1. (a)

The data collection serves to examine the hypothesis that increased screen time, particularly on social media, is correlated with higher levels of stress and reduced well-being. This hypothesis stems from research suggesting that excessive engagement with social media can lead to increased feelings of anxiety (Lin et.al, 2019). The gathered data will allow us to analyze the relationship between daily screen time dedicated to social media and its potential psychological impacts, which could be further explored in the initial group project.

Lin, W.S., Chen, H.R., Lee, T.S.H. and Feng, J.Y., 2019. Role of social anxiety on high engagement and addictive behavior in the context of social networking sites. Data Technologies and Applications, 53(2), pp.156-170.

(b)

The Informed Consent Form is a fundamental document in research that ensures participants are fully informed about the study's aims, procedures, risks, and benefits, and that their participation is voluntary. It attempts to safeguard participants' rights and privacy while clearly outlining how their data will be used, stored, and safeguarded. This process also upholds the ethical principle of respecting the autonomy and welfare of research subjects.

(c)

Timing of Data Collection: Data is collected daily from January 6th, 2024, to January 26th, 2024. Variables Collected: The variables include total screen time (Total.ST), total screen time in minutes (Total.ST.min), social media screen time (Social.ST), social media screen time in minutes (Social.ST.min), number of times the phone is picked up (Pickups), and the time of the first pickup of the day (Pickup.1st). Data Collection Source: Data is collected from the participants' smartphones using built-in tracking software or applications that log screen time and usage statistics. Volume of Data: Data points are collected for each variable every day over the 21-day period, resulting in 21 data points per variable before the data freeze.

The data freeze date is set for January 26th, 2024, after which no further data for this period will be collected or altered, ensuring the integrity of the data set for analysis. This approach allows for a comprehensive view of the participants' screen time and interactions with their smartphones over a continuous period.

```r
#prepare packages
library(ggplot2)
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
```

```r
library(GGally)
```

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

```r
library(forecast)
```

```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```r
library(circular)
```

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

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

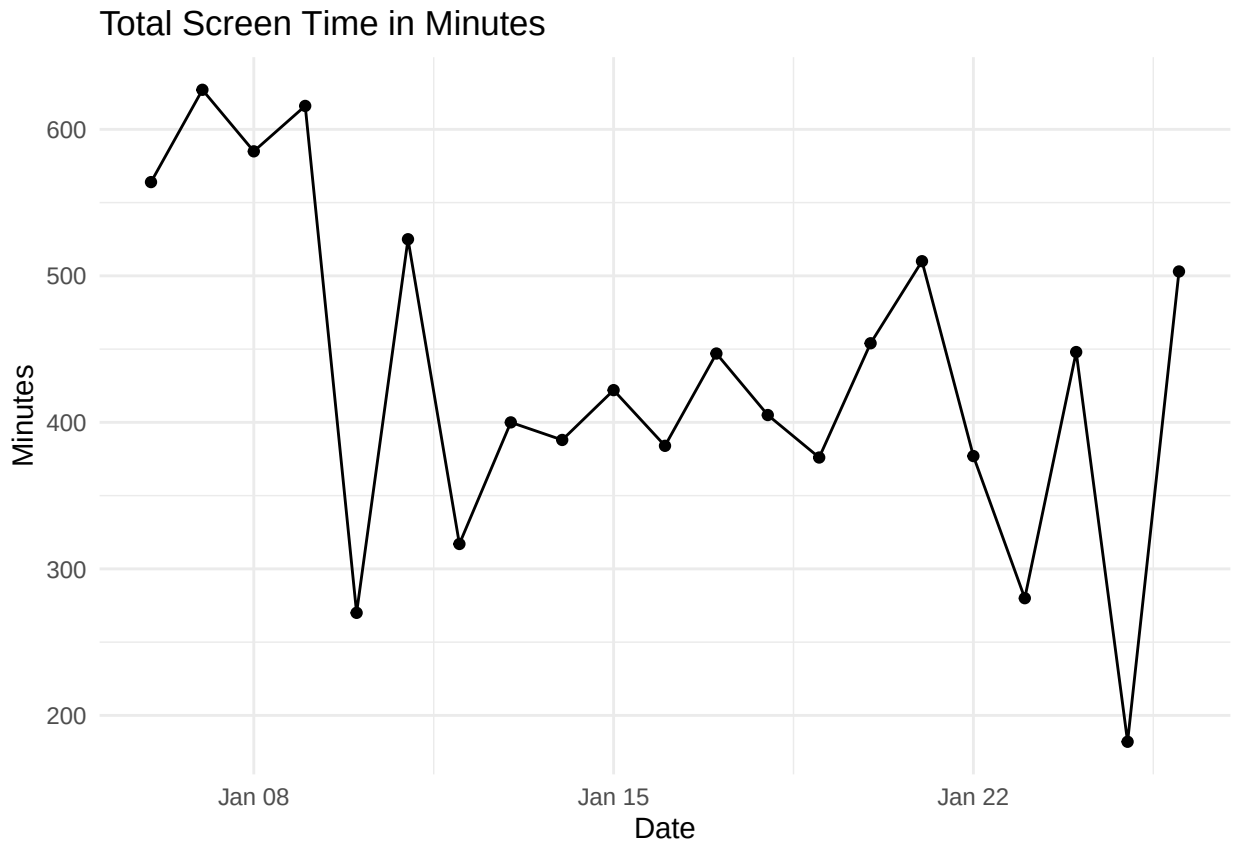
```r
file_path <- "~/Desktop/hw1620.xlsx"
screen_data <- read_excel(file_path)
head(screen_data)
```

```
## # A tibble: 6 x 9
##   Date      Total.ST Total.ST.min Social.ST Social.ST.min Pickups
##   <chr>     <chr>           <dbl> <chr>             <dbl>   <dbl>
## 1 1/6/2024  9h24m             564 5h25m               325      62
## 2 1/7/2024  10h27m            627 6h9m                369      50
## 3 1/8/2024  9h45m             585 5h59m               359      44
## 4 1/9/2024  10h16m            616 5h6m                306      70
## 5 1/10/2024 4h30m             270 2h2m                122     130
## 6 1/11/2024 6h45m             525 4h5m                245      89
## # i 3 more variables: Pickup.1st <dttm>,
## #   'Daily proportion of social screen time' <dbl>,
## #   'daily duration per use' <dbl>
```

