Search Costs Email Randomization

June 22, 2024

Items

calculate seminar bins based on number of seminars per department

```
# Count the number of seminars per department
seminar_counts <- data %>%
  group_by(department) %>%
  summarize(seminar_count = n()) %>%
  arrange(seminar_count)

# Define bins based on the seminar counts by department
bins <- cut(seminar_counts$seminar_count,
            breaks = c(0, 1, 3, 5,7,11,17, 26),
            include.lowest = TRUE,
            right = TRUE)

# Add bin information to the seminar counts
seminar_counts <- seminar_counts %>%
  mutate(bin_category = bins)

# Summarize and print bins
bin_summary <- seminar_counts %>%
  group_by(bin_category) %>%
  summarize(department_count = n(),
            total_seminars = sum(seminar_count))

# Print the summary
print(bin_summary)
```

```
## # A tibble: 7 x 3
##   bin_category department_count total_seminars
##   <fct>                   <int>          <int>
## 1 [0,1]                     279            279
## 2 (1,3]                     117            281
## 3 (3,5]                      68            300
## 4 (5,7]                      43            278
## 5 (7,11]                     32            298
## 6 (11,17]                    23            323
## 7 (17,26]                     6            128
```

randomization within each bin

```
# Set seed for reproducibility
set.seed(114)

# Function to perform stratified randomization
stratified_randomize <- function(data, strata_col, group_col, num_groups) {
  data %>%
    group_by(across(all_of(strata_col))) %>%
    mutate(
      {{group_col}} := sample(rep(c("control", "treatment"), each = ceiling(n() / num_groups), length.ou
    ) %>%
    ungroup()
}

# Define number of groups
num_groups <- 2

# Apply stratified randomization
randomized_data <- stratified_randomize(seminar_counts, "bin_category", "condition", num_groups)

# Check the resulting distribution
randomized_distribution <- randomized_data %>%
  group_by(bin_category, condition) %>%
  summarize(department_count = n(), total_seminars = sum(seminar_count), .groups = 'drop')

print(randomized_distribution)
```

```
## # A tibble: 14 x 4
##    bin_category condition department_count total_seminars
##    <fct>        <chr>                <int>          <int>
##  1 [0,1]        control                140            140
##  2 [0,1]        treatment              139            139
##  3 (1,3]        control                 59            137
##  4 (1,3]        treatment               58            144
##  5 (3,5]        control                 34            152
##  6 (3,5]        treatment               34            148
##  7 (5,7]        control                 22            141
##  8 (5,7]        treatment               21            137
##  9 (7,11]       control                 16            146
## 10 (7,11]       treatment               16            152
## 11 (11,17]      control                 12            175
## 12 (11,17]      treatment               11            148
## 13 (17,26]      control                  3             60
## 14 (17,26]      treatment                3             68
```

chisq tests measuring whether randomization worked

seminars in each discipline

```
##            discipline
## condition   Chemistry Computer Science Mathematics Mechanical Engineering
##   control         136               84         481                     51
##   treatment       162               87         438                     49
##            discipline
## condition   Physics
##   control       199
##   treatment     200
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(merged_data$discipline, merged_data$condition)
## X-squared = 4.2566, df = 4, p-value = 0.3724
```

seminars in each department bin

```
## # A tibble: 14 x 2
##    department_count condition
##               <int> <chr>
##  1              140 control
##  2              139 treatment
##  3               59 control
##  4               58 treatment
##  5               34 control
##  6               34 treatment
##  7               22 control
##  8               21 treatment
##  9               16 control
## 10               16 treatment
## 11               12 control
## 12               11 treatment
## 13                3 control
## 14                3 treatment
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(randomized_distribution$department_count, randomized_distribution$condition)
## X-squared = 8, df = 10, p-value = 0.6288
```

```
## "x"
## "randomized_data.csv"
```

seminars in each bin

```
##
##  Pearson's Chi-squared test
##
## data:  table(department, condition)
## X-squared = 279, df = 278, p-value = 0.4718
##
##
##  Pearson's Chi-squared test
##
## data:  table(department, condition)
## X-squared = 117, df = 116, p-value = 0.4565
##
##
##  Pearson's Chi-squared test
##
## data:  table(department, condition)
## X-squared = 68, df = 67, p-value = 0.4429
##
##
##  Pearson's Chi-squared test
##
## data:  table(department, condition)
## X-squared = 43, df = 42, p-value = 0.4282
##
##
##  Pearson's Chi-squared test
##
## data:  table(department, condition)
## X-squared = 32, df = 31, p-value = 0.4167
##
##
##  Pearson's Chi-squared test
##
## data:  table(department, condition)
## X-squared = 23, df = 22, p-value = 0.4017
##
##
##  Pearson's Chi-squared test
##
## data:  table(department, condition)
## X-squared = 6, df = 5, p-value = 0.3062
```

```
## # A tibble: 2 x 2
##   condition  mean
##   <chr>     <dbl>
## 1 control    3.33
## 2 treatment  3.32
```