Assignment 5, Social Science Inquiry II (SOSC13200-W22-3)

Tiffanie Huang

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.*

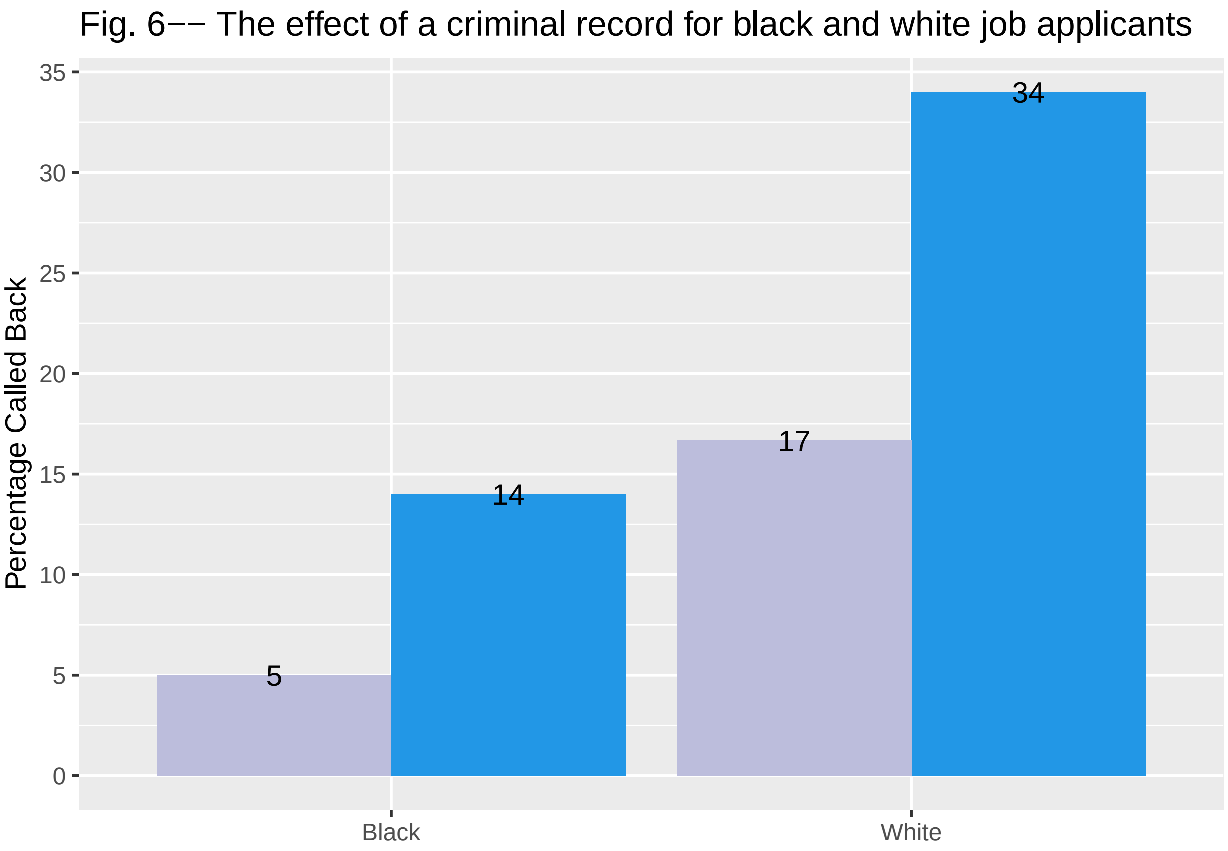
```
data <- data.frame(
  black=rep(c(0,1),times=c(300,400)),
  record=c(rep(c(0,1),each=150),rep(c(0,1), each = 200)),
  call_backs=c(rep(c(0,1),times=c(99,51)),
                rep(c(0,1),times=c(125,25)),
                rep(c(0,1),times=c(172,28)),
                rep(c(0,1),times=c(190,10))
  )
)
```

(1b)

Recreate Figure 6 in the paper.

```r
#make factors
agg <- aggregate(data$call_backs,by=list(data$black,data$record),mean)
colnames(agg) <- c("race","record","call_backs")
agg$race <- factor(agg$race,levels=c(1,0), labels=c("Black","White"))
agg$record <- factor(agg$record,levels=c(1,0),labels = c("record","no record"))

#figure 6 barplot
library(ggpattern)
ggplot(agg,aes(x=race,y=call_backs,fill=record))+
  geom_col(position="dodge")+
  geom_text(aes(label=signif(100*call_backs,2)),
    position = position_dodge(width = .9))+
  scale_fill_manual(values=c("#bcbddc","756bb1"))+
  scale_x_discrete(labels=c("Black","White"))+
  ylab('Percentage Called Back')+
  ggtitle("Fig. 6-- The effect of a criminal record for black and white job applicants")+
  scale_y_continuous(breaks=seq(0,0.4,0.05),labels=seq(0,40,5))+
  theme(legend.position="",axis.title.x=element_blank())
```



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.

```
#new variables
data$W <- data$record
data$Y <- data$call_backs

#number audits assigned T and C if record is T
sum(data$W) #350 audits assigned treatment
```

```
## [1] 350
```

```
sum(data$W==0) #350 audits assigned control
```

```
## [1] 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.