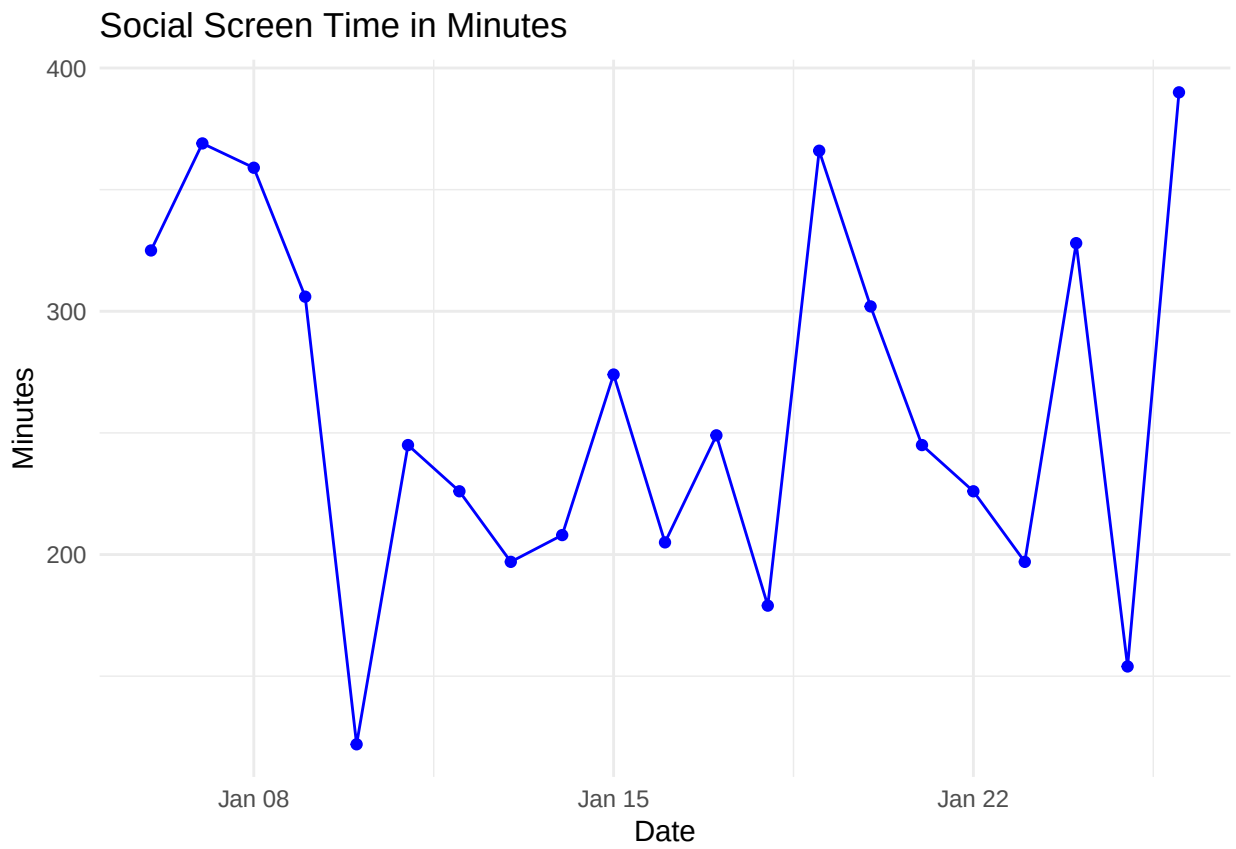
2. (a)
3. "Total Screen Time in Minutes": The plot shows fluctuations in total screen time with a recurring pattern of peaks and troughs, suggesting variability in daily screen use with a noticeable drop on January 10th and the end of the period.
4. "Social Screen Time in Minutes": Social screen time displays a high degree of variability with sharp increases and decreases, indicating inconsistent daily use patterns.
5. "Number of Pickups": The number of pickups fluctuates significantly over time, with some peaks far exceeding the general trend, which may suggest occasional days of very high frequency of phone usage.
6. "Daily Proportion of Social Screen Time": This time series shows a volatile pattern with several spikes on around January 10th and 19th, indicating days where the proportion of social screen time compared to total screen time was exceptionally high.

7. “Daily Duration Per Use”: The plot indicates a generally consistent range of duration per use despite a huge drop on around January 10th, reflecting a somewhat stable average use time with infrequent longer usage periods.

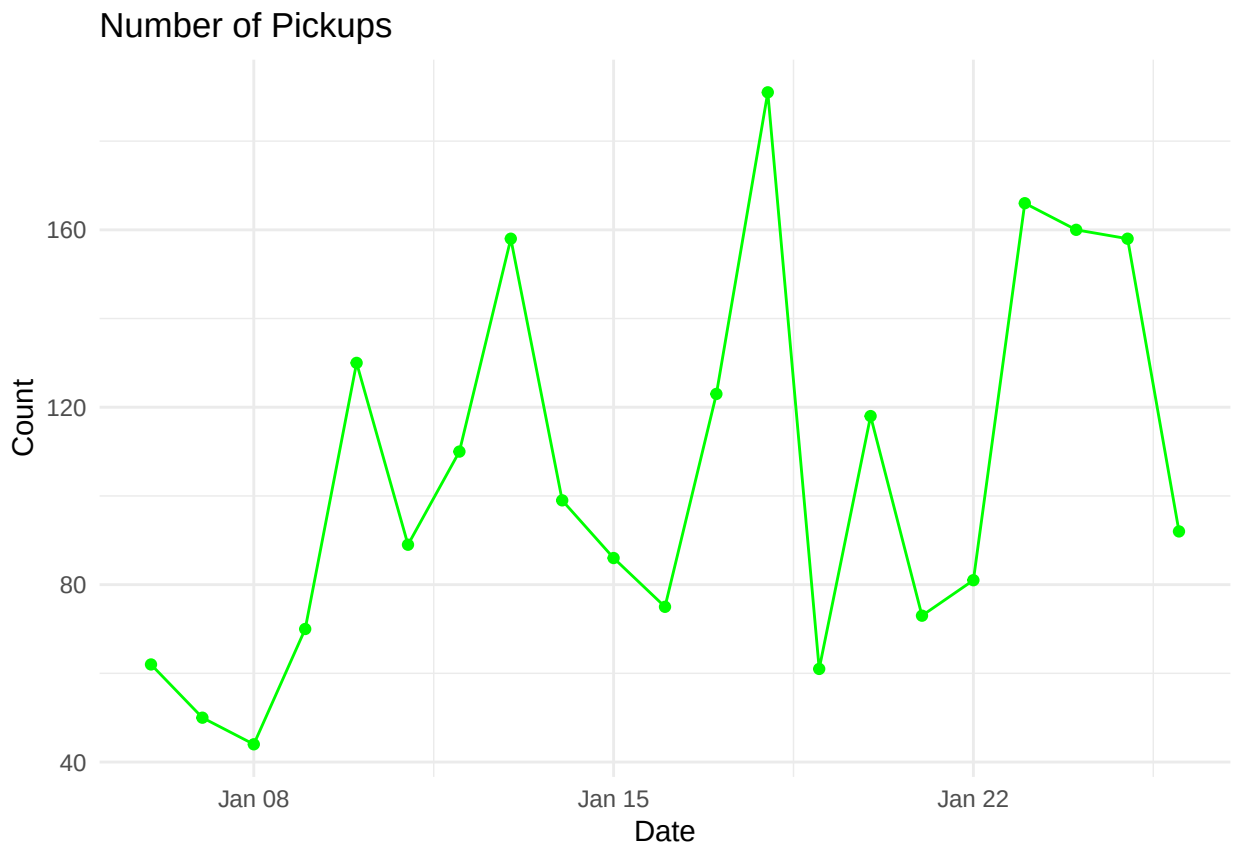
```
screen_data$Date <- as.Date(screen_data$Date, format="%m/%d/%Y")

#for Total Screen Time in Minutes
ggplot(screen_data, aes(x=Date, y=Total.ST.min)) +
  geom_line() +
  geom_point() +
  labs(title="Total Screen Time in Minutes", x="Date", y="Minutes") +
  theme_minimal()
```



