A/B Testing

Daniel Shang

```
# Load the necessary packages
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)
```

```
## Warning: package 'ggplot2' was built under R version 4.0.3
```

```
library(ggthemes)
```

```
## Warning: package 'ggthemes' was built under R version 4.0.3
```

```
# Import the data and show the first 10 lines
data = read.csv('test_data.csv')
head(data, 10)
```

```
##    user_id   cpgn_id   group email open click  purch  chard sav_blanc  syrah
## 1  1000001 1901Email    ctrl FALSE    0     0   0.00   0.00      0.00  33.94
## 2  1000002 1901Email email_B  TRUE    1     0   0.00   0.00      0.00  16.23
## 3  1000003 1901Email email_A  TRUE    1     1 200.51 516.39      0.00  16.63
## 4  1000004 1901Email email_A  TRUE    1     0   0.00   0.00      0.00   0.00
## 5  1000005 1901Email email_A  TRUE    1     1 158.30 426.53   1222.48   0.00
## 6  1000006 1901Email email_B  TRUE    1     0   0.00   0.00      0.00   0.00
## 7  1000007 1901Email email_B  TRUE    1     1  26.52   0.00      0.00 124.31
## 8  1000008 1901Email email_B  TRUE    1     0   0.00   0.00      0.00  32.12
## 9  1000009 1901Email    ctrl FALSE    0     0   0.00   0.00      0.00 148.59
## 10 1000010 1901Email email_A  TRUE    1     0   0.00   0.00      0.00   0.00
##      cab past_purch days_since visits
## 1   0.00      33.94        119     11
## 2  76.31      92.54         60      3
```

```
## 3   0.00     533.02          9      9
## 4  41.21      41.21        195      6
## 5   0.00    1649.01         48      9
## 6   0.00       0.00        149      6
## 7  58.19     182.50        118      8
## 8  62.67      94.79        125      7
## 9   0.00     148.59        100      7
## 10  0.00       0.00         50      6
```

```
# Summarize the data. The 'user_id' uniquely identifies each row, and each row is
## a customer in the CRM database. The 'cpgn_id' is the same of each row, as the
## whole data set represents a single campaign. The 'group' variable tells if the
## customer is in the control group (receive no email), email group A (receive email
## version A), or email group B (receive email version B). The 'email' means if the
## customer receives the email or not. The 'open' means if the pictures in the email
## are downloaded. 'Click' indicates if the customer click through the link in the
## email. 'Purchase' indicates the amount of purchase the customer made. 'Chard,'
## 'sav_blanc,' 'syrah,' and 'cab' are product categories names. They indicate the
## amount of purchase customers made before under each product category. 'Past_purch'
## is the total amount of past purchases. 'Days_since' indicates the
## number of days past since the last purchase, and 'visits' indicates the number
## of time the customers visited the website.
summary(data)
```

```
##     user_id          cpgn_id             group              email
##  Min.   :1000001   Length:123988      Length:123988      Mode :logical
##  1st Qu.:1030998   Class :character   Class :character   FALSE:41330
##  Median :1061995   Mode  :character   Mode  :character   TRUE :82658
##  Mean   :1061995
##  3rd Qu.:1092991
##  Max.   :1123988
##       open             click              purch              chard
##  Min.   :0.0000   Min.   :0.00000   Min.   :   0.00   Min.   :   0.00
##  1st Qu.:0.0000   1st Qu.:0.00000   1st Qu.:   0.00   1st Qu.:   0.00
##  Median :0.0000   Median :0.00000   Median :   0.00   Median :   0.00
##  Mean   :0.4567   Mean   :0.07503   Mean   :  21.30   Mean   :  73.31
##  3rd Qu.:1.0000   3rd Qu.:0.00000   3rd Qu.:  21.86   3rd Qu.:  54.06
##  Max.   :1.0000   Max.   :1.00000   Max.   :1607.40   Max.   :9636.92
##    sav_blanc           syrah              cab            past_purch
##  Min.   :   0.00   Min.   :   0.00   Min.   :   0.00   Min.   :   0.00
##  1st Qu.:   0.00   1st Qu.:   0.00   1st Qu.:   0.00   1st Qu.:   0.00
##  Median :   0.00   Median :   0.00   Median :   0.00   Median :  91.22
##  Mean   :  72.45   Mean   :  26.68   Mean   :  16.35   Mean   : 188.79
##  3rd Qu.:  57.42   3rd Qu.:  20.91   3rd Qu.:  12.96   3rd Qu.: 246.87
##  Max.   :6609.92   Max.   :2880.15   Max.   :2365.90   Max.   :9636.92
##    days_since         visits
##  Min.   :  0.00   Min.   : 0.000
##  1st Qu.: 26.00   1st Qu.: 4.000
##  Median : 63.00   Median : 6.000
##  Mean   : 89.98   Mean   : 5.946
##  3rd Qu.:125.00   3rd Qu.: 7.000
##  Max.   :992.00   Max.   :51.000
```

```r
# Shows the structure of the data set
str(data)
```

