Gov 2001: Problem Set 7 Answers

Due Wednesday, April 13th at 6pm

Instructions

You should submit your answers and R code to the problems below using the Quizzes section on Canvas.

Problem 1 - Estimating an ATE

In this Problem Set, we'll look at comparing matching methods for estimating a causal effect to an experimental benchmark. The classic comparison of the two is the Lalonde (1986) evaluation of the National Supported Work (NSW) job training program, which examines the effect of a job training program on individual's wages in two settings: an experiment where individuals were randomly assigned to the job program, and in a hypothetical observational setting, where we observed individuals assigned to the program and a series of potential controls drawn from the overall US population. While Lalonde shows that nonexperimental approaches fail to meet the experimental benchmark or are highly sensitive to modeling choices, subsequent work by Dehejia and Wahba (1999) shows that matching estimators can help over-come this model sensitivity.

We're going to replicate some of these analyses here.

The datasets should be pulled from the `causalsens` R package (we're not going to use any functions in this package, only the datasets). Install `causalsens` from CRAN and load it into your workspace. Load the **experimental** benchmark data using the command `data(lalonde.exp)`. This will load the `lalonde.exp` dataset into your workspace. Do the same for the `lalonde.psid` dataset (the **observational** dataset) using the command `data(lalonde.psid)`. Each of the dataset has the same set of variables:

- `age` - age in years.
- `education` - number of years of schooling.
- `black` - 1 if black, 0 otherwise.
- `hispanic` - 1 if Hispanic, 0 otherwise.
- `married` - 1 if married, 0 otherwise.
- `nodegree` - 1 if no high school degree, 0 otherwise.
- `re74` - earnings in 1974.
- `re75` - earnings in 1975.
- `re78` - earnings in 1978.

- `u74` - if unemployed in 1974, 0 otherwise.
- `u75` - 1 if unemployed in 1975, 0 otherwise.
- `treat` - 1 if treated, 0 otherwise.

1A

The causal effect of interest is the the effect of assignment to the job training program `treat` on earnings in 1978 `re78`. The variable `treat` takes on a value of 1 if the individual was assigned to the job program and 0 if they were not.

Use a simple difference-in-means estimator to estimate the average treatment effect in the **experimental** dataset. Form a 95% confidence interval for the average treatment effect.

The estimated ATE is 1794.343. The 95% confidence interval is [479.2137, 3109.4725]

```
# ATE Point estimate
point <- mean(lalonde.exp$re78[lalonde.exp$treat == 1]) - mean
   (lalonde.exp$re78[lalonde.exp$treat == 0])
# Standard error
se <- sqrt(var(lalonde.exp$re78[lalonde.exp$treat == 1])/
   length(lalonde.exp$re78[lalonde.exp$treat == 1]) +
              var(lalonde.exp$re78[lalonde.exp$treat == 0])/
                 length(lalonde.exp$re78[lalonde.exp$treat ==
                 0]))
# 95% CI
ci <- c(point - qnorm(.975)*se, point + qnorm(.975)*se)
```

1B

Now apply the same estimator to the **observational** dataset, `lalonde.psid`. Compute the point estimate and large-sample 95% confidence interval

The estimated ATE is −15204.78. The 95% confidence interval is [−16492.62, −13916.93]

```
# ATE Point estimate
point <- mean(lalonde.exp$re78[lalonde.exp$treat == 1]) - mean
   (lalonde.exp$re78[lalonde.exp$treat == 0])
# Standard error
se <- sqrt(var(lalonde.exp$re78[lalonde.exp$treat == 1])/
   length(lalonde.exp$re78[lalonde.exp$treat == 1]) +
              var(lalonde.exp$re78[lalonde.exp$treat == 0])/
                 length(lalonde.exp$re78[lalonde.exp$treat ==
                 0]))
# 95% CI
ci <- c(point - qnorm(.975)*se, point + qnorm(.975)*se)
```

1C

How does your estimated treatment effect differ between 1A and 1B and why do you think this is the case?

