Investigating the association between daily social screen time proportion and daily phone pickups with federated learning

Team: XSQ

(Team members: Yuyang Xia, Sangyi Su and Xuheng Qiang)

Abstract:

Background: Federated learning is an effective way to solve data sharing issues when doing statistical analysis. Our study's aim is to prove the unbiasedness of the method by investigating the association between daily social screen time proportion and the daily phone pickups. **Method:** Three investigators' screen time data for 30 days were collected. Linear regression is used to find the association. **Results:** There is a significant association between social screen time proportion and daily phone pickups (β=0.0011， 95%CI:(0.0003,0.0018), $p = 0.009 < 0.05$). The residual standard errors and beta estimations from confirmation analysis are the same as the federated learning. **Conclusion:** Our study proved the unbiasedness of federated learning in linear regression. The positive association between daily social screen time and phone pickups is also informative and intriguing.

Key Phases: Linear regression, Federated learning, Phone pickups, Social screen time

Introduction:

Data analysis tends to become more and more important for us to find trends, solve problems and make decisions. However, raw data might be stored in a distributed fashion and no data sharing is allowed between devices due to data security and data privacy. Thus, we want to analyze the distributed data in a collective way without affecting its data privacy. Since there is an increasing concern of the impact of young people using social media[1], We choose to analyze the screen using activity data collected from a group of three people to operate federated learning techniques and distributed computing and we want to compare the results using these two strategies. Our hypothesis for this project is that there is a positive linear relationship between daily social screen time proportion and daily phone pickups. Also, we are trying to prove the unbiasedness with or without using the federated learning method by investigating the association between daily social screen time proportion and the daily phone pickups.

We collected data of everyday total screen time and social screen time, number of pickups, first pickup time (we use the record after the user's wake-up to mark the beginning and end of the user's day) and we also create variables to record whether that day is snowy, and whether it is weekday. After observing the data, we find that screen time for different users varies quite differently, thus, we derive a new variable called 'daily proportion of social screen time', defined as the ratio of daily total social screen time over daily total screen time, so that the resulting proportion is comparable among individuals for better interpretations, and we set this variable to be our outcome (i.e. dependent variables), we also try to find the independent variables (i.e. variables other than 'daily proportion of social screen time') that are meaningful for the dependent variable.

To characterize participants and teams, we create some basic variables, including number of team members we have worked previously for any other group projects, talk to regularly about academic matters, talked to about topics other than academic matters, to see if these variables that measures participants mutual relationships will affect their daily habits of using mobile devices; variables also including whether the participants live with pets at home that they will look after, sex, age, course credit hours in the winter semester, country where previous degree received, whether currently have a job, number of siblings and the self-reported procrastination score assessed online, to see if daily habits of using mobile devices have something to do with these more personal variables; variable also including number of social apps installed on their major device and number of personal mobile devices possessed, to see if these will affect our habit of using mobile devices.

For our data analysis outline, we first explored each individual's collected screen time data and then we did the variable selection by our primary outcome. Linear regression is used for federated learning validation and confirmation analysis. Model diagnosis is finally conducted to check the assumptions of the model.

Our study is significant because we are trying to improve our understanding of behavioral patterns on the use of electronic devices.

Data Description:

Table 1 lists summary descriptive statistics for all collected variables to describe the team.

Table 1: summary descriptive statistics for all collected variables

Characteristic	Overall
Weekday	90
Yes	15
No	75
Snowy	90
Yes	36
No	54
Total screen time	463.97(279.61)
Social screen time	95.15(80.67)
Pickups	70.96(57.02)
Proportion	0.21(0.16)
Duration	12.30(14.00)
Jobs	3
Yes	1
No	2
Apps	2.33(1.25)
Workmate	0(0)
Academic	0(0)
Nonacademic	0(0)
Pets	3
Yes	0
No	3
Sex	3
female	2
male	1
Age	22.67(0.47)
Course hours	14.5(1.48)
Degree	3
US	0
Non-US	3
Siblings	0(0)
Procrastinations	28.33(3.42)

