Data wrangling

Banana

Data Cleaning

```
#data is given in a long format
d <- fread('banana.csv',header=FALSE, sep=",")
names(d) <- c('block', 'banana_number', 'treatment', 'day', 'humidity',
              'temperature', 'black_ratio', 'weight')
```

- Figure B1(a) indicates there exists both individual fixed effect (i.e., value that varies among differnt subjects, but fixed over time) and block effect (i.e., value that varies between blocks but fixed within blocks)
- The individual fixed effect could have caused by how the picture was taken by different individuals and block fixed effect could have been the result of different enviromental condition (i.e., temperature of the house or humidity or lighting).
- The individual and block fixed had been taken out in Figure B1(b)

```
#remove the fixed effect
d[,b_mu:=mean(black_ratio), by = .(block,banana_number)]
d[,y := black_ratio - b_mu]

p1 <- d %>% ggplot(aes(x = day, y = black_ratio)) +
  geom_point(aes(color = treatment)) + ggtitle("Figure B1(a)") +
  xlab("Day") + ylab("Black Ratio") + theme(plot.title = element_text(hjust = 0.5))

p2 <- d %>% ggplot(aes(x = day, y = y)) +
  geom_point(aes(color = treatment)) + ggtitle("Figure B1(b)") +
  xlab("Day") + ylab("Black Ratio") + theme(plot.title = element_text(hjust = 0.5))

p1 / p2
```

Figure B1(a)

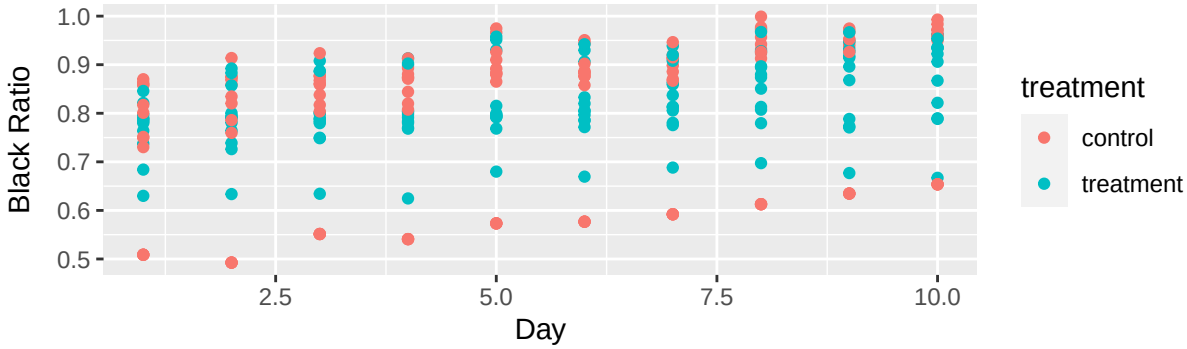
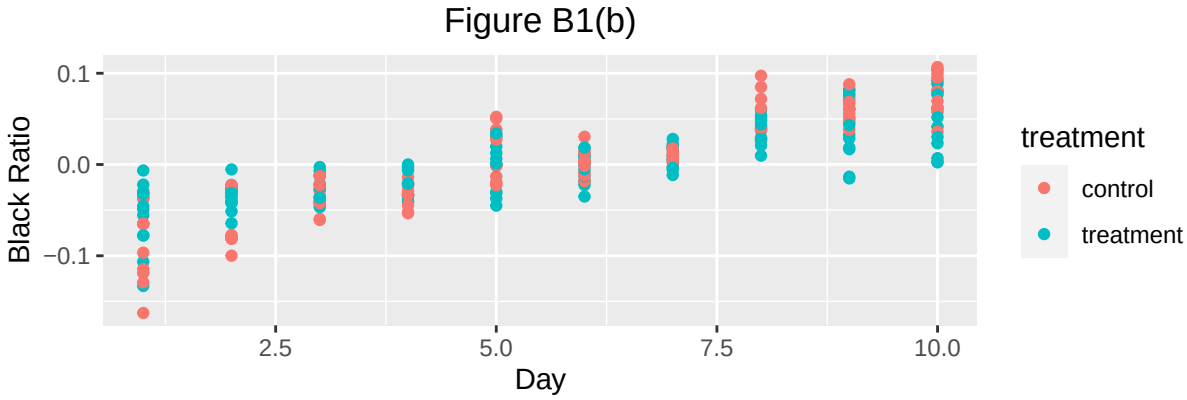