> The estimated treatment effect in 1B is extremely negative while the estimate in 1A is moderate and positive. The difference between the observational and experimental estimates is likely due to significant confounding on relevant pre-treatment covariates. For example, because controls are drawn from the overall pool of U.S. residents, they are likely to be generally wealthier on average than those who were participating in the job training program experiment. Selection into the training program is conditional on income, so the mean observed income of control observations in the observational dataset is not a good counterfactual for the treated observations.

1D

In order for the difference-in-means estimator $E[Y_i|T_i = 1] - E[Y_i|T_i = 0]$ to be unbiased for the true causal effect, what is the key assumption that must be made about the relationship between treatment T_i and the potential outcomes $Y_i(1)$ and $Y_i(0)$? For which of the two datasets is this assumption more likely to hold and why?

> Potential outcomes $Y_i(1)$ and $Y_i(0)$ are independent of treatment assignment T_i – this means that all treatment is doing is "revealing" one of the potential outcomes via Y_i and we can equate $E[Y_i|T_i = a]$ with $E[Y_i(a)]$. This is likely to hold in the experimental dataset since we have assigned treatment at random. Conversely, we think this is unlikely to hold in the observational dataset since we know that poorer individuals are more likely to receive treatment and pre-treatment income is a strong predictor of post-treatment potential outcomes.

1E

A common way of adjusting for observed confounders when estimating a causal effect in observational data is to fit a regression model for outcome with treatment and your confounding confounding covariates as regressors.

Run a linear regression of earnings in 1978 (`re78`) on treatment and your 10 covariates using the **observational** dataset. Assume that there are no interactions or polynomial terms in the model and that all of the covariates enter into the model additively. What is your estimate of the average treatment effect using this regression model (the coefficient on treatment)?[1]

How does this estimated treatment effect compare to your experimental benchmark in 1A?

[1]Note that the coefficient is not *quite* an estimate of the ATE. See Aronow and Samii (2015) "Does Regression Produce Representative Estimates of Causal Effects?" for a discussion of this regression weighting problem. Assuming constant effects, however, these two quantities are equal.

Using regression, the estimated treatment effect is 4.15 (and the 95% confidence interval includes zero). While regression adjustment gets rid of a lot of the bias, it still under-estimates the true treatment effect (based on the experimental benchmark) since we don't know what the "true" model for the outcome is.

```
regress.full <- lm(re78 ~ treat + age + education + black +
    hispanic + married + nodegree + re74 + re75 + u74 + u75,
    data=lalonde.psid)
summary(regress.full)
```

Problem 2 - Assessing Balance

When making causal inferences from observational data, we typically want to approximate a hypothetical randomized experiment. The goal of randomization is to generate balance between treated and control groups on potentially confounding covariates. Typically, when first approaching an observational dataset to estimate a causal effect, we will investigate the degree of balance or imbalance on relevant covariates.

2A

For each of the ten covariates, calculate the absolute standardized difference in means (that is, the difference divided by the sample standard deviation) between treated and control observations using the **observational** dataset.

What is the value of the largest absolute standardized difference? For which variable is it the largest?

The largest absolute standardized difference is 1.85 for variable `u74`

```
#### Calculate absolute standardized difference

abs.std.diff.obs <- abs(apply(lalonde.psid, 2, function(x) (
    mean(x[lalonde.psid$treat==1]) - mean(x[lalonde.psid$treat
    ==0]))/sd(x)))
#### Drop "treat" and "re78"

abs.std.diff.obs <- abs.std.diff.obs[-c(9,12)]

#### Largest value
max(abs.std.diff.obs)
### Which one?
names(abs.std.diff.obs)[which.max(abs.std.diff.obs)]
```

2B

Now repeat 2A, but instead, use the **experimental** dataset.

What is the value of the largest absolute standardized difference? For which variable is it the largest?

