the statistical perspective of model fitting, model diagnosis, and model prediction.

2. Introduction

2.1. Research Question

College students spend a large amount of time doing exercises in the gym. School gym is the place where we can keep fit, stay healthy and make friends. A growing amount of college students tend to spend more time in gym in recent years. College students who go to the gym frequently sometimes face a problem, that during certain time periods, gym is extremely crowded that students have to stay in lines waiting for equipments. Therefore, an analysis about how crowded is the gym during different time periods is necessary.

We are interested in what factors affect the crowdedness of gym. Most of our factors are related to time, but in different dimensions, including hour of the day, day of the week, month of the year, on weekend or not, on holiday or not, at the start of semester or not and during the semester or not. Besides time factors, we also look at how would temperature affect
crowdedness in gym.

2.2. Data Description

The dataset “Crowdedness at campus gym” came from Kaggle and the details of this dataset can be found here: https://www.kaggle.com/nsrose7224/crowdedness-at-the-campus-gym.

The original dataset consists of 26,000 people counts from 2015-08-14 to 2017-03-08. The features include:

Variable	**Data Type**	**Classification**	**Description**
number_people	integer	Continuous	Number of people
timestamp	integer	Continuous	Number of seconds since beginning of day
day_of_week	integer	Categorical	0 (monday) - 6 (sunday)
is_weekend	integer	Binary	boolean, 1 = saturday or sunday, 0 = otherwise
is_holiday	integer	Binary	boolean, 1 = federal holiday, 0 = otherwise
temperature	float	Continuous	Temperature outside (degrees Fahrenheit)
is_start_of_semester	integer	Binary	1 = the beginning of a school semester, 0 = otherwise

is_during_semester	integer	Binary	1 = during the semester, 0 = otherwise
month	integer	Categorical	Month in a year: 1(jan) - 12(dec)
hour	integer	Categorical	Hour in a day: 0 - 23
day	string	Categorical	Day in a month (1-31)
year	string	Categorical	year

2.3. Dataset Modification

Since the dataset includes more than 60k observations (from 2015-08-14 to 2017-03-08), it might be hard to analyze in R. Therefore, we only take 4 months observations from 2016-11-01 to 2017-02-28 that include 10396 rows. Since the date variable contains unnecessary information, is was excluded in our analysis. Instead, we added an attribute “day” that extract information from data. Therefore, our Y variable to analysis is number of people, and our X variables are day_of_week, is_weekend, is_holiday, temperature, is_start_of_semester, is_during_semester, month, hour and day.

ò

3. Data Analysis and Visualizations

3.1. Variation of Single Variables

3.1.1. number_people and temperature

The variable of number_people is roughly normally distributed (Figure 1) except that there are some values that are close to zero. This actually make sense for our dataset. Because our dataset records number of people every 10 minutes for 24 hours, which means it would contain observations that is 3:10 AM in the morning and this would not be a common time that people go to gym. It has a median of 16 and mean of 16.07. Maximum value is 88 and minimum is 0.

The variable of temperature is roughly normally distributed and skewed to the right (Figure 2). It has a median of 52.33 fahrenheit and mean of 53.34 fahrenheit. Maximum value is 73.39 fahrenheit and minimum is 38.92 fahrenheit.

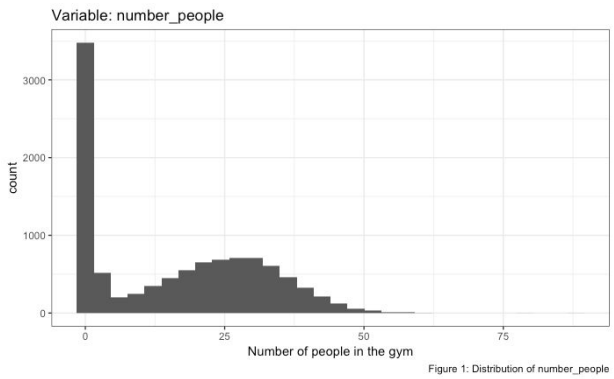

Figure 1: Distribution of number_people

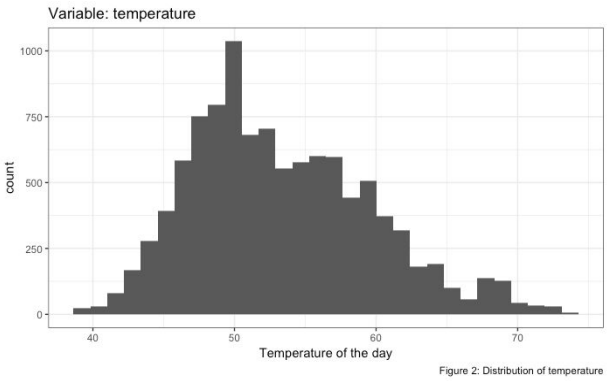

Figure 2: Distribution of temperature

3.1.2. day_of_week, is_weekend, is_holiday

"day_of_week" is a categorical variables. "is_weekend" and "is_holiday" are binary variables. The boxplots in figures 3, 4 & 5 show the distribution of our "day_of_week", "is_weekend", and "is_holiday" variables. As we can imagine, the "day_of_week" variable has a roughly uniform distribution; "is_weekend" variable's distribution is roughly about 5:2.

Figure 3: number_people corresponding to weekdays

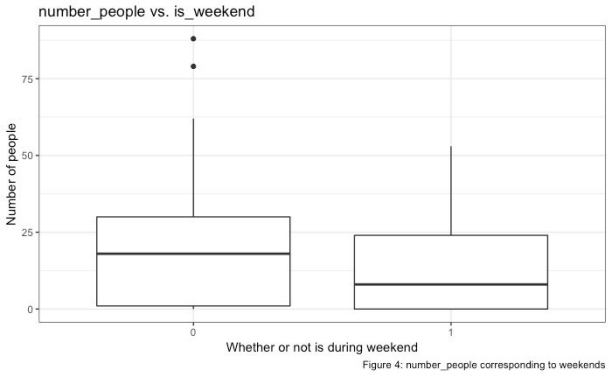

Figure 4: number_people corresponding to weekends

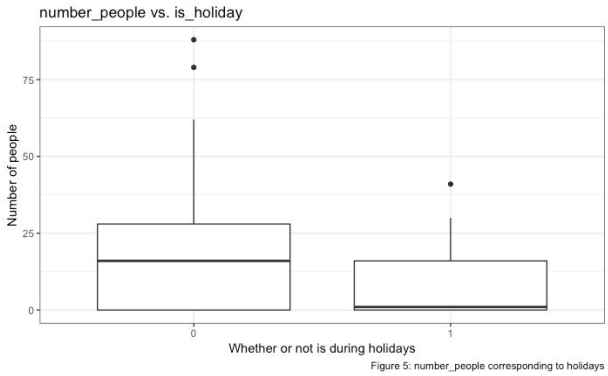

Figure 5: number_people corresponding to holidays

3.1.3. month, hour, day

Figures 6, 7 & 8 show the distribution of variables month, hour and day. As it was mentioned above, we are taking the data from Nov. 2016 to Feb. 2017. In Figure 6, we can see there is less people in the gym in Dec and Jan, which is the time of a winter break for universities. In Figure 7, we can see the gym have no people visiting during 1 AM to 5 Am. Figure 8 shows the distribution of number of people in the gym corresponding to the day of the month. There is not any obvious trend for this relationship.

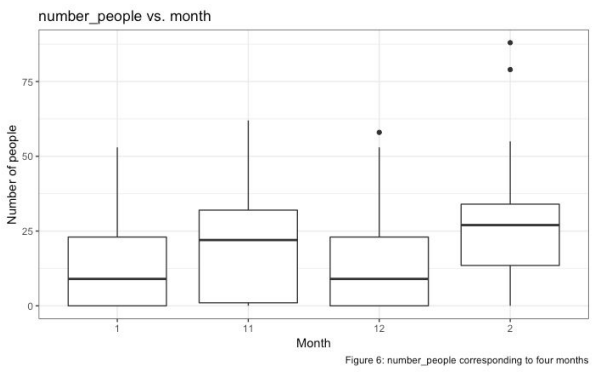

Figure 6: number_people corresponding to four months

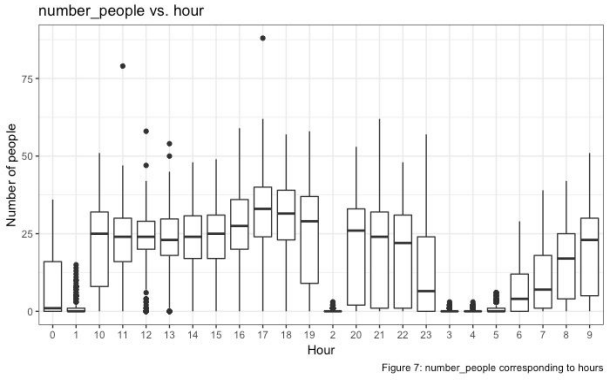

Figure 7: number_people corresponding to hours

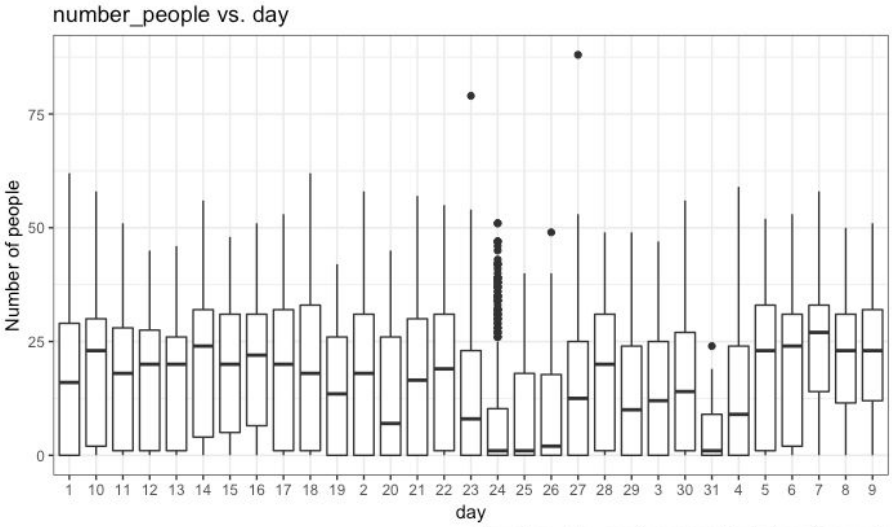

Figure 8: number_people corresponding to days of the month

3.1.4. is_start_of_semester, is_during_semester

Figures 9 & 10 show the distribution of number of people corresponding to whether it is the start of the semester or during the semester. We can tell that there are more people in the gym in the beginning of the semester comparing to other times. Also, there are more people in the gym in during the semester than not during the school year.

