Homework 2

Yuya Furusawa (29-186036)

2019/07/03

```r
# read data
data <- read.csv("corruption.csv");
```

Question 1

(a)

In this design, forcing variable is the population of each village, and outcome variable is whether any irregularity was found or not.

(b)

We can't simply compare all "treated" and "non-treated" villages because we have multiple cutoffs (or treatment is not categorical). We cannot clearly distinguish "treated" and "non-treated". In addition, population is not random because some village has tendency to have large population (e.g. village beside the river may have large population than village in the forest because it plays a role as trading city). Also, FPM cutoff may not be random if the amount of transfer is not totally determined by the observed characterristics of the village.

(c)

Estimand of this design is the causal relationship between additional transfer on corruption. But, in this RD design,

$$\tau = E[Y_i(1) - Y_i(0)|X_i = c]$$

where c is the cutoff, 1 indicates the treatment(higer FPM transfer), 0 indicates the control(lower FPM transfer).

The weakness of this design is the fact that FPM transfer is not deterministically decided.

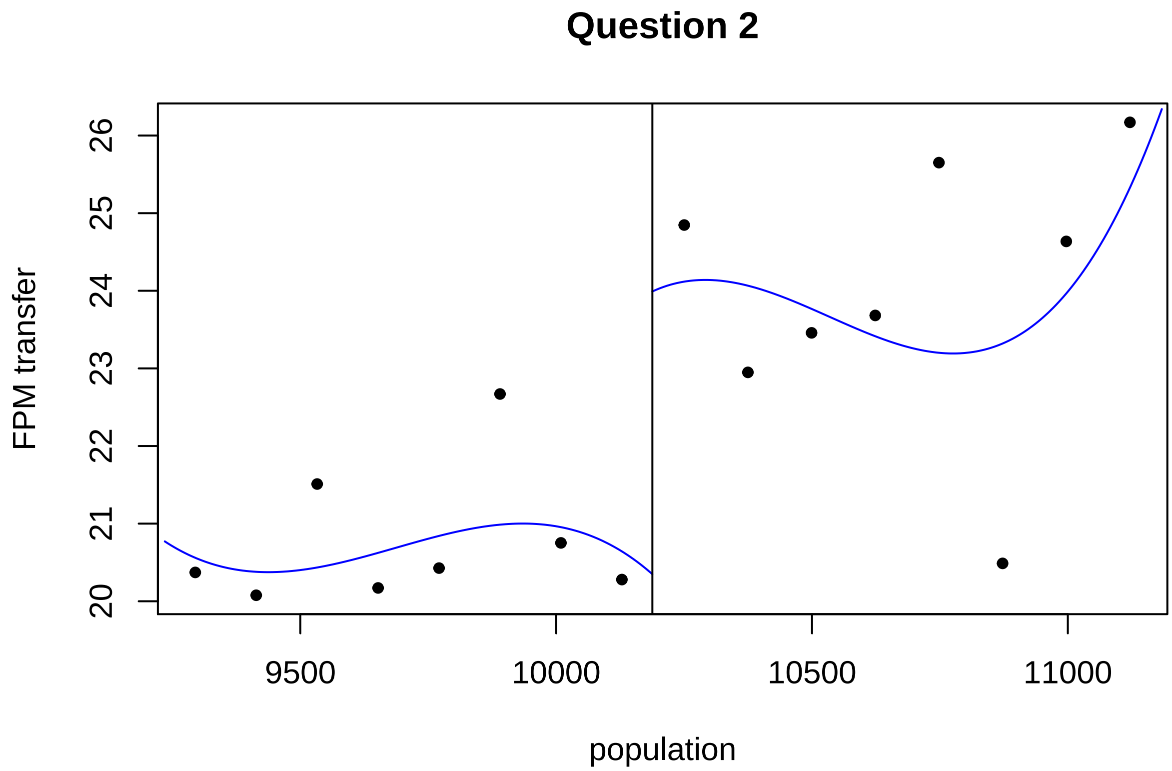
External validity is also problematic. The result cannot be inmediately applied to other countries or other situations.