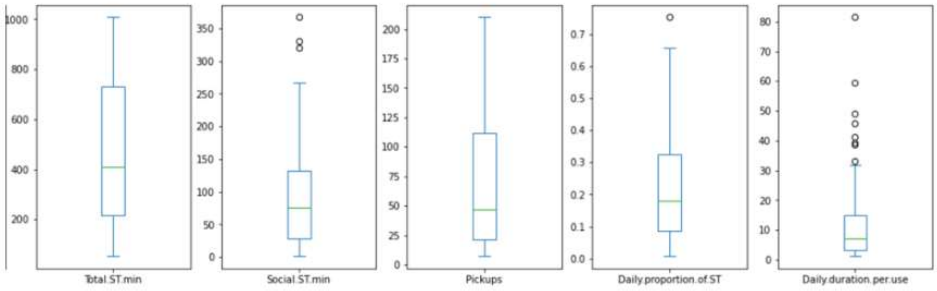

Figure 1: Boxplots of total screen time, social screen time, daily number of pickups, daily proportion of social screen time and daily duration per use, respectively.

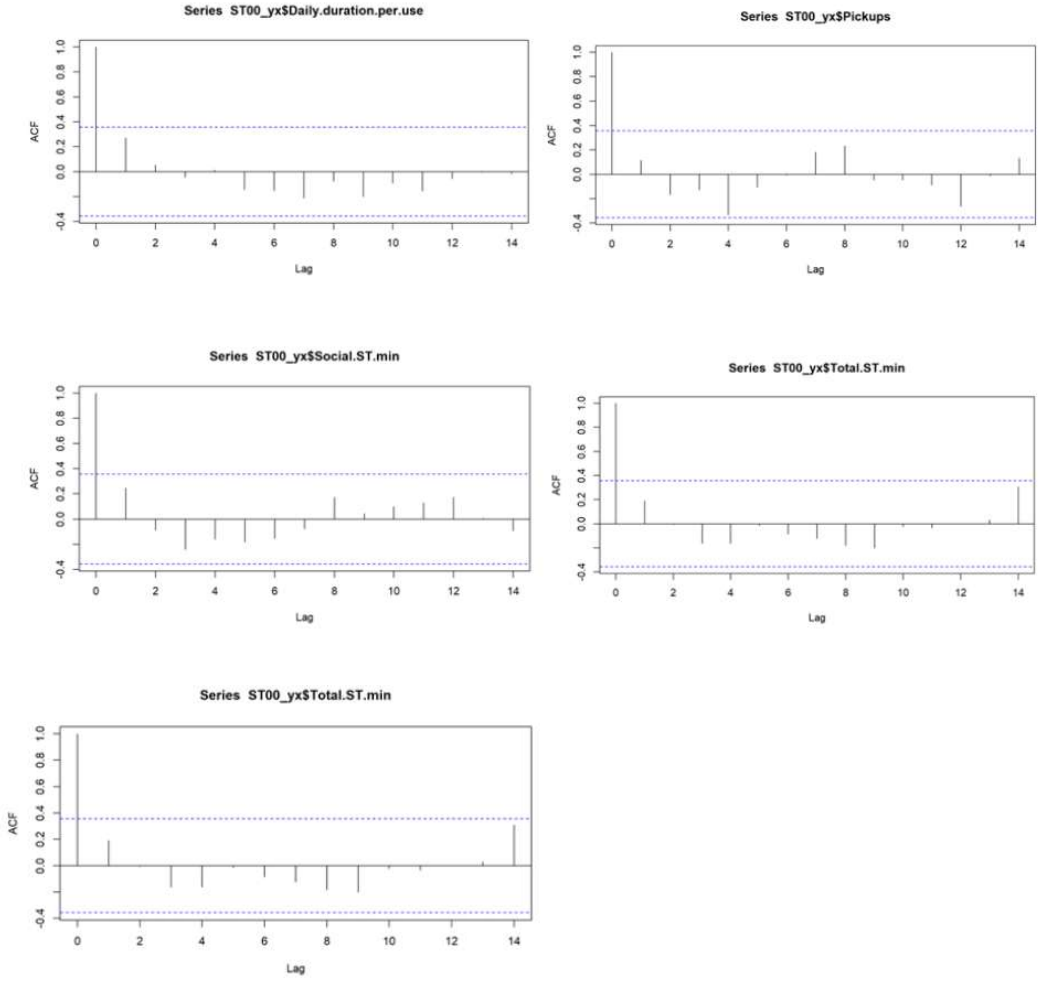

Figure 2: Autocorrelation plots of total screen time, social screen time, daily number of pickups, daily proportion of social screen time and daily duration per use, respectively.