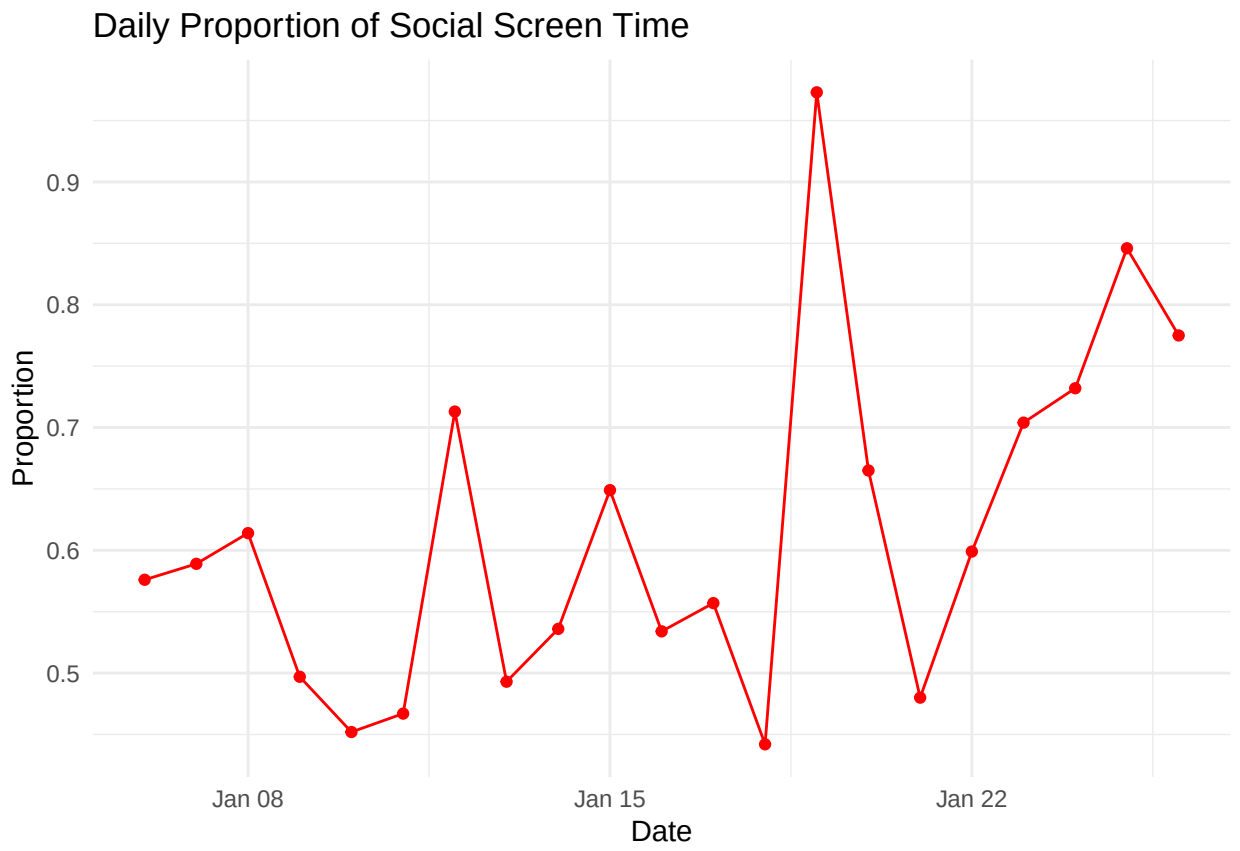

```
#for Social Screen Time in Minutes
ggplot(screen_data, aes(x=Date, y=Social.ST.min)) +
  geom_line(color="blue") +
  geom_point(color="blue") +
  labs(title="Social Screen Time in Minutes", x="Date", y="Minutes") +
  theme_minimal()
```



