Assignment 5, Social Science Inquiry II (SOSC13200-W23-2)

Monday 2/6/23 at 5pm

Packages

```
library(ggplot2)
set.seed(60637)
```

Analysis is based on:

Pager, Devah. The mark of a criminal record. *American Journal of Sociology* 108, no. 5 (2003): 937-975.

1.

(1a)

Re-generate the data used in Pager (2003) based on a reading of the text. Create a data set that has the following variables:

- `black`, which is an indicator that is 1 if the respondent is black, and 0 otherwise.
- `record`, which is an indicator that is 1 if the respondent has a criminal record, and 0 otherwise.
- `call_back`, which is an indicator that is 1 if the respondent was called back, and 0 otherwise.

The data set should have one row for every observation, where an observation is an individual audit. I.e., the data set should have 700 rows, and 3 columns. *Note: total number of call backs for whites with criminal records could plausibly take on two values.*

```
pager_df <- data.frame(
  black = rep(c(0, 1), times = c(300, 400)),
  record = c(rep(c(0, 1), each = 150),
             rep(c(0, 1), each = 200)),
  call_back = c(
    # whites without criminal records
    rep(c(0, 1), times = c(99, 51)), # 150
    # whites with criminal records
    rep(c(0, 1), times = c(125, 25)), # 150: callbacks could be 25 or 26
    # blacks without criminal records
    rep(c(0, 1), times = c(172, 28)), # 200
    # blacks with criminal records
    rep(c(0, 1), times = c(190, 10)) # 200
  )
)

dim(pager_df)
```

```
## [1] 700   3
```

(1b)

Recreate Figure 6 in the paper.

You can install the package **`ggpattern`** *from https://github.com/coolbutuseless/ggpattern if you want to get diagonal stripes as in the original plot.*

```
pager_agg <- aggregate(call_back~black + record, data = pager_df, mean)
pager_agg$race <- factor(pager_agg$black,
                         levels = c(1, 0),
                         labels = c('Black', 'White'))
pager_agg$criminal_record <- factor(pager_agg$record,
                         levels = c(1, 0),
                         labels = c('Record', 'No Record'))

ggplot(pager_agg, aes(x = race, y = call_back, fill = criminal_record)) +
  geom_col(position = 'dodge') +
  geom_text(aes(label = round(call_back,2)*100),
            position = position_dodge(width = .9))+
  scale_fill_brewer(palette="BuPu") +
  # you can play around with color schemes here: https://colorbrewer2.org/
  theme_bw() +
  ylab('Percentage Called Back') +
  labs(caption = 'Fig. 6.') +
  theme(legend.position = '', axis.title.x=element_blank())
```