The data used to derive figure 1 and figure 2 are average data collected from our participants. From figure 1, we can derive that participants have similar regular habits at certain parts of the scale, but in other parts of the scale the regularity varies more. From figure 2, only all of the plots indicate non-significant autocorrelation over lags. The data used in figure 3 and figure 4 are raw data collected from our participants. The scatterplot of the first pickup data on a 24-hour clock circle is shown in Figure 3 and Figure 4, indicating most of the 1st pick-up times occur after 6:30am, which could be reflective of the wake-up time of the user. There are also a few 1st pick-up times observed between 2am-7am.

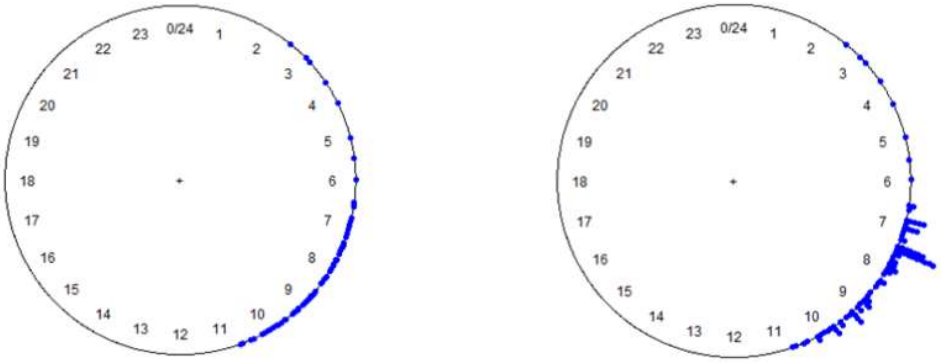

Figure 3: Circular scatter plot of 1st pickup time; histogram of 1st pickup time

Based on the primary outcome - proportion of social screen time, we used simple linear regression to find potential predictors that have correlation with the outcome. Procrastination, phone pickups, and apps installed on the phone are selected to include in our further regression model from the baseline variables and the screen time data. The correlation and distribution of the four variables are shown below where strong correlations are observed (Figure 5).

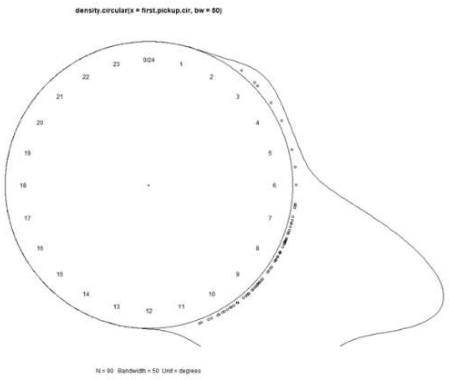

Figure 4: Density plot of 1st pickup time.

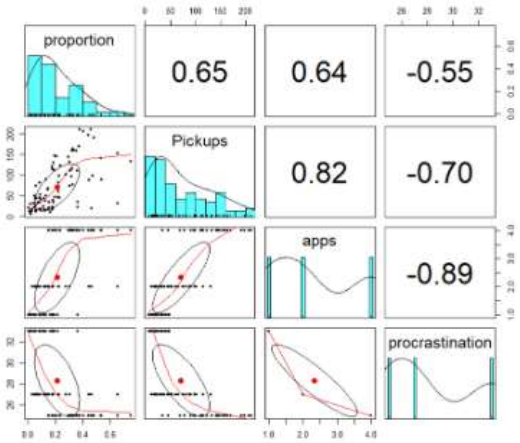

Figure 5: Data description for proportion, pickups, apps, and procrastination

Data Preprocessing:

The separated data from three group members and the baseline variables are combined into a merged dataset. The data collection time period is 30 days in total from Jan 5th to Feb 3rd. Therefore, the sample size of the study is 90. Besides the required baseline variables in the document, "weekday" (binary, 0 indicates weekend, 1 indicates weekday) and "snowy" (binary, 0 indicates didn't snow on the day, 1 indicates snow on the day. Data source: https://www.wunderground.com/, weather based on zip code: 48105.) were also added to the dataset for regression predictor purposes.

Federated Learning:

Based on our study and hypothesis given in the introduction section as well as preliminary data analysis in the Data Description section, in this case we choose Daily proportion of social screen time as our outcome variable y.

Our variables are Pickups, apps and procrastination and we assume that they are independent and identically distributed.