```
## 'data.frame':    123988 obs. of  14 variables:
##  $ user_id   : int  1000001 1000002 1000003 1000004 1000005 1000006 1000007 1000008 1000009 1000010
##  $ cpgn_id   : chr  "1901Email" "1901Email" "1901Email" "1901Email" ...
##  $ group     : chr  "ctrl" "email_B" "email_A" "email_A" ...
##  $ email     : logi  FALSE TRUE TRUE TRUE TRUE TRUE ...
##  $ open      : int  0 1 1 1 1 1 1 1 0 1 ...
##  $ click     : int  0 0 1 0 1 0 1 0 0 0 ...
##  $ purch     : num  0 0 201 0 158 ...
##  $ chard     : num  0 0 516 0 427 ...
##  $ sav_blanc : num  0 0 0 0 1222 ...
##  $ syrah     : num  33.9 16.2 16.6 0 0 ...
##  $ cab       : num  0 76.3 0 41.2 0 ...
##  $ past_purch: num  33.9 92.5 533 41.2 1649 ...
##  $ days_since: int  119 60 9 195 48 149 118 125 100 50 ...
##  $ visits    : int  11 3 9 6 9 6 8 7 7 6 ...
```

```r
# These are the three treatment groups. The number of occurrence of each group
## is, and should be, roughly the same.
table(data$group)
```

```
##
##    ctrl email_A email_B
##   41330   41329   41329
```

```r
# Check the means of different variables within each group. One critical piece
## of A/B Testing a valid randomization. Given a large enough sample size, we would
## expect the data within each baseline variables to be similar. Here, we can see that
## the three variables have roughly the same average. Thus, the randomization used
## when assigning groups is valid.
data %>% group_by(group) %>% summarize(mean(days_since), mean(visits), mean(past_purch))
```

```
## # A tibble: 3 x 4
##   group   `mean(days_since)` `mean(visits)` `mean(past_purch)`
## * <chr>                <dbl>          <dbl>              <dbl>
## 1 ctrl                  90.0           5.95               188.
## 2 email_A               90.2           5.95               188.
## 3 email_B               89.8           5.94               190.
```

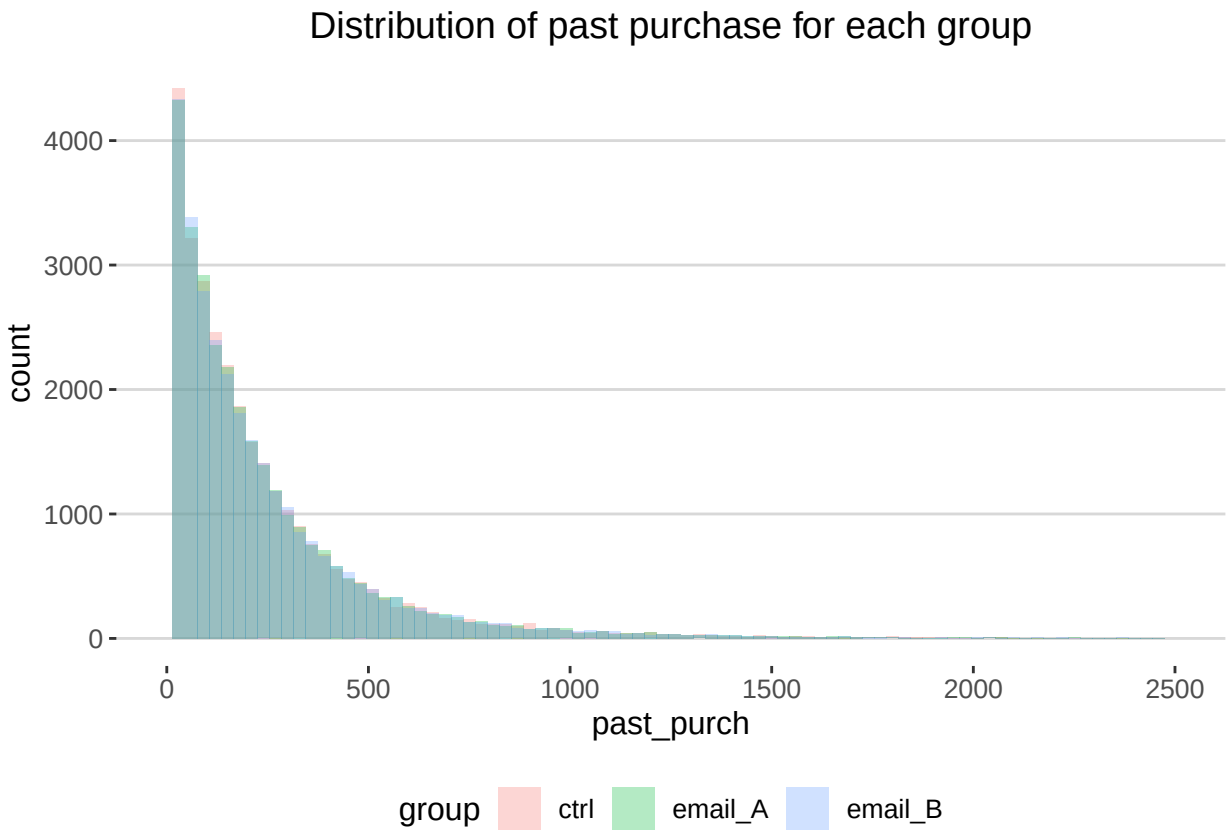
```r
# Similarly, the portion of customers who have purchase history within each group
## is roughly the same.
data %>% group_by(group) %>% summarize(mean(past_purch > 0))
```

```
## # A tibble: 3 x 2
##   group   `mean(past_purch > 0)`
## * <chr>                    <dbl>
## 1 ctrl                     0.744
## 2 email_A                  0.741
## 3 email_B                  0.741
```

```r
# We can also visualize the distribution of 'past_purch' of each group to verify
## if the randomization is valid. Since they have a very similar distribution,
## the randomization was properly conducted.
data %>% filter(past_purch > 0) %>%
  ggplot(aes(x = past_purch, fill = group)) +
  geom_histogram(binwidth = 30, alpha = 0.3, position = 'identity') +
  xlim(0, 2500) +
  labs(title = 'Distribution of past purchase for each group') +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme_hc()
```

```
## Warning: Removed 225 rows containing non-finite values (stat_bin).
```