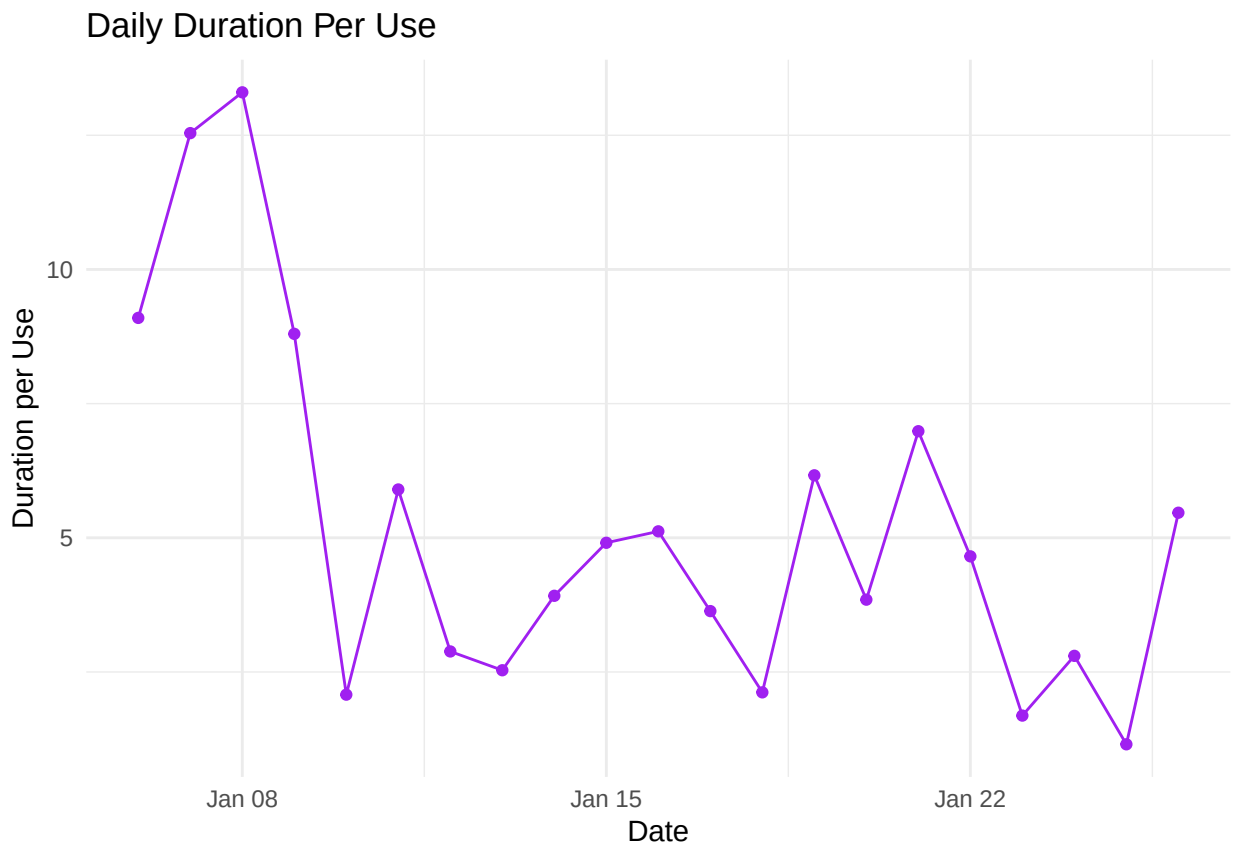

```
#for Number of Pickups
ggplot(screen_data, aes(x=Date, y=Pickups)) +
  geom_line(color="green") +
  geom_point(color="green") +
  labs(title="Number of Pickups", x="Date", y="Count") +
  theme_minimal()
```




```
#for Daily Proportion of Social Screen Time
ggplot(screen_data, aes(x=Date, y=`Daily proportion of social screen time`)) +
  geom_line(color="red") +
  geom_point(color="red") +
  labs(title="Daily Proportion of Social Screen Time", x="Date", y="Proportion") +
  theme_minimal()
```



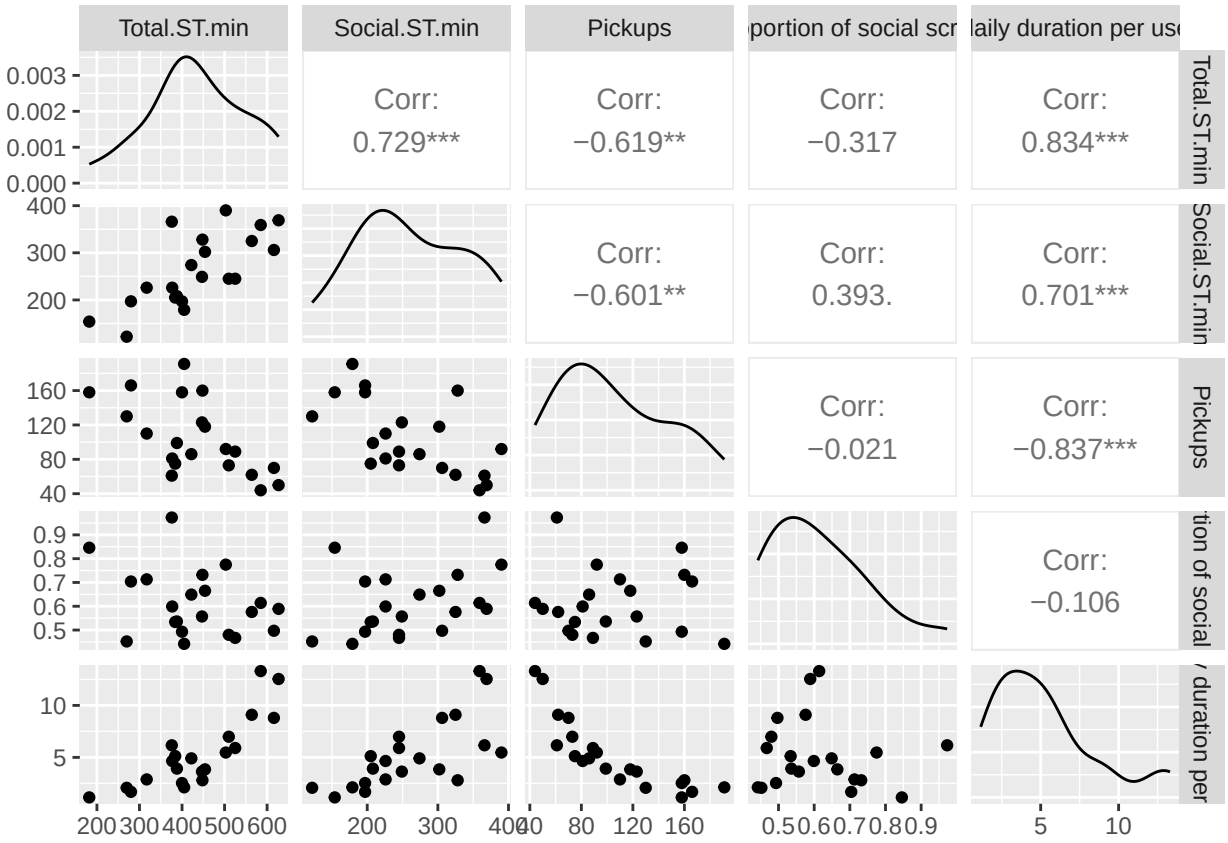

```
#for Daily Duration Per Use
ggplot(screen_data, aes(x=Date, y=`daily duration per use`)) +
  geom_line(color="purple") +
  geom_point(color="purple") +
  labs(title="Daily Duration Per Use", x="Date", y="Duration per Use") +
  theme_minimal()
```



(b)

1. "Total.ST.min" and "Social.ST.min" show a strong positive correlation with a coefficient of 0.729, indicating a significant statistical relationship where increases in total screen time are associated with increases in social screen time.
2. "Pickups" and "Total.ST.min" have a negative correlation of -0.601, suggesting that more frequent pickups are associated with a decrease in total screen time.
3. "Pickups" and "Social.ST.min" also have a negative correlation with a coefficient of -0.619**, indicating that more frequent pickups correlate with less social screen time.
4. "Daily duration per use" shows a strong positive correlation with "Total.ST.min" (Corr: 0.834) and "Social.ST.min" (Corr: 0.701), suggesting that longer durations per use are strongly associated with higher total and social screen time.
5. Conversely, "Pickups" and "daily duration per use" have a strong negative correlation of -0.837, indicating that more frequent pickups are associated with shorter durations of use.

```
ggpairs(screen_data[, c('Total.ST.min', 'Social.ST.min', 'Pickups', 'Daily proportion of social screen
```



(c)

1. The "Social.ST.min Occupation Time Curve" graph displays identifiable patterns of time spent on social media, as evident by the flat regions of the curve which imply particular durations are more frequent.