The largest absolute standardized difference is 0.306 for the variable `nodegree`

```
###### Problem 2 B
#### Do the same for experimental
abs.std.diff.exp <- abs(apply(lalonde.exp, 2, function(x) (
    mean(x[lalonde.exp$treat==1]) - mean(x[lalonde.exp$treat
    ==0]))/sd(x)))

#### Drop "treat" and "re78"
abs.std.diff.exp <- abs.std.diff.exp[-c(9,12)]

#### Largest value
max(abs.std.diff.exp)
### Which one?
names(abs.std.diff.exp)[which.max(abs.std.diff.exp)]
```

2C

Now, make a plot comparing the level of imbalance in the observational dataset to the imbalance in the experimental one. Your y-axis should be the absolute standardized difference in means. For each of your ten covariates, plot the imbalance measures in the experimental data and the observational data side-by-side as two vertical columns of points. Add a line for each variable that connects the two points for each variable (to visualize the differences in imbalance between observational and experimental data).

Note: This is meant to be challenging and to get you to think about useful visual methods of conveying balance. If you are stuck on how to do this, note that your plot should look similar in style to Figure 2 from Stuart (2010) "Matching Methods for Causal Inference: A Review and a Look Forward" which you can download by going here Hint: use `type = "b"` to get R's plot command to plot both lines and points.

```
### For each point, plot a line segment that goes from obs to
    experimental
plot(y=c(abs.std.diff.exp[1], abs.std.diff.obs[1]), x=c(1, 0),
     type="b", pch=18, col="grey", ylim=c(0, 2), xlim=c(-0.2,
    1.2), xaxt="n", xlab="", ylab="Absolute standardized
    difference-in-means")
for (q in 2:length(abs.std.diff.exp)){
  points(y=c(abs.std.diff.exp[q], abs.std.diff.obs[q]), x=c(1,
       0), type="b", pch=18, col="grey")
}
axis(side = 1, at = c(0, 1), labels = c("Observational\
    nDataset", "Experimental\nDataset"), lwd=0)
```

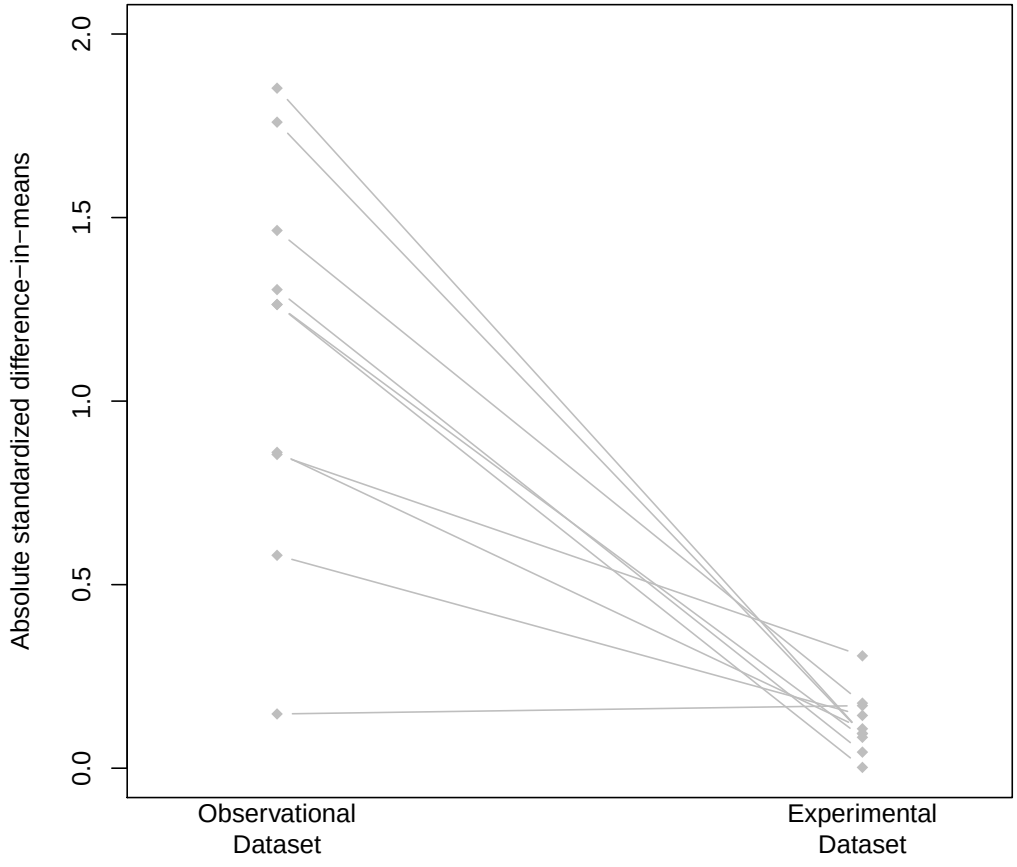

Figure 1: Comparison in imbalance between observational and experimental datasets