Table. 2 the summary statistics for each data source

sources	SSX			SSY	SSXY
Data 1(Qiang)					
	21971	729	24057	0.5966	90.4317
	729	30	990		3.2122
	24057	990	32670		106.0031
Data 2(Xia)					
	601135	16108	100675	4.3738	1500.2083
	729	30	990		42.6026
	24057	990	32670		266.2664
Data 3(Su)					
	119366	3260	44010	1.4929	305.7117
	3260	120	1620		10.5745
	44010	1620	21870		142.7665
TOTAL(SUM)					
	742472	20097	168742	6.4634	0.0010
	20097	630	5610		0.0464
	168742	5610	73290		0.0010

Each device user in our team represents a data source where raw data is not available but summary statistics can be shared. Thus , we consider Federated learning methods with no need of merging raw data from different data sources to acquire information that we need. The Federated learning procedure is represented in the appendix. And the summary statistics for each group member are shown in Table.2.

Table.3 inference in Federated Learning

	Pickups	Apps	Procrastination
Estimate	0.0011	0.0464	0.0010
Standard error	0.0004	0.0165	0.0008
Residual standard error		0.1229	

Confirmation analysis:

From the results of federated learning, beta estimates for procrastination, apps, and

pickups are equal to the combined data analysis (0.0011, 0.0464, 0.0010). The residual standard error is also the same (0.1229).

When checking normality assumptions, although the Q-Q plot (Figure 6) shows that the residuals are a little right skewed, it won't be much of a problem since our sample size is relatively large (n=90). The constant variance assumption is met since the residuals are randomly distributed (Figure 7). The linearity assumption met since the partial regression plots for three predictors all show linear associations (Figure 8). There is no multicollinearity (VIF=4.90, 7.43, 3.03).

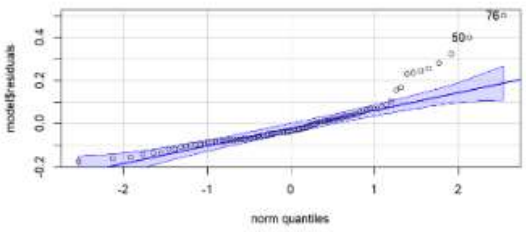

Figure 6: Q-Q plot for linear regression residuals

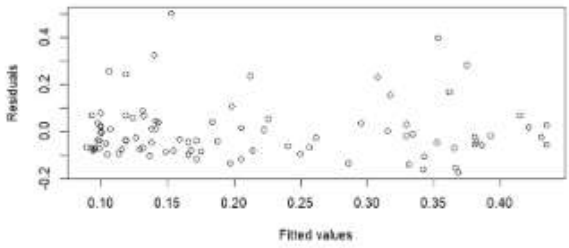

Figure 7: Residual plot

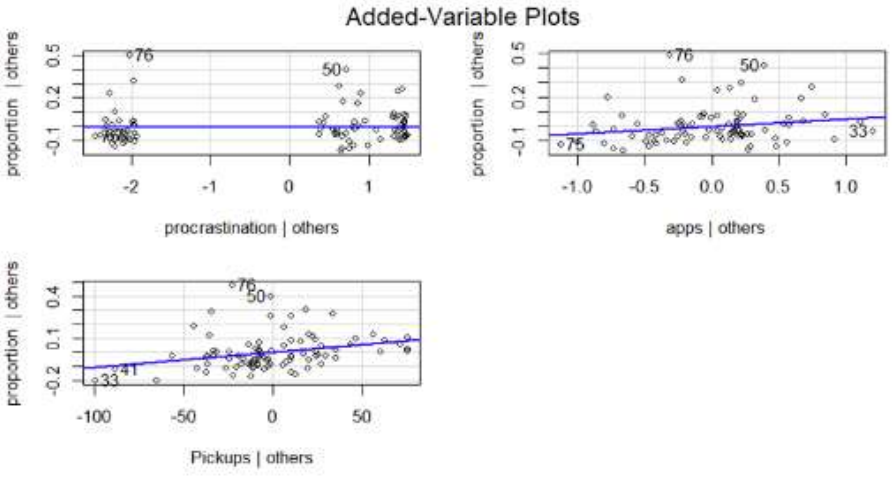

Figure 8: Partial regression plot

Conclusion & Discussion:

We discovered that there is a positive linear relationship between daily social screen time proportion and daily phone pickups adjusted for procrastination score, and applications installed on the phone. However, the results may not be convincing since the data is only collected from three individuals within a relatively short period (30 days) and thus repeated measurements considered as independent data is used for each individual. Future study on this topic should use larger sample size and consider using aggregated values such as mean or median to summarize each individual's data or just use repeated data analysis methods to improve the overall effectiveness of the results. Since the primary aim of the study is to validate the unbiasedness of beta estimation and residual errors of federated analysis in linear regression and our results have proved that successfully, the purpose of the study can be considered as well met.