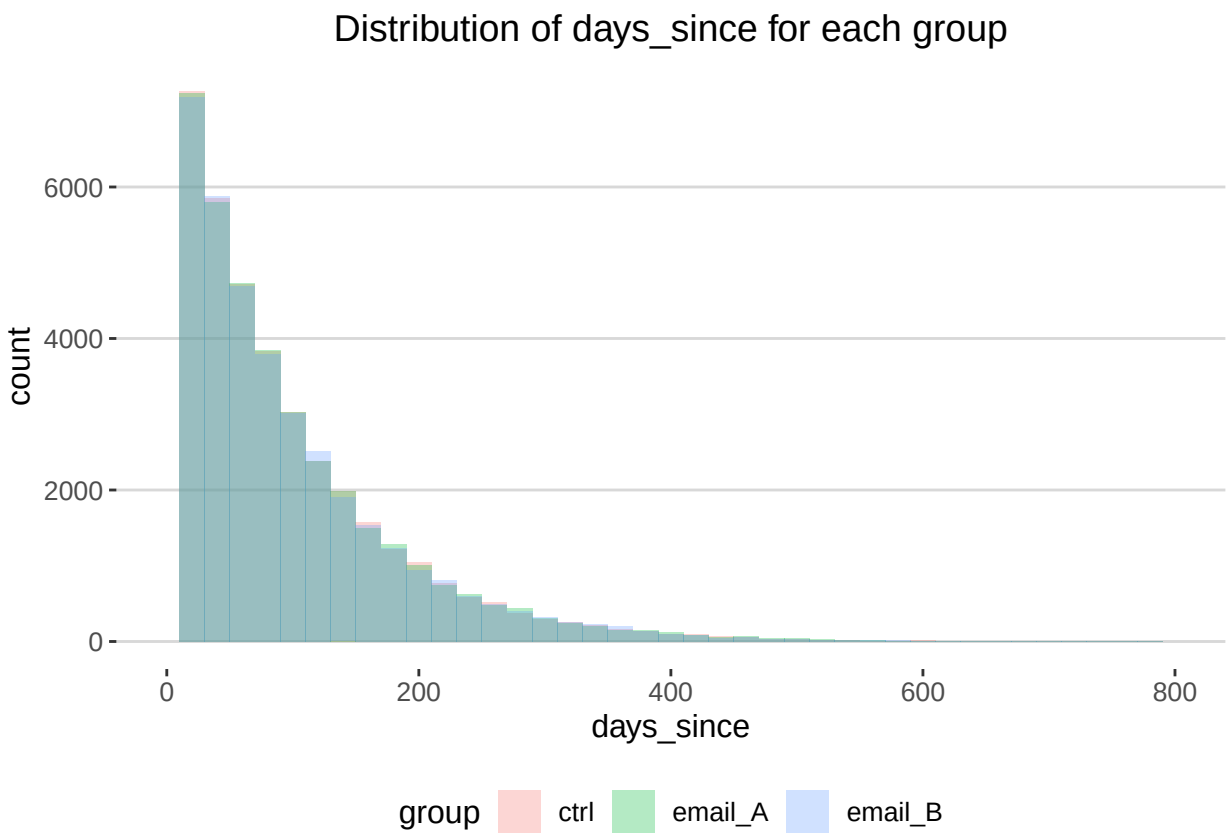
```
## Warning: Removed 6 rows containing missing values (geom_bar).
```




```r
# Check the distribution of 'days_since' for each group
data %>%
  ggplot(aes(x = days_since, fill = group)) +
  geom_histogram(binwidth = 20, alpha = 0.3, position = 'identity') +
  xlim(0, 800) +
  labs(title = 'Distribution of days_since for each group') +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme_hc()
```

```
## Warning: Removed 18 rows containing non-finite values (stat_bin).
```

