Biostat620-homework1

Zhengrui Huang

2024-02-04

GitHub Link: https://github.com/huangzr1228/Biostat620_hw1

```
library(readxl)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
library(lubridate)
```

```
##
## Attaching package: 'lubridate'

## The following objects are masked from 'package:base':
##
##     date, intersect, setdiff, union
```

```
library(circular)
```

```
##
## Attaching package: 'circular'

## The following objects are masked from 'package:stats':
##
##     sd, var
```

```
# Import the data
data <- read_xlsx("ScreenTime_SPH.xlsx")
data
```

```
## # A tibble: 27 x 7
##    Date                Total.ST Total.ST.min Social.ST Social.ST.min Pickups
##    <dttm>              <chr>           <dbl> <chr>             <dbl>   <dbl>
##  1 2023-12-31 00:00:00 7h01m             421 2h12m               122     220
##  2 2024-01-01 00:00:00 4h11m             251 1h36m                96     215
##  3 2024-01-02 00:00:00 7h09m             429 1h39m                99     137
##  4 2024-01-03 00:00:00 7h51m             471 58m                  58     132
##  5 2024-01-04 00:00:00 4h23m             263 1h56m               116     277
##  6 2024-01-05 00:00:00 7h39m             459 1h25m                85     174
##  7 2024-01-06 00:00:00 4h45m             285 1h51m               111     169
##  8 2024-01-07 00:00:00 4h19m             259 2h47m               167     174
##  9 2024-01-08 00:00:00 4h40m             280 2h09m               129     174
## 10 2024-01-09 00:00:00 5h31m             331 1h22m                82     183
## # i 17 more rows
## # i 1 more variable: Pickup.1st <dttm>
```

```
# Convert the type of Date
data <- data %>%
  mutate(Date = as.Date(Date))

# Convert the type of Pickup.1st
data <- data %>%
  mutate(
    Pickup.1st = as.POSIXct(paste(as.Date(Date),
                                  format(strptime(Pickup.1st,
                                  format="%Y-%m-%d %H:%M:%S"), "%H:%M:%S")),
                             format="%Y-%m-%d %H:%M:%S", tz="America/New_York")
  )
```

PATT I: DATA COLLECTION AND DATA PROCESSING

Problem 1

a

The purpose of the data collection is to explore the association between screen time and sleep duration among children and adolescents.
Hypothesis: The increase in screen time will decrease the sleep duration among children and adolescents.
Cite the Reference:
Hale, L., & Guan, S. (2015). Screen time and sleep among school-aged children and adolescents: a systematic literature review. Sleep medicine reviews, 21, 50–58. https://doi.org/10.1016/j.smrv.2014.07.007

b

The role of Informed Consent Form is to make possible participants know the purpose of the planned study, understand how their data will be used, and obtain their consent to participate in the study, in order that researchers could collect their data and apply their data to the study appropriately.

c

Table 1: Data Collection Plan

Data Collection Time	Collected Variables and Types	Data Source	The number of data collected before the data freeze (2024-1-26)
From 2023-12-31 to 2024-1-26	Total Screen Time (Text), Social Screen Time (Social.ST-text), Total Pickups (Pickups-numeric), first pickup time (Pickup.1st-date)	From the SPH Students	27

d

```r
# Create and add two new variables: daily proportion of social screen time & "daily duration per use
data <- data %>%
  mutate(Daily_prop_social_ST = Social.ST.min/Total.ST.min,
         Daily_duration_per_use =  Total.ST.min/Pickups)
```

Problem 2

a. Make a time series plot of each of the five variables in your data.

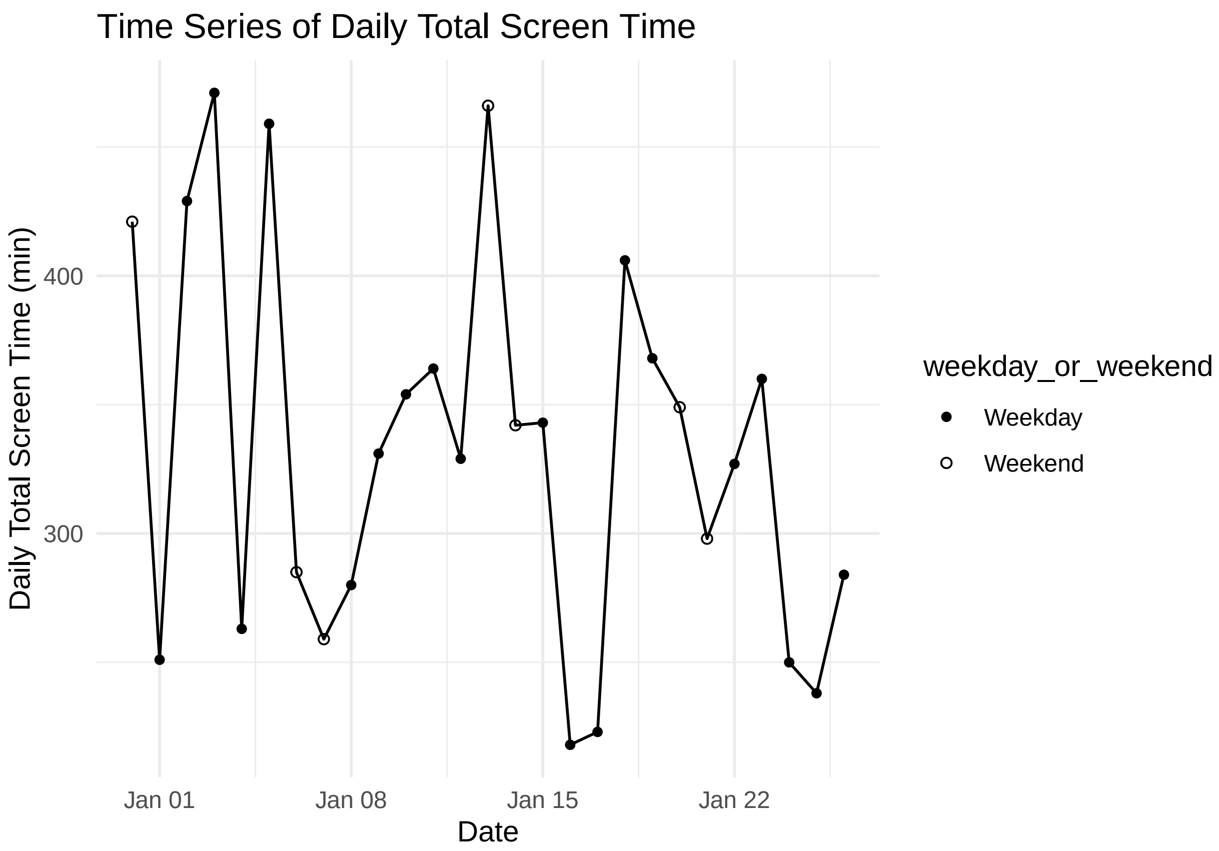
The general temporal patterns of these five figures are similar, since the lines are very convoluted. There are some extremely high or low points in each figure, which is not highly related to whether the day is the weekday or weekend.

```r
data <- data %>%
  mutate(
    weekday_or_weekend = if_else(weekdays(Date) %in% c('Saturday', 'Sunday'), 'Weekend', 'Weekday')
  )
```

Time Series Plot of Total Screen Time

Specifically, the Daily Total Screen Time on Saturday is higher than that on Sunday, indicating that the participant tends to decrease the daily total screen time near next weekday.