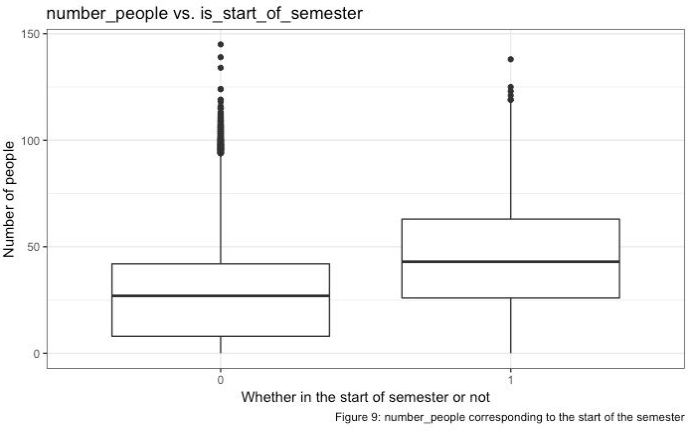

Figure 9: number_people corresponding to the start of the semester

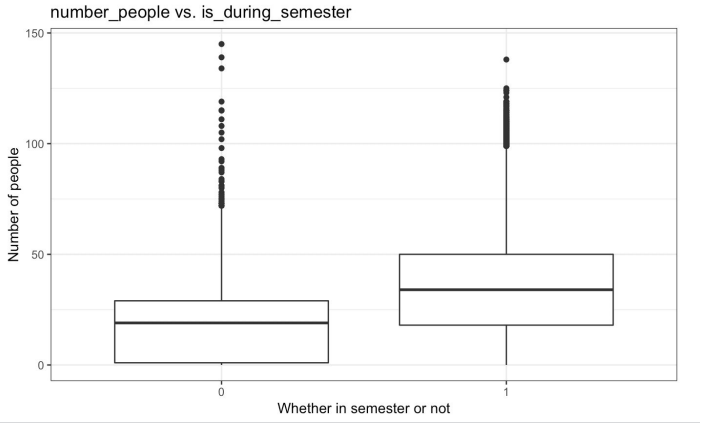

Figure 9.2: number_people corresponding to the time interval during the semester.

4. Covariation between Multiple Variables

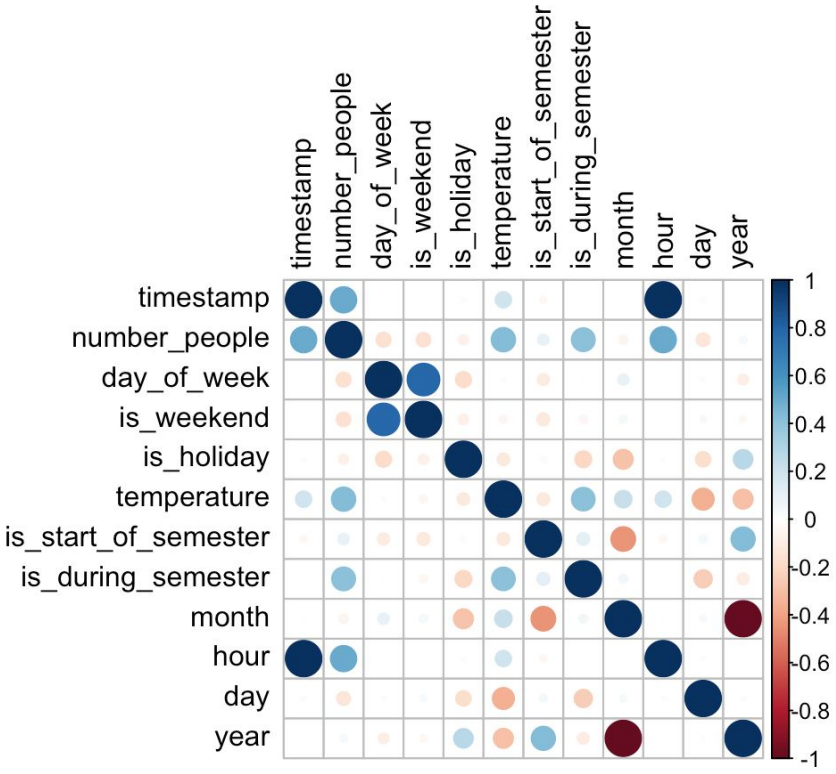

Figure 11: Correlation Matrix

A correlation matrix in R computes correlation majorly between continuous variables, but it also computes a corrected pearson's correlation between a mix of categorical and continuous variables. According to our correlation matrix of the dataframe, Timestamp, temperature, is_during_semester, and hour are relatively more correlated to crowdedness of the gym than other variables. Also, independent variable hour and timestamp, year and month seems to be closely correlated. This might increase the risk of collinearity while doing statistical modelling. Therefore we would focus on one of the two variables in the pair hour and timestamp, and pair year and month. Choosing the variable would be implemented in later analysis. As we mentioned, correlation computed for categorical and continuous variables might not be as accurate as the ones computed for continuous variables. Therefore, we need more evidence for the correlation between month and year, hour and timestamp. In this case, looking at vif gives us more information:

	Variables	VIF
1	timestamp	434.162209
2	number_people	1.950164
3	day_of_week	2.921658
4	is_weekend	2.785965
5	is_holiday	1.248821
6	temperature	2.089330
7	is_start_of_semester	1.408524
8	is_during_semester	1.529943
9	month	93.293390
10	hour	434.340602
11	day	1.189706
12	year	95.027919

Figure 12: VIF Table 1

	Variables	VIF
1	timestamp	1.471319
2	number_people	2.014602
3	day_of_week	2.985973
4	is_weekend	2.860126
5	is_holiday	1.286495
6	temperature	1.649939
7	is_start_of_semester	1.333965
8	is_during_semester	1.549142
9	month	1.479483
10	day	1.274489

Figure 13: VIF Tables 2

Year, month, timestamp, and hour all have a vif above 100, which is a strong indicator of collinearity. Since timestamp and month captures time variations in greater detail, we choose to delete hour and year from our dataset. vifs of variables in the new dataset all goes down to below 4, which is an acceptable range by convention.

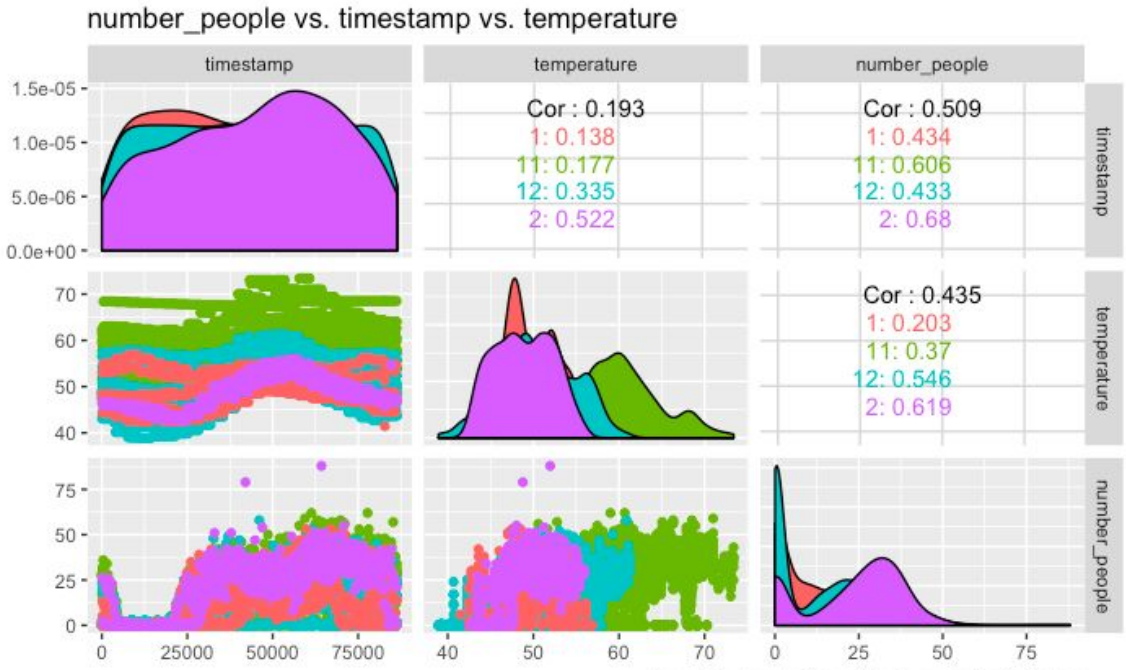

Figure 14: Correlation matrix for numerical variables

For covariation within continuous variables, we can see that dependent variables have moderate correlations of around 0.5 with our independent variable, and dependent variables have low correlation of

0.19. More importantly, it seems like as timestamp or temperature goes up, the crowdedness of the gym increases as well. Coloring the trend with month, we can see that the temperature is highest in Nov. Also, Nov has the highest crowdedness possible. This may suggest an effect of temperature on the crowdedness of the gym.