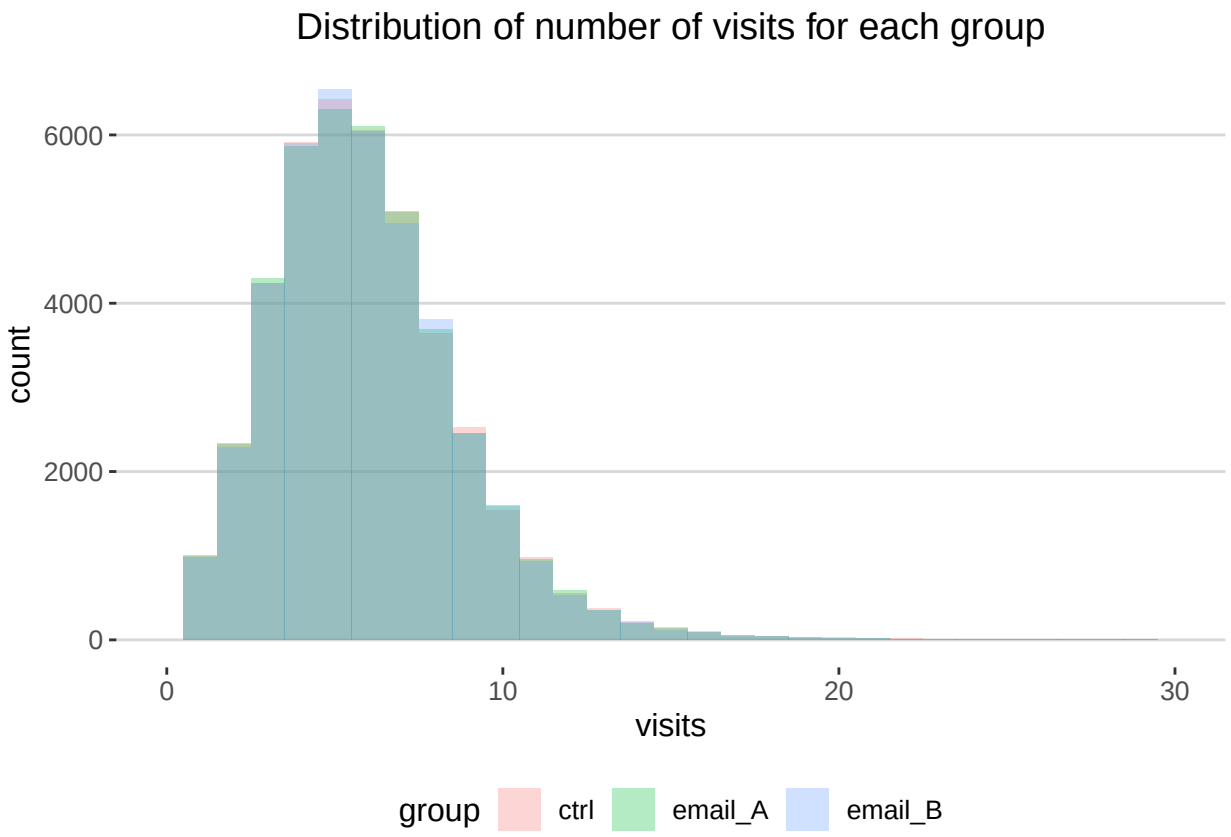
```
## Warning: Removed 6 rows containing missing values (geom_bar).
```




```
# Check the distribution of 'visits' for each group. One thing to note is that
## it is improper to check the distribution of response variables. In this case,
## the three response variables are 'open,' 'click,' and 'purch.' Since we expect
## our campaign to work, the response variable of each group should be different.
## Otherwise, the campaign is not working.
data %>% filter(visits > 0) %>%
  ggplot(aes(x = visits, fill = group)) +
  geom_histogram(binwidth = 1, alpha = 0.3, position = 'identity') +
  xlim(0, 30) +
  labs(title = 'Distribution of number of visits for each group') +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme_hc()
```

```
## Warning: Removed 25 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 6 rows containing missing values (geom_bar).
```




```
# Next, I would like to compare the effect of different emails in terms of customers'
## response and purchasing behavior. We can see that, on average, the possibility
## that customers who received email A downloaded the pictures in the email is 71.8%,
## while the possibility for customers who received email B is 65.2%. Additionally,
## the click through rates for customers who received email A and B are 13.2% and
## 9.3%, respectively. However, it seems that the higher opening and click through
## rates of email A does not convert to more purchase.
data %>% group_by(group) %>% summarize(mean(open), mean(click), mean(purch))
```

```
## # A tibble: 3 x 4
##   group   `mean(open)` `mean(click)` `mean(purch)`
## * <chr>          <dbl>         <dbl>         <dbl>
## 1 ctrl           0             0              12.4
## 2 email_A        0.718         0.132          25.6
## 3 email_B        0.652         0.0934         25.9
```

