Assignment 4
Econometrics I

Universidad Carlos III de Madrid

Gabriel Merlo

```
# install packages (if missing)
list_packages <- c("aod", "dplyr", "glmnet", "quantreg", "tidyr")
new_packages <- list_packages[!(list_packages %in% installed.packages()[,
    "Package"])]
if (length(new_packages)) install.packages(new_packages)

# Load packages
sapply(list_packages, require, character.only = TRUE)
```

Part 1

(a) Read the data and estimate the ATE using the standard difference of sample means and a linear regression using as controls X.

```
# Load data
penn <- as.data.frame(read.table("penn_jae.dat", header = TRUE))

# Keep control group, and treatment group 4
penn4 <- penn %>% filter(tg == 0 | tg == 4)

# Recode treatment variable
penn4$tg <- recode(penn4$tg, `4` = 1L)

# Control variables
x <- c("female", "black", "othrace", "dep", "q2", "q3", "q4",
    "q5", "q6", "agelt35", "agegt54", "durable", "lusd", "husd")

# Log transformation of dependent variable
penn4$l_inuidur1 <- log(penn4$inuidur1)

# ATE from difference of sample means
penn4_summary <- penn4 %>% group_by(tg) %>% summarize(mean = mean(l_inuidur1),
    sd = sd(l_inuidur1), n = n())
ate_diff_mean <- penn4_summary %>% dplyr::select(tg, mean) %>%
    spread(tg, mean) %>% summarize(diff = `1` - `0`)
(ate_diff_mean <- as.numeric(ate_diff_mean))
```

```
## [1] -0.08545541
```

```
# ATE from linear regression with controls
ate_ols <- lm(as.formula(paste0("l_inuidur1 ~ tg + ", paste0(x,
```

```
    collapse = " + "))), data = penn4)
summary(ate_ols)
```

```
##
## Call:
## lm(formula = as.formula(paste0("l_inuidur1 ~ tg + ", paste0(x,
##     collapse = " + "))), data = penn4)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -2.6195 -0.9966  0.3133  1.0400  2.0883
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.178441   0.159001  13.701  < 2e-16 ***
## tg          -0.071659   0.035460  -2.021 0.043349 *
## female       0.125810   0.034780   3.617 0.000301 ***
## black       -0.293971   0.052967  -5.550 3.00e-08 ***
## othrace     -0.470387   0.198281  -2.372 0.017713 *
## dep          0.045993   0.022535   2.041 0.041308 *
## q2           0.073251   0.156807   0.467 0.640420
## q3          -0.039092   0.156454  -0.250 0.802704
## q4          -0.055596   0.156534  -0.355 0.722478
## q5          -0.144996   0.155854  -0.930 0.352243
## q6           0.003035   0.166438   0.018 0.985453
## agelt35     -0.162642   0.036960  -4.401 1.10e-05 ***
## agegt54      0.227801   0.058892   3.868 0.000111 ***
## durable      0.126551   0.048142   2.629 0.008597 **
## lusd        -0.175602   0.040972  -4.286 1.85e-05 ***
## husd        -0.105557   0.044893  -2.351 0.018746 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.199 on 5083 degrees of freedom
## Multiple R-squared:  0.02912,    Adjusted R-squared:  0.02625
## F-statistic: 10.16 on 15 and 5083 DF,  p-value: < 2.2e-16
```

```
# Checking balance of covariates
penn4 %>% group_by(tg) %>% dplyr::select(x) %>% summarise_all(funs(mean(.)))
```

```
## # A tibble: 2 x 15
##      tg female black othrace   dep    q2    q3    q4    q5     q6 agelt35
##   <int>  <dbl> <dbl>   <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>  <dbl>   <dbl>
## 1     0  0.405 0.121 0.00566 0.438 0.205 0.237 0.220 0.256 0.0698   0.535
## 2     1  0.402 0.123 0.0103  0.442 0.202 0.233 0.237 0.264 0.0499   0.564
## # ... with 4 more variables: agegt54 <dbl>, durable <dbl>, lusd <dbl>,
## #   husd <dbl>
```

The difference in the mean of log of duration of unemployment between treated and control groups is -0.085. This implies that those that receive the treatment spend less time being unemployed than those who don't get the treatment.

Controlling by observable characteristics of the individuals, the log of duration of unemployment is 0.072 smaller for the individuals that receive the treatment. Once we control by our vector of observables `x`, the

effect of the treatment is 0.014 smaller than when comparing using the difference of means (without controls).

Ideally, randomization should balance the distribution of covariates among treated and untreated. One way to check if this is true is by calculating the sample mean difference in covariates between treatment and control groups. The balance is in general quite good but some characteristics are still not very well balanced (we could test if the differences are significant). This can explain the difference in the ATE by the two previous methods.