2. In the "Total.ST.min Occupation Time Curve" graph, the progression of the curve denotes a variety of standard total screen times within the data, punctuated by leaps in cumulative probability at certain time intervals.

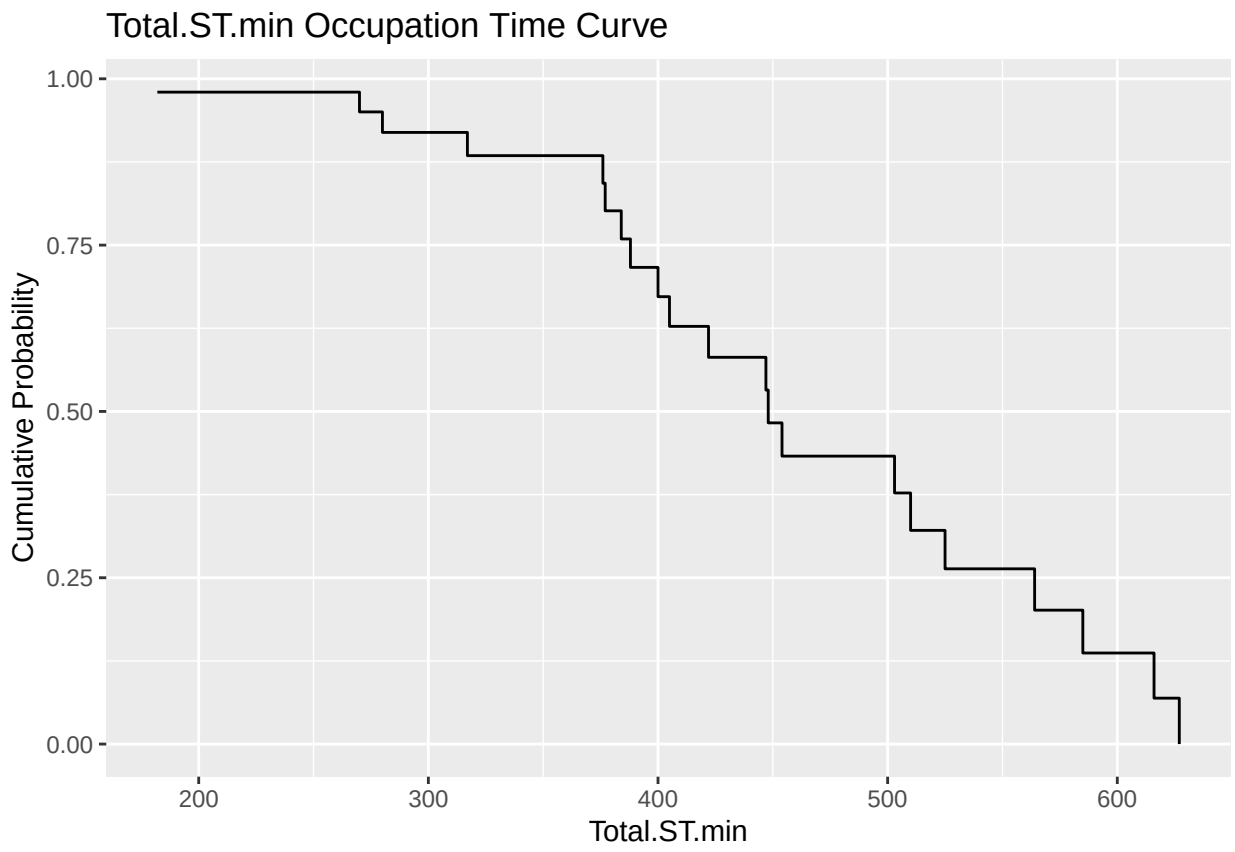
3. The "Pickups Occupation Time Curve" chart reveals a patterned distribution of phone pickups, with the flat segments of the curve indicating repeated pickup counts.

4. The "Daily proportion of social screen time Occupation Time Curve" chart exhibits fluctuations in the daily share of social media screen time, with distinct levels revealing common proportional segments.

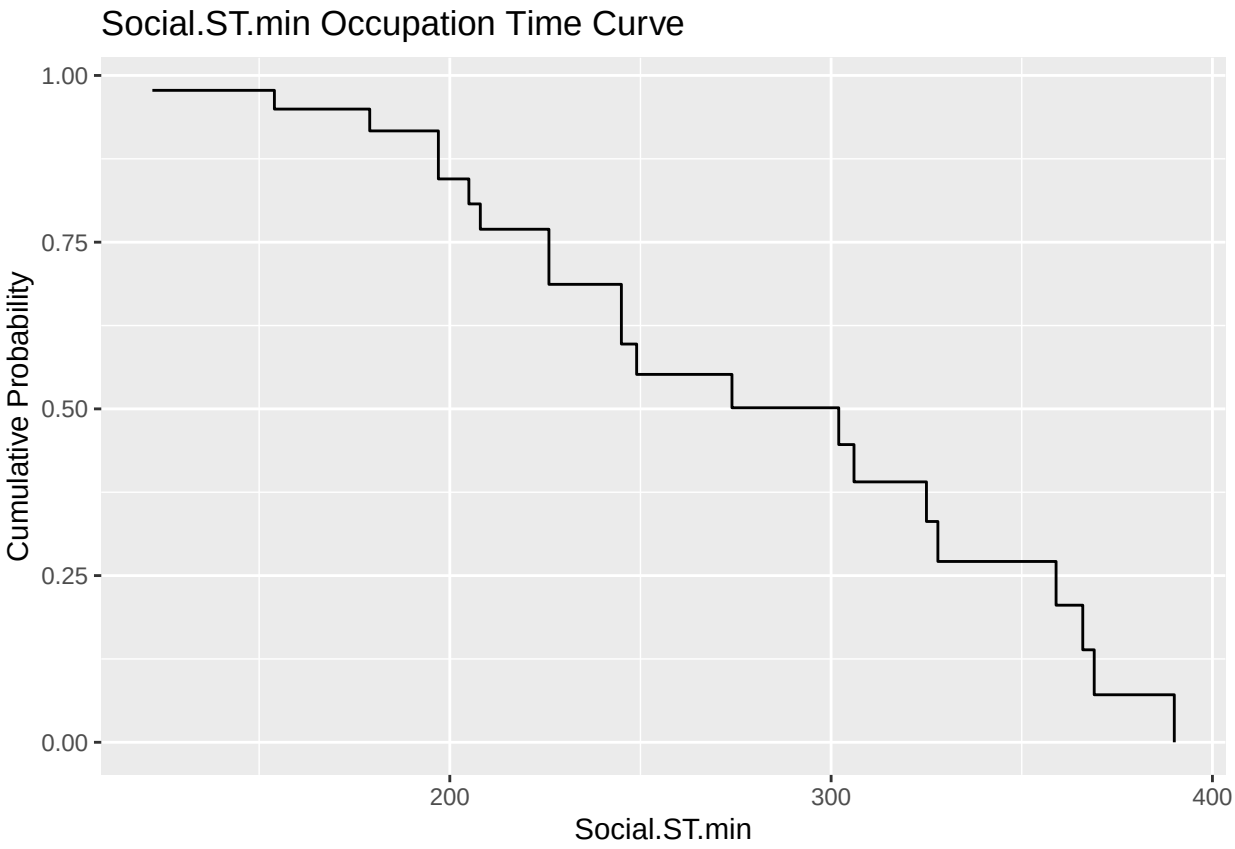
5. The "daily duration per use Occupation Time Curve" chart's stepped configuration indicates that there are specific mean durations of phone use that recur more frequently in the collected data.

```
# Define the function 'plot' to calculate and plot the CDF
plot_c <- function(data, variable_name) {
# calculate cumulative probability using dplyr
  data <- data %>%
    arrange(!!sym(variable_name)) %>%
    mutate(cum_prob = 1 - cumsum(!!sym(variable_name)) / sum(!!sym(variable_name)))
```