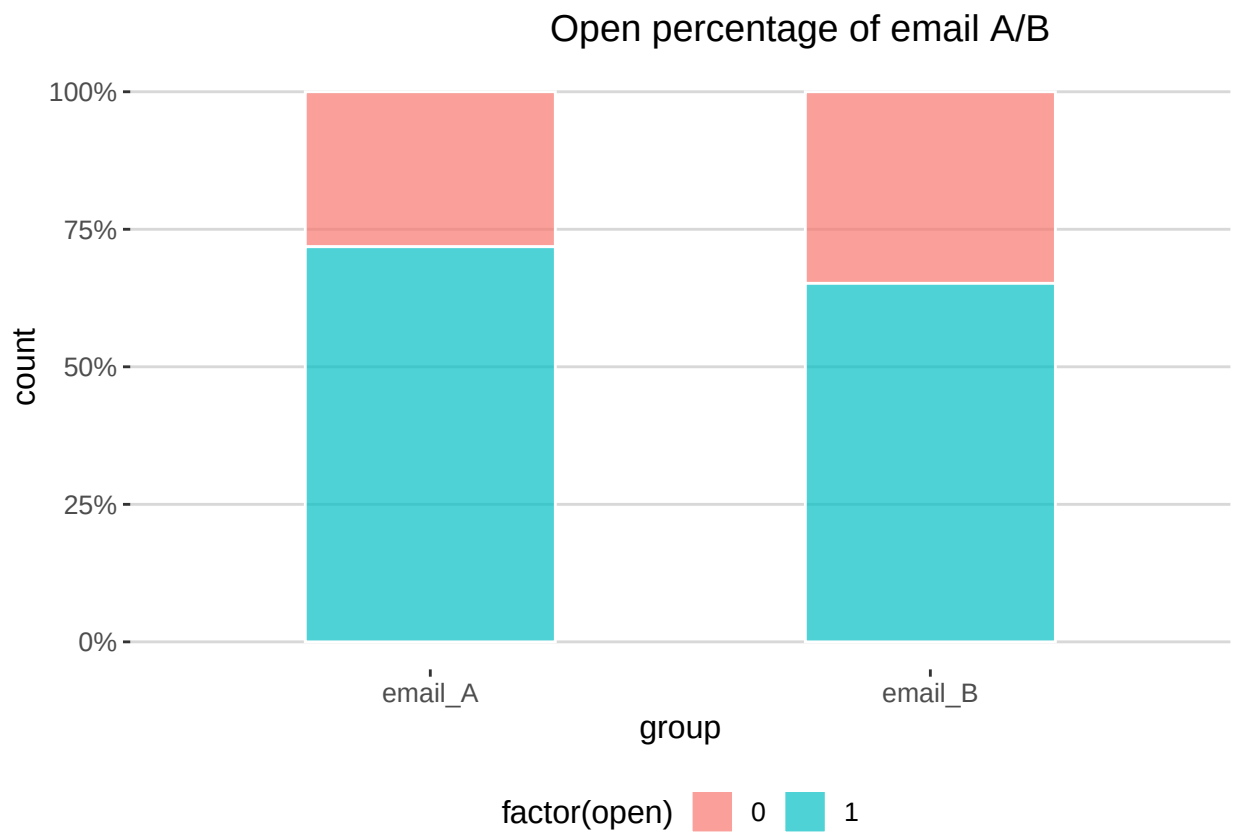
```
# Since we are more interested in the effect difference between email A and email
## B, we remove 'ctrl' from the group. Then, we switch the position of values of
## 'open' field for easier comparison.

# Although we know from the previous table that customers who received email A are
## more likely to open the email, we want to know whether that difference is statistically
## significant. To do that, we use the prop.test function. The null hypothesis of
## this test is that the proportions in the two groups are the same.
data_noctrl = data[data[, 'group'] != 'ctrl', ]
table(data_noctrl$group, data_noctrl$open)[, 2:1]
```

```
##
##               1     0
##   email_A 29686 11643
##   email_B 26934 14395
```

```
# A two-sided test returns a p-value of 2.2 * 10 ^ -16, a value much less than 0.05.
## Therefore, we have strong evidence to reject the null hypothesis and conclude
## that the proportions of the two groups are different. In addition, we get a
## 95% confidence interval of the percentage difference, meaning that we are 95%
## confident that the difference between the two proportions will fall into that
## interval. Thus, we know that email A convert to a higher open rate than email B.
prop.test(table(data_noctrl$group, data_noctrl$open)[, 2:1], alternative = 'two.sided')
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  table(data_noctrl$group, data_noctrl$open)[, 2:1]
## X-squared = 424.32, df = 1, p-value < 2.2e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.06024628 0.07292897
## sample estimates:
##    prop 1    prop 2
## 0.7182850 0.6516974
```