Figure B1(b)



- The percent of `black-ratio` should be non-decreasing function of days, but we have noticed that `black_ratio` can decrease when the tip of the banana withers which cause size of the banana cropped during the image processing part.
- To account for such error introduced while measuring the `black_ratio`, one of off-line abrupt change detection algorithm was employeed to detect changes in `blac_ratio` while accounting for such small noise.

```
#included the detrended data
db <- d[,c('y','block', 'banana_number', 'treatment', 'day', 'humidity', 'temperature','black_ratio','w

#save the result of the detrended data
write.csv(db,"db.csv")

## change the format to process that data
db[,b_name := paste0(block,banana_number)]
db <- db[order(rank(b_name), y)]

## create empty data.frame to store d_turn
b_result <- data.frame(block = numeric(0),banana_number= numeric(0),
                       treatment = numeric(0),
                       d_turn = numeric(0) )
```

Estimate the treatment effect

- The response is the day at which rapid deteroriation of banana condition was observed and the following figure shows there were detected

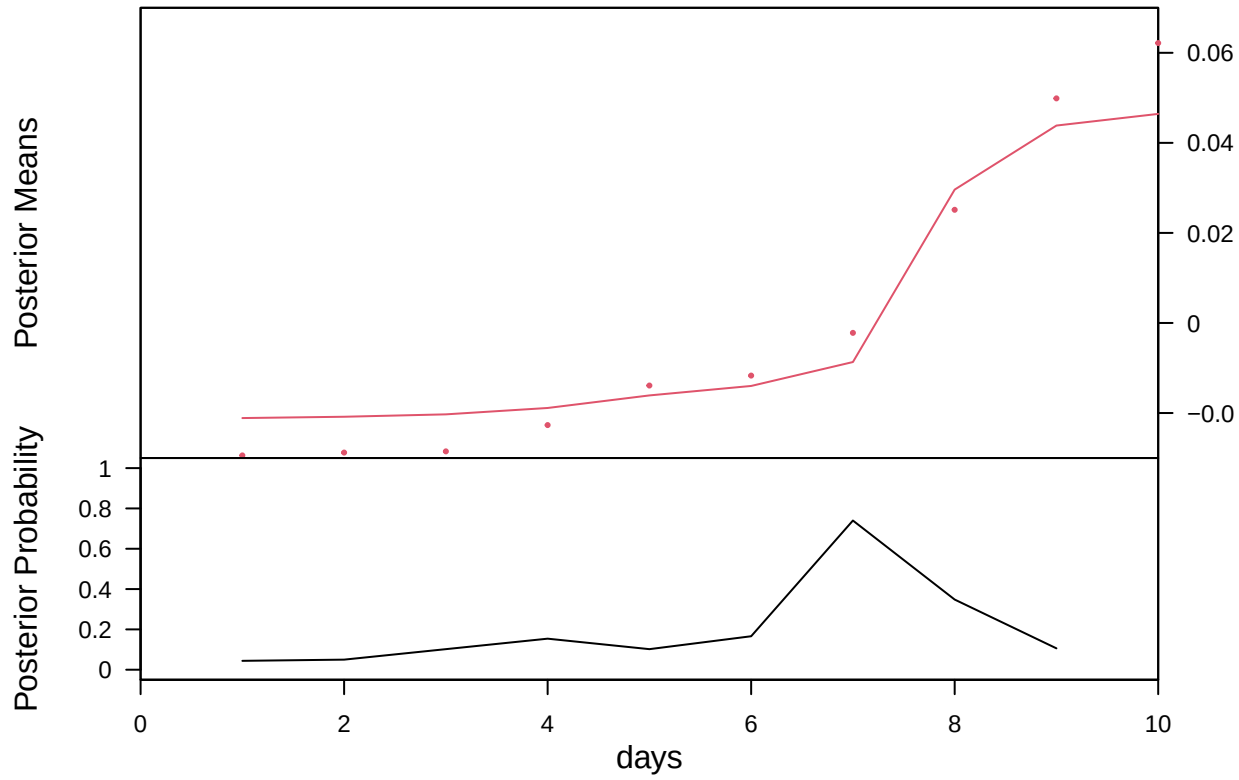
Code for mannual confirmation

```
#try 11 and 24 for demonstration
id <- 25
start <- 1 + (id-1)*10
end <- start + 9
d_temp <- as.data.frame(db[start:end,])

dg <- d_temp$y
dg
```

```
##  [1] -0.029421402 -0.028783308 -0.028521919 -0.022681982 -0.013893121
##  [6] -0.011689147 -0.002195621  0.025132820  0.049883101  0.062170579
```

```
set.seed(1000)
bcp_x <- bcp(dg, return.mcmc = TRUE)

plot(bcp_x,main="Figure B2",
        xlab="days",
        ylab="Black Ratio")
```