Question 2

(a)

```r
cutoff <- 10188
data_Q2 <- filter(data, pop >= cutoff - 1000, pop <= cutoff + 1000)
rdplot(y = data_Q2$fpm, x = data_Q2$pop,
       x.label = "population", y.label = "FPM transfer",
       c = cutoff, title="Question 2", p=3)
```

Question 2



(b)

From the RD plot, we can see that there exists a discontinuity of FPM transfer at the cutoff. Treatment assignment mechanism here is deterministic from RD plot, but in fact assignment seems to be random. The estimand in the follwing analysis is

$$\tau = \lim_{\epsilon \to 0} E[Y_i | X_i = c_i + \epsilon] - \lim_{\epsilon \to 0} E[Y_i | X_i = c_i - \epsilon]$$

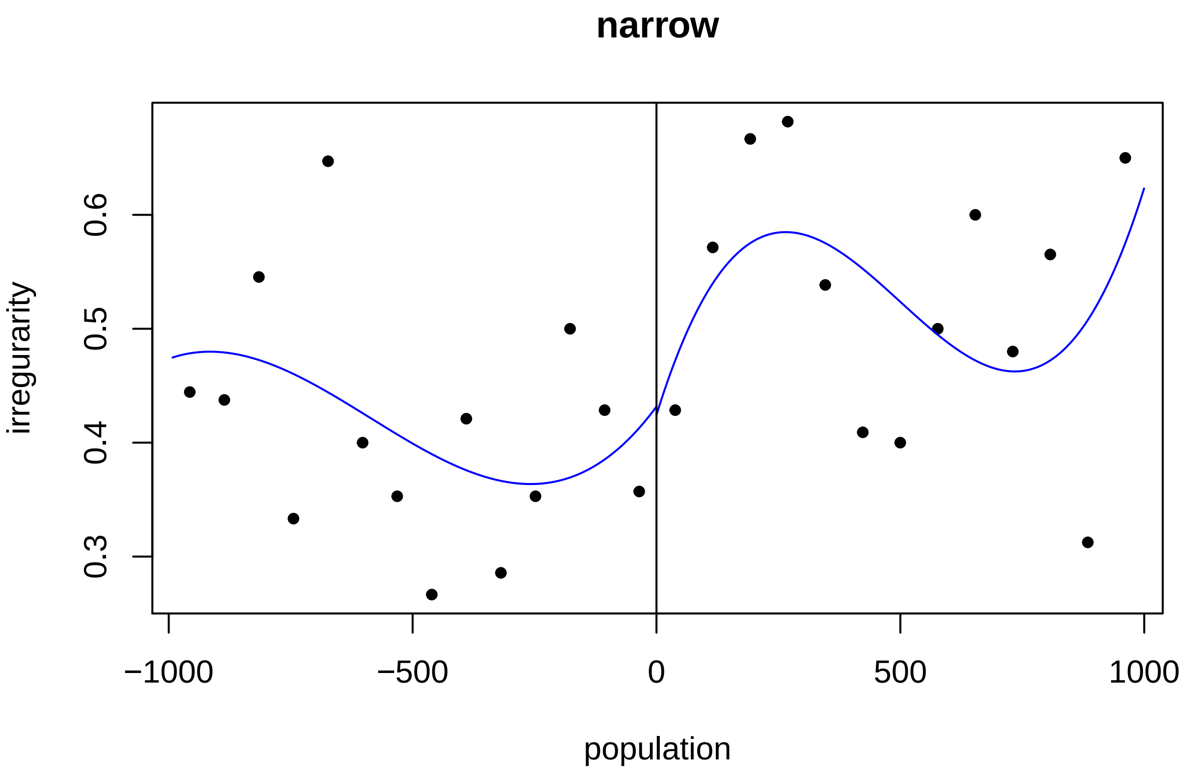
where c_i is the i's corresponding cutoff.