```
# Visualize the two groups to show the difference between their open rates
data %>% filter(group != 'ctrl') %>%
  ggplot(aes(x = group, fill = factor(open))) +
  geom_bar(width = 0.5, alpha = 0.7, position = 'fill', color = 'white') +
  scale_y_continuous(labels = scales::percent) +
  labs(title = 'Open percentage of email A/B') +
  theme(plot.title = element_text(hjust = 0.7)) +
  theme_hc()
```




```
# Similarly, we build a table and run a test to show if email A does a better job
## at boosting click rate than email B, and whether the difference is statistically
## different
table(data_noctrl$group, data_noctrl$click)[, 2:1]
```

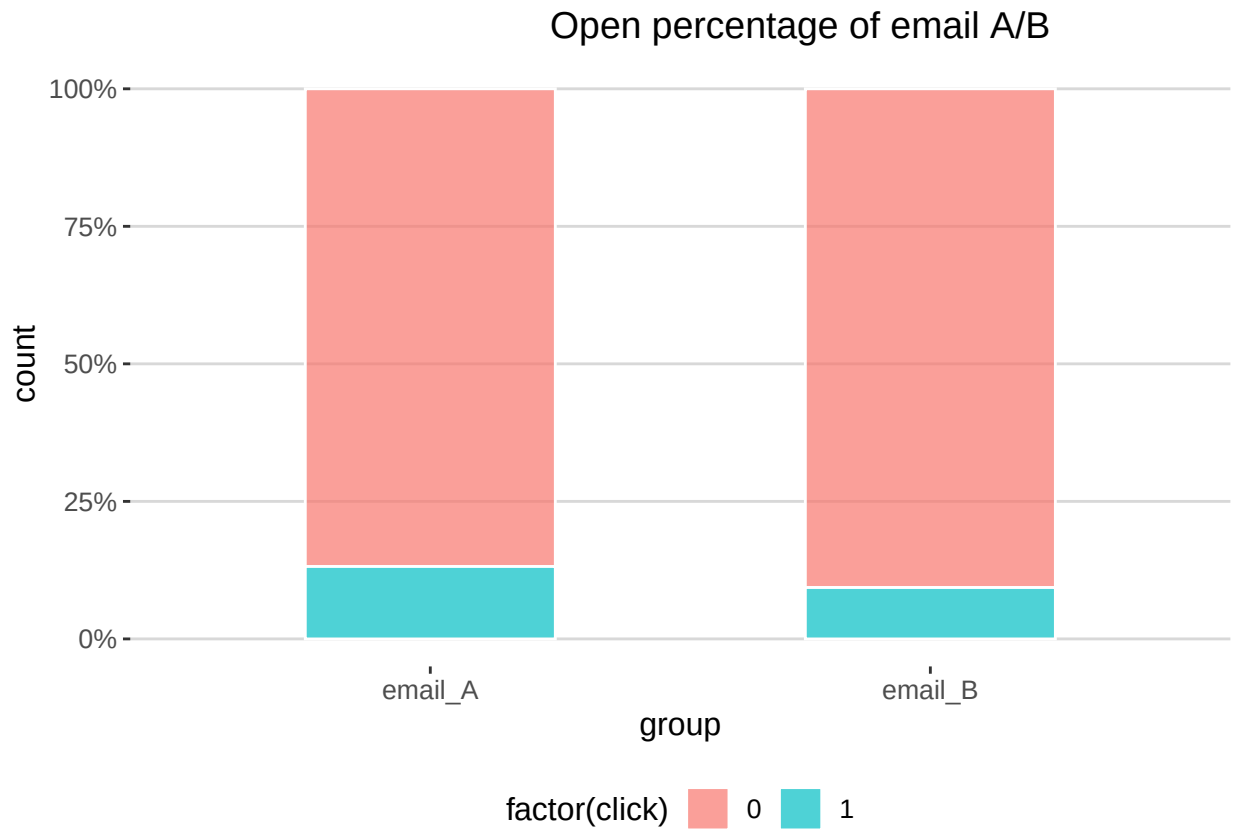
```
##
##               1     0
##   email_A  5442 35887
##   email_B  3861 37468
```

```
# The test result returns a p-value small enough that we can reject the null
## hypothesis and conclude that the click rate of email A and email B is different.
## Specifically, we are 95% confident that the difference between the click rate
## falls in the range from 0.034 to 0.043.
prop.test(table(data_noctrl$group, data_noctrl$click)[, 2:1], alternative = 'two.sided')
```

```
##
##  2-sample test for equality of proportions with continuity correction
##
## data:  table(data_noctrl$group, data_noctrl$click)[, 2:1]
## X-squared = 302.38, df = 1, p-value < 2.2e-16
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  0.03392871 0.04257931
## sample estimates:
```

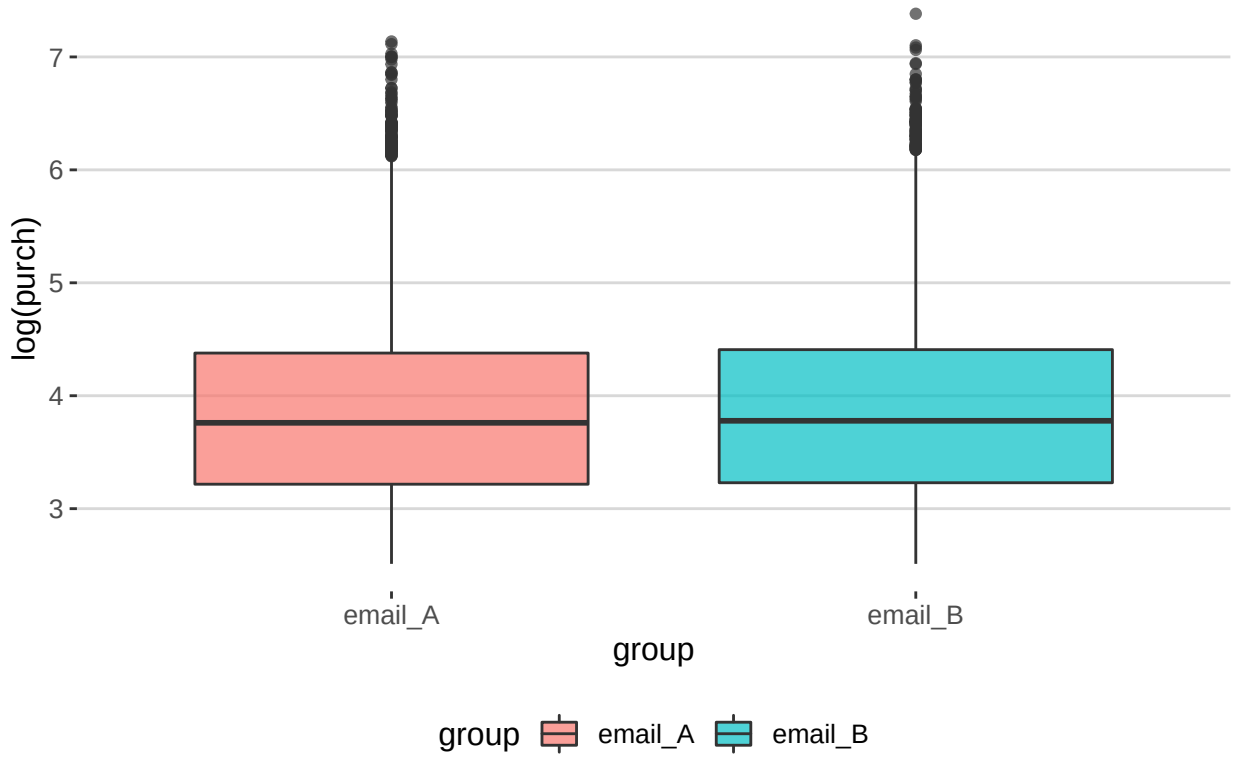
```
##     prop 1     prop 2
## 0.13167509 0.09342108
```

```r