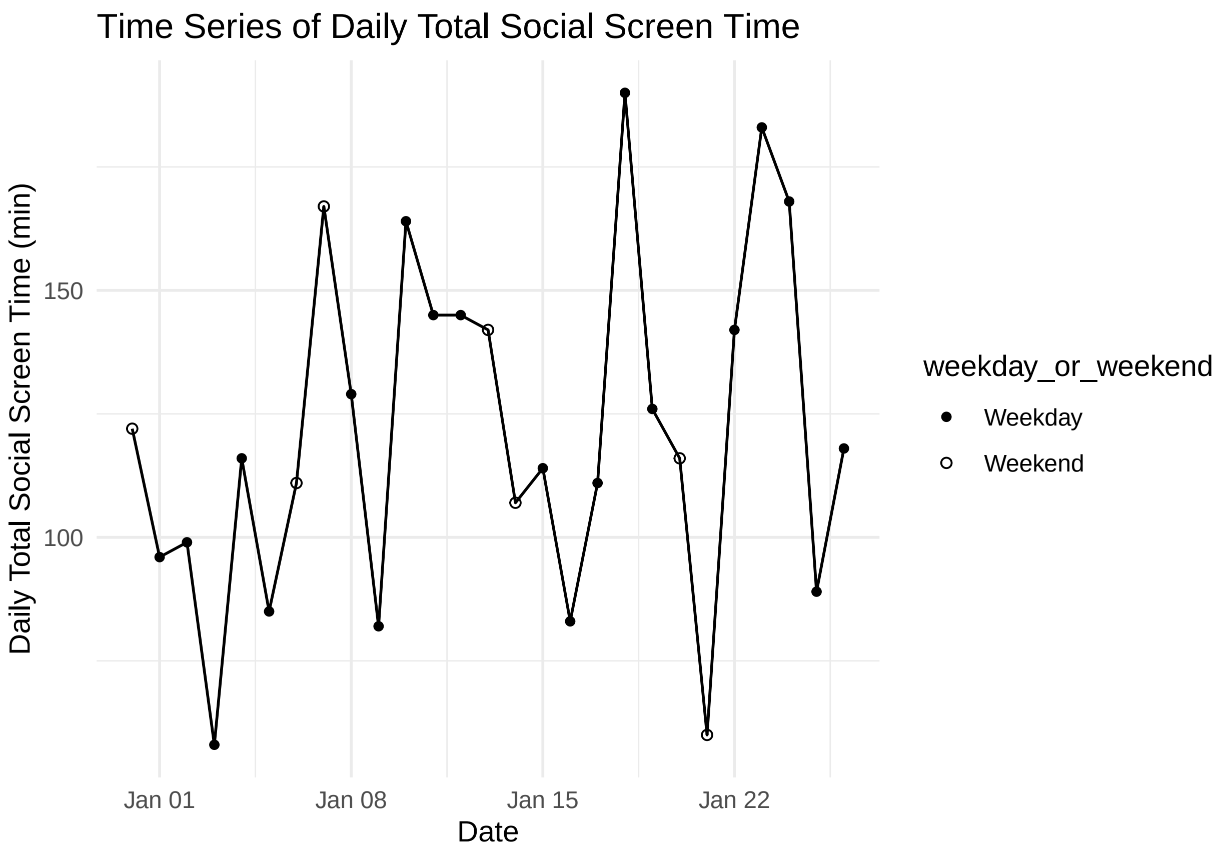
```r
ggplot(data, aes(x = Date, y = Total.ST.min)) +
  geom_line() +
  geom_point(aes(shape = weekday_or_weekend)) +
  scale_shape_manual(values = c("Weekday" = 16, "Weekend" = 1)) +
  theme_minimal() +
  labs(title = "Time Series of Daily Total Screen Time",
       x = "Date",
       y = "Daily Total Screen Time (min)")
```



Time Series Plot of Total Social Screen Time

In general, the Daily Number of Pickups on Saturday is higher than that on Sunday, indicating that the participant tends to decrease the daily number of pickups near next weekday, which is similar as the figure of Daily Total Screen Time. But there are exceptions, where the daily total social screen time on Sunday is higher than that on Saturday.

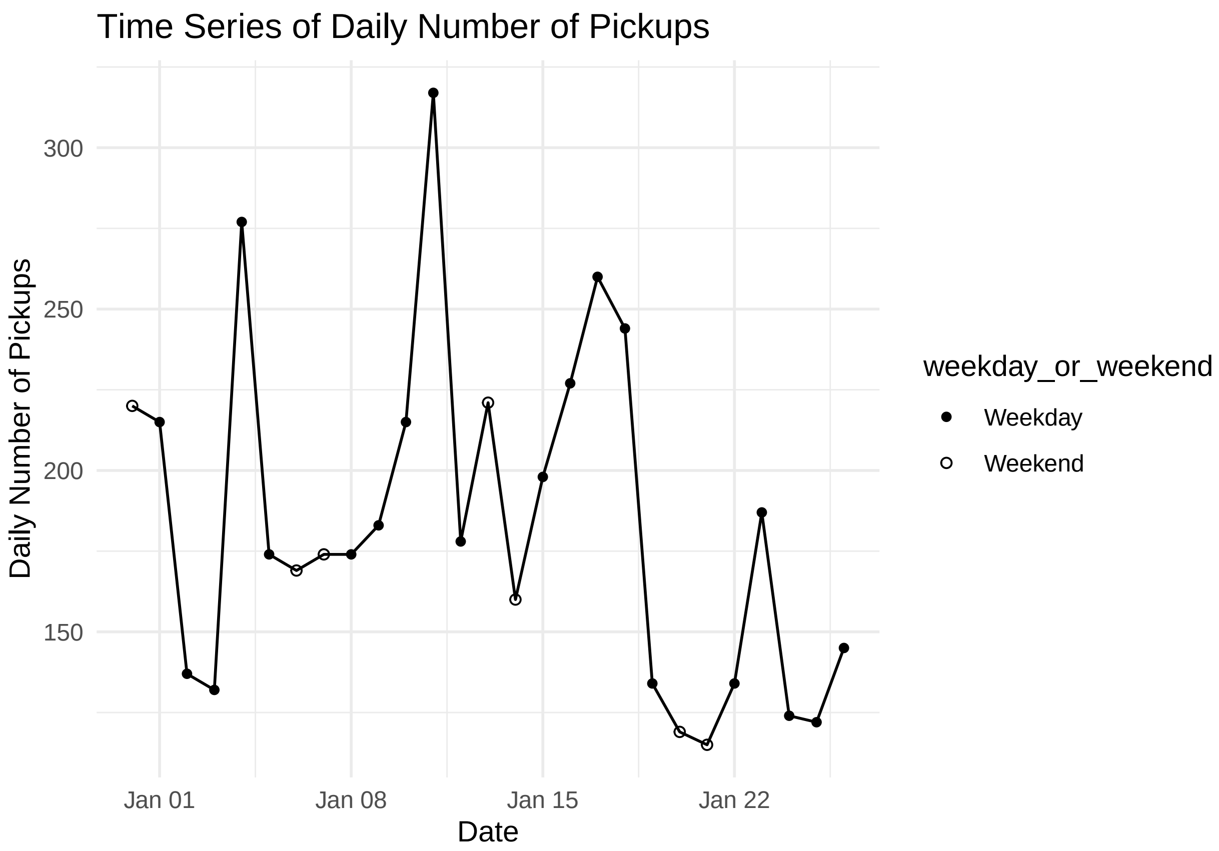
```r
ggplot(data, aes(x = Date, y = Social.ST.min)) +
  geom_line() +
  geom_point(aes(shape = weekday_or_weekend)) +
  scale_shape_manual(values = c("Weekday" = 16, "Weekend" = 1)) +
  theme_minimal() +
  labs(title = "Time Series of Daily Total Social Screen Time",
       x = "Date",
       y = "Daily Total Social Screen Time (min)")
```



Time Series Plot of Pickups

In general, the Daily Total Social Screen Time on Saturday is higher than that on Sunday, indicating that the participant tends to decrease the daily total social screen time near next weekday, which is similar as the figure of Daily Total Screen Time. But there are exceptions, where the daily total social screen time on Sunday is a little higher than that on Saturday.

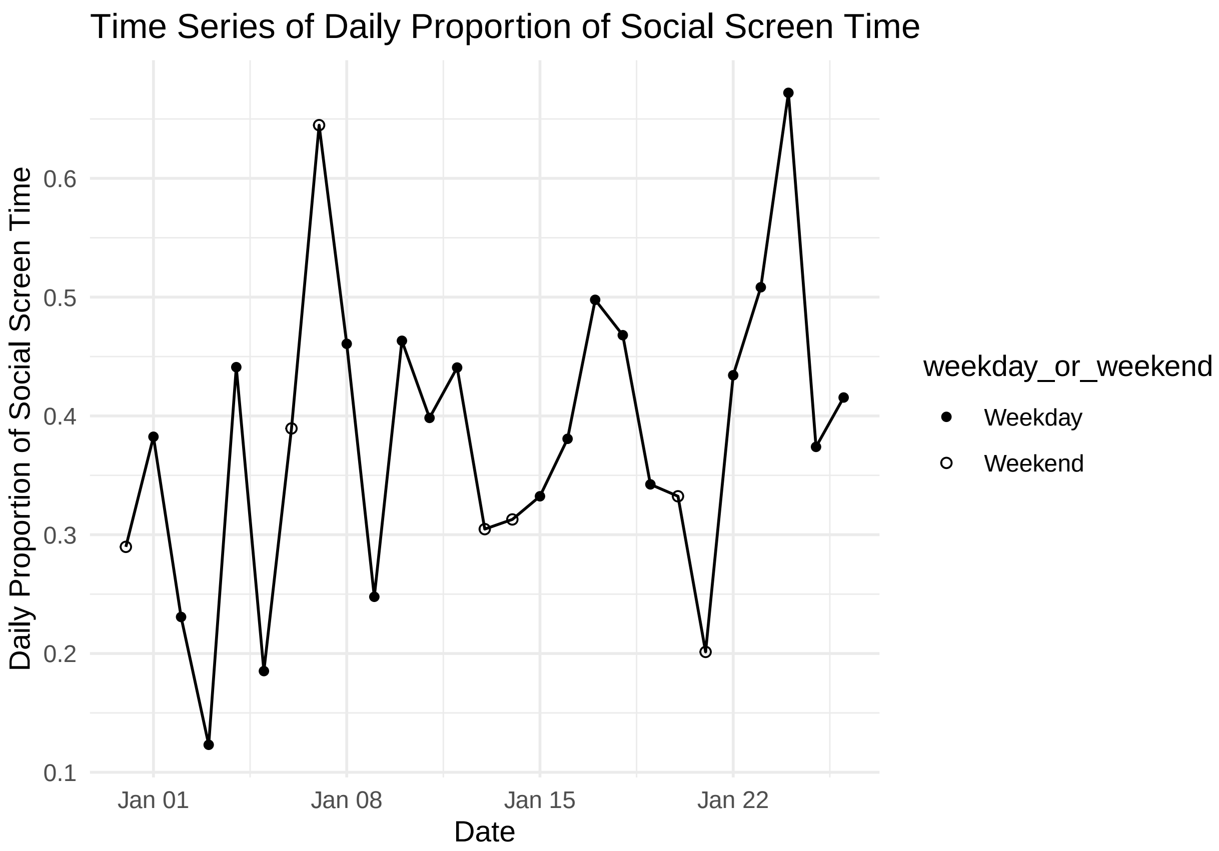
```
ggplot(data, aes(x = Date, y = Pickups)) +
  geom_line() +
  geom_point(aes(shape = weekday_or_weekend)) +
  scale_shape_manual(values = c("Weekday" = 16, "Weekend" = 1)) +
  theme_minimal() +
  labs(title = "Time Series of Daily Number of Pickups",
       x = "Date",
       y = "Daily Number of Pickups")
```



Time Series Plot of Daily Proportion of Social Screen Time

Conversely, the Daily Proportion of Social Screen Time on Sunday is higher than that on Saturday, but there are exceptions where the Daily Proportion of Social Screen Time on Saturday is obviously higher.

