

1. What happens when pilgrims attend the Hajj pilgrimage to Mecca?

What happens when a diverse set of people are brought together toward a common purpose? Maybe it brings people together, or maybe instead it highlights the differences between groups. Clingingsmith, Khwaja and Kremer (2009) investigate the question. by asking Pakistani nationals to provide information about their views about people from other nations.

The random assignment and data is collected in the following way (detailed in the paper):

- Pakistani nationals apply for a chance to attend the Hajj at a domestic bank. Saudi Arabia agreed in the time period of the study (2006) to grant 150,000 visas.
- Of the 135,000 people who applied for a visa, 59% of those who applied were successful.
- The remainder of the visas were granted under a different allocation method that was not randomly assigned, and so this experiment cannot provide causal evidence about other allocation mechanisms.
- Among the 135,000 who apply, the authors conduct a random sample and survey about these respondents views about others.

Using the data collected by the authors, test, using randomization inference, whether there is a change in beliefs about others as a result of attending the Hajj.

- Use, as your primary outcome the **views** variable. This variable is a column-sum of each respondent's views toward members of other countries.
- Use, as your treatment feature **success**. This variable encodes whether the respondent successfully attended the Hajj.

```
library(data.table)
```

```
d <- fread("../data/clingingsmith_2009.csv")
d <- data.table(d)
d[success == "1", sum(success)]
```

```
## [1] 510
```

1. State the sharp-null hypothesis that you will be testing.

Fill this in with your null.

2. Using `data.table`, group the data by **success** and report whether views toward others are generally more positive among lottery winners or lottery non-winners. This answer should be of the form `d[, .(mean_views = ...), keyby = ...]` where you have filled in the `...` with the appropriate functions and variables.

```
hajj_group_mean <- d[, .(mean_views = mean(views)), keyby = .(success)] ##'fill this in' # the result
hajj_ate <- hajj_group_mean[, mean(mean_views), keyby = .(success)][, diff(V1)] ##'fill this in
```

```
class(hajj_group_mean)
```

```
## [1] "data.table" "data.frame"
```

```
hajj_group_mean
```

```
##      success mean_views
## 1:         0    1.868304
## 2:         1    2.343137
```

```
hajj_ate
```

```
## [1] 0.4748337
```

```
class(hajj_ate)
```

```
## [1] "numeric"
```

But is this a “meaningful” difference? Or, could a difference of this size have arisen from an “unlucky” randomization? Conduct 10,000 simulated random assignments under the sharp null hypothesis to find out. (Don’t just copy the code from the async, think about how to write this yourself.)

```
## do your work to conduct the randomiation inference here.
## as a reminder, RI will randomly permute / assign the treatment variable
## and recompute the test-statistic (i.e. the mean difference) under each permutation

randomize <- function(units_per_group) {
  assignment_vector <- rep(c(0, 1), each = units_per_group)
  sample(assignment_vector)
}

ri <- function(simulations) {
  res <- NA
  for(sim in 1:simulations) {
    res[sim] <- d[, .(group_mean = mean(views)), keyby = .(randomize(958/2))][, diff(group_mean)]
  }
  return(res)
}

# d <- d[, .(group_mean = mean(views)), keyby = .(randomize(958/2))]
# d
# randomize(958/2)
# dist_sharp_null <- ri(simulations = 1)
# dist_sharp_null

hajj_ri_distribution <- ri(simulations = 10000) #'fill this in' # this should be a numeric vector that
length(hajj_ri_distribution)
```

```
## [1] 10000
```

3. How many of the simulated random assignments generate an estimated ATE that is at least as large as the actual estimate of the ATE? Conduct your work in the code chunk below, saving the results into `hajj_count_larger`, but also support your coding with a narrative description. In that narrative description (and throughout), use R’s “inline code chunks” to write your answer consistent with each time you run your code.

```
hajj_count_larger <- sum(hajj_ri_distribution >= hajj_ate) #'fill this in' # length 1 numeric vector fr
```

There are only `hajj_count_larger` instances where simulated random assignments generate an estimated ATE that is at least as large as the actual estimate of the ATE

4. If there are `hajj_count_larger` randomizations that are larger than `hajj_ate`, what is the implied *one-tailed* p-value? Both write the code in the following chunk, and include a narrative description of the result following your code.

```
hajj_one_tailed_p_value <- 'fill this in' # length 1 numeric vector
```

5. Now, conduct a similar test, but for a two-sided p-value. You can either use two tests, one for larger than and another for smaller than; or, you can use an absolute value (`abs`). Both write the code in the following chunk, and include a narrative description of the result following your code.

```
hajj_two_tailed_p_value <- 'fill this in' # length 1 numeric vector
```