Question 3

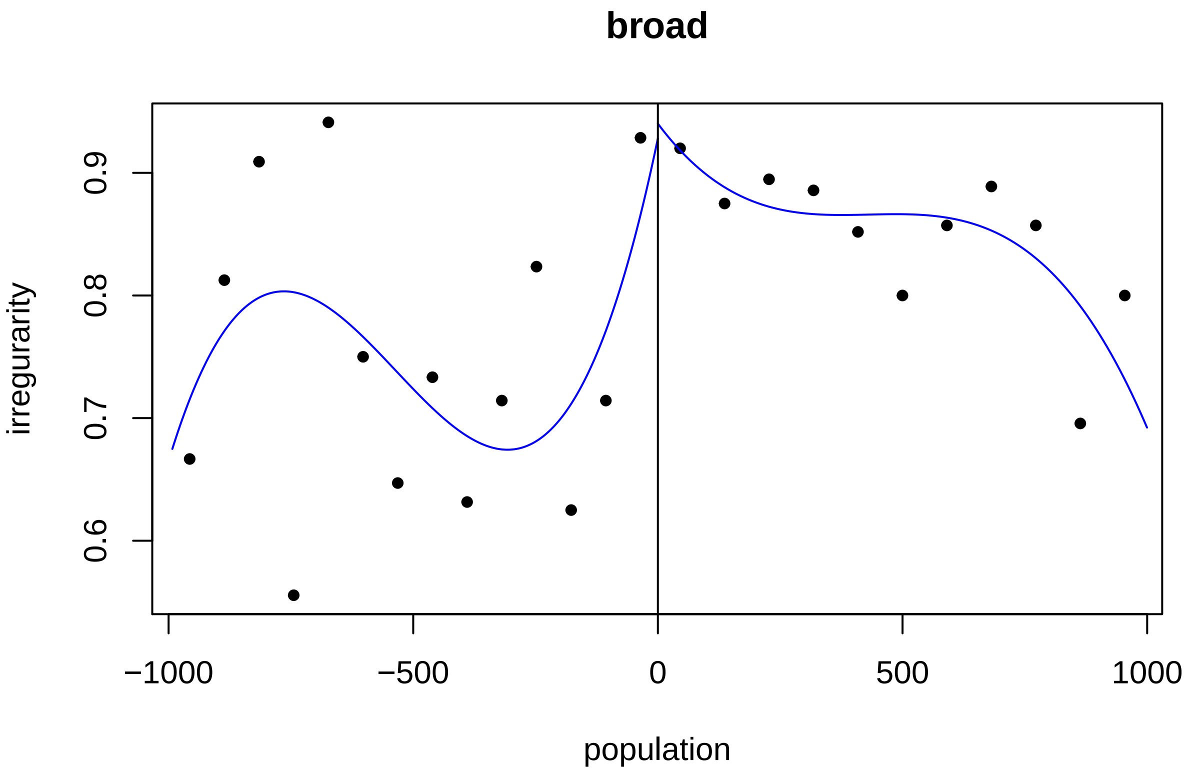
(a)

```r
# create standardized data
cutoffs <- c(10188, 13584, 16980, 23772, 30564, 37356, 44148)
cdata <- vector("list", length(cutoffs))
for (i in 1:length(cutoffs)){
  cdata[[i]] <- filter(data, pop >= cutoffs[i] - 1000, pop <= cutoffs[i] + 1000)
  cdata[[i]]$pop <- cdata[[i]]$pop - cutoffs[i]
}
stan_data <- rbind(cdata[[1]], cdata[[2]])
for (i in 3:length(cutoffs)){
  stan_data <- rbind(stan_data, cdata[[i]])
};
```

```r
# RD plot : narrow
rdplot(y=stan_data$narrow, x=stan_data$pop,
       x.label = "population", y.label = "irregurarity",
       title="narrow", p=3)
```