Problem 3 - Propensity score matching

Now we'll use matching to try to get better balance on our covariates in the observational data and try to approximate balance levels in the experimental data. First we'll use nearest neighbor propensity score matching to create a dataset matched on all ten covariates. The propensity score is defined as $\pi_i = Pr(T_i = 1|X_i)$.

3A

Estimate the propensity score for each observation (π_i) in the **observational** dataset using a logit regression model on all ten covariates (with no interactions or polynomial terms – just an additive function of the ten covariates).[2] Your outcome variable in the propensity score model should be whether or not the observation is treated. Use the model to predict $\hat{\pi}_i$ for each observation in your dataset.

Plot two histograms of the estimated propensity scores, one for the treated units and one for the control units. Make sure the x-axes of the two histograms are the same so you can compare the distributions of the propensity scores.

```
### First fit the propensity score model
pscore.model <- glm(treat ~ age + education + black + hispanic
    + married + nodegree + re74 + re75 + u74 + u75, data=
   lalonde.psid, family=binomial(link="logit"))

### Next predict propensity of treatment for each
lalonde.psid$pscore <- predict(pscore.model, type="response")

### Histogram 1:
pdf("treated_pscore.pdf", width=4, height=7)
hist(lalonde.psid$pscore[lalonde.psid$treat == 1], col="blue",
    xlim=c(0,1), xlab="Propensity Score", freq=F, main="
   Treated Units")
dev.off()
### Histogram 2
pdf("control_pscore.pdf", width=4, height=7)
hist(lalonde.psid$pscore[lalonde.psid$treat == 0], col="red",
   xlim=c(0,1), xlab="Propensity Score", freq=F, main="Control
    Units")
dev.off()
```

3B

Compare the two plots from 3A. How well do the propensity scores seem to predict whether the unit is in the treatment or control group? What does that tell you about

[2]Note, you may get a warning from R regarding possible separation in the model - for now, just ignore the warning for the purposes of this exercise. However, do take note of possible issues that can arise when fitting logit models with near-perfect separation.