Acknowledgement:

Thank Yuyang Xia for the introduction and data description part, Sangyi Su for the federated learning part, and Xuheng Qiang for the confirmation analysis part. Thanks to all three group members for the contribution to the abstract and conclusion & discussion writing part.

References:

[1]Barthorpe A, Windstone L, Mars B, Moran P. Is social media screen time really associated with poor adolescent mental health? A time use diary study. J Affect Disord. 2020; 274: 864-870. doi: 10.1016/j.jad.2020.05.106

Appendix:

Aggregated value using as predictor and outcome as an improved analysis method when more data are available for future study as described in the conclusion and discussion part is shown in the last part of R code (Improvement).

Python code:

```
# boxplot
import pandas as pd
screentime_all = pd.read_csv('screentime_all.csv')
screentime_all.plot(kind='box',
             subplots=True,
             sharey=False,
         figsize=(20, 5));
```

R code:

```
#BIOSTAT 620 Project 1
#################################################################################
#Data description
# time series plots
# read data
#install.packages('readxl')
library(readxl)
library(lubridate)
library(dplyr)
setwd('C:/Users/lenovo/Desktop')
# Participants' average data
ST00_yx = read_excel(path = 'screentime_average.xlsx',col_types = c('date', 'numeric',  'numeric'
'numeric', 'numeric','numeric'))
ST00_yx$weekday = weekdays(ST00_yx$Date, abbreviate = T)
ST00_yx = ST00_yx %>% mutate(if_weekend = weekday %in% c('Sunday', 'Saturday'))
library(ggplot2)
library(dplyr)
# total screen time
total = ggplot(ST00_yx, aes(x=Date, y=Total.ST.min, color = if_weekend))+
  geom_line(color = 'steelblue')+
  geom_point()+
  xlab("")+ylab("total_screen_time_(min)")+ylim(300, 650)+
  scale_color_manual(labels = c("weekdays", "weekends"),
                     values=c("black", "red"))+
  theme_minimal()+
  theme(axis.text.x =element_text(angle=60, hjust=1),legend.title = element_blank(),
              panel.background = element_rect(fill = "white"))

ggsave("HD1_b1_total.png")

# social screen time
social = ggplot(ST00_yx, aes(x=Date, y=Social.ST.min, color = if_weekend))+
  geom_line(color = 'steelblue')+
  geom_point()+
  xlab("")+ylab("social_screen_time_(min)")+
  ylim(0, 250)+
```

```
  scale_color_manual(labels = c("weekdays", "weekends"), values=c("black", "red"))+
  theme_minimal()+

  theme(axis.text.x = element_text(angle=60, hjust=1), legend.title = element_blank(), panel.background =
element_rect(fill = "white"))
ggsave("HD1_b1_social.png")
# pickups
pickups = ggplot(ST00_yx, aes(x=Date, y=Pickups, color = if_weekend))+
  geom_line(color = 'steelblue')+
  geom_point()+
  xlab("")+ylab("total_numbers_of_pickups")+
  ylim(0, 250)+
  scale_color_manual(labels = c("weekdays", "weekends"), values=c("black", "red"))+
  theme_minimal()+

  theme(axis.text.x = element_text(angle=60, hjust=1), legend.title = element_blank(), panel.background =
element_rect(fill = "white"))


ggsave("HD1_b1_pickups.png")

# daily proportion of social screen time

dailyproportion = ggplot(ST00_yx, aes(x=Date, y=Daily.proportion.of.ST, color = if_weekend))+
  geom_line(color = 'steelblue')+
  geom_point()+
  xlab("")+ylab("daily_proportion_of_ST")+
  ylim(0, 1)+
  scale_color_manual(labels = c("weekdays", "weekends"), values=c("black", "red"))+
  theme_minimal()+

  theme(axis.text.x = element_text(angle=60, hjust=1), legend.title = element_blank(), panel.background =
element_rect(fill = "white"))
ggsave("HD1_b1_Daily_proportion_of_ST.png")