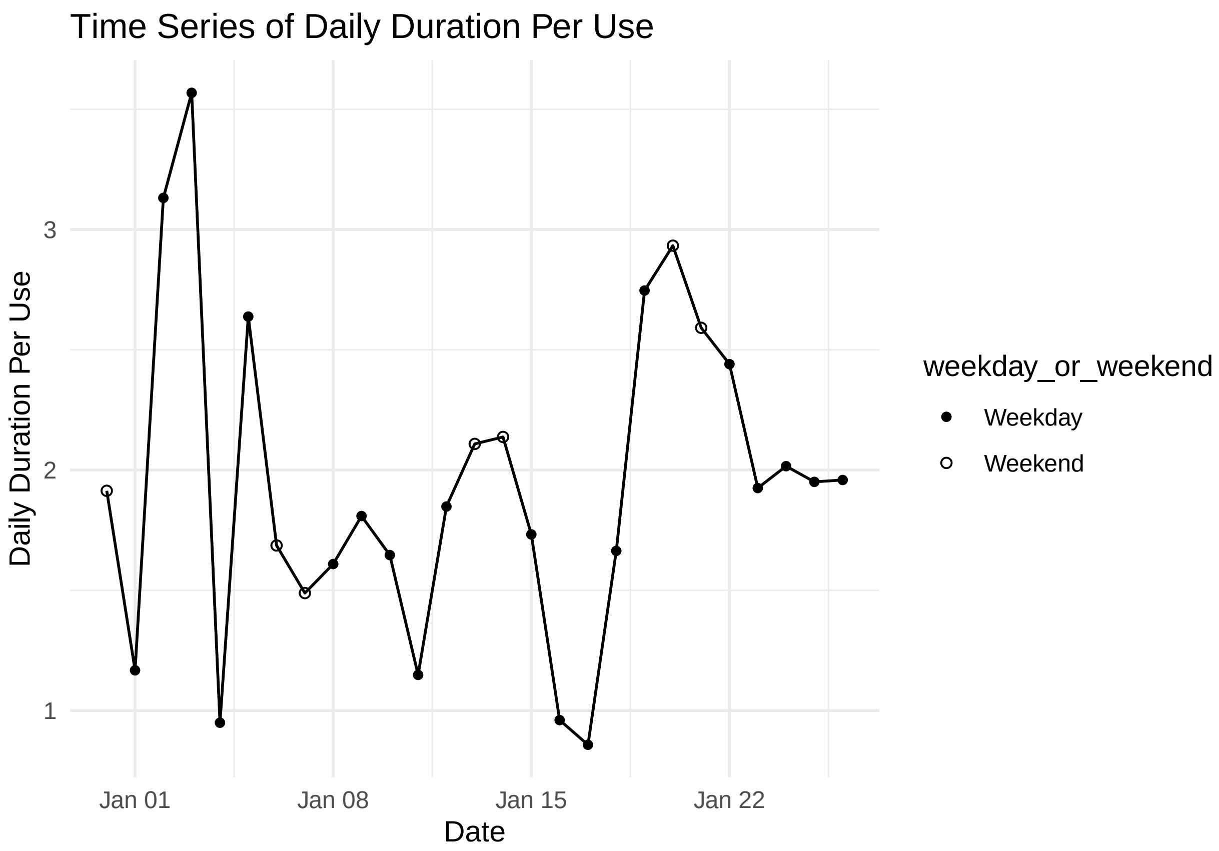
```r
ggplot(data, aes(x = Date, y = Daily_prop_social_ST)) +
  geom_line() +
  geom_point(aes(shape = weekday_or_weekend)) +
  scale_shape_manual(values = c("Weekday" = 16, "Weekend" = 1)) +
  theme_minimal() +
  labs(title = "Time Series of Daily Proportion of Social Screen Time",
       x = "Date",
       y = "Daily Proportion of Social Screen Time")
```



Time Series Plot of Daily Duration Per Use

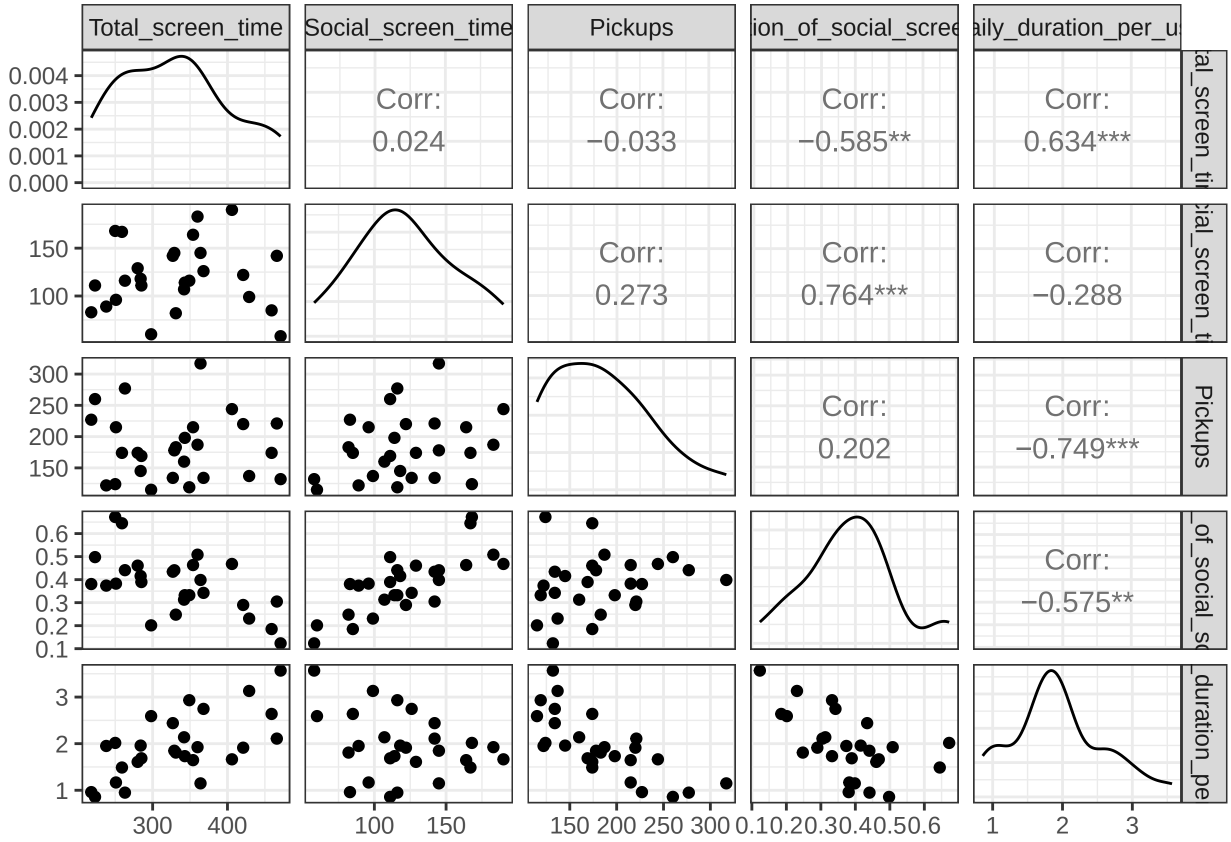
In general, the Daily Duration Per Use on Saturday is higher than that on Sunday, indicating that the participant tends to decrease the daily duration per use near next weekday, which is similar as the figure of Daily Total Screen Time. But there are exceptions, where the daily duration per use on Sunday is a little higher than that on Saturday.

```
ggplot(data, aes(x = Date, y = Daily_duration_per_use)) +
  geom_line() +
  geom_point(aes(shape = weekday_or_weekend)) +
  scale_shape_manual(values = c("Weekday" = 16, "Weekend" = 1)) +
  theme_minimal() +
  labs(title = "Time Series of Daily Duration Per Use",
       x = "Date",
       y = "Daily Duration Per Use")
```



b. Make pairwise scatterplots of five variables

```
ggpairs(data, columns = c("Total.ST.min", "Social.ST.min",
                          "Pickups", "Daily_prop_social_ST",
                          "Daily_duration_per_use"),
        columnLabels = c("Total_screen_time", "Social_screen_time",
                         "Pickups", "Proportion_of_social_screen_time",
                         "Daily_duration_per_use")) +
          theme_bw()
```