# Visualize the data for better understanding
data %>% filter(group != 'ctrl') %>%
  ggplot(aes(x = group, fill = factor(click))) +
  geom_bar(width = 0.5, alpha = 0.7, position = 'fill', color = 'white') +
  scale_y_continuous(labels = scales::percent) +
  labs(title = 'Open percentage of email A/B') +
  theme(plot.title = element_text(hjust = 0.7)) +
  theme_hc()
```




```r
# Next, we look at the purchase made by customers after they received either email
## A or email B. Since the purchase amount shows some outliers, we use the log
## transformation to condense the plot. The box plots of both groups are similar
## to each other.
data_noctrl %>%
  ggplot(aes(x = group, y = log(purch), fill = group)) +
  geom_boxplot(alpha = 0.7) +
  labs(title = 'Distribution of purchase on log scale in each group') +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme_hc()
```

```
## Warning: Removed 51719 rows containing non-finite values (stat_boxplot).
```




```
# To verify that the purchase amounts are similar between the two groups, we run a
## two sample t-test. Since the p-value is large and the 95% confidence interval
## includes 0, we do not have enough evidence to reject the null hypothesis and
## hence conclude that the mean purchases of the two groups are not significantly
## different.
t.test(purch ~ group, data = data_noctrl)
```

```
##
##  Welch Two Sample t-test
##
## data:  purch by group
## t = -0.59169, df = 82644, p-value = 0.5541
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -1.0498820  0.5629813
## sample estimates:
## mean in group email_A mean in group email_B
##              25.62284              25.86629
```