```
y_w1 <- mean(data$Y[which(data$W==1)])
y_w0 <- mean(data$Y[which(data$W==0)])
ate <- y_w1-y_w0
print(ate) #ATE estimate on Y under W: -0.1257143
```

```
## [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?

```
data$newW <- sample(data$W)
sum(data$newW) #350 individuals assigned treatment
```

```
## [1] 350
```

```
sum(data$newW==0) #350 individuals assigned control
```

```
## [1] 350
```

```
#It is the same because we are just resampling from the values in W and will generate a random permutat
```

(2d)

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

```
y_neww1 <- mean(data$Y[which(data$newW==1)])
y_neww0 <- mean(data$Y[which(data$newW==0)])
ate_new <- y_neww1-y_neww0
print(ate_new) #ATE estimate on Y under newW: 0.02285714
```

```
## [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.

```
randinf <- function(df){
  df1 <- df
  df1$newW <- sample(df$W)
  y1 <- mean(df1$Y[which(df1$newW==1)])
  y0 <- mean(df1$Y[which(df1$newW==0)])
  return(y1-y0)
}
```

(2f)

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

```
rep1k <- replicate(1000, randinf(data))
```

(2g)

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

```
mean(abs(rep1k)>abs(ate)) #no values greater --> 0
```

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

(2h)

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

```
#answer:
```

H0: treatment effect of criminal record on callbacks = 0 for all i in our population HA: treatment effect of criminal record on callbacks =/= 0 for all i in our population I would interpret the p-value as statistically significant, as 0 is lower than our significance level, 0.1. This means that our parameter value fell outside the estimated 99% confidence interval. Under the assumption that our null hypothesis is true, if we took many many random samples we would observe an ATE value as extreme as 0.1257143 0% of the time, so it is very unlikely and the discrepancy in callbacks between those with a criminal record/those without is significant. We reject H0. My interpretation is consistent with what Pager reports as a large and significant effect. # (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')`

```
#from documentation:
#inputs: Ys, perms, X, Ypre, prob, HT
# - Ys: list consisting of two N-length numeric vectors labeled Y0 and Y1, as output by genouts()
# - perms: N-by-r permutation matrix
# - X: N-by-k numeric matrix of covariates for regression adjustment
# - Ypre: numeric vector of length N, pretreatment measure of the outocme variable for difference estim
# - prob: numeric vector within the (0,1) interval of length N, probability of treatment assignment, as
# when HT= TRUE, invokes horvitz-thompson (difference-in-totals) estimator. When HT = FALSE, invokes th
#output: r-length vector of estimated ATEs
```

```
#data.frame
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)
)

#under sharp null of no effect:
y1 <- df$Y[which(df$W == 1)]
y0 <- df$Y[which(df$W == 0)]

#add hypothetical data
df <-  cbind(df, Y0 = df$Y, Y1 = df$Y)

#Ys input: list
```

```r
Ys_arg <- list(Y0 = df$Y0, Y1 = df$Y1)

#permutations function (?)
perm <- function(w_vec){
  n_treat <- sum(w_vec) #number units assigned treatment
  n_tot <- length(w_vec) #total units
  return(combn(n_tot,n_treat,function(x) replace(numeric(n_tot),x,1)))
}
perm_t <- perm(df$W)

#ate sampling dist function
ate_sampdist <- function(Ys, perms){ #leaving out prob
  ate_dm <- numeric(dim(perms)[2])
  for(i in 1:(dim(perms)[2])){
    ate_dm[i] <- mean(Ys$Y0[which(perms[,i]==1)])-mean(Ys$Y0[which(perms[,i]==0)])
  }
  return(ate_dm)
}

#distribution with values from df
dm <- ate_sampdist(Ys_arg,perm_t)

#check with original package function
library('ri')
perms1 <- genperms(df$W)
dm1 <- gendist(Ys_arg,
              perms1,
              prob=df$probs)

(identical(dm,dm1)) #true
```

```
## [1] TRUE
```