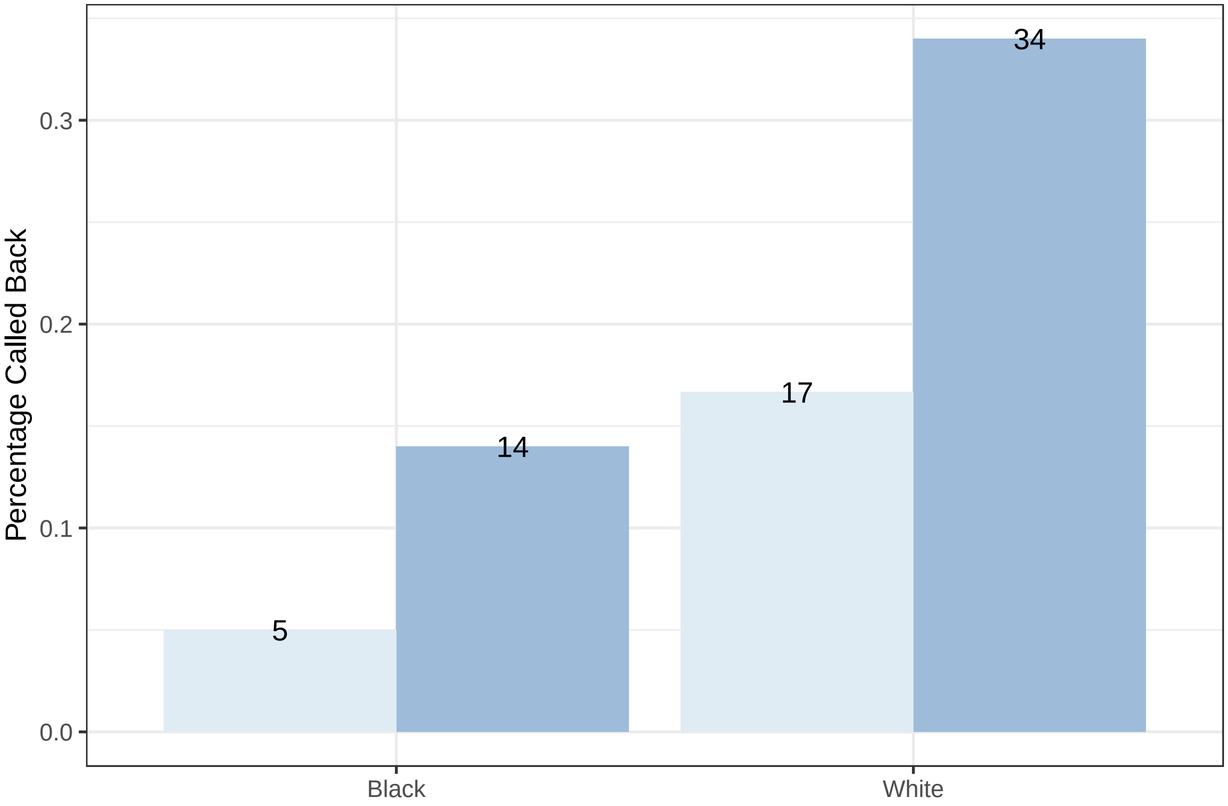

Fig. 6.

2. Randomization inference.

Pager reports that "The main effects of race and criminal record are statically significant (P <.01)."

(2a)

Create a new variable called `W`, which is a copy of `record`. Create a new variable called `Y` which is a copy of `call_back`. Report the number of audits assigned treatment and control if we consider having a criminal record to be the treatment condition.

```r
pager_df$W <- pager_df$record
pager_df$Y <- pager_df$call_back

table(pager_df$W)
```

```
##
##   0   1
## 350 350
```

(2b)

Get the difference-in-means estimate of the ATE on `Y`, and save the estimate as an object called `ate`. Report the value of your difference-in-means estimate of the ATE.

```r
(ate <- mean(pager_df$Y[which(pager_df$W == 1)]) - mean(pager_df$Y[which(pager_df$W == 0)]))
```

```
## [1] -0.1257143
```

(2c)

Create a new column called `newW` which resamples from `W` *without* replacement. Report the number of individuals assigned treatment and control under `newW`. Is it the same as under W?

```r
pager_df$newW <- sample(pager_df$W)

table(pager_df$newW)
```

```
##
##   0   1
## 350 350
```

(2d)

Calculate the difference-in-means estimate of the average treatment effect UNDER THE RE-SAMPLED TREATMENT, `newW`.

```r
(ate_new <- mean(pager_df$Y[which(pager_df$newW == 1)]) - mean(pager_df$Y[which(pager_df$newW == 0)]))
```

```
## [1] -0.03428571
```

(2e)

Write a randomization inference function that takes a data frame `df` as an argument, then:

- Creates a new column called `newW` which resamples from W.
- Calculates the difference in means estimate of the average treatment effect UNDER THE RE-SAMPLED TREATMENT, `newW`.
- Returns the value of estimated ATE.

Apply your randomization inference function to the pager data and report the estimated ATE.

```r
# randomization inference function
my_ri <- function(df){
  df_ri <- df
  df_ri$newW <- sample(df$W)
  Y1_ri <- df$Y[which(df_ri$newW == 1)]
  Y0_ri <- df$Y[which(df_ri$newW == 0)]
  ate_hat <- mean(Y1_ri)-mean(Y0_ri)
  return(ate_hat)
}
my_ri(pager_df)
```

```
## [1] 2.775558e-17
```

(2f)

Using `replicate()`, apply your function to the pager data 1000 times. Save the output but DO NOT print it out here.

```
dm <- replicate(1000, my_ri(pager_df))
```

(2g)

Report the portion of your results from question 2f that have a larger *absolute value* than the *absolute value* of the object `ate`.

```
(pval <- mean(abs(dm)>abs(ate)))
```

```
## [1] 0
```

(2h)

How do you interpret the p-value in 2g? Is your answer consistent with what Pager reports?

```
# your answer here
```

(XX) Extra credit

Worth 2 points.

Consider the function `gendist()` in the `ri` package. Look at the inputs, and what the function outputs. Using the toy data set from class (recreated below), write your own function that takes the same inputs and produces the same output.

If you have issues downloading the package because of your R version, you should be able to access a version following the below commands (uncommented).

```
# install.packages('remotes')
```

```
# library(remotes)
```

```
# install_github('cran/ri')
```

```
df <- data.frame(
  # our initial treatment vector
  W = c(1, 0, 0, 0, 0, 0, 1),
  # our initial response vector
  Y = c(15, 15, 20, 20, 10, 15, 30),
  # treatment assignment probability
  probs = rep(2/7, 7)
)
```