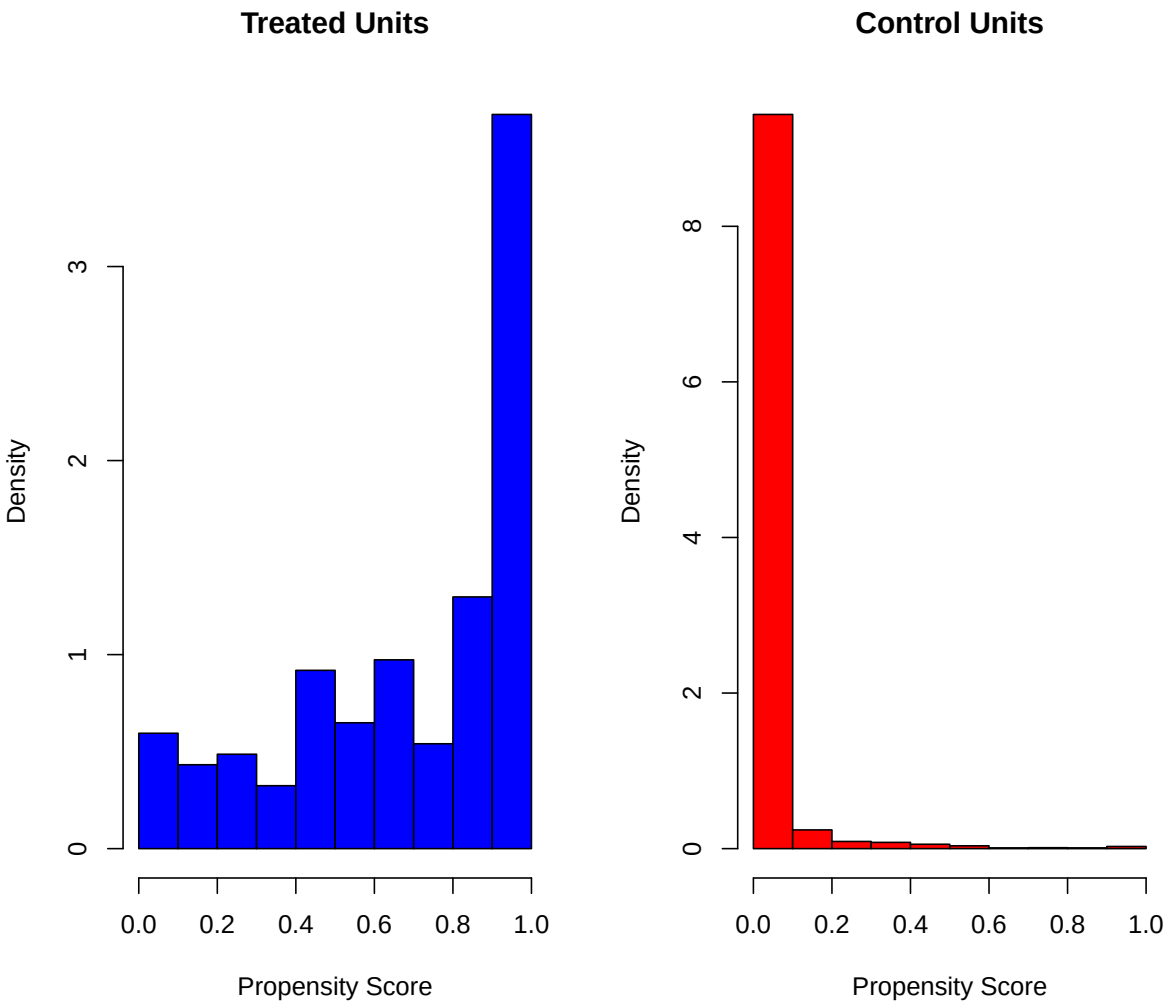

Figure 2: Treated vs. control propensity scores - unmatched dataset

the data?

> The plots show strong evidence that receipt of treatment is strongly predicted by the variables that we've identified in the model. If those variables are also strongly predictive of the outcome, we will get biased treatment effect estimates if we just use a raw comparison of outcome for treated and control arms. It tells us that the raw observational data by itself is going to give misleading results for analyzing the causal effect of treatment on outcome.

3C

Now we're going to use matching to try to improve balance on the covariates. Start by pruning the data so as to drop all control units with propensity scores that lie outside of the support of the propensity scores for treated units. How many units did you drop?

We drop 1282 observations.

```
### Now we match - first prune all controls that fall outside
    of the range of treated Pscores
treat.range <- range(lalonde.psid$pscore[lalonde.psid$treat
    ==1])
# Get the indices
prune.controls <- lalonde.psid$treat == 0  & (lalonde.psid$
    pscore < treat.range[1] | lalonde.psid$pscore > treat.range
    [2])
# How many prunes?
sum(prune.controls)
lalonde.psid.pruned <- lalonde.psid[prune.controls,]
```

3D

Now we're going to try to estimate the average treatment effect on treated units (ATT) by searching for appropriate counterfactuals to use for each treated observation. We'll start by doing one-to-one matching *with* replacement. Again, we are using the **observational** dataset.

Start by subsetting out all of the treated observations. Then, for each treated observation, search through the set of all controls to find the control observation with a propensity score closest to that of the treated unit (in terms of absolute distance). Store that control observation. Combine the treated and matched control observations to make your "matched" dataset.

How many individual rows (observations) are in your matched dataset?