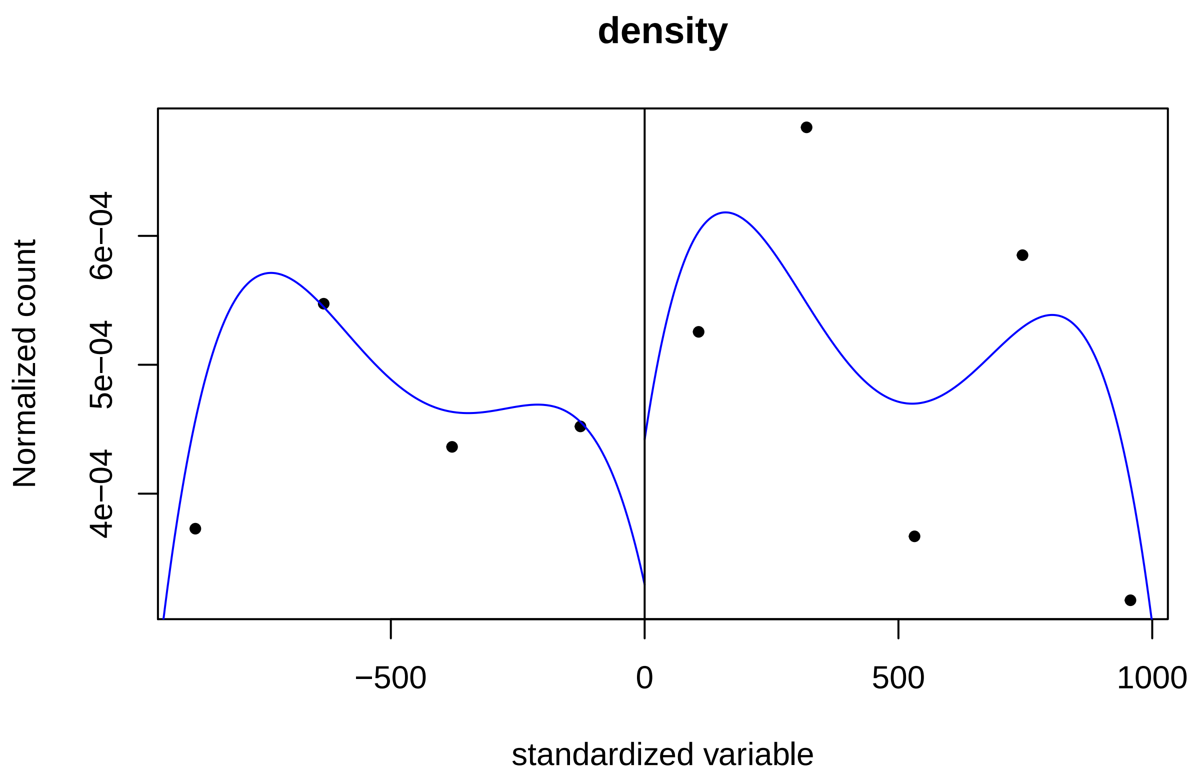



```r
# RD plot : broad
rdplot(y=stan_data$broad, x=stan_data$pop,
       x.label = "population", y.label = "irregurarity",
       title="broad", p=3)
```



(b)

```
# Compute Density
n <- length(stan_data$pop)
b <- 2 * sd(stan_data$pop) / sqrt(n)
Xdisc <- floor(stan_data$pop/b) * b + b/2
m <- (max(Xdisc) - min(Xdisc)) / min(diff(sort(unique(Xdisc)))) + 1
Xj <- min(Xdisc) + (0:m * min(diff(sort(unique(Xdisc)))))
Yj <- table(factor(round(Xdisc,3), levels=round(Xj,3))) / (n*b)
rdplot(y=Yj, x=Xj, title = "density",
       x.label = "standardized variable", y.label = "Normalized count")
```



This implies that around the cutoff, population is not randomly distributed.

(c)

```
# RD Regression : broad
fit <- rdrobust(y=stan_data$broad, x=stan_data$pop, p=1, kernel="tri")
res_tab <- cbind(fit$coef, fit$se)
rownames(res_tab) <- rownames(fit$coef)
colnames(res_tab) <- c("estimate", "se")
print(res_tab)
```

```
##                  estimate        se
## Conventional   -0.01303467 0.1159968
## Bias-Corrected -0.05768286 0.1159968
## Robust         -0.05768286 0.1330161
```

```
# RD Regression : narrow
fit <- rdrobust(y=stan_data$narrow, x=stan_data$pop, p=1, kernel="tri")
res_tab <- cbind(fit$coef, fit$se)
rownames(res_tab) <- rownames(fit$coef)
colnames(res_tab) <- c("estimate", "se")
print(res_tab)
```

```
##                  estimate        se
## Conventional   0.05170782 0.2322657
## Bias-Corrected 0.05420390 0.2322657
## Robust         0.05420390 0.2871732
```

These results indicates that we cannot reject the hypothesis that there exists a discontinuous effect on irregularity, and corruption is not caused by large FPM transfer.

Question 4

(a)

$$
\begin{aligned}
E[Y_i|\tilde{X}_i = \epsilon] &= \sum_{c\in\mathcal{C}} E[Y_{ic}|\tilde{X}_i = \epsilon, C_i = c]P(C_i = c|\tilde{X}_i = \epsilon) \\
&= \sum_{c\in\mathcal{C}} E[Y_{ic}|X_i = c+\epsilon, C_i = c]P(C_i = c|\tilde{X}_i = \epsilon) \\
&= \sum_{c\in\mathcal{C}} E[Y_{ic}(1)|X_i = c+\epsilon, C_i = c]P(C_i = c|\tilde{X}_i = \epsilon)
\end{aligned}
$$

Last equality holds because $\epsilon > 0$.