# daily duration per use
dailyduration = ggplot(ST00_yx, aes(x=Date, y=Daily.duration.per.use, color = if_weekend))+
  geom_line(color = 'steelblue')+
  geom_point()+
  xlab("")+ylab("daily_duration_per_use")+
  ylim(0,40)+
  scale_color_manual(labels = c("weekdays", "weekends"), values=c("black", "red"))+
  theme_minimal()+

  theme(axis.text.x = element_text(angle=60, hjust=1), legend.title = element_blank(), panel.background =
element_rect(fill = "white"))


ggsave("HD1_b1_Daily_duration_per_use.png")
 # ACF plots
acf(ST00_yx$Total.ST.min, plot = TRUE)
acf(ST00_yx$Social.ST.min, plot = TRUE)
acf(ST00_yx$Pickups, plot = TRUE)
```

```
acf(ST00_yx$Daily.proportion.of.ST, plot = TRUE)
acf(ST00_yx$Daily.duration.per.use, plot = TRUE)
# circular plots
setwd('C:/Users/lenovo/Desktop')
# Participants' combined data
ST00_yx = read_excel(path = 'screentime_all.xlsx',
                     col_types = c('date', 'numeric', 'text', 'numeric',
                                   'text', 'numeric', 'numeric', 'date',
                                   'numeric', 'numeric'))
# Correct the errorenous dates
ST00_yx = ST00_yx %>% mutate(Pickup.1st =
                     as.POSIXct
                     (paste(as.character(Date),
                                   unlist
                          (lapply(Pickup.1st,
                                        function(x)
                                  {strsplit(
                                    as.character(x),
                                    split='')[[1]][2]})))))
# Sanity check the minute format variables: Total.ST.min and Social.ST.min
hm_to_min = function(hm){unlist
  (lapply(hm, function(x){splt=strsplit(x, "h")[[1]];
  hr=as.numeric(splt[1]); mn=as.numeric(strsplit(splt[2], "m")[[1]][1]);
  return(60*hr+mn)}))}
ST00_yx = ST00_yx %>% mutate(Total.ST.min.true = hm_to_min(Total.ST),
                     Social.ST.min.true = hm_to_min(Social.ST),
                     Total.ST.match = Total.ST.min.true
                     == Total.ST.min, Social.ST.match =
                    Social.ST.min.true == Social.ST.min) %>%
  relocate(Date, snowy, Total.ST, Total.ST.min,
           Total.ST.min.true, Total.ST.match, Social.ST,
           Social.ST.min, Social.ST.min.true, Social.ST.match)
ST00_yx1 = read_excel(path = 'screentime_all.xlsx',
                     col_types = c('date', 'numeric', 'text', 'numeric',
                                   'text', 'numeric', 'numeric', 'date',
                                   'numeric', 'numeric'))
ST00_yx$Pickup.1st = ST00_yx1$Pickup.1st
ST00_yx$weekday = weekdays(ST00_yx$Date, abbreviate = T)
ST00_yx = ST00_yx %>% mutate(if_weekend = weekday %in% c('Sunday', 'Saturday'))
library(ggplot2)
library(dplyr)

ST00_yx = ST00_yx %>% mutate(Pickups.1st.angular = (hour(Pickup.1st)*60+minute(Pickup.1st))/(24*60)*360)
head(ST00_yx)

#install.packages("circular")
library(circular)
first.pickup.cir = circular(ST00_yx$Pickups.1st.angular, units = "degrees", template="clock24")
png("HD1_3b.png")
plot(first.pickup.cir, col="blue")
```

```
dev.off()
png("HD1_3c0.png")
plot(first.pickup.cir, stack = TRUE, bins=288, col="blue")
dev.off()
# density circular plot
first.pickup.cir.den = density(first.pickup.cir, bw = 50)
png("HD1_3d.png", height = 800, width = 1500)
plot(first.pickup.cir.den, points.plot = T)
dev.off()

################################################################################
#Data preprocessing
library(car)
library(psych)
library(readxl)
Qiang<-read_excel("C:/Users/qiang/OneDrive/Desktop/BIOS 620/proj/1/ScreenTime_Qiang.xlsx")