We have 370 matched observations in our dataset.

```
##### Let's do matching!

### Grab the treateds
lalonde.psid.pruned.treated <- lalonde.psid.pruned[lalonde.
    psid.pruned$treat == 1,]
### Grab the controls
lalonde.psid.pruned.control <- lalonde.psid.pruned[lalonde.
    psid.pruned$treat == 0,]

### Initialize the placeholder for matched controls
lalonde.psid.pruned.matched.control <- NULL

### For each observation that's treated:
for (ind in 1:nrow(lalonde.psid.pruned.treated)){

  ### What's the treated pscore
  treat.pscore <- lalonde.psid.pruned.treated[ind,]$pscore

  ### What's the closest control?
  match.index <- which.min(abs(lalonde.psid.pruned.control$
      pscore - treat.pscore))
```

```
  ### Add to the placeholder dataset
  lalonde.psid.pruned.matched.control <- rbind(lalonde.psid.
     pruned.matched.control, lalonde.psid.pruned.control[match
     .index,])


}

### Merge into a single matched dataset
pscore.matched.data <- rbind(lalonde.psid.pruned.treated,
    lalonde.psid.pruned.matched.control)
```

3E

Create the same histograms as you did in 2A, but now using the matched dataset. Compare the distribution of propensity scores. What did matching do to the distribution?

The new distributions are *very* similar to one another!