(b)

$$
\begin{aligned}
P(C_i = c|\tilde{X}_i = x) &= \frac{P(C_i = c \wedge \tilde{X}_i = x)}{P(\tilde{X}_i = x)} \\
&= \frac{P(C_i = c \wedge \tilde{X}_i = x)}{\sum_{c'\in\mathcal{C}} P(C_i = c', \tilde{X}_i = x + c')} \\
&= \frac{f_{X|C}(c+x|c)P(C_i = c)}{\sum_{c'\in\mathcal{C}} f_{X|C}(c+x|c')P(C_i = c')}
\end{aligned}
$$

Second equality holds by Bayes rule.

(c)

$$
\begin{aligned}
\lim_{\epsilon\downarrow 0} E[Y_i|\tilde{X}_i = \epsilon] &= \lim_{\epsilon\downarrow 0} \sum_{c\in\mathcal{C}} E[Y_{ic}(1)|X_i = c+\epsilon, C_i = c]P(C_i = c|\tilde{X}_i = \epsilon) \\
&= \lim_{\epsilon\downarrow 0} \sum_{c\in\mathcal{C}} E[Y_{ic}(1)|X_i = c+\epsilon, C_i = c]\frac{f_{X|C}(c+\epsilon|c)P(C_i = c)}{\sum_{c'\in\mathcal{C}} f_{X|C}(c+\epsilon|c')P(C_i = c')} \\
&= \sum_{c\in\mathcal{C}} E[Y_{ic}(1)|X_i = c, C_i = c]\frac{f_{X|C}(c|c)P(C_i = c)}{\sum_{c'\in\mathcal{C}} f_{X|C}(c|c')P(C_i = c')} \\
&= \sum_{c\in\mathcal{C}} E[Y_{ic}(1)|X_i = c, C_i = c]w_c
\end{aligned}
$$

First equation comes from (a). Second equation comes from (b). Last equation comes from the definition of w_c.

Similary, we can show, for any $\epsilon < 0$,

$$
E[Y_i|\tilde{X}_i = \epsilon] = \sum_{c\in\mathcal{C}} E[Y_{ic}(0)|X_i = c+\epsilon, C_i = c]P(C_i = c|\tilde{X}_i = \epsilon)
$$

From this and (b), we have

$$
\lim_{\epsilon\uparrow 0} E[Y_i|\tilde{X}_i = \epsilon] = \sum_{c\in\mathcal{C}} E[Y_{ic}(0)|X_i = c, C_i = c]w_c
$$

Thus, we have

$$
\begin{aligned}
\tau &= \lim_{\epsilon \downarrow 0} E[Y_i|\tilde{X}_i = \epsilon] - \lim_{\epsilon \uparrow 0} E[Y_i|\tilde{X}_i = \epsilon] \\
&= \sum_{c \in \mathcal{C}} E[Y_{ic}(1)|X_i = c, C_i = c]w_c - \sum_{c \in \mathcal{C}} E[Y_{ic}(0)|X_i = c, C_i = c]w_c \\
&= \sum_{c \in \mathcal{C}} E[Y_{ic}(1) - Y_{ic}(0)|X_i = c, C_i = c]w_c
\end{aligned}
$$

Finally, we can get desired result.

Question 5

(a)

Estimand in a fuzzy RD is

$$\tau = E[Y_i(1, T_i(1)) - Y_i(0, T_i(0))|X_i = c]$$

where c is the coorreponding cutoff, $T_i(z)$ is the FPM transfer(z is 1 if population is above cutoff, 0 otherwise), $Y_i(z, T_i(z))$ is potential outcome for z.

We need the following assumptions.

1. Monotonicity, village i gets more FMP transfer when the population is above cutoff than that when it is below cutoff.
2. Exclusion restriction, irregurarity level is same if FMP transfer is same.
3. $E[Y_i|\tilde{X}_i = x]$ and $E[D_i|\tilde{X}_i = x]$ are continuous in x.