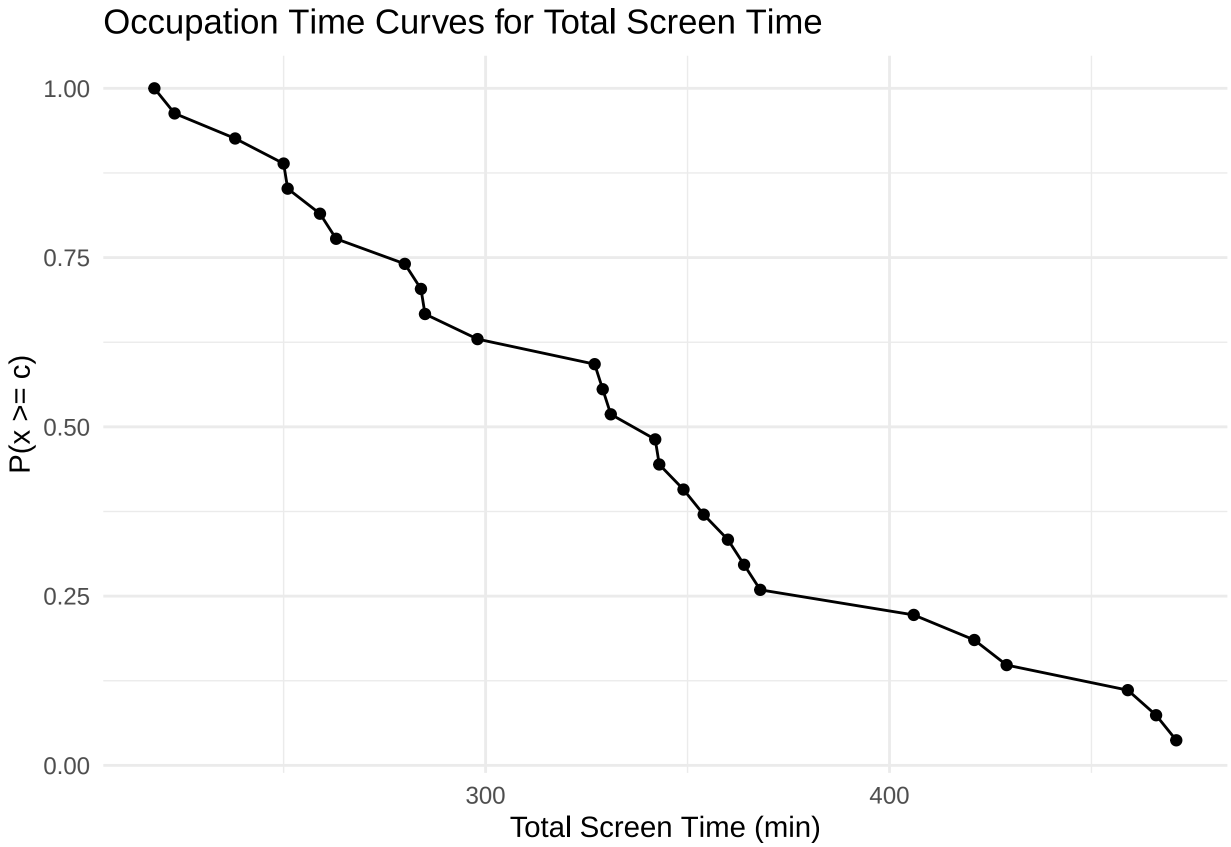
In general, the strongest positive correlation is between Daily Proportion of Social Screen Time and Daily Total Social Screen Time, where the Pearson correlation is 0.764. A significant positive correlation is also observed between Total Screen Time and Daily Duration Per Use (Pearson correlation is 0.634), indicating longer usage periods on days with more screen time.
The strongest negative correlation is between Social Screen Time and Pickups, where the Pearson correlation is -0.749. There is a moderate negative correlation between Total Screen Time and the Proportion of Social Screen Time, indicating a lower proportion of social screen time on days with higher overall screen time.

c. Make an occupation time curve for each of the five time series.

Occupation Time Curves for Total Screen Time

The probability of total screen time exceeding the thresholds gradually decreases as the threshold increases. Specifically, there is a sharp decline around the 350-minute, indicating that the situation of total screen time exceeding this duration are significantly less common.

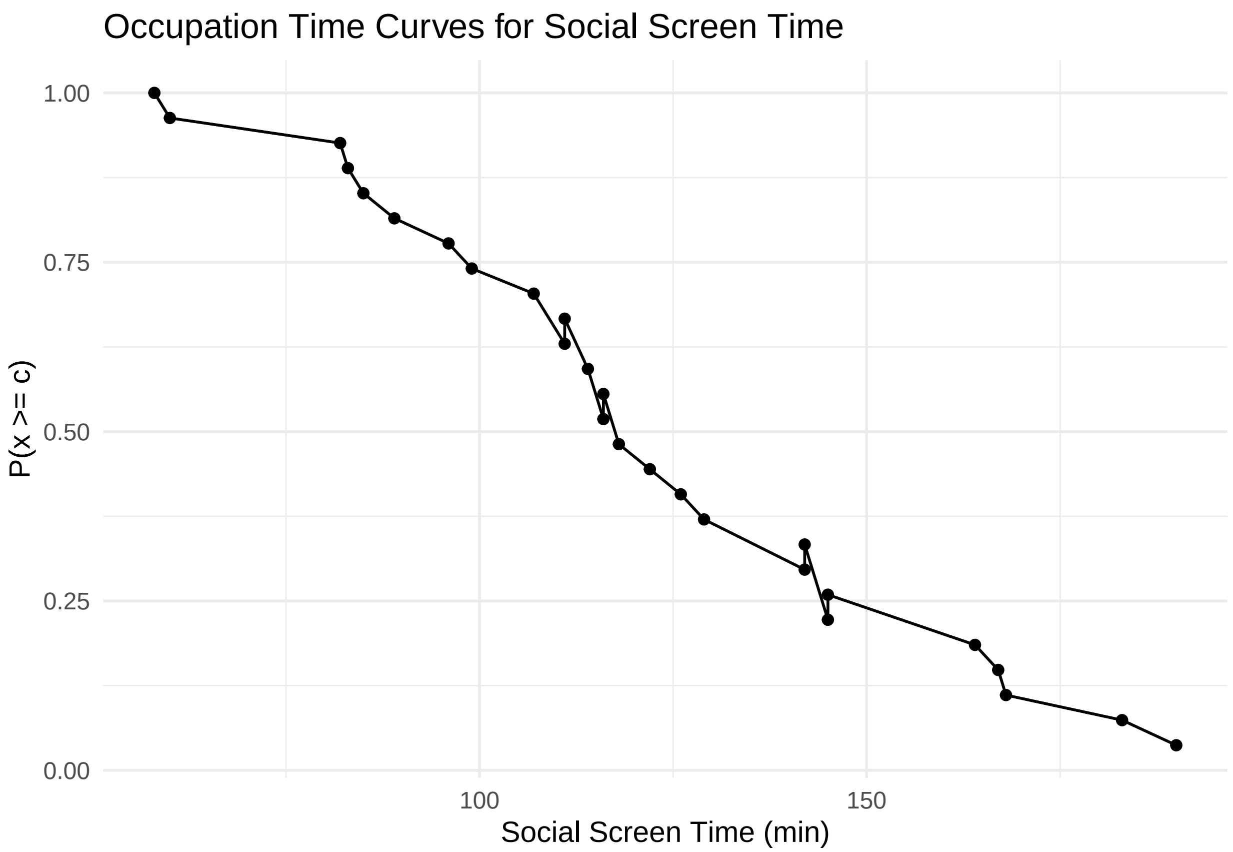
```
data1 <- arrange(data, desc(Total.ST.min))
data1$Total.ST_prob <- seq_along(data1$Total.ST.min) / nrow(data1)

ggplot(data1, aes(x = Total.ST.min, y = Total.ST_prob)) +
  geom_line() +
  geom_point() +
  labs(x = "Total Screen Time (min)",
       y = "P(x >= c)",
       title = "Occupation Time Curves for Total Screen Time") +
  theme_minimal()
```



Occupation Time Curves for Total Social Screen Time

The probability of total social screen time exceeding the thresholds generally decreases as the amount of total social screen time increases, indicating a higher probability of shorter social screen time periods. However, there are several points where the curve becomes steady, indicating the situation that social screen time duration is consistent before dropping to lower probabilities at higher thresholds.