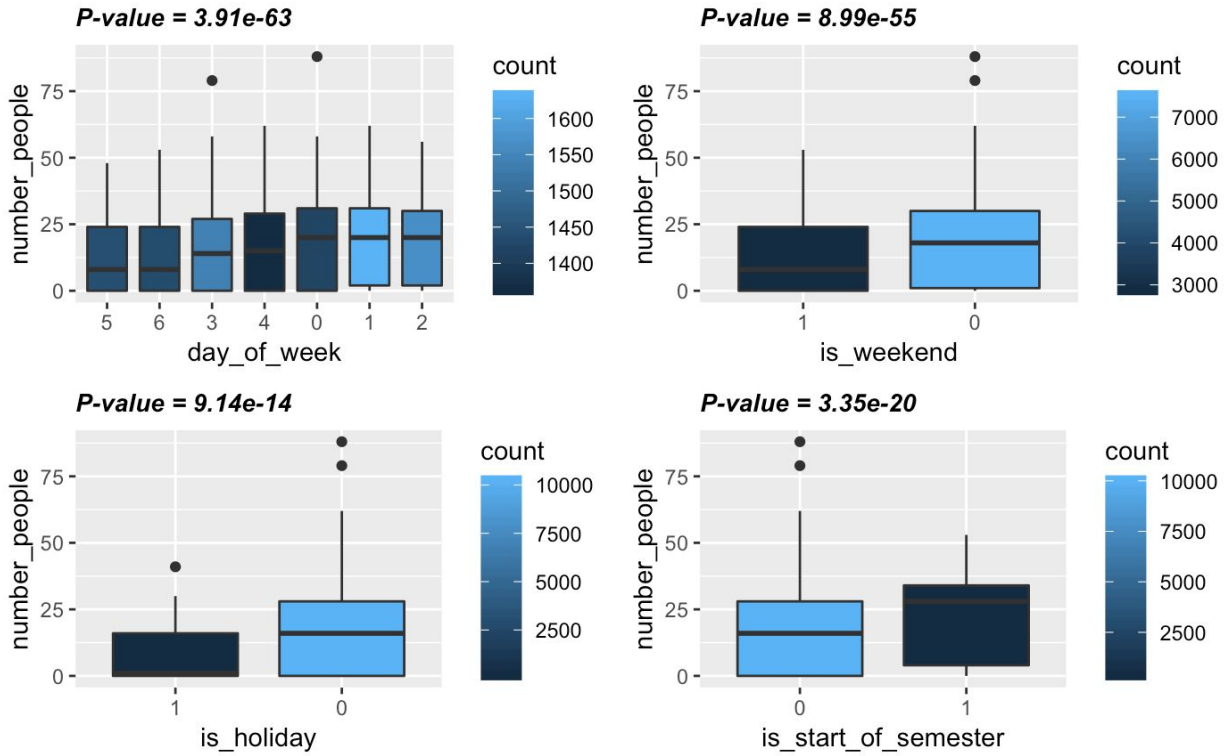

Figure 15: Boxplots with ANOVA 1

Above each boxplot are the p-value for the anova analysis of statistical significance between two variables. A p-value below 0.05 by convention suggests statistical significance, or a strong correlation between two variables. As we can see from the boxplots, there is significant difference between crowdedness in weekend and weekdays. The count for weekdays are significantly higher. Suggesting that the weekly routine for most people are during weekdays. The same situation applies to is_holidays. However, for during semester and start of semester. Both counts seems to be really high. The two boxplots seems to be contradictory, and are something we should investigate further into the analysis. Also, by looking at month, there seems to be less crowdedness at the end and the beginning of the year. This is probably due to the Christmas and winter holidays.

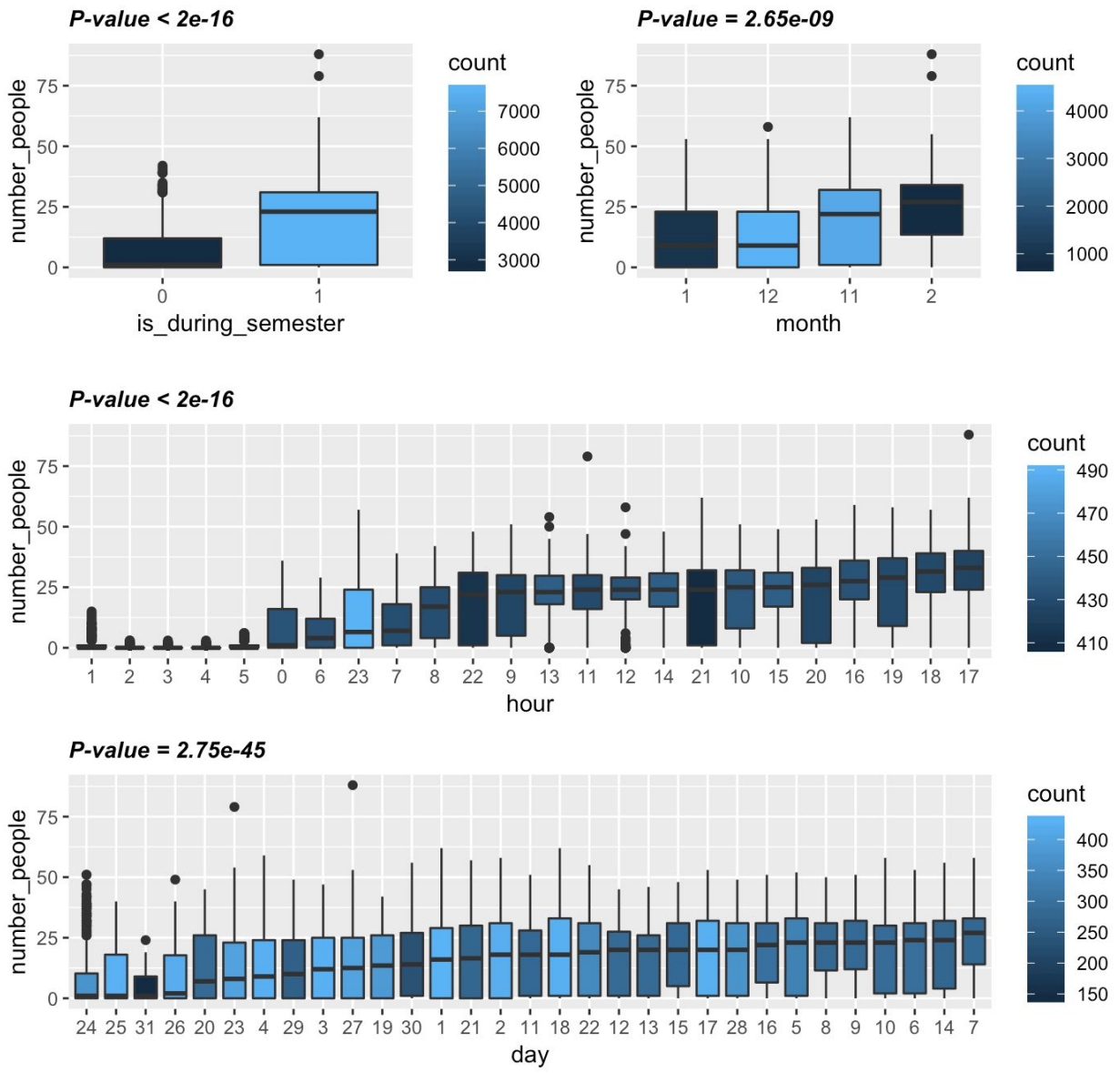

Figure 15: Boxplots with ANOVA 2

From boxplot for hours and day, we can see that crowdedness in gym are typically between 9 and 22. And there's significantly low activities in 23th, 24th, 25th, and 26th of each month.

5. Modeling and Analysis

5.1. Fit model

Given that independent variables hour and year have been excluded from the pool of considered independent variables for modelling, our full model would be set to all the covariates that are left. Moreover, it is common to think that during weekdays people tends to go to the gym in the early morning before school/work or at evening after work/school, whereas during weekends people may go there during the day. So we believe an interaction term between variable timestamp (which indicated the time of the day) and variable day_of_week could be statistically significant to our model, thus we decide to include this interaction term in our pool of independent variables.For model selection, there has been much written about the relative merits of AIC, AICC, and BIC: "AIC and AICc have the desirable property that they are efficient model selection criteria. As the sample gets larger, the error obtained in making predictions using the model chosen using these criteria becomes indistinguishable from the error obtained using the best possible model among all candidate models. That is, in large sample predictive sense, it is as if the best approximation was known to the data analyst. Other criteria, such as the Bayesian Information Criterion, BIC ... do not have this property."(Simonoff 2003, p.46) A common data analysis strategy is to utilize and compare AIC/AICc, BIC, and adjusted R square by comparing the models which minimize AIC/AICc and BIC with the model that maximizes adjusted R square. In our case since we have sufficient sample data we would not be using AICc, and for AIC and BIC the procedure we would be using is backward, since we believe forward method has a higher chance of incurring the risk of overfitting thus lead to risky decisions.

Full model:

```
lm(formula = d$number_people ~ d$timestamp + d$day_of_week +
    d$is_weekend + d$is_holiday + d$temperature + d$is_start_of_semester +
    d$is_during_semester + d$month + d$day)

Coefficients:
                         Estimate Std. Error t value Pr(>|t|)
(Intercept)            -2.559e+01  1.056e+00 -24.227  < 2e-16 ***
d$timestamp             2.715e-04  4.138e-06  65.602  < 2e-16 ***
d$day_of_week          -1.315e+00  8.561e-02 -15.364  < 2e-16 ***
d$is_weekend            5.889e-01  3.753e-01   1.569  0.11663
d$is_holiday           -6.485e+00  9.223e-01  -7.031 2.17e-12 ***
d$temperature           5.719e-01  1.921e-02  29.767  < 2e-16 ***
d$is_start_of_semester  1.921e+00  6.267e-01   3.065  0.00218 **
d$is_during_semester    1.029e+01  2.601e-01  39.549  < 2e-16 ***
d$month                -4.775e-01  3.188e-02 -14.975  < 2e-16 ***
d$day                   2.234e-02  1.253e-02   1.782  0.07473 .
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 10.24 on 10386 degrees of freedom
Multiple R-squared:  0.5118, Adjusted R-squared:  0.5114
F-statistic:  1210 on 9 and 10386 DF,  p-value: < 2.2e-16
```

Figure 16: Summary of full model

Final model after backward AIC and BIC:

```
lm(formula = d$number_people ~ d$timestamp + d$day_of_week +
    d$is_holiday + d$temperature + d$is_start_of_semester + d$is_during_semester +
    d$month + d$timestamp:d$day_of_week)

Coefficients:
              (Intercept)               d$timestamp             d$day_of_week               d$is_holiday
                -2.563e+01                 2.962e-04                -8.650e-01                -6.830e+00
             d$temperature     d$is_start_of_semester       d$is_during_semester                   d$month
                 5.569e-01                 1.905e+00                 1.022e+01                -4.748e-01
d$timestamp:d$day_of_week
                -8.038e-06
```