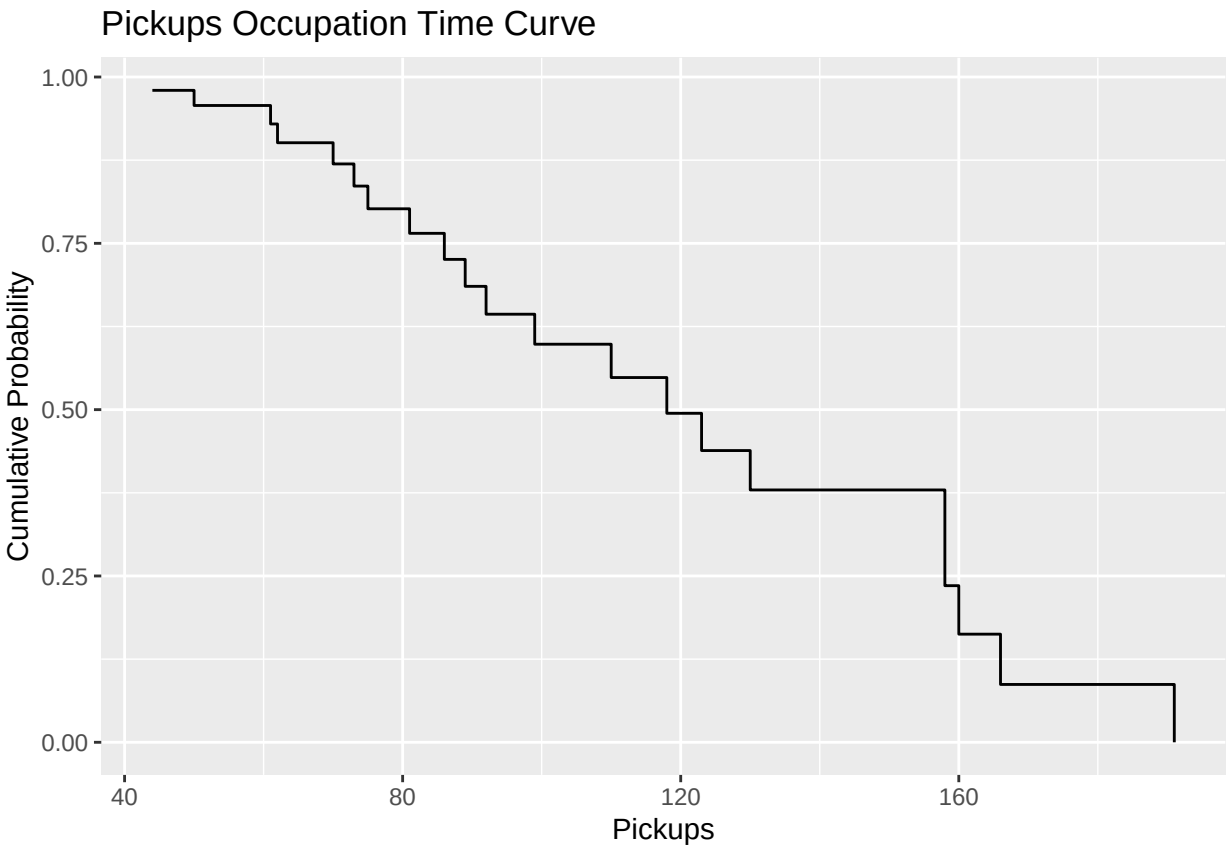
```
  ggplot(data, aes(x = !!sym(variable_name), y = cum_prob)) +
    geom_step() +
    labs(x = variable_name, y = "Cumulative Probability") +
    ggtitle(paste(variable_name, "Occupation Time Curve"))
}


plot_c(screen_data, 'Total.ST.min')
```




```
plot_c(screen_data, 'Social.ST.min')
```