(b) One way to evaluate if the randomization is sucessful is to test the significance of θ_0 in a Probit specification of the propensity score $p(x) = \Phi(x'\theta_0)$. Run such a test and interpret the results. Discuss the type of test, critical value, etc.

```
# Probit model estimation
penn4_ps <- glm(as.formula(paste0("tg ~", paste0(x, collapse = " + "))),
    family = binomial(link = "probit"), data = penn4)
summary(penn4_ps)
```

```
##
## Call:
## glm(formula = as.formula(paste0("tg ~", paste0(x, collapse = " + "))),
##     family = binomial(link = "probit"), data = penn4)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2447  -0.9291  -0.8823   1.4296   1.7155
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -0.36877    0.16948  -2.176   0.0296 *
## female      -0.01550    0.03753  -0.413   0.6797
## black        0.02176    0.05709   0.381   0.7031
## othrace      0.38716    0.20762   1.865   0.0622 .
## dep          0.01345    0.02430   0.553   0.5801
## q2          -0.12192    0.16760  -0.727   0.4670
## q3          -0.12428    0.16721  -0.743   0.4573
## q4          -0.06672    0.16724  -0.399   0.6900
## q5          -0.09220    0.16653  -0.554   0.5798
## q6          -0.31679    0.17919  -1.768   0.0771 .
## agelt35      0.09104    0.03999   2.277   0.0228 *
## agegt54      0.06200    0.06364   0.974   0.3299
## durable     -0.04423    0.05217  -0.848   0.3965
## lusd         0.08107    0.04402   1.842   0.0655 .
## husd        -0.01044    0.04864  -0.215   0.8300
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 6552.2  on 5098  degrees of freedom
## Residual deviance: 6529.4  on 5084  degrees of freedom
## AIC: 6559.4
##
```

```
## Number of Fisher Scoring iterations: 4
```

```
# Extract coefficients
penn4_ps_coef <- coef(penn4_ps)

# Extract variance-covariance matrix
penn4_ps_sigma <- vcov(penn4_ps)

# Testing joint significance (Wald test)
penn4_ps_wt <- wald.test(Sigma = penn4_ps_sigma, b = penn4_ps_coef,
    Terms = 2:15)
penn4_ps_wt
```

```
## Wald test:
## ----------
##
## Chi-squared test:
## X2 = 22.6, df = 14, P(> X2) = 0.067
```

To test if randomization was done correctly we can test the joint significance of the covariates on the probability of being treated. If the assignment of the treatment was truly random, no observable characteristic should be significant. To that end, we can apply a Wald test to the vector of coefficients θ_0.

The test result indicates that we can not reject the null of all coefficients being simultaneously equal to zero for a significance level of 0.05. So far, the evidence is not strong against the success of the randomization process.

(c) Estimate the ATE by DML based on Lasso.

```
set.seed(123)

# Double Debiased Machine Learning function with lasso (DML)
b_DML <- function(Y, X, D) {
    DML1 <- cv.glmnet(X, Y, alpha = 1)
    yhat <- predict(DML1, X)
    res1 <- Y - yhat
    DML2 <- cv.glmnet(X, D, alpha = 1)
    Dhat <- predict(DML2, X)
    res2 <- D - Dhat
    DML <- lm(res1 ~ 0 + res2)
    DML_coef <- as.numeric(coef(DML))
    DML_se <- as.numeric(summary(DML)$coefficients["res2", "Std. Error"])
    b_DML <- c(DML_coef, DML_se)
    return(b_DML)
}

# Calculating ATE using DML
ate_dml <- b_DML(penn4$l_inuidur1, as.matrix(penn4[, x]), penn4$tg)
ate_dml
```

```
## [1] -0.08266116  0.03564880
```

The ATE obtained using DML technique with lasso is -0.083.

(d) Construct 95% CI for the ATE using the previous estimates.

```
# CI ATE from difference in means

# Lower
ate_diff_mean_ci_l <- ate_diff_mean - qnorm(0.975) * sqrt(penn4_summary$sd[1]^2/penn4_summary$n[1] +
    penn4_summary$sd[2]^2/penn4_summary$n[2])
# Upper
ate_diff_mean_ci_u <- ate_diff_mean + qnorm(0.975) * sqrt(penn4_summary$sd[2]^2/penn4_summary$n[2] +
    penn4_summary$sd[2]^2/penn4_summary$n[2])

ate_diff_mean_ci <- c(ate_diff_mean_ci_l, ate_diff_mean_ci_u)
ate_diff_mean_ci
```

```
## [1] -0.155734324 -0.004801921
```

```
# CI ATE from OLS

# Lower
ate_ols_ci_l <- as.numeric(ate_ols$coefficients[2] - qnorm(0.975) *
    summary(ate_ols)$coefficients["tg", "Std. Error"])
# Upper
ate_ols_ci_u <- as.numeric(ate_ols$coefficients[2] + qnorm(0.975) *
    summary(ate_ols)$coefficients["tg", "Std. Error"])

ate_ols_ci <- c(ate_ols_ci_l, ate_ols_ci_u)
ate_ols_ci
```

```
## [1] -0.141159224 -0.002158592
```

```
# CI ATE from DML

# Lower
ate_dml_ci_l <- ate_dml[1] - qnorm(0.975) * ate_dml[2]
# Upper
ate_dml_ci_u <- ate_dml[1] + qnorm(0.975) * ate_dml[2]

ate_dml_ci <- c(ate_dml_ci_l, ate_dml_ci_u)
ate_dml_ci
```

```
## [1] -0.15253153 -0.01279079
```