Figure 17: Final Model

By comparing AIC, BIC, and adjusted R square, we decide to choose the model which also excludes hour and year in addition to the variables that we have already excluded in our frame of independent variables. We build an regression of number_people on timestamp, day_of_week, is_holiday, temperature, is_start_of_semester, is_during_semester, month. The adjusted R square is moderate(0.512). All the variables are significant (all below 0.001). In this model, the betas of variables day_of_week, is_holiday and month are negative, which makes sense. When it gets closer to the weekends and holidays, people tends to go to gyms less and spend their time in other things. The month variable has an negative effects here can be interpret as that since we are only taking Jan, Feb, Nov and Dec, Nov and Dec are typical final exam period for school and students spend less time in the gym. Therefore, it makes sense to see there is an negative for month variable. For variables timestamp, temperature, is_start_of_semester and is_during_semester, they all have an coefficient larger than zero. They also make sense. Because more people tend to go to gym during evening after school/work; people tends to go to gym when the weather is warm; students typically have less school work and more time for gym in the start of the semester; for a college gym, there will be more people during the semester than not during the semester. In addition, The interaction term is significant. The beta of it is negative, meaning that during the same time period of each day, there are less people going to the gym when that day is closer to the weekends:

```
Coefficients:
                             Estimate Std. Error t value Pr(>|t|)
(Intercept)                -2.563e+01  9.504e-01 -26.962  < 2e-16 ***
d$timestamp                 2.962e-04  7.316e-06  40.486  < 2e-16 ***
d$day_of_week              -8.650e-01  1.015e-01  -8.523  < 2e-16 ***
d$is_holiday               -6.830e+00  8.957e-01  -7.625 2.66e-14 ***
d$temperature               5.569e-01  1.822e-02  30.560  < 2e-16 ***
d$is_start_of_semester      1.905e+00  6.238e-01   3.054  0.00226 **
d$is_during_semester        1.022e+01  2.569e-01  39.804  < 2e-16 ***
d$month                    -4.748e-01  3.175e-02 -14.952  < 2e-16 ***
d$timestamp:d$day_of_week -8.038e-06  2.027e-06  -3.965 7.38e-05 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 10.24 on 10387 degrees of freedom
Multiple R-squared:  0.5123,    Adjusted R-squared:  0.5119
F-statistic:  1364 on 8 and 10387 DF,  p-value: < 2.2e-16
```

Figure 16: Summary of final model

5.2. Diagnosis

A crucial step after fitting the model is to check the residual for assumptions. From the plot of standard residuals versus observed values, we can see an upward going trend, indicating that we still need more

features and information to better fit the model. From the plot of standard residuals versus fitted values, we can see that besides the eccentric straight line at the bottom left of the plot, the points seems to be randomly distribute around zero thus indicating that our model met the normality assumption. From the plot of standard residuals versus observed value and the plot of standard residual versus fitted value, we can see that this issue is common, that part of the points seems to be in the pattern of a straight line. The straight line observed is horizontal in rstandard versus observed value and steep in rstandard versus fitted value plot. One major reason is that there is a lot of constant response variables versus independent variables. After investigating the dataset, we discovered that during 1am to 5am, the gym is closed, therefore no one is visiting the gym and the number of people during that time frame is always 0. Thus, as an attempt to improve the model to make it more adhere to the assumption of normality, we removed all data from 1am to 5am and refitted the model and plotted the diagnostic plots. In the standard residuals versus fitted value/observed value plot below, green points correspond to the dataset without data points from 1am to 5am and blue points correspond to the dataset with the data points in the early morning:

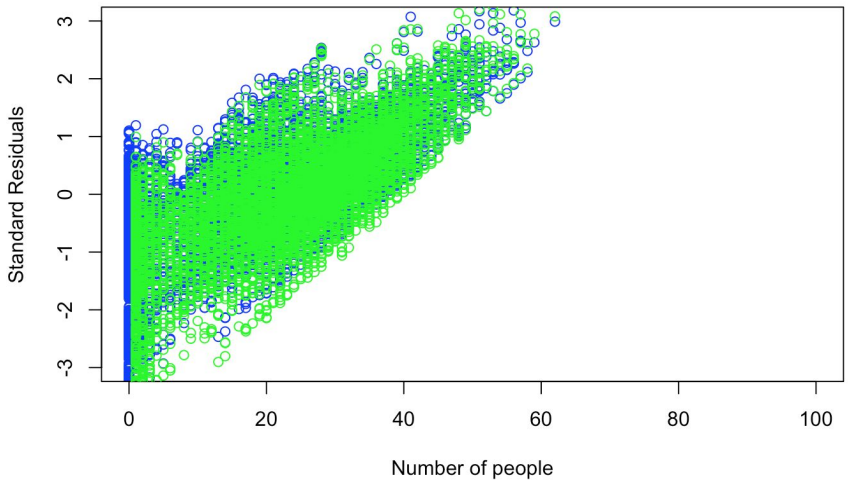

Figure 17: Standard residuals versus observed value

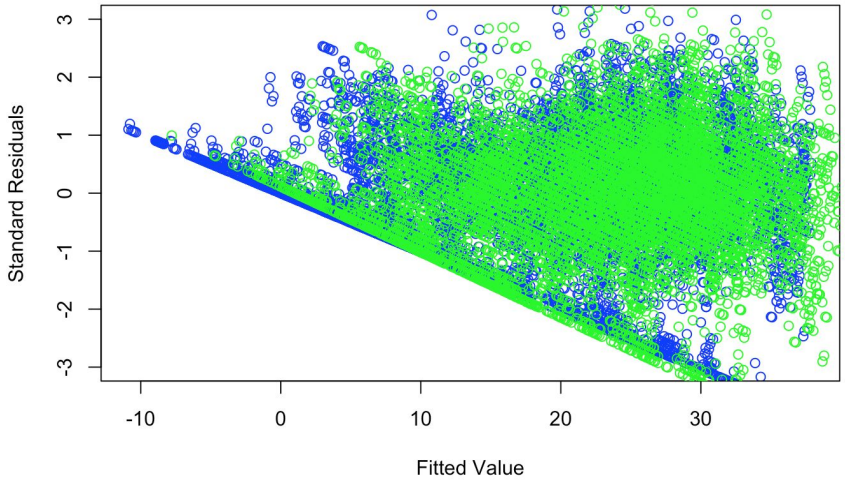

Figure 18: Standard residuals versus fitted value

As we can see, the residual plots still obtain the straight line pattern. This is because that after deleting the data from 1 am to 5 am, there are still a lot of duplicated 0's (796), along with a lot of duplicate values from other number of people that we have:

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
796	276	153	91	73	69	57	77	80	89	114	116	119	150	142	163	179	181	192	217	203	230	206	241	235	237	239	231	229	243	235
32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	62	79	88
188	210	204	159	154	153	127	109	93	73	63	43	38	50	30	38	23	24	13	17	7	7	5	6	5	4	3	1	2	1	1

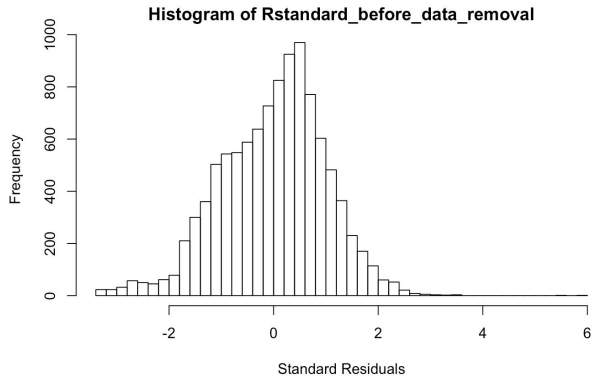

Figure 19: Standard residuals with early morning data

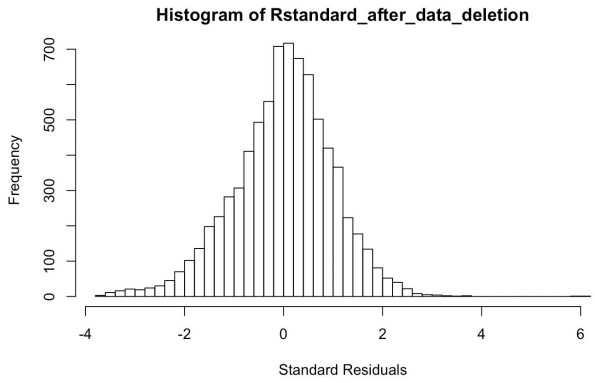

Figure 20: Standard residuals without early morning data

Comparing the histogram of the standard residuals before(r2) and after(r1) the data deletion, we can see that the histogram of standard residuals after data deletion looks more normally distributed than the histogram of that before data deletion. Also, from the qq plot we can further verify that the normality assumption is met to a certain extent, as the qq points adhere to the qq line in general.