```
# Knowing the difference between the two email versions, I would like to know more
## about the email and no email group. To do that, I visualized the data and found
## that those customers who were sent an email were constantly making more purchases
## than those who did not receive email at all. Therefore, I would recommend the
## company to design a marketing campaign that sends email to promote sales.
data %>%
```

```
  ggplot(aes(x = purch, fill = email)) +
  geom_histogram(binwidth = 5, alpha = 0.5, position = 'identity') +
  xlim(0, 500) +
  ylim(0, 3500) +
  labs(title = 'Distribution of purchase by email') +
  theme(plot.title = element_text(hjust = 0.5)) +
  theme_hc()
```

```
## Warning: Removed 214 rows containing non-finite values (stat_bin).
```

```
## Warning: Removed 4 rows containing missing values (geom_bar).
```


```
# As a verification, I used a t-test to check if the purchase amount difference
## between email and non-email group is significant. A p-value much smaller than 0.05
## shows that the difference is statistically. Thus, we have strong evidence to
## reject the null hypothesis that the mean purchase amounts between the two groups
## are the same.
t.test(purch ~ email, data = data)
```

```
##
##  Welch Two Sample t-test
##
## data:  purch by email
## t = -44.823, df = 107015, p-value < 2.2e-16
```

```
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -13.90691 -12.74164
## sample estimates:
## mean in group FALSE  mean in group TRUE
##            12.42029            25.74456
```