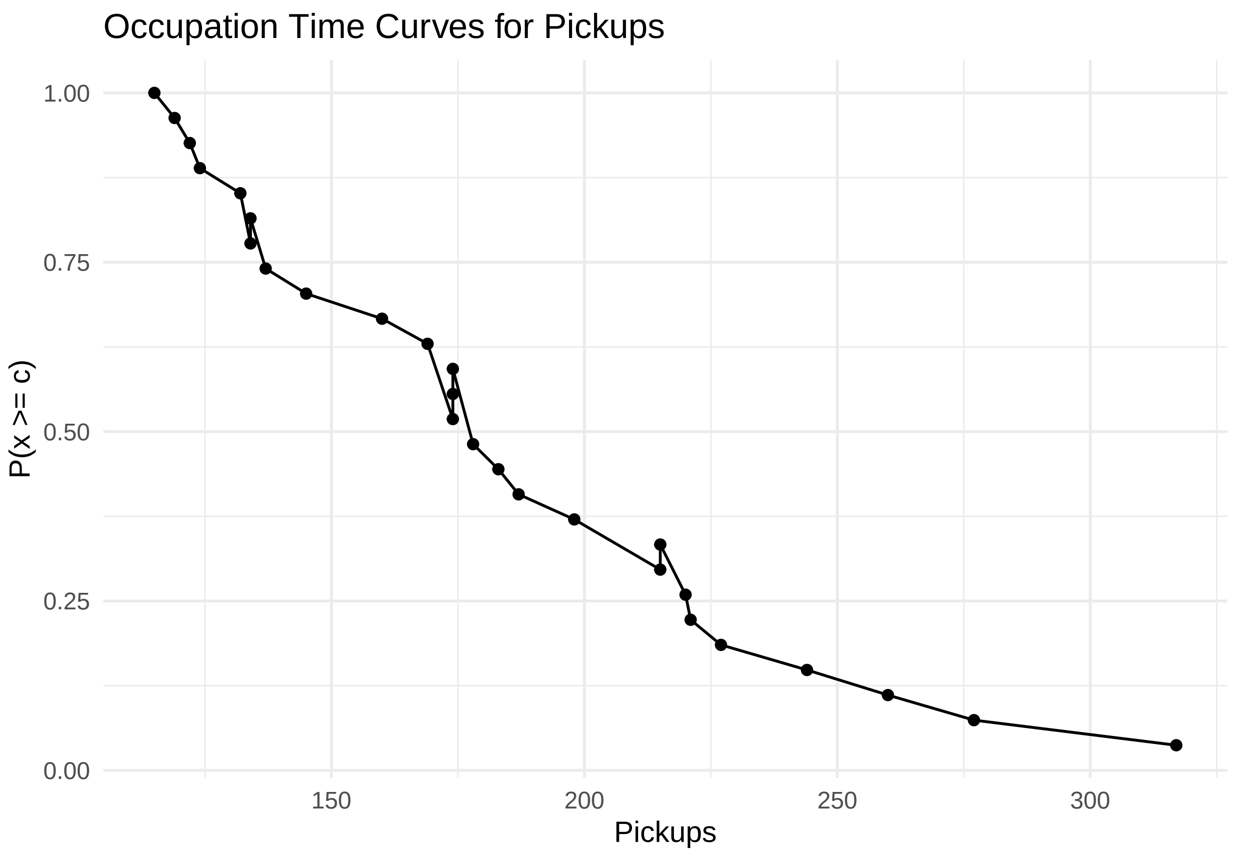
```
data2 <- arrange(data, desc(Social.ST.min))
data2$Social.ST_prob <- seq_along(data2$Social.ST.min) / nrow(data2)

ggplot(data2, aes(x = Social.ST.min, y = Social.ST_prob)) +
  geom_line() +
  geom_point() +
  labs(x = "Social Screen Time (min)",
       y = "P(x >= c)",
       title = "Occupation Time Curves for Social Screen Time") +
  theme_minimal()
```



Occupation Time Curves for Pickups

The probability of total pickups exceeding the thresholds decreases as the amount of total pickups increases, indicating that a higher probability of less pickups. There are points on the curve where the decrease in probability becomes more obvious(at about 175 and 220), indicating too many pickups are relatively uncommon.

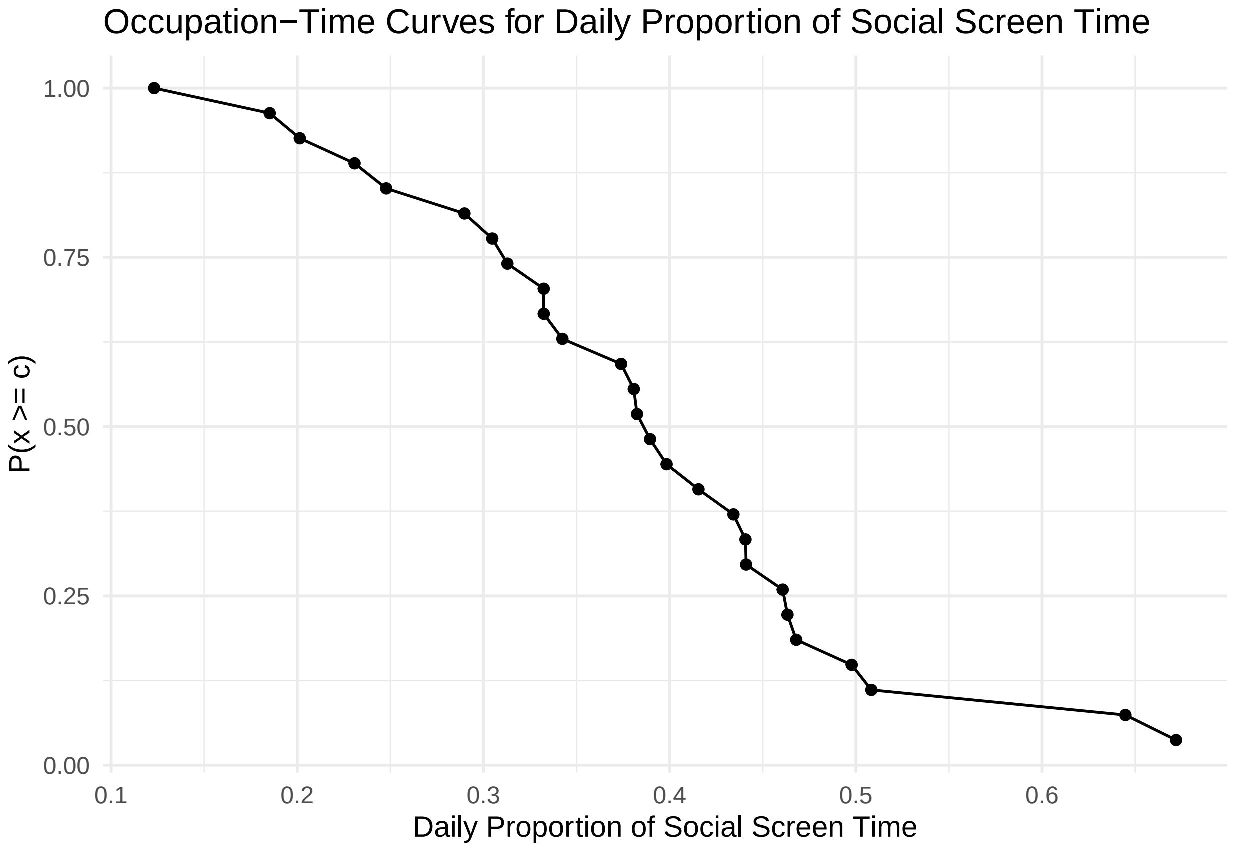
```r
data3 <- arrange(data, desc(Pickups))
data3$Pickups_prob <- seq_along(data3$Pickups) / nrow(data3)

ggplot(data3, aes(x = Pickups, y = Pickups_prob)) +
  geom_line() +
  geom_point() +
  labs(x = "Pickups",
       y = "P(x >= c)",
       title = "Occupation Time Curves for Pickups") +
  theme_minimal()
```



Occupation-Time Curves for Daily Proportion of Social Screen Time

The probability of the daily proportion of social screen time decreases as the proportion increases, which shows that lower proportions are more common than higher ones. There is a sharp decline from about 0.4 to 0.5, indicating that higher proportions of social screen time become increasingly rare when the proportion exceeds the thresholds.