5.3. Predictions

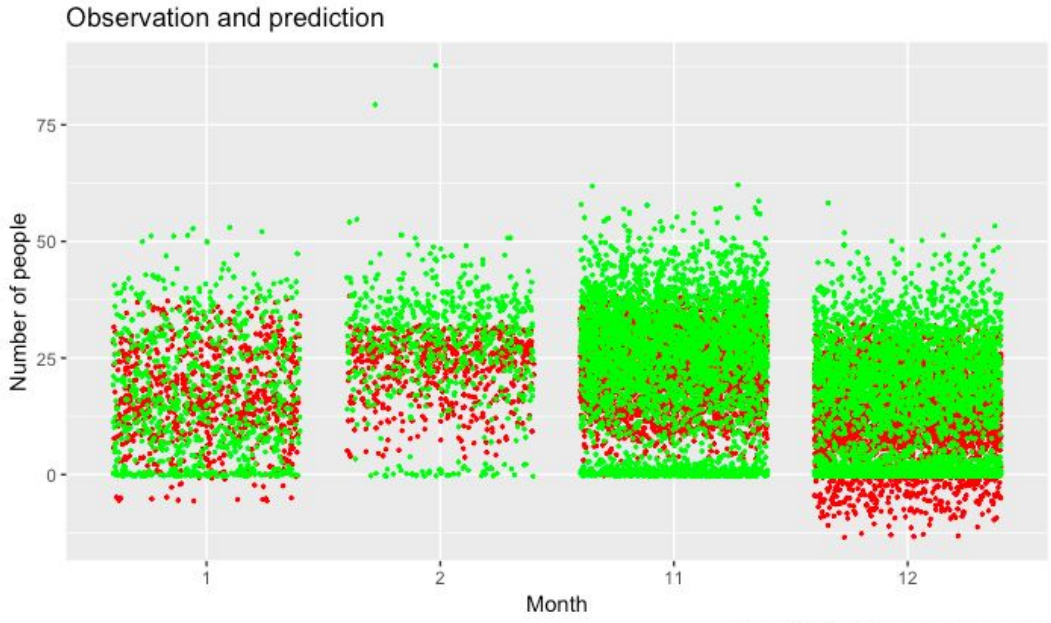

Figure 21: Predictions based on models

From figure 25, we can see how points in prediction (red) and actual (green) number of people distributed. The plot shows that generally, the actual observations are more spread out than prediction of number of people. As we can see in the graphs, the green points and red points generally overlaps in all four months. At the same time, no number of people in prediction(red points) was higher than 40, while there were a certain amount of actual number of people(green points) larger than 40 in all 4 months. We also noticed that there were certain amount of red points fall below y=0. Since we know that number of people cannot be negative, these red negative points might correspond to "0" in reality.

6. Discussion

In response to our research question, as we summarized in the modeling section, timestamp, day of week, interaction term between timestamp and day of week, is_holiday, temperature, is_start_of_semester, is_during_semester, and month are statistically significant indicators for number of people. However during modelling diagnosis, we discovered a straight line pattern in the plot of residuals versus observed values and residuals versus fitted values. Based on our observation in the dataset, there're a lot of points with number of people equals to 0. From further investigations into the origin of dataset, we discovered that the gym is closed in the early morning therefore none of the students would go to gym during that time period. Therefore, we filtered out hours between 1am and 5am and discovered that it slightly improved the normality of the residuals however it still did not resolve the straight line patterns within the residuals versus observed value/fitted value plot. This is because that many of our response variables are constant as independent variable varies. For instance after removing data from 1 am to 5 am, there are still more than 700 data points indicating 0 people at the gym, along with 200+ data points for 14 out of 88 unique responses. Essentially, our response variable are consists of nonnegative counts of an event in an interval of time or space. Therefore, modelling it with poisson distribution might be a more appropriate method. However, as we discovered in our exploratory data analysis, temperature affects the rate at which number of people is arriving each day. Since our dataset consists of data form November, December, January, and February, it is obvious that changes in general temperature occurs. In addition, according to our data, the range of temperature changes from day to day as well. Therefore, modelling our response variable with poisson regression might incur the risk of violating one of its fundamental assumptions - The rate at which events occur is the same for all intervals. Since our residual plots exhibits that a certain level of normality assumption are met, we decided to conclude with the results from our multiple linear regression model.