Xia<-read_excel("C:/Users/qiang/OneDrive/Desktop/BIOS 620/proj/1/Screentime_Yuyang_origin.xlsx")
Su<-read_excel("C:/Users/qiang/OneDrive/Desktop/BIOS 620/proj/1/ScreenTime_Sangyi_origin.xlsx")
colnames(Xia)[8]<-"proportion"
colnames(Xia)[9]<-"duration"
colnames(Su)[8]<-"proportion"
colnames(Su)[9]<-"duration"
Xia<-Xia[12:41,]
Su<-Su[4:33,]
data<-read_excel("C:/Users/qiang/OneDrive/Desktop/BIOS 620/proj/1/ScreenTime.xlsx")
data<-subset(data,select = c(proportion, Pickups,apps,procrastination))
################################################################################Univariate
analysis
attach(data)
lm1<-lm(proportion~procrastination)
lm2<-lm(proportion~Pickups)
lm3<-lm(proportion~apps)
summary(lm1)
summary(lm2)
summary(lm3)
vif(model)
#Describe data
pairs.panels(data)
################################################################################
#Model fitting
model<-lm(proportion~procrastination+apps+Pickups)
confint(model, 'Pickups', level=0.95)
summary(model)
detach(data)
################################################################################
#Model Diagnosis:

#not much violation detected,
#only a bit right skewed when checking normality from Q-Q plot,
#but can be ignored with relatively large n (n=90))
avPlots(model)
plot(model$residuals~model$fitted.values, xlab = "Fitted values", ylab = "Residuals")
qqPlot(model$residuals)
################################################################################
#Improvement:
#Aggregated values(mean value in this case) are used to perform regression due to
```

```
#underlying problems produced by using independent data analysis method to deal with
#repeated measured data.
#This is just an example since the sample size is limited.
mprop<-c(mean(Qiang$proportion),mean(Xia$proportion),mean(Su$proportion))
procrast<-c(33,25,27)
temp<-as.data.frame(cbind(mprop,procrast))
cor(temp$mprop,temp$procrast)
mprocrast<-lm(mprop~procrast,data=temp)
summary(mprocrast)
temp$app<-c(1,4,2)
cor(temp$mprop,temp$app)
mapp<-lm(mprop~app,data=temp)
summary(mapp)
temp$mpick<-c(mean(Qiang$Pickups),mean(Xia$Pickups),mean(Su$Pickups))
cor(temp$mprop,temp$mpick)
mpick<-lm(mprop~mpick,data=temp)
summary(mpick)
modelnew<-lm(mprop~procrast+app+mpick,data=temp)
summary(modelnew)
##########################################################################################divide into 3
subset by ID
data.1 = subset(FLdata, FLdata$id == 'Qiang')
data.2 = subset(FLdata, FLdata$id == 'Xia')
data.3 = subset(FLdata, FLdata$id == 'Su')
##########################################################################################Federated
Learning
#(1)data preparing###########
X1 = cbind(data.1$Pickups,data.1$apps,data.1$procrastination)
X2 = cbind(data.2$Pickups,data.2$apps,data.2$procrastination)
X3 = cbind(data.3$Pickups,data.3$apps,data.3$procrastination)
Y1 = data.1$proportion
Y2 = data.2$proportion
Y3 = data.3$proportion
Y = c(data.1$proportion,data.2$proportion,data.3$proportion)
X = rbind(X1,X2,X3)
#(2)summary statistics###########
(SSX1 = t(X1)%*%X1)
(SSX2 = t(X2)%*%X2)
(SSX3 = t(X3)%*%X3)
(SSY1 = t(Y1)%*%Y1)
(SSY2 = t(Y2)%*%Y2)
(SSY3 = t(Y3)%*%Y3)
(SSXY1 = t(X1)%*%Y1)
(SSXY2 = t(X2)%*%Y2)
(SSXY3 = t(X3)%*%Y3)
(YTY = SSY1+SSY2+SSY3)
(XTY = SSXY1+SSXY2+SSXY3)
(XTX = SSX1+SSX2+SSX3)
#(3)linear regression##########
(beta.hat = solve(XTX)%*%XTY)
(sigma.hat.sq = (YTY-2*t(beta.hat)%*%(XTY)+t(beta.hat)%*%(XTX)%*%beta.hat)/(90-3))
(standard_error.1 = sqrt(sigma.hat.sq)*sqrt(solve(XTX)[1,1]))
(standard_error.2 = sqrt(sigma.hat.sq)*sqrt(solve(XTX)[2,2]))
(standard_error.3 = sqrt(sigma.hat.sq)*sqrt(solve(XTX)[3,3]))
```