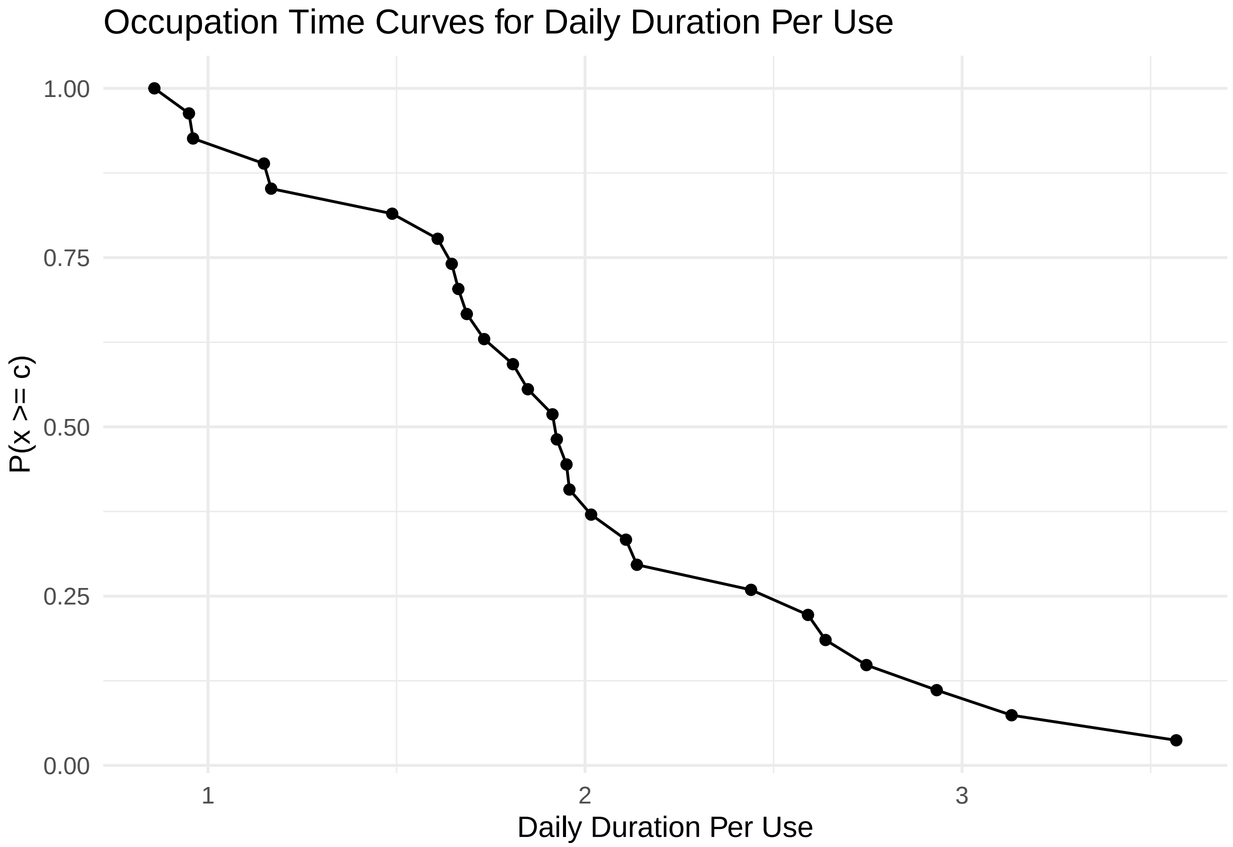
```
data4 <- arrange(data, desc(Daily_prop_social_ST))
data4$Daily_prop_social_ST_prob <- seq_along(data4$Daily_prop_social_ST) / nrow(data4)

ggplot(data4, aes(x = Daily_prop_social_ST, y = Daily_prop_social_ST_prob)) +
  geom_line() +
  geom_point() +
  labs(x = "Daily Proportion of Social Screen Time",
       y = "P(x >= c)",
       title = "Occupation-Time Curves for Daily Proportion of Social Screen Time") +
  theme_minimal()
```



Occupation Time Curves for Daily Duration Per Use

The probability of the daily duration per use decreases as the proportion increases, which shows that lower duration per use are more common than higher ones. There is a sharp decline from about 1.6 to 2, indicating that higher duration per use become increasingly rare when the duration per use exceeds the thresholds.

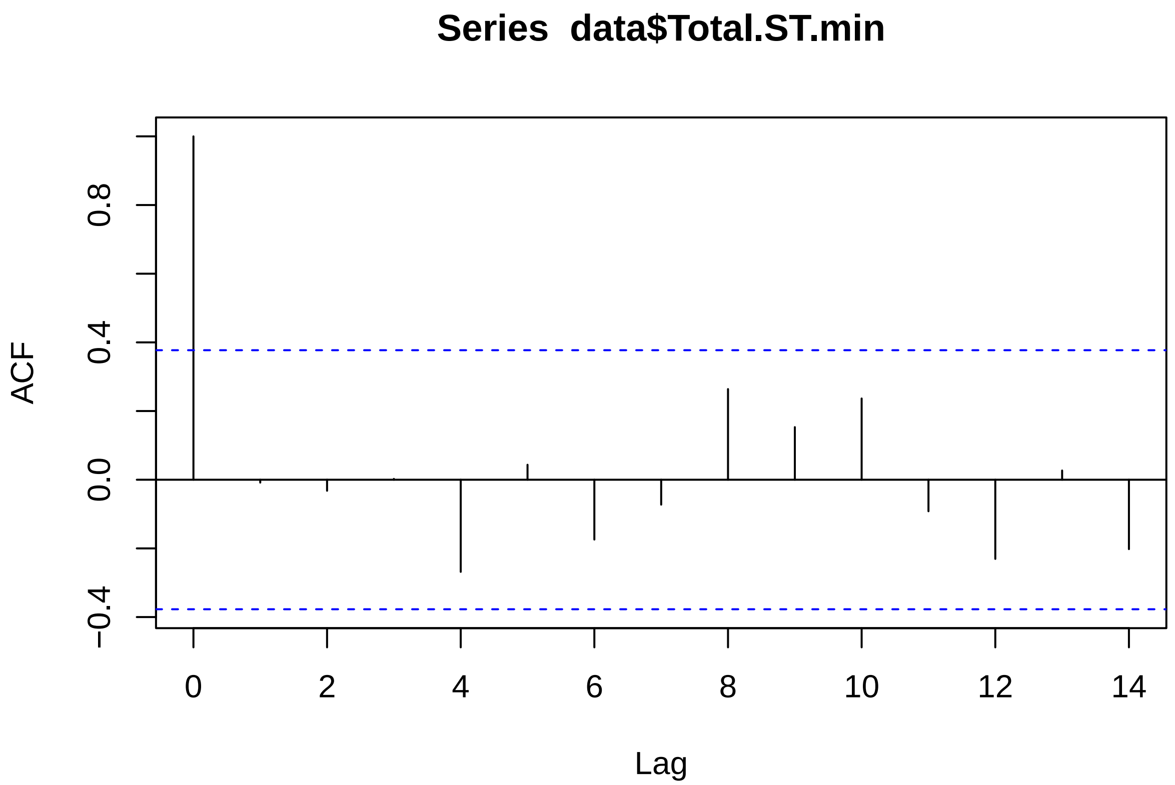
```
data5 <- arrange(data, desc(Daily_duration_per_use))
data5$Daily_duration_per_use_prob <- seq_along(data5$Daily_duration_per_use) / nrow(data5)

ggplot(data5, aes(x = Daily_duration_per_use, y = Daily_duration_per_use_prob)) +
  geom_line() +
  geom_point() +
  labs(x = "Daily Duration Per Use",
       y = "P(x >= c)",
       title = "Occupation Time Curves for Daily Duration Per Use") +
  theme_minimal()
```



d. Display the serial dependence for each of the five time series

There are no significant autocorrelations, since for five figures of series, all the bars at different lags are within the bounds of blue dashed lines.

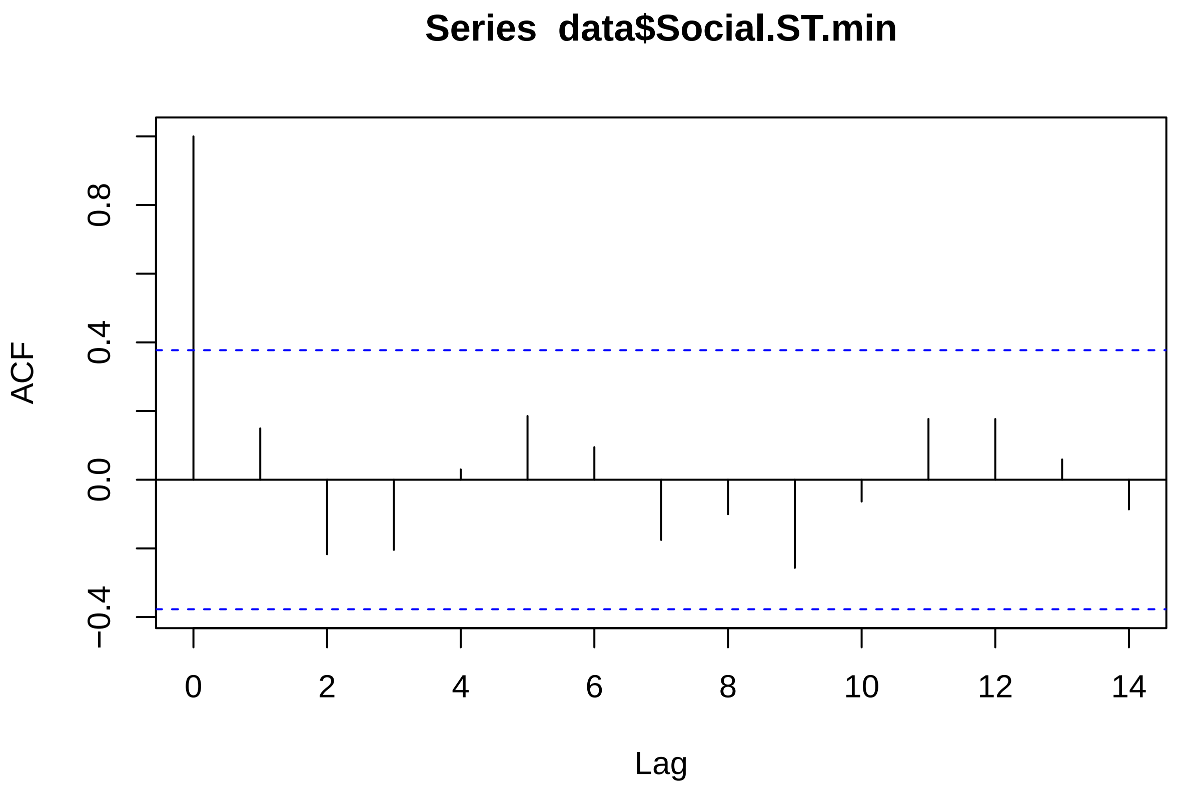
```
acf(data$Total.ST.min)
```



```
acf(data$Total.ST.min, plot = FALSE)
```

```
##
## Autocorrelations of series 'data$Total.ST.min', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000 -0.008 -0.032  0.002 -0.268  0.044 -0.174 -0.073  0.264  0.153  0.237
##     11     12     13     14
## -0.092 -0.231  0.027 -0.202
```