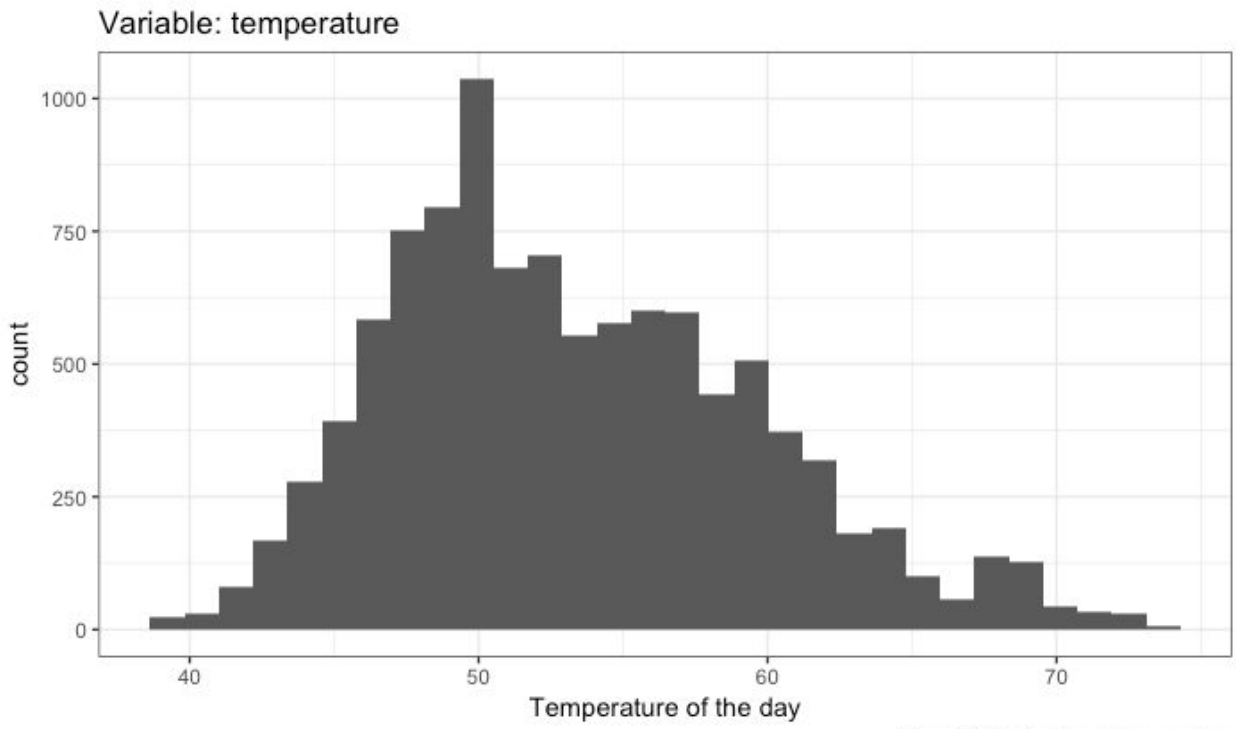

Figure 2: Distribution of temperature

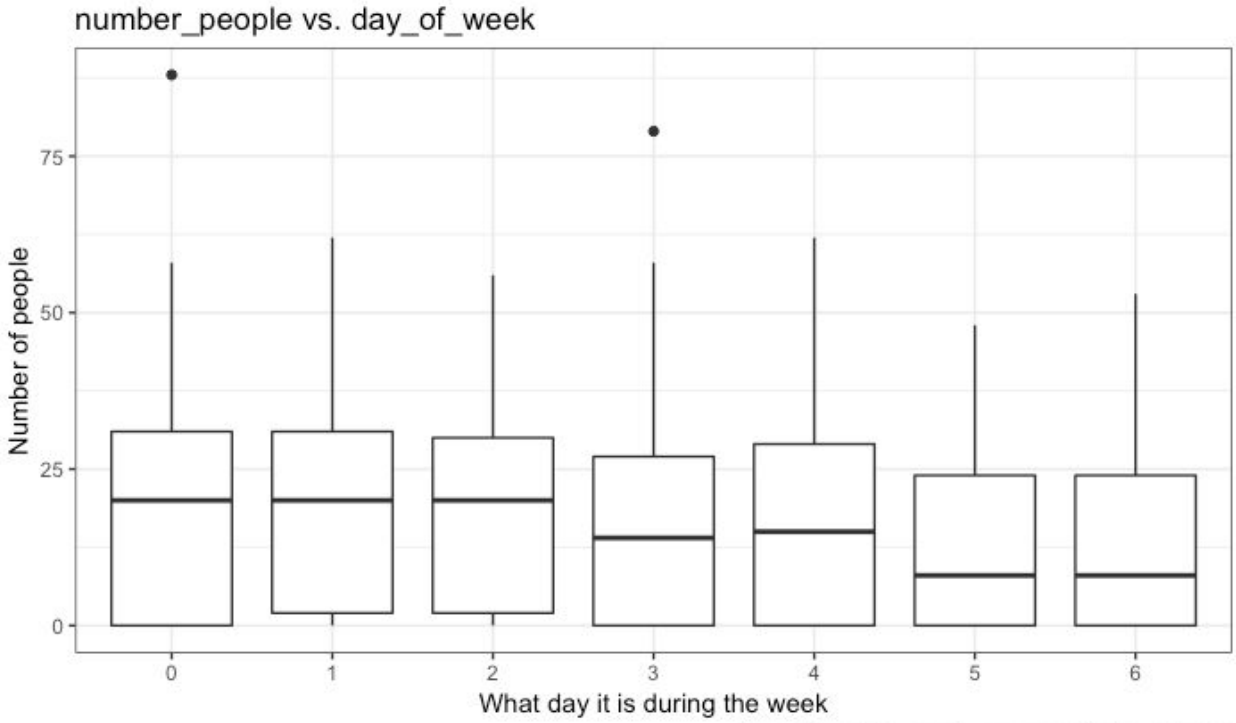

Figure 3: number_people corresponding to weekdays

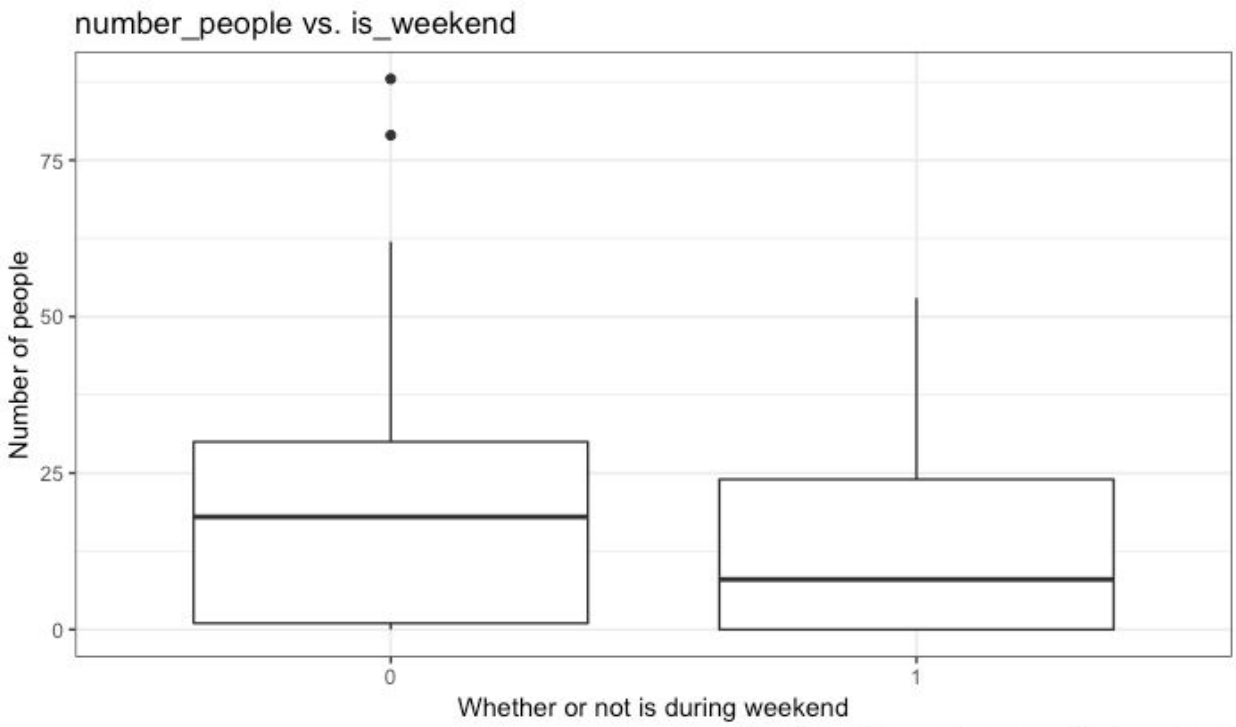

Figure 4: number_people corresponding to weekends

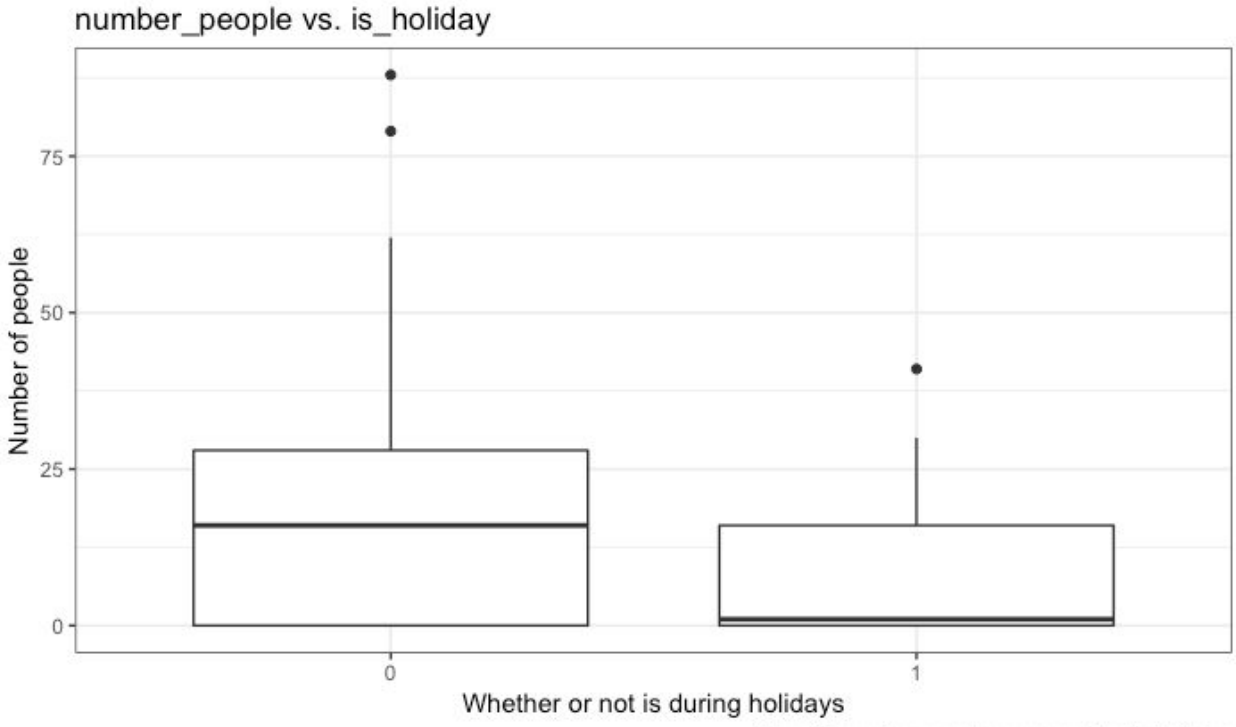

Figure 5: number_people corresponding to holidays

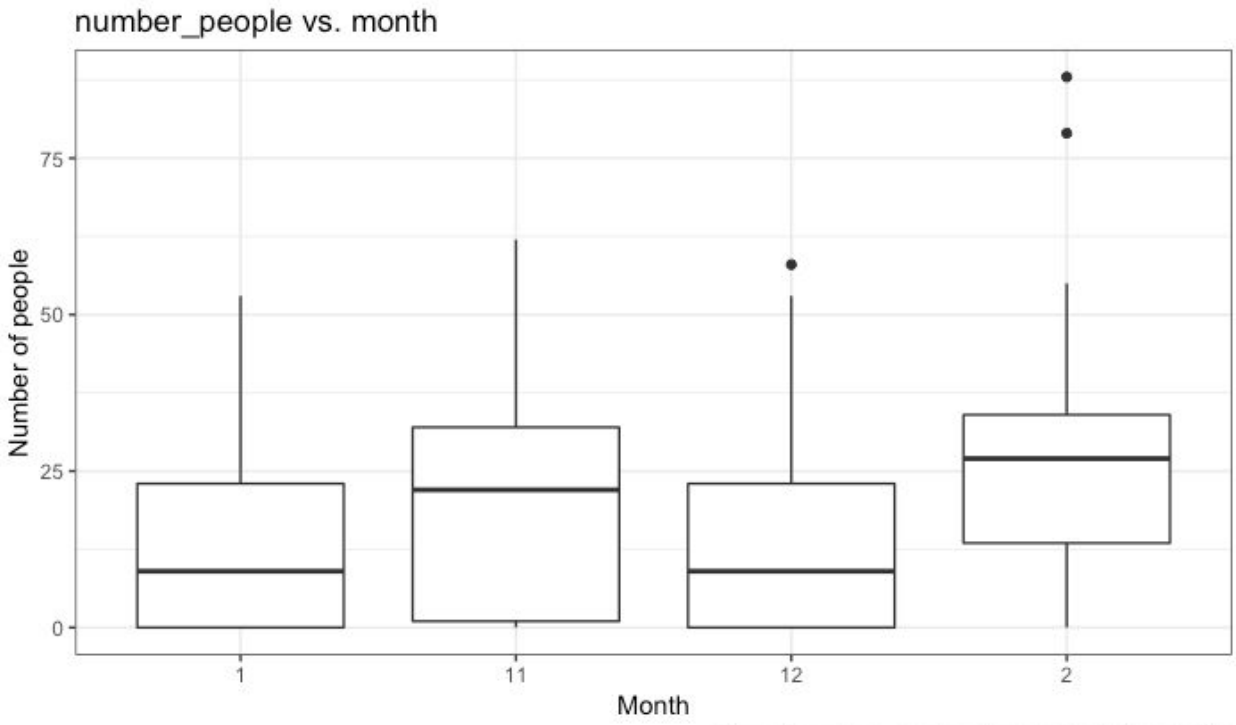

Figure 6: number_people corresponding to four months

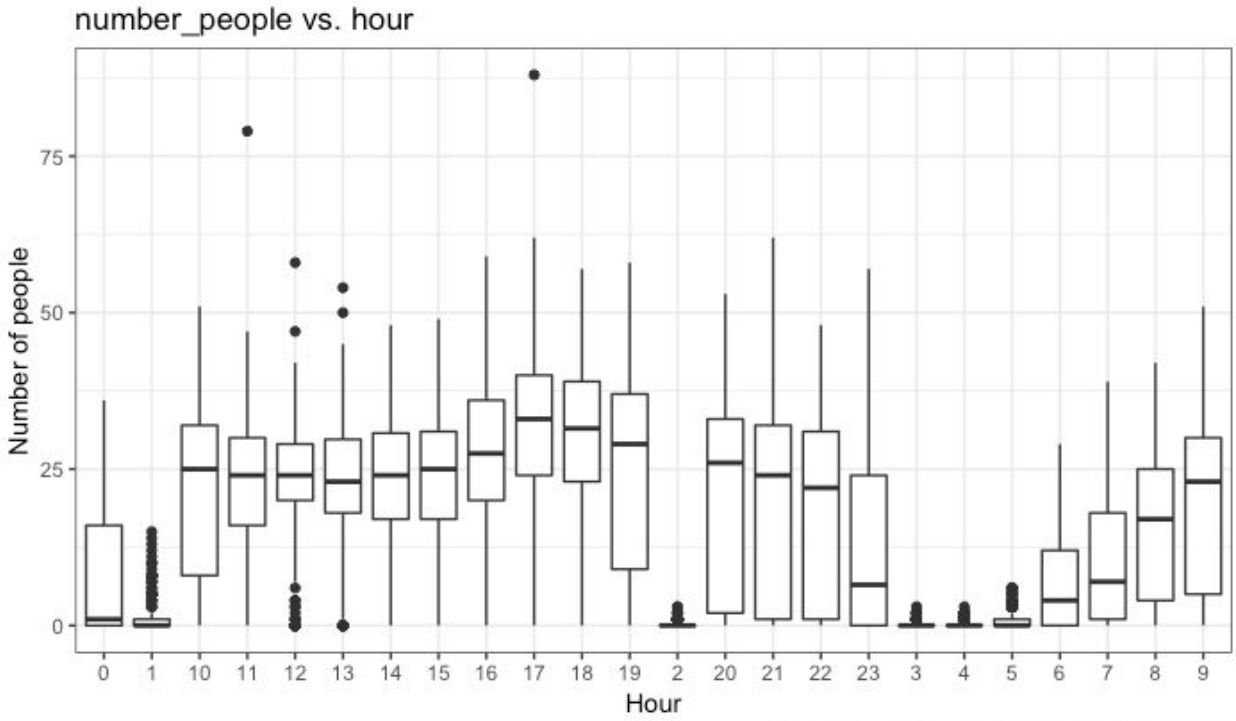

Figure 7: number_people corresponding to hours

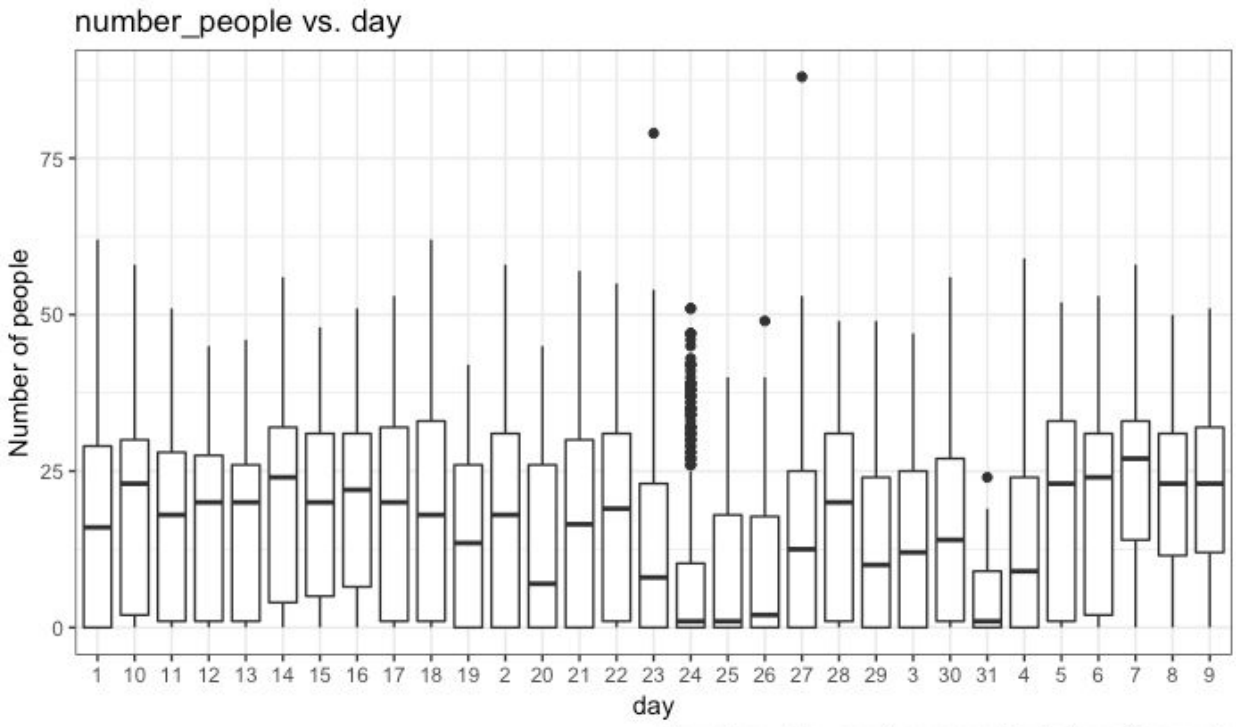

Figure 8: number_people corresponding to days of the month

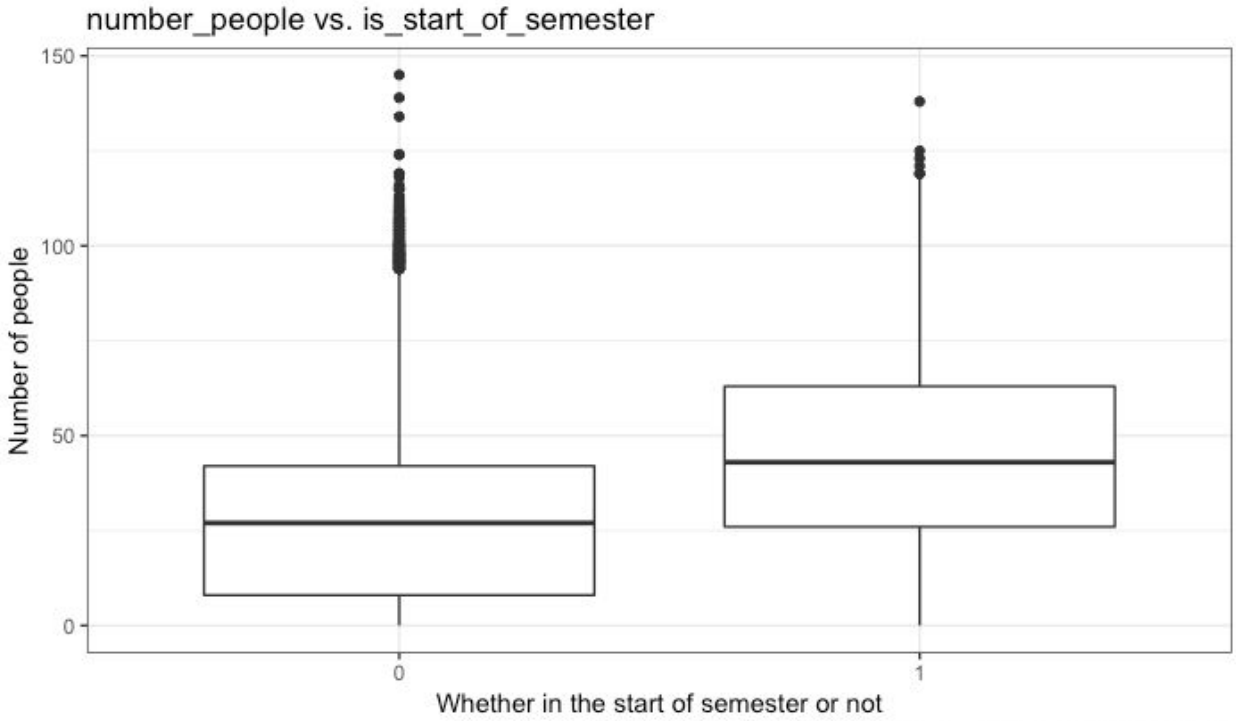

Figure 9: number_people corresponding to the start of the semester

2. R code

###Data Cleaning

```{r, echo=FALSE, message=FALSE, warning=FALSE}
#Creating columns for year and day from the date column
gym_data <- gym_data %>% mutate(day = substr(date, 9, 10)) %>% mutate(year = substr(date, 1, 4))
# Assert type of year and day to be numeric
#typeof(gym_data$year)
#typeof(gym_data$day)
gym_data$year<-as.numeric(gym_data$year)
gym_data$day <- as.numeric(gym_data$day)
#Getting the observations between Nov. 2016 and Feb. 2017
data_set <- gym_data[c(50466:60861),c(1:13)]
data_set<-data_set%>%dplyr::select(timestamp,number_people,day_of_week,is_weekend,is_holiday,tem
perature,is_start_of_semester,is_during_semester,month,hour,day,year)
d<-data_set
d$date<-NULL
```

#Variation of Single Variables

number_people and temperature

```{r,warning=FALSE,message=F}
(np<-ggplot(data_set)+ geom_histogram(aes(x = number_people))+theme_bw()+
  ggtitle("Variable: number_people")+
  xlab("Number of people in the gym") + labs(caption="Figure 1: Distribution of number_people"))
 # xlim(2,max(data_set$number_people))
#summary(data_set$number_people)

(t<-ggplot(data_set)+ geom_histogram(aes(x = temperature))+theme_bw()+
  ggtitle("Variable: temperature")+
  xlab("Temperature of the day") +labs(caption="Figure 2: Distribution of temperature"))

#hist(data_set$temperature)
#summary(data_set$temperature)

library(ggpubr)