Figure B2




```
invisible(capture.output(bcp_sum <- as.data.frame(summary(bcp_x))))

bcp_sum$id <- 1:length(dg)
#selecting draft_number with posterior probabltiy greater than 0.2
sel <- as.data.frame(bcp_sum[which(bcp_x$posterior.prob > 0.2), ])
sel <- sel[sel$id <9,]
sel <- sel[which.max(sel$Probability),]
#get the id
time_of_change <- time(dg)[sel$id]
time_of_change[1]
```

```
## [1] 7
```

Get the data

```
db[,b_name := paste0(block,banana_number)]
db <- db[order(rank(b_name), y)]

b_result <- data.frame(block = numeric(0),banana_number= numeric(0),
                       treatment = numeric(0),
                       d_turn = numeric(0) )
#has 160 rows
for (val in 1:24)
```

```
{
 start <- 1 + (val-1)*10
 end <- (10*val)
 #get the subject information
 val_block   <- db[start,block]
 val_subject <- db[start,banana_number]
 val_treat   <- db[start,treatment]

 # # print(paste(start,":",end))
 d_temp <- db[start:end,]
 d_turn <- getChangeDate(d_temp)
 b_result[val,] <- c(val_block,val_subject,val_treat,d_turn)
}

# b_result
```

Analysis

We have used block random assignement design.
We have three blocks, `B`, `N`, and `J` and the probability of assignement to treatment, $p(d = 1) = 0.5$ for all three which allows us to later pool the data across the block and estimate ATE (page 76 FE)

We have selected block radomization to reduce sampling variablity (see page 72 FE). The subjects within the group are from the same brunch LINK such that subjects in each block have similiar potential outcomes. This also ensured that certain subgroups are available for seperate analysis (see page 71 FE)