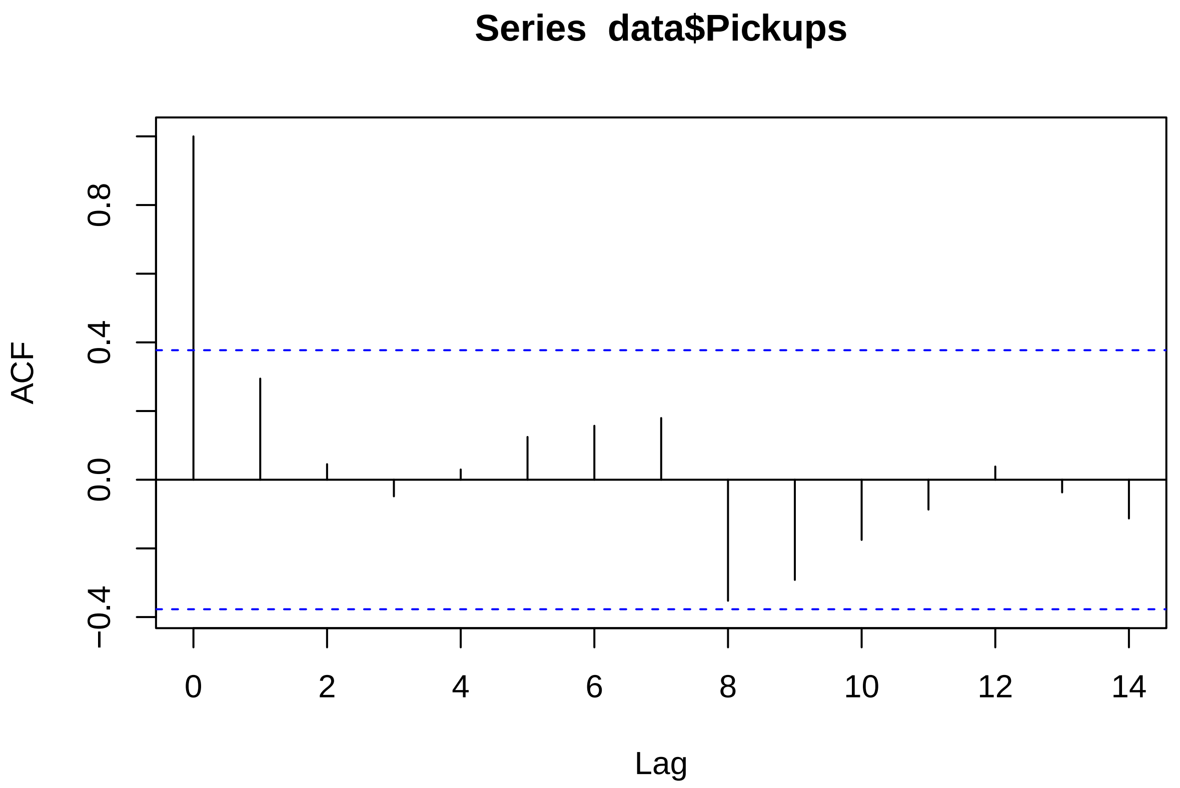
```
acf(data$Social.ST.min)
```



```
acf(data$Social.ST.min, plot = FALSE)
```

```
##
## Autocorrelations of series 'data$Social.ST.min', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.150 -0.217 -0.204  0.030  0.186  0.095 -0.175 -0.101 -0.257 -0.064
##     11     12     13     14
##  0.177  0.177  0.059 -0.086
```

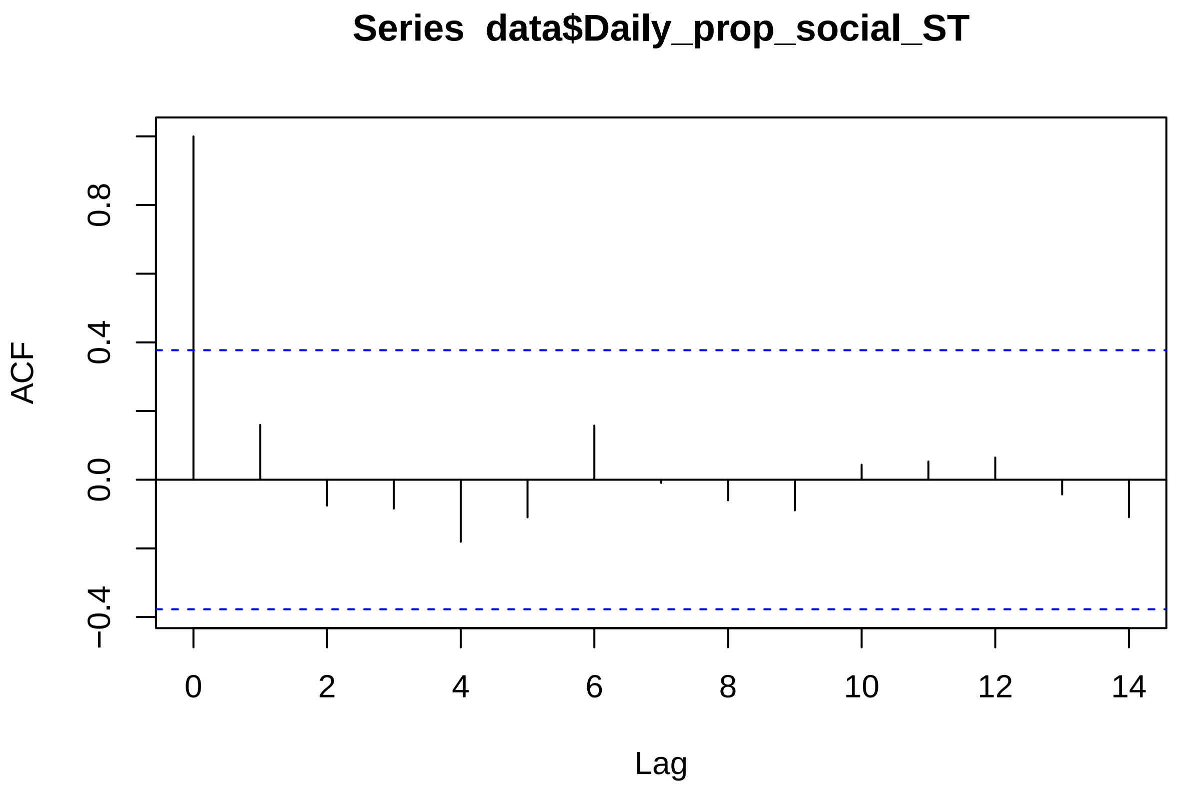
```
acf(data$Pickups)
```



```
acf(data$Pickups, plot = FALSE)
```

```
##
## Autocorrelations of series 'data$Pickups', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.295  0.045 -0.048  0.030  0.125  0.157  0.180 -0.353 -0.292 -0.175
##     11     12     13     14
## -0.087  0.038 -0.037 -0.113
```

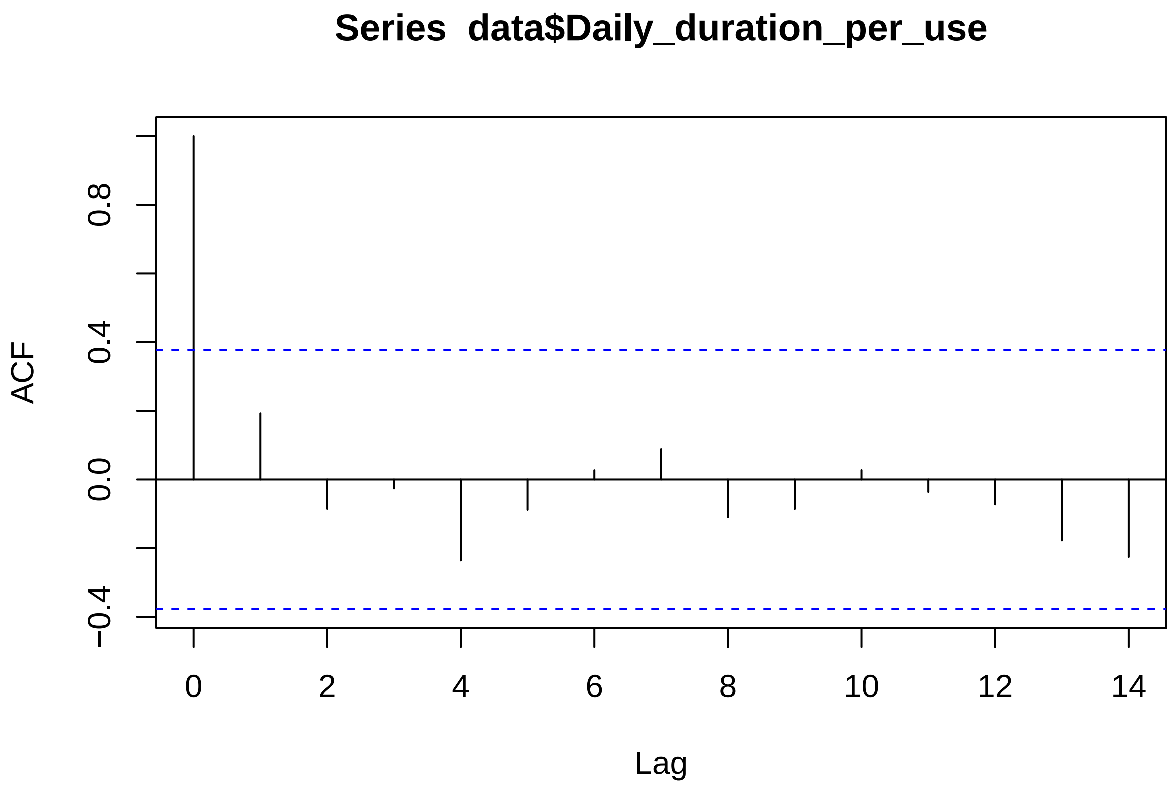
```
acf(data$Daily_prop_social_ST)
```



```
acf(data$Daily_prop_social_ST, plot = FALSE)
```

```
##
## Autocorrelations of series 'data$Daily_prop_social_ST', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.160 -0.076 -0.084 -0.181 -0.110  0.158 -0.009 -0.060 -0.089  0.044
##     11     12     13     14
##  0.053  0.065 -0.043 -0.109
```

```
acf(data$Daily_duration_per_use)
```



```
acf(data$Daily_duration_per_use, plot = FALSE)
```

```
##
## Autocorrelations of series 'data$Daily_duration_per_use', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.193 -0.085 -0.026 -0.236 -0.088  0.027  0.088 -0.110 -0.086  0.027
##     11     12     13     14
## -0.036 -0.073 -0.178 -0.225
```

Problem 3

a. Transform (or covert) the time of first pickup to an angle ranged from 0 to 360 degree

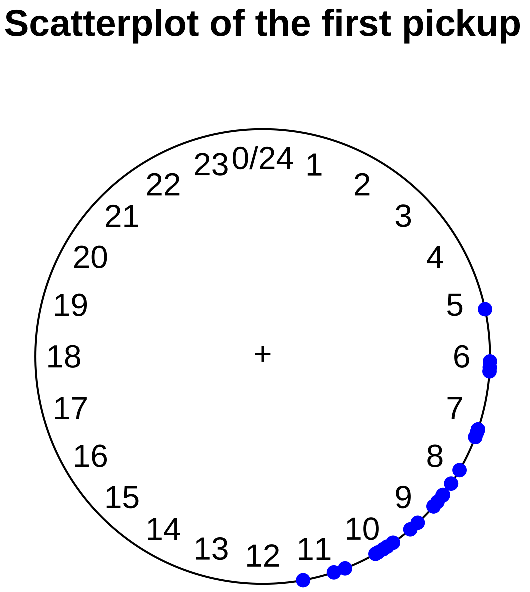
```
data <- data %>%
  mutate(Pickup.1st.angular = (hour(Pickup.1st) * 60 + minute(Pickup.1st)) / (24 * 60) * 360)
head(data)
```

```
## # A tibble: 6 x 11
##   Date       Total.ST Total.ST.min Social.ST Social.ST.min Pickups
##   <date>     <chr>           <dbl> <chr>             <dbl>   <dbl>
## 1 2023-12-31 7h01m             421 2h12m               122     220
## 2 2024-01-01 4h11m             251 1h36m                96     215
## 3 2024-01-02 7h09m             429 1h39m                99     137
## 4 2024-01-03 7h51m             471 58m                  58     132
## 5 2024-01-04 4h23m              263 1h56m                116     277
## 6 2024-01-05 7h39m              459 1h25m                 85     174
## # i 5 more variables: Pickup.1st <dttm>, Daily_prop_social_ST <dbl>,
## #   Daily_duration_per_use <dbl>, weekday_or_weekend <chr>,
## #   Pickup.1st.angular <dbl>
```

b. Make a scatterplot of the first pickup data on a 24-hour clock circle.

The scatterplot shows that the first pickup time concentrate on around 6 AM to 9 AM, which is within this early morning period. The absence of points in the late night to early morning period indicates that pickups during these times are rare, suggesting a typical pattern of the participant firstly picking up the phone early and not using the phone overnight.

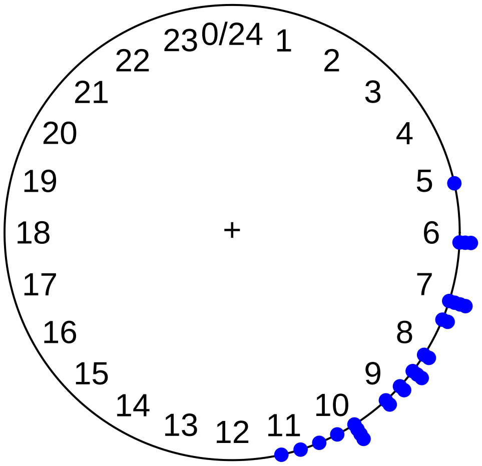
```
first.pickup.cir = circular(data$Pickup.1st.angular, units = "degrees", template = "clock24")
plot(first.pickup.cir, col="blue", main="Scatterplot of the first pickup")
```



c. Make a histogram plot on the circle in that you may choose a suitable bin size to create stacking

During a day, there are 24 hours * 60 minutes = 1440 minutes in a day. Since I would like to set each bin to represent 20 minutes, the bins equals to 1440/20=72.

```
plot(first.pickup.cir, stack = TRUE, bins = 72, col = "blue")
```



Problem 4

a. Explain why the factor St is needed in the Poisson distribution above

Yt represents the daily number of pickups at day t and St represents the daily total screen time at day t. Since the daily total screen would influence the daily number of pickups at day t, St is needed to scale the rate parameter Lambda to reflect the difference of daily total screen time during days.

b. Use the R function glm to estimate the rate parameter lambda in which ln(St) is included in the model as an offset

```r
data <- data %>%
  mutate(S_t = Total.ST.min / 60)

model <- glm(Pickups ~ offset(log(S_t)), family = "poisson", data = data)
summary(model)
```

```
##
## Call:
## glm(formula = Pickups ~ offset(log(S_t)), family = "poisson",
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -8.7905  -2.9855  -0.8345   1.4431  10.7120
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.50108    0.01421   246.4   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 648.97  on 26  degrees of freedom
## Residual deviance: 648.97  on 26  degrees of freedom
## AIC: 840.33
##
## Number of Fisher Scoring iterations: 4
```

c

```
data$Xt <- ifelse(weekdays(as.Date(data$Date)) %in% c("Saturday", "Sunday"), 0, 1)
data$Zt <- ifelse(as.Date(data$Date) >= as.Date("2024-01-10"), 1, 0)

model1 <- glm(Pickups ~ Xt + Zt + offset(log(S_t)), family = poisson(link = "log"), data = data)
summary(model1)
```

```
##
## Call:
## glm(formula = Pickups ~ Xt + Zt + offset(log(S_t)), family = poisson(link = "log"),
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -9.0471  -2.3405  -0.6308   2.0903   9.9610
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  3.34968    0.03421  97.902  < 2e-16 ***
## Xt           0.16893    0.03338   5.061 4.16e-07 ***
## Zt           0.04078    0.02936   1.389    0.165
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for poisson family taken to be 1)
##
##     Null deviance: 648.97  on 26  degrees of freedom
## Residual deviance: 620.40  on 24  degrees of freedom
## AIC: 815.77
##
## Number of Fisher Scoring iterations: 4
```

c.1

There is data evidence for significantly different behavior of daily pickups between weekdays and weekends, since the p-value 4.16e-07 is smaller than 0.05.

c.2

There is not any data evidence for a significant change on the behavior of daily pickups after the winter semester began, since the p-value 0.165 is larger than 0.05.

Problem 5

a

```r
# Convert angle to radian
data$Pickup.1st.angular.radians <- data$Pickup.1st.angular * (pi / 180)
fit <- mle.vonmises(data$Pickup.1st.angular.radians)

mu <- fit$mu
kappa <- fit$kappa
print(mu)
```

```
## Circular Data:
## Type = angles
## Units = radians
## Template = none
## Modulo = asis
## Zero = 0
## Rotation = counter
## [1] 2.206185
```

```r
print(kappa)
```

```
## [1] 6.634791
```

b

```r
time_8_30_am <- (8.5 / 24) * 2 * pi
prob_after_8_30_am <- 1 - pvonmises(time_8_30_am, mu, kappa)
print(prob_after_8_30_am)
```

```
## [1] 0.4807702
```