#ggarrange(np, t, ncol = 2)

```

day_of_week, is_weekend, is_holiday

```{r}
```

```
#unique(data_set$day_of_week)
#unique(data_set$is_weekend)
#unique(data_set$is_holiday)

(pp1<-ggplot(data_set,mapping = aes(x=as.character(day_of_week),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("What day it is during the week")+
  ylab("Number of people")+
  ggtitle("number_people vs. day_of_week")+labs(caption="Figure 3: number_people corresponding to
weekdays"))

(pp2<-ggplot(data_set,mapping = aes(x=as.character(is_weekend),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("Whether or not is during weekend")+
  ylab("Number of people")+
  ggtitle("number_people vs. is_weekend")+labs(caption="Figure 4: number_people corresponding to
weekends"))

(pp3<-ggplot(data_set,mapping = aes(x=as.character(is_holiday),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("Whether or not is during holidays")+
  ylab("Number of people")+
  ggtitle("number_people vs. is_holiday")+labs(caption="Figure 5: number_people corresponding to
holidays"))

#ggarrange(pp1,pp2,pp3,ncol = 2, nrow = 2)
```


month,hour,day
```{r}

(p1 <- ggplot(data_set,mapping = aes(x=as.character(month),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("Month")+
  ylab("Number of people")+
  ggtitle("number_people vs. month")+labs(caption="Figure 6: number_people corresponding to four
months"))
```

```
(p2 <- ggplot(data_set,mapping = aes(x=as.character(hour),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("Hour")+
  ylab("Number of people")+
  ggtitle("number_people vs. hour")+ labs(caption="Figure 7: number_people corresponding to hours"))
#+ coord_flip()

(p3 <- ggplot(data_set,mapping = aes(x=as.character(day),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("day")+
  ylab("Number of people")+
  ggtitle("number_people vs. day")+labs(caption="Figure 8: number_people corresponding to days of the
month"))# + coord_flip()

#ggarrange(p1,p2,p3,ncol = 3,nrow=1)
#hist(data_set$month)
#summary(data_set$month)
#hist(data_set$hour)
#summary(data_set$hour)
#hist(as.numeric(data_set$day))
#summary(as.numeric(data_set$day))
```