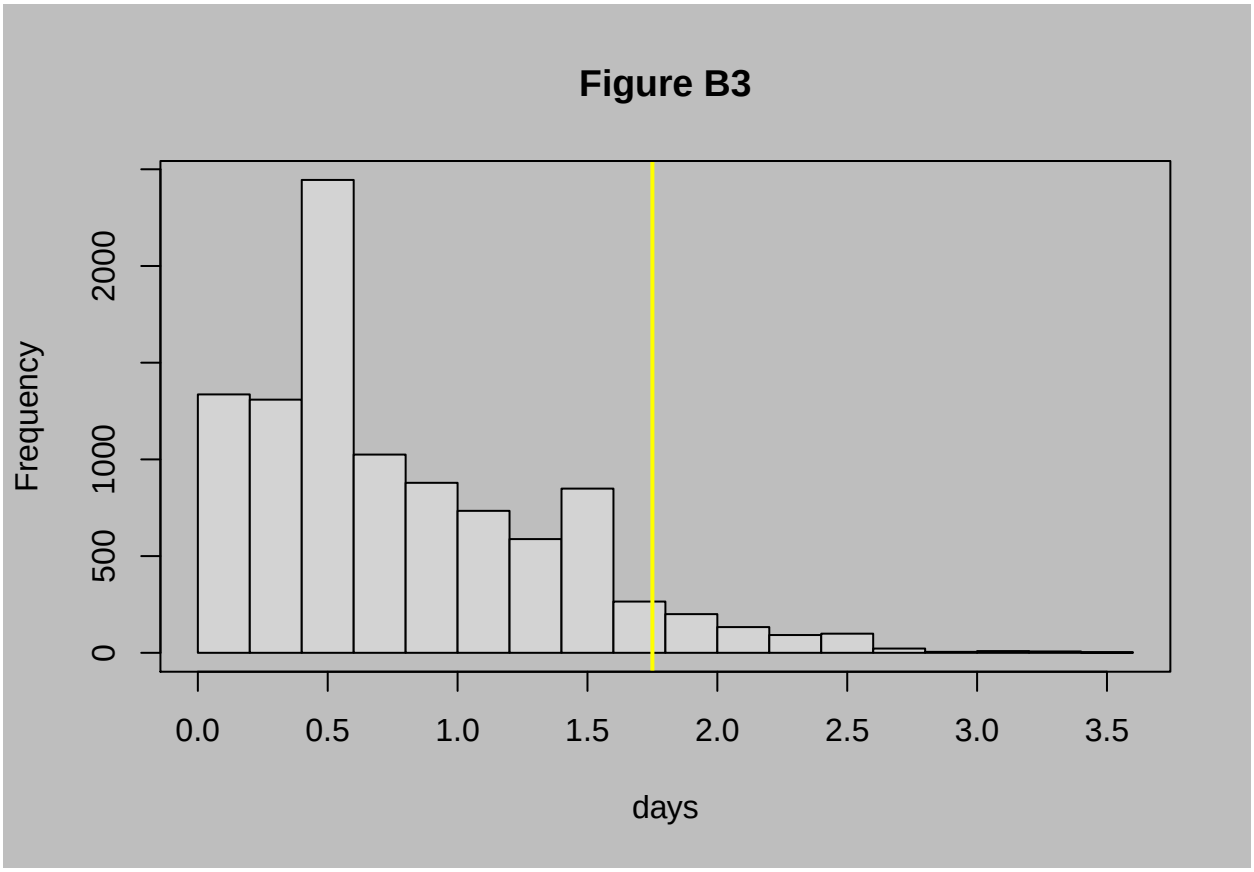
Sharp null hypothesis

We began by testing sharp null hypothesis of no treatment effect

the treatment effect was measured by the day beyond which the black ratio rapidly increased.

```
#get control and treatment data
b_control <- as.integer(b_result[b_result[,"treatment"] == "control","d_turn"])
b_treatment <- as.integer(b_result[b_result[,"treatment"] == "treatment","d_turn"])

#put them into a vector
Z <- c(b_control,b_treatment)

#get the length of each vectors
n <- length(b_control)
m <- length(b_treatment)
N <- length(Z)

# Number of permutations
K = 10000

# Test statistic
```

```r
getMean <- function(A,B) abs(mean(A) - mean(B))

# Test statistic for the observed sample
b_mean <- getMean(b_control,b_treatment)

# Vector of test statistics for each permutation
TT <- vector()

# Permutation test
for(i in 1:K){
 #set.seed(i)
 Z.pi <- sample(Z, N, replace = FALSE)
 TT[i] <- getMean(Z.pi[1:n], Z.pi[(n+1):(n+m)])
}

# Visualising the permuted test statistics
par(bg = 'grey')
hist(TT,main="Figure B3",
        xlab="days",
        ylab="Frequency")
abline(v = b_mean , lwd = 2, col = "yellow")
box()
```




```r
# approximate p-value
b_pvalue <- mean(TT>b_mean )
```

The p-value associated with the observed statistic is 0.057.

Regression Estimator for Block Randomization

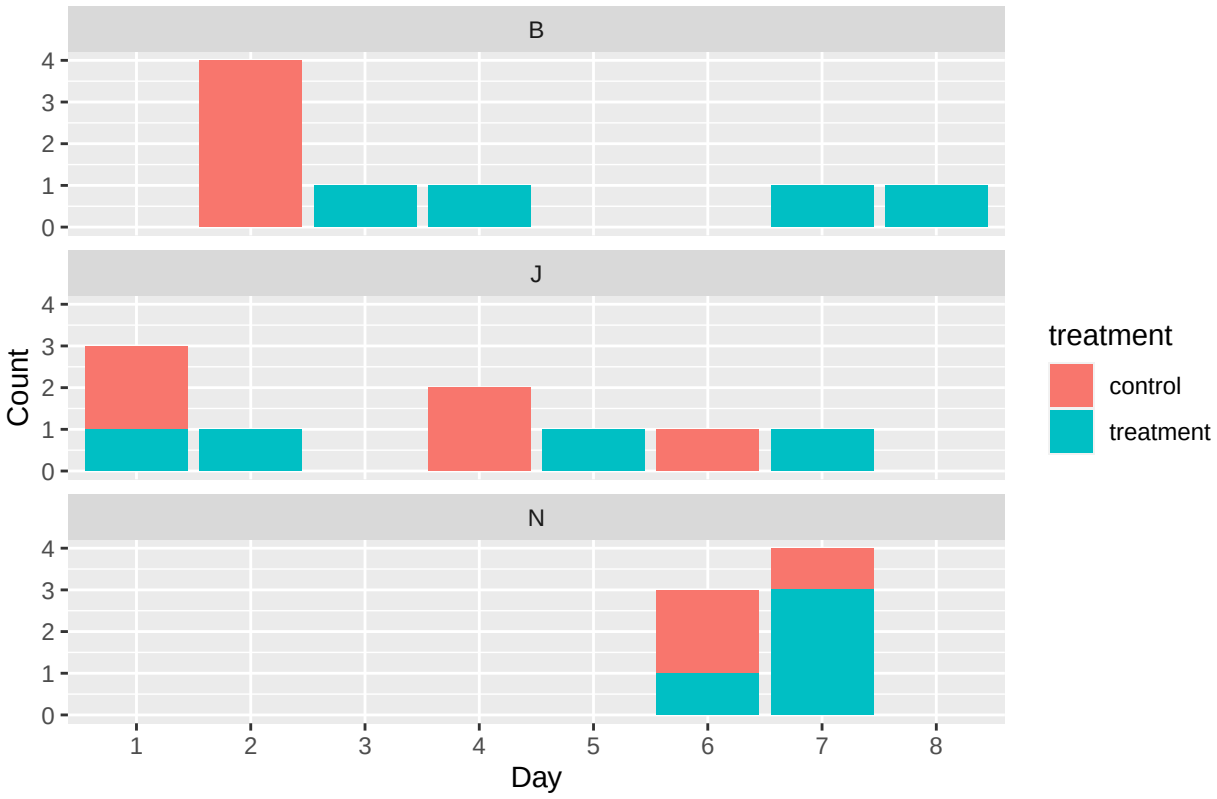
We have 3 blocks, B, N, and J with 8 samples in each block.

```
mod_b <- lm(d_turn ~ treatment + as.factor(block), data = b_result)
coeftest(mod_b, vcov = vcovHC(mod_b, type = "HC1"))
```

```
##
## t test of coefficients:
##
##                     Estimate Std. Error t value  Pr(>|t|)
## (Intercept)          2.99733    0.64013  4.6824 0.0001432 ***
## treatmenttreatment   1.50535    0.79142  1.9021 0.0716616 .
## as.factor(block)J   -0.22193    1.05018 -0.2113 0.8347748
## as.factor(block)N    2.71390    0.76663  3.5400 0.0020553 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
b_result %>% ggplot(aes(x = d_turn, fill=treatment)) + geom_bar() +  facet_wrap(~block,ncol = 1)+ ggtitl
  xlab("Day") + ylab("Count") + theme(plot.title = element_text(hjust = 0.5))
```

Figure B4



```r
temp <- as.data.table(b_result)

b_result2 <- temp[, .(round(mean(as.integer(d_turn))),0),
                  by = .(block,treatment)][,1:3]
print(b_result2)
```

```
##    block treatment V1
## 1:     B   control  2
## 2:     B treatment  6
## 3:     J treatment  4
## 4:     J   control  3
## 5:     N   control  6
## 6:     N treatment  7
```