(b)

(1)

From Question 4, we have

$$\tau_{wald} = \frac{\sum_{c \in \mathcal{C}} E[Y_{ic}(1) - Y_{ic}(0)|X_i = c, C_i = c]w_c}{\sum_{c \in \mathcal{C}} E[D_{ic}(1) - D_{ic}(0)|X_i = c, C_i = c]w_c}$$

Therefore,

$$\hat{\tau}_{wald} = ...$$

(2)

```r
# fuzzy RD : broad
fit_frd <- rdrobust(y=stan_data$broad, x=stan_data$pop, fuzzy=stan_data$fpm, p=1, kernel="tri")
summary(fit_frd)
```

```
## Call: rdrobust
##
## Number of Obs.                   486
## BW type                        mserd
## Kernel                    Triangular
## VCE method                        NN
##
## Number of Obs.                  228          258
## Eff. Number of Obs.              53           64
## Order est. (p)                    1            1
## Order bias  (p)                   2            2
## BW est. (h)                 235.490      235.490
## BW bias (b)                 397.476      397.476
## rho (h/b)                     0.592        0.592
##
## =============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
## =============================================================================
##   Conventional     0.018     0.027     0.677     0.499    [-0.034 , 0.070]
##         Robust         -         -     0.597     0.550    [-0.044 , 0.082]
## =============================================================================
```

Therefore, $\hat{Var}(\hat{\tau}_{wald}) = (0.027)^2 \approx 0.0007$

```
# fuzzy RD : narrow
fit_frd <- rdrobust(y=stan_data$narrow, x=stan_data$pop, fuzzy=stan_data$fpm, p=1, kernel="tri")
summary(fit_frd)
```

```
## Call: rdrobust
##
## Number of Obs.                   486
## BW type                        mserd
## Kernel                    Triangular
## VCE method                        NN
##
## Number of Obs.                  228          258
## Eff. Number of Obs.              58           68
## Order est. (p)                    1            1
## Order bias  (p)                   2            2
## BW est. (h)                 261.804      261.804
## BW bias (b)                 419.719      419.719
## rho (h/b)                     0.624        0.624
##
## =============================================================================
##         Method     Coef. Std. Err.         z     P>|z|      [ 95% C.I. ]
## =============================================================================
##   Conventional    -0.005     0.044    -0.117     0.907    [-0.092 , 0.081]
##         Robust         -         -    -0.008     0.993    [-0.107 , 0.106]
## =============================================================================
```

Therefore, $\hat{Var}(\hat{\tau}_{wald}) = (0.044)^2 \approx 0.0019$

(3)

These results indicate that we cannot reject the hypothesis $\hat{\tau}_{eald} = 0$.

(c)

```
## Warning: The `printer` argument is deprecated as of rlang 0.3.0.
## This warning is displayed once per session.
```

```r
# 2SLS : broad
result_broad <- lm(stan_data$broad ~ D_hat * stan_data$pop)
summary(result_broad)
```

```
##
## Call:
## lm(formula = stan_data$broad ~ D_hat * stan_data$pop)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.9208  0.0920  0.1759  0.2409  0.2732
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.161e+00  1.077e+00  -2.007  0.04528 *
## D_hat                9.411e-02  3.390e-02   2.776  0.00572 **
## stan_data$pop       -1.775e-04  1.020e-03  -0.174  0.86191
## D_hat:stan_data$pop -7.590e-07  3.142e-05  -0.024  0.98074
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.3953 on 482 degrees of freedom
## Multiple R-squared:  0.02252,    Adjusted R-squared:  0.01644
## F-statistic: 3.702 on 3 and 482 DF,  p-value: 0.01177
```

```r
# 2SLS : narrow
result_narrow <- lm(stan_data$narrow ~ D_hat * stan_data$pop)
summary(result_narrow)
```

```
##
## Call:
## lm(formula = stan_data$narrow ~ D_hat * stan_data$pop)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -0.5466 -0.5097 -0.3669  0.4824  0.6368
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)         -2.354e+00  1.356e+00  -1.736   0.0831 .
## D_hat                8.851e-02  4.269e-02   2.073   0.0387 *
## stan_data$pop       -1.646e-03  1.284e-03  -1.282   0.2004
## D_hat:stan_data$pop  4.701e-05  3.956e-05   1.188   0.2353
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.4977 on 482 degrees of freedom
## Multiple R-squared:  0.01498,    Adjusted R-squared:  0.008853
## F-statistic: 2.444 on 3 and 482 DF,  p-value: 0.06336
```

These results show that τ is greater than zero, that is, there is a positive effect of FPM transfer on corruption.

This is an evidence of