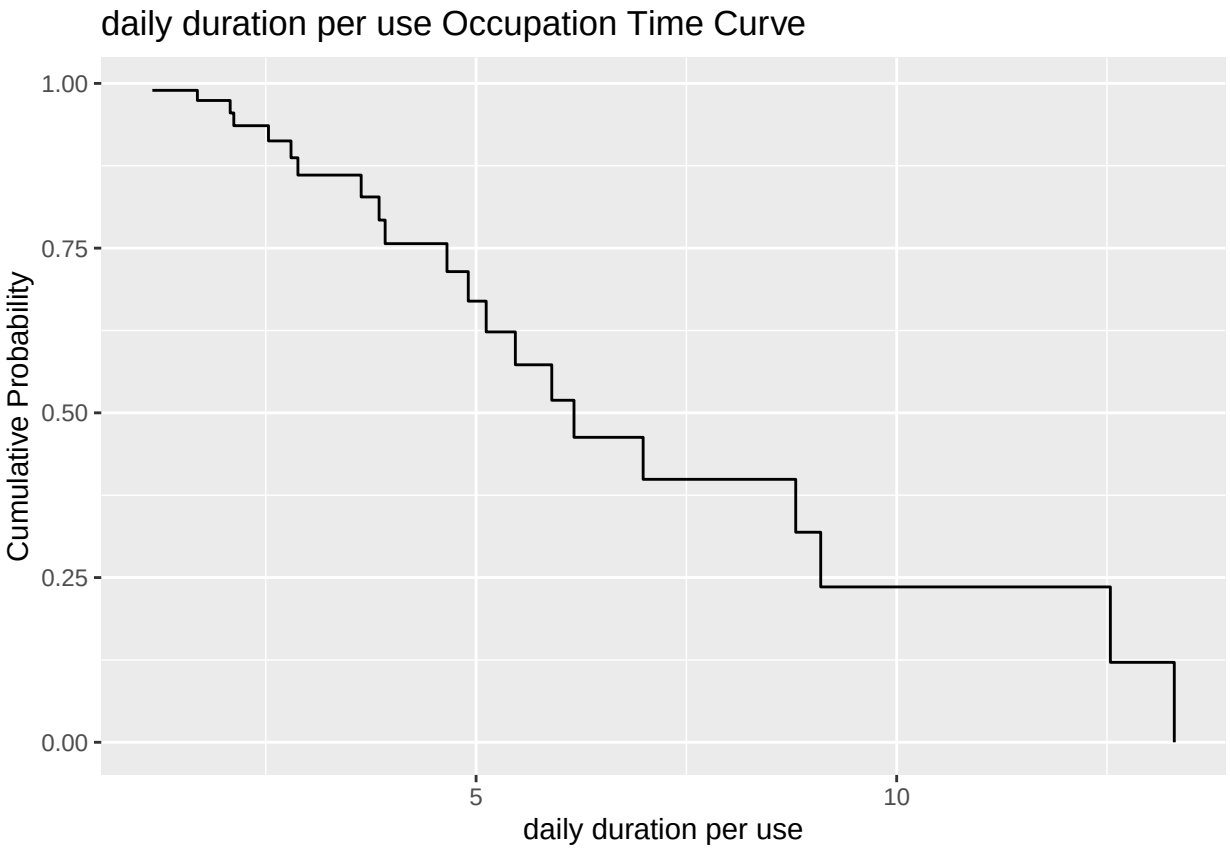



```
plot_c(screen_data, 'Pickups')
```




```
plot_c(screen_data, 'Daily proportion of social screen time')
```


```
plot_c(screen_data, 'daily duration per use')
```



(d)

For 'Total.ST.min', the autocorrelation coefficients are generally low and values are around zero, which suggests no significant autocorrelation.

For 'Social.ST.min', the autocorrelation coefficients are generally low and values are around zero, which suggests no significant autocorrelation.

For 'Pickups', the autocorrelation coefficients are generally low and values are around zero, which suggests no significant autocorrelation.

For 'Daily proportion of social screen time', there is a small positive correlation at lag 7 (0.243), and other lags showing insignificant coefficients, which suggests no significant autocorrelation.

For 'daily duration per use', the highest autocorrelation is seen at lag 1 (0.572), this might imply a possible short-term consistency in daily usage duration.

While small significant autocorrelation in this data is indicated by coefficients close to 1, the coefficients here are not consistently strong, indicating weak autocorrelation in these times.

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

```
##
## Autocorrelations of series 'screen_data$Total.ST.min', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.055  0.340 -0.022 -0.098 -0.096 -0.008 -0.055 -0.076  0.038 -0.017
```

```
##     11     12     13
##  0.006  0.004  0.189
```

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

```
##
## Autocorrelations of series 'screen_data$Social.ST.min', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000 -0.028  0.150 -0.172 -0.065 -0.057 -0.179  0.095 -0.093 -0.119 -0.177
##     11     12     13
##  0.091  0.112  0.098
```

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

```
##
## Autocorrelations of series 'screen_data$Pickups', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.230 -0.006 -0.117 -0.207  0.182 -0.012  0.132  0.025 -0.212 -0.012
##     11     12     13
## -0.015  0.018  0.116
```

```
acf(screen_data$'Daily proportion of social screen time', plot = FALSE)
```

```
##
## Autocorrelations of series 'screen_data$"Daily proportion of social screen time"', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.159 -0.063  0.066  0.175  0.093  0.038  0.243 -0.106 -0.225 -0.136
##     11     12     13
##  0.017 -0.069 -0.108
```

```
acf(screen_data$'daily duration per use', plot = FALSE)
```

```
##
## Autocorrelations of series 'screen_data$"daily duration per use"', by lag
##
##      0      1      2      3      4      5      6      7      8      9     10
##  1.000  0.572  0.262 -0.010 -0.142 -0.059 -0.117 -0.016  0.012 -0.120 -0.058
##     11     12     13
## -0.054  0.026  0.107
```

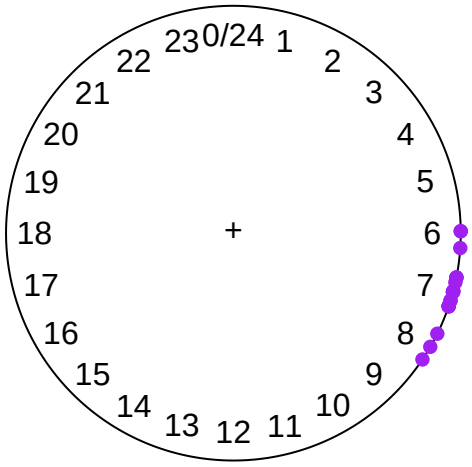
3. (a)

```
times <- strptime(screen_data$Pickup.1st, format="%Y-%m-%d %H:%M:%S")

angles <- (as.numeric(format(times, "%H")) +
                  as.numeric(format(times, "%M")) / 60) * 360 / 24
```

(b)

The most frequent points are around 90 to 120 degrees indicating the range around 6 AM to 8 AM, this means that the consistency of first pick time. This also reflects personal habit of pick the phone once wake up around 6AM to 8 AM, and mostly around 7 AM.