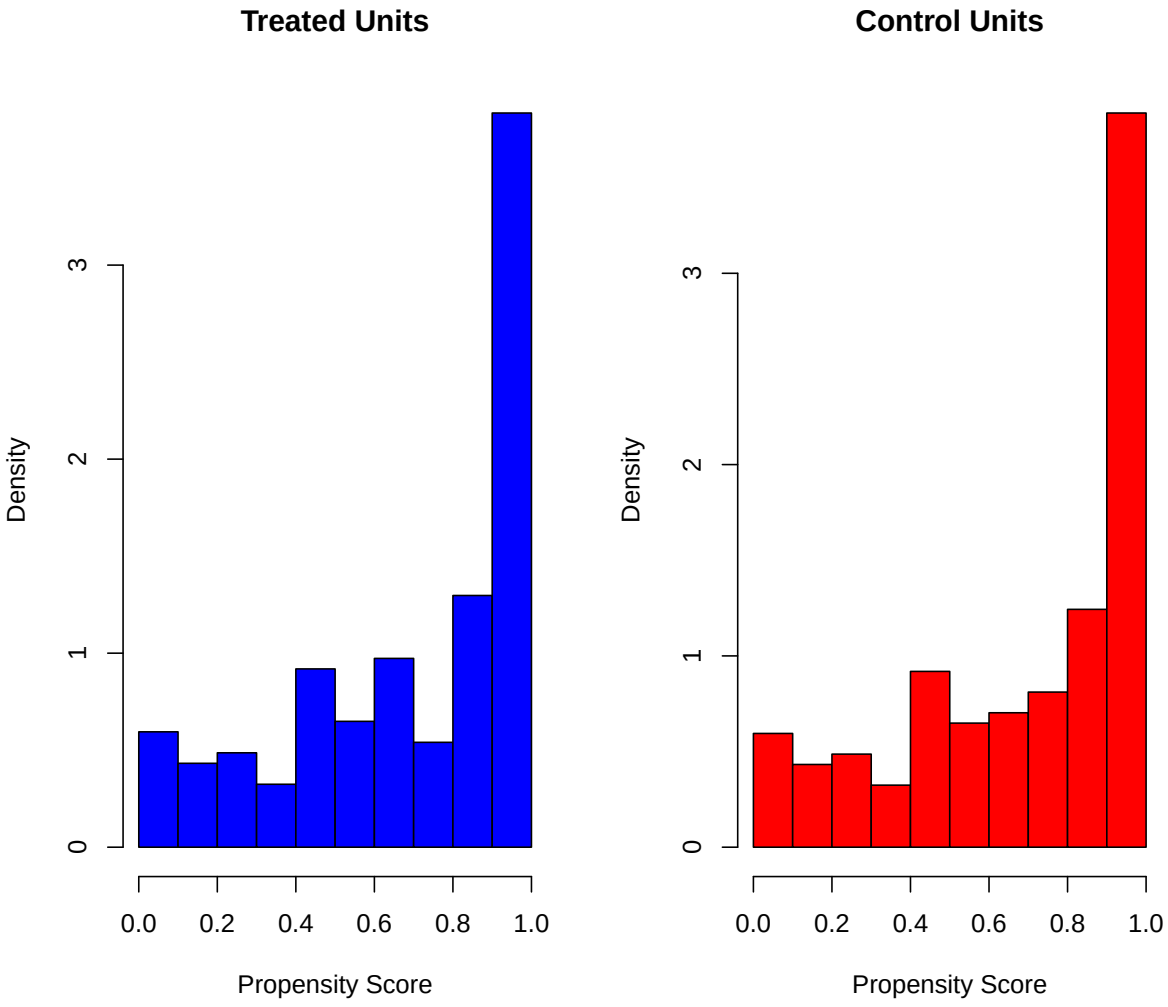

Figure 3: Treated vs. control propensity scores - matched dataset

3F

Using the same standardized difference-in-means that you used in Problem 2, calculate the imbalance in the matched dataset for each of your ten covariates. What is the largest difference-in-means that you calculated? For which variable is it the largest?

The largest absolute standardized difference-in-means is .41 for the variable u75.

```
abs.std.diff.match <- abs(apply(pscore.matched.data, 2,
    function(x) (mean(x[pscore.matched.data$treat==1]) - mean(x
    [pscore.matched.data$treat==0]))/sd(x)))
#### Drop "treat" and "re78" and any other variables

abs.std.diff.match <- abs.std.diff.match[c(1:8, 10, 11)]

### Which one?
names(abs.std.diff.match)[which.max(abs.std.diff.match)]

#### Largest value
max(abs.std.diff.match)
```

3G

Calculate a difference-in-means estimate of the treatment effect using your matched dataset.[3]

Compare the point estimate to your original estimates on the observational and experimental data from part 1. What has matching done to your estimate? Did matching alleviate the problem of unobserved confounding?

The matching estimate of the treatment effect is 941.79.

```
### Estimate an effect
match.point <- mean(pscore.matched.data$re78[pscore.matched.
    data$treat == 1]) - mean(pscore.matched.data$re78[pscore.
    matched.data$treat == 0])
```

Matching has allowed us to non-parametrically adjust for known observed confounders, bringing the observational difference-in-means estimate closer to the experimental truth. It *does not* adjust for unobserved confounders, so we are still possibly seeing some omitted variable bias in the estimate as we obtain an estimate slightly smaller than the experimental one.

[3]Note that standard errors for these types of matching estimators are a bit more complicated to work out so we're not having you calculate them them here. Intuitively, since we have repeated observations that are being matched, we would need to take into account some non-independence across units.

Problem 4 - Mahalanobis matching

Now we'll compare the balance changes using an alternative distance metric for matching – Mahalanobis matching.

The Mahalanobis distance between two covariate column vectors X_i and X_j is defined as

$$M(X_i, X_j) = [(X_i - X_j)^T \mathbf{S}^{-1}(X_i - X_j)]^{1/2}$$

where $\mathbf{S}$ is the sample variance-covariance matrix of X (which you can compute using `cov()` on a the matrix of covariates).

4A

Write a function that takes two vectors and the S variance-covariance matrix and returns the Mahalanobis distance between those two vectors.[4] Use it to calculate the Mahalanobis distance between observation 1 and observation 350 on the 10 covariates (that is, variables that are not treatment or outcome).

The function is

```
### Mahalanobis distance matching
mahal <- function(x1, x2, S){
  return(sqrt(t(x1 - x2)%*%solve(S)%*%(x1 - x2)))
}
```

The Mahalanobis distance between observations 1 and 350 is 4.54.

```
mahal(x1=t(as.matrix(lalonde.psid[1,c(1:8, 10, 11)])),
      x2=t(as.matrix(lalonde.psid[350,c(1:8, 10, 11)])),
      S=cov(lalonde.psid[,c(1:8, 10, 11)]))
```

4B

We're going to again do one-to-one matching *with* replacement using the observational data, but instead of matching each using the propensity score, we're going to match using the Mahalanobis distance between the covariates of treated and control units.