###is_start_of_semester, is_during_semester
```{r}
(s<-ggplot(gym_data, mapping = aes(x=as.character(is_start_of_semester),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("Whether in the start of semester or not")+
  ylab("Number of people")+
  ggtitle("number_people vs. is_start_of_semester")+labs(caption="Figure 9: number_people
corresponding to the start of the semester"))

(d<-ggplot(gym_data,mapping = aes(x=as.character(is_during_semester),y=number_people))+
  geom_boxplot() +
  theme_bw() +
  xlab("Whether in semester or not")+
  ylab("Number of people")+
  ggtitle("number_people vs. is_during_semester")+ labs(caption="Figure 10: number_people
corresponding to semesters"))
```

#ggarrange(s, d, ncol = 2, nrow = 1)

```

#Covariation between Multiple Variables
<br>
```{r, echo=FALSE, message=FALSE, warning=FALSE}
library(GGally)
library(ggpubr)
library(corrplot)
d <- data_set # Abbreviation
d$date<-NULL
corrplot(cor(d), tl.cex=1,tl.col = "black")#+ labs(caption="Figure 11: Correlation Matrix"))
```


```{r, echo=FALSE, message=FALSE, warning=FALSE}
library(usdm)
tes <- d
vif(d)
tes$year<-NULL
tes$hour<-NULL
vif(tes)
```

```{r, echo=FALSE, message=FALSE, warning=FALSE}
(d %>% dplyr::select(timestamp, temperature,number_people) %>%
  ggpairs(mapping=ggplot2::aes(colour = as.character(d$month)))+ ggtitle("number_people vs.
timestamp vs. temperature")+
  labs(caption="Figure 14: Correlation matrix for numerical variables"))
```

```{r, echo=FALSE, message=FALSE, warning=FALSE}

sub <- function(a){
 if(a==0){
   res2="P-value < 2e-16"
   return(res2)
 }
 else{
   a=as.character(a)
   a1=substr(a,1,4)
   a2=substr(a, nchar(a)-4+1, nchar(a))

```
   res1 = paste(a1,a2,sep="")
   res2=paste("P-value =", res1, sep=" ")
   return(res2)
  }

}
cor1 <- sub(summary(aov(d$number_people ~d$day_of_week))[[1]][["Pr(>F)"]][1])
dow<- d %>% group_by(day_of_week) %>%
                mutate(count=n()) %>%
                ggplot(aes(x=reorder(factor(day_of_week), number_people, FUN = median),
y=number_people)) +
                labs(x = "day_of_week") +
geom_boxplot(aes(fill=count))+ggtitle(cor1)+theme(plot.title=element_text(size=10.5,
face="bold.italic"))

cor2<- sub(summary(aov(d$number_people ~d$is_weekend))[[1]][["Pr(>F)"]][1])
isw<-d %>% group_by(is_weekend) %>%
                mutate(count=n()) %>%
                ggplot(aes(x=reorder(factor(is_weekend), number_people, FUN = median),
y=number_people)) +
                labs(x = "is_weekend") + geom_boxplot(aes(fill=count))
+ggtitle(cor2)+theme(plot.title=element_text(size=10.5, face="bold.italic"))

cor3<- sub(summary(aov(d$number_people ~d$is_holiday))[[1]][["Pr(>F)"]][1])
ish<-d %>% group_by(is_holiday) %>%
                mutate(count=n()) %>%
                ggplot(aes(x=reorder(factor(is_holiday), number_people, FUN = median),
y=number_people)) +
                labs(x = "is_holiday") + geom_boxplot(aes(fill=count))
+ggtitle(cor3)+theme(plot.title=element_text(size=10.5, face="bold.italic"))

cor4<- sub(summary(aov(d$number_people ~d$is_start_of_semester))[[1]][["Pr(>F)"]][1])
issfs<-d %>% group_by(is_start_of_semester) %>%
                mutate(count=n()) %>%
                ggplot(aes(x=reorder(factor(is_start_of_semester), number_people, FUN = median),
y=number_people)) +
                labs(x = "is_start_of_semester") + geom_boxplot(aes(fill=count))
+ggtitle(cor4)+theme(plot.title=element_text(size=10.5, face="bold.italic"))

cor5<- sub(summary(aov(d$number_people ~d$is_during_semester))[[1]][["Pr(>F)"]][1])
isds<-d %>% group_by(is_during_semester) %>%
                mutate(count=n()) %>%
```

```
        ggplot(aes(x=reorder(factor(is_during_semester), number_people, FUN = median),
y=number_people)) +
        labs(x = "is_during_semester") + geom_boxplot(aes(fill=count))
+ggtitle(cor5)+theme(plot.title=element_text(size=10.5, face="bold.italic"))

cor6<- sub(summary(aov(d$number_people ~d$month))[[1]][["Pr(>F)"]][1])
mon <-d %>% group_by(month) %>%
        mutate(count=n()) %>%
        ggplot(aes(x=reorder(factor(month), number_people, FUN = median), y=number_people)) +
        labs(x = "month") + geom_boxplot(aes(fill=count))
+ggtitle(cor6)+theme(plot.title=element_text(size=10.5, face="bold.italic"))

cor7<- sub(summary(aov(d$number_people ~d$hour))[[1]][["Pr(>F)"]][1])
hr <-d %>% group_by(hour) %>%
        mutate(count=n()) %>%
        ggplot(aes(x=reorder(factor(hour), number_people, FUN = median), y=number_people)) +
        labs(x = "hour") + geom_boxplot(aes(fill=count))
+ggtitle(cor7)+theme(plot.title=element_text(size=10.5, face="bold.italic"))

cor8<- sub(summary(aov(d$number_people ~d$day))[[1]][["Pr(>F)"]][1])
day <-d %>% group_by(day) %>%
        mutate(count=n()) %>%
        ggplot(aes(x=reorder(factor(day), number_people, FUN = median), y=number_people)) +
        labs(x = "day") + geom_boxplot(aes(fill=count))
+ggtitle(cor8)+theme(plot.title=element_text(size=10.5, face="bold.italic"))

ggarrange(dow,isw, ish, issfs, ncol = 2, nrow = 2)+ labs(caption="Figure 14: Correlation matrix for
categorical variables")


```



```{r, echo = FALSE, message=FALSE,warning=FALSE}
ggarrange(isds, mon, ncol = 2, nrow = 2)+labs(caption="Figure 15: Correlation matrix for categorical
variables")
```



```{r, echo = FALSE, message=FALSE,warning=FALSE}
ggarrange(hr, day, ncol = 1, nrow = 2)

```
```
```

#Modeling and Analysis

#Modeling and Analysis

Fit model

```{r, echo=FALSE, message=FALSE, warning=FALSE}
mls <- lm(d$number_people ~
d$timestamp+d$day_of_week+d$is_weekend+d$is_holiday+d$temperature+d$is_start_of_semester+d$is
_during_semester+d$month+d$day)
summary(mls)

```

```{r, echo=FALSE, message=FALSE, warning=FALSE}
mls_interactions <- lm(d$number_people~d$timestamp*d$day_of_week+
                       d$is_weekend+
                       d$is_holiday+
                       d$temperature+
                       d$is_start_of_semester+
                       d$is_during_semester+
                       d$month+
                       d$day)

step(mls_interactions, direction="backward")
step(mls_interactions, direction="backward",k=log(length(d$number_people)))
summary(mls_interactions)
```

```{r, echo=FALSE, message=FALSE, warning=FALSE}

mls2_interactions <- lm(d$number_people~d$timestamp*d$day_of_week+
                        d$is_holiday+
                        d$temperature+
                        d$is_start_of_semester+
                        d$is_during_semester+
                        d$month)
summary(mls2_interactions)
```

Multiple Linear Regression with an Interaction Term of timestamp * day_of_week

Diagnosis

```{r, echo=FALSE, message=FALSE, warning=FALSE}
r2 <- rstandard(mls2_interactions)
# standard residuals vs. observed value
d1<- d %>% dplyr:: filter(number_people !=0)
mls1_interactions <- lm(d1$number_people~d1$timestamp*d1$day_of_week+
                        d1$is_holiday+
                        d1$temperature+
                        d1$is_start_of_semester+
                        d1$is_during_semester+
                        d1$month)

r1 <- rstandard(mls1_interactions)
# standard residuals vs. observed value


{plot(d$number_people, r2, xlim=c(0,100), ylim=c(-3,3), ylab="Standard Residuals", xlab="Number of people", col="blue")
points(d1$number_people, r1, xlim=c(0,100), ylim=c(-3,3), ylab="Standard Residuals", xlab="Number of people", col="green")}

# standard residuals vs. fitted value
{plot(mls2_interactions$fitted.values, r2, ylim=c(-3,3), ylab="Standard Residuals", xlab="Fitted Value", col="blue")
points(mls1_interactions$fitted.values, r1, ylim=c(-3,3), ylab="Standard Residuals", xlab="Fitted Value", col="green")}
```

As we can see, the residual plots still obtain the straight line pattern. This is because that after deleting the data from 1 am to 5 am, there are still a lot of duplicatied 0's (796), along with a lot of duplicated values from other number of people that we have:

```{r, echo=FALSE, message=FALSE, warning=FALSE}
table(d1$number_people)
```

Comparing the histogram of the standard residuals before and after the data deletion, we can see that the histogram of standard residuals after data deletion looks more normally distributed than the histogram of

that before data deletion. Also, from the qq plot we can further verify that the normality assumption is met, as the qq points adhere to the qq line.

```{r, echo=FALSE, message=FALSE, warning=FALSE}
# Histogram and QQ-plot
hist(r2, xlab="Standard Residuals", breaks=50)
```

```{r, echo=FALSE, message=FALSE, warning=FALSE}

# Histogram and QQ-plot
hist(r1, xlab="Standard Residuals", breaks=50)
```

```{r, echo=FALSE, message=FALSE, warning=FALSE}
{qqnorm(r2)
 qqline(r2)}
```

Prediction
```{r,echo=FALSE, message=FALSE, warning=FALSE}
library(modelr)
require(ggplot2)
d <- filter(d, hour > 5 | hour == 0)
g <- d%>% add_predictions(mls)
ggplot(g, aes(x= factor(month))) +                    # basic graphical object
  geom_point(aes(y=pred), colour="red",position = "jitter", size=0.5) +  # first layer
  geom_point(aes(y=number_people), colour="green", position = "jitter",size=0.5) + # second layer
  xlab("Month")+
  ylab("Number of people")+
  ggtitle("Observation and prediction")+
  labs(caption = "Figure 25: Predictions based on models")

```