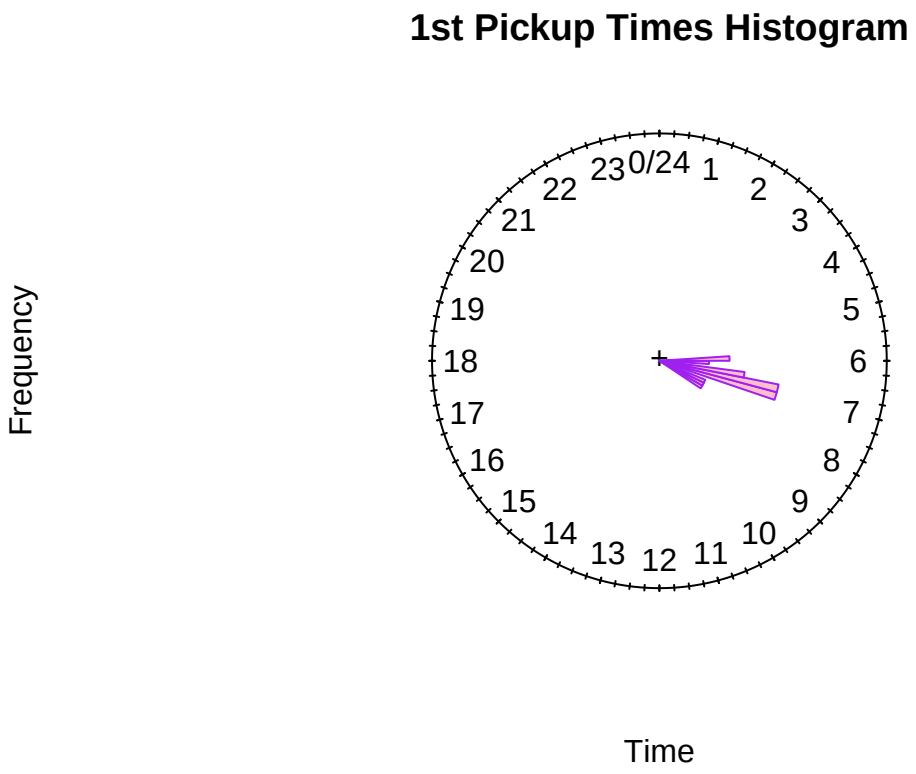
```
# Convert angles to circular data
circle <- circular(angles, units = "degrees")

first_pickup_circle <- circular(circle, units ="degrees", template ="clock24")

plot(first_pickup_circle ,col = "purple")
```



(c)

Upon reviewing the provided data, I chose a bin size of 3.75 degrees for the final histogram plot because the daily fluctuation in wake-up times is approximately 15 minutes, evident in times like 7:15 AM, 8:15 AM, and 6:45 AM. Since there are 96 fifteen-minute intervals in a day (24 hours * 4 intervals/hour), and a full circle is 360 degrees, dividing 360 by 96 yields a bin size of 3.75 degrees. This bin size aligns each bin with a 15-minute interval, which is the observed variability in the wake-up times, thereby allowing the histogram to accurately reflect the distribution of first pickup times throughout the day.

```
bin_size = 3.75
bins = 360/bin_size
rose.diag(first_pickup_circle, bins = bins,  col = "pink", border = "purple", xlab = "Time", ylab = "Fr
```



4. (a)

The factor S(t) is needed in the Poisson distribution because it represents the exposure or time-at-risk for pickups. It adjusts the expected number of pickups lambda based on the total time spent on screens each day. It accounts for the fact that the likelihood of pickups occurring is influenced by the amount of time individuals spend on screens, which means more screen time increases the opportunity for pickups, making it an essential factor in modeling the daily number of pickups.

(b)

The rate parameter lambda is around 2.675.

```
screen_data$Total.ST.hr <- screen_data$Total.ST.min / 60

poisson_model <- glm(Pickups ~ offset(log(Total.ST.hr)), family = poisson, data = screen_data)

summary(poisson_model)
```

```
##
## Call:
## glm(formula = Pickups ~ offset(log(Total.ST.hr)), family = poisson,
##     data = screen_data)
##
## Coefficients:
```

```
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  2.67491    0.02134   125.3   <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: 816.33  on 20  degrees of freedom
## Residual deviance: 816.33  on 20  degrees of freedom
## AIC: 952.93
##
## Number of Fisher Scoring iterations: 5
```

(c)

(c.1) For the variable Xt, which represents whether the day is a weekday (1) or a weekend (0), the p-value is 0.0015083. Since this p-value is less than the significance level of alpha = 0.05, we have evidence to reject the null hypothesis and conclude that there is a statistically significant difference in the behavior of daily pickups between weekdays and weekends.

(c.2) For the variable Zt, which indicates whether the day is during the winter semester (1) starting January 10 or during the winter holiday (0), the p-value is 6.319299e-54, which is also far less than the significance level of alpha = 0.05. This suggests that there is a statistically significant change in the behavior of daily pickups after the winter semester began compared to the winter holiday.

```
screen_data$Xt <- ifelse(screen_data$Date %in% c('2024-01-06', '2024-01-07','2024-01-13','2024-01-14','1

screen_data$Zt <- ifelse(screen_data$Date %in% c('2024-01-06', '2024-01-07', '2024-01-08', '2024-01-09'

log_linear_model <- glm(Pickups ~ Xt + Zt + offset(log(Total.ST.hr)), family = poisson, data = screen_da

summary_log_linear_model <- summary(log_linear_model)
summary_log_linear_model$coefficients
```

```
##              Estimate Std. Error   z value      Pr(>|z|)
## (Intercept) 1.6501445 0.07175207 22.997865 4.896289e-117
## Xt          0.1624334 0.04957554  3.276484  1.051083e-03
## Zt          1.0995526 0.07111590 15.461417  6.319299e-54
```

5. (a)

The value mu is 1.839 and lambda is 49.01

```
# Extract just the hour and minute part
pickup_times <- with(screen_data, as.numeric(substr(Pickup.1st, 12, 13)) + as.numeric(substr(Pickup.1st

# Convert hours and minutes to radians
pickup_angles <- circular::circular(pickup_times * 2 * pi / 24, units='radians', template='none', zero=0

estimates <- mle.vonmises(pickup_angles)

print(estimates)
```

```
##
## Call:
## mle.vonmises(x = pickup_angles)
##
## mu: 1.839  ( 0.03133 )
##
## kappa: 49.01  ( 15.05 )
```

(b)

Based on the estimated parameters, the probability of the first pickup being at 8:30 AM or later to be approximately 0.0037, indicating a very low likelihood of the event of picking up phone occurring at or after 8:30 AM. This is consistent with the data, in the 21-days interval, no day's first pick up time is later than 8:30 AM and only 2 days' first pick up time are around 8 AM, which are 8:15 AM and 8:00 AM. Therefore it is reasonable to have a low likelihood of around 0.0037.

```
mu <- 1.839
kappa <- 49.01

time_830am <- (8.5 / 24) * 2 * pi

if(time_830am > pi) {
  time_830am <- time_830am - 2 * pi
}

# Calculate the cumulative probability up to 8:30 AM
p_before_830am <- pvonmises(time_830am, mu, kappa)
```

```
## Warning in as.circular(x): an object is coerced to the class 'circular' using default value for the
##   type: 'angles'
##   units: 'radians'
##   template: 'none'
##   modulo: 'asis'
##   zero: 0
##   rotation: 'counter'
## conversion.circularqradians0counter
```

```
## Warning in as.circular(x): an object is coerced to the class 'circular' using default value for the
##   type: 'angles'
##   units: 'radians'
##   template: 'none'
##   modulo: 'asis'
##   zero: 0
##   rotation: 'counter'
## conversion.circularmuradians0counter
```

```
p_at_or_after_830am <- 1 - p_before_830am

p_at_or_after_830am
```

```
## [1] 0.003675538
```