Start by subsetting out all of the treated observations. Then, for each treated observation, calculate the Mahalanobis distance between that unit's covariates and the the covariates of all of the control observations. Pick the control observation with the smallest Mahalanobis distance. Store that control observation. Combine the treated and matched control observations to make your "matched" dataset.

Using your matched dataset, calculate the difference-in-means estimate of the average treatment effect on the treated.

[4] Hint, you may find it useful to convert the vectors and S to matrix form in order to get R to correctly do matrix multiplication. Pay attention to what is a column and what is a row vector.

Our estimated treatment effect using Mahalanobis distance matching is 2023.073.

```
##### Let's do matching!

whichCovs <- c(1:8, 10, 11) ### These are the column numbers
    for the covariates
### Grab the treateds
lalonde.psid.treated <- lalonde.psid[lalonde.psid$treat == 1,]
### Grab the controls
lalonde.psid.control <- lalonde.psid[lalonde.psid$treat == 0,]

### Initialize the placeholder for matched controls
lalonde.psid.matched.control <- NULL

### What's the S matrix?
S.mat <- cov(lalonde.psid[,c(1:8, 10, 11)])

### For each observation that's treated:
for (ind in 1:nrow(lalonde.psid.treated)){

  ### What's the treated covariate profile
  treat.cov <- t(as.matrix(lalonde.psid.treated[ind,c(1:8, 10,
      11)]))

  ### Calculate the mahalanobis distance for each row in the
     control dataset
  mahal.dist <- apply(lalonde.psid.control[,c(1:8, 10, 11)],
     1, function(x) mahal(treat.cov, as.matrix(x), S=S.mat))

  ### What's the closest control?
  match.index <- which.min(mahal.dist)

  ### Add to the placeholder dataset
  lalonde.psid.matched.control <- rbind(lalonde.psid.matched.
     control, lalonde.psid.control[match.index,])

}

### Merge into a single matched dataset
mahal.matched.data <- rbind(lalonde.psid.treated, lalonde.psid
    .matched.control)

### Estimate an effect
mahal.match.point <- mean(mahal.matched.data$re78[mahal.
    matched.data$treat == 1]) - mean(mahal.matched.data$re78[
    mahal.matched.data$treat == 0])

mahal.match.point
```

4C

Finally, we're going to compare the balance on covariates between the propensity score matched dataset from Problem 3 and the Mahalanobis matched dataset. Calculate the

average of the absolute standardized differences-in-means between treated/control for each of the covariates in your propensity score matched dataset. Do the same thing for the matched Mahalanobis dataset. Based on your results, which matching metric produced better covariate balance? From what we learned in class, why do you think it worked better?

The average standardized difference-in-means between treated/control in the propensity score matched dataset is .24.

However, the average standardized difference-in-means using Mahalanobis matching is 0.061 – and in fact, we get perfect balance on some of the covariates! Mahalanobis did better in achieving covariate balance because it approximates blocking rather than a fully randomized experiment.

Indeed, Mahalanobis matching yields better balance than the experimental benchmark, suggesting that matching can help improve efficiency by approximating blocking on observed covariates *even in* a fully randomized experiment.

```
#### Pscore average standardized difference
mean(abs.std.diff.match)

#### Mahalanobis standardized differences
abs.std.diff.mahal <- abs(apply(mahal.matched.data, 2,
    function(x) (mean(x[mahal.matched.data$treat==1]) - mean(x[
    mahal.matched.data$treat==0]))/sd(x)))
#### Drop "treat" and "re78" and any other variables

abs.std.diff.mahal <- abs.std.diff.mahal[c(1:8, 10, 11)]

mean(abs.std.diff.match)
```

Problem 5 - Coarsened Exact Matching

In this part you will use Coarsened Exact Matching to estimate the causal effect. You'll do this by using the `cem()` function in the `cem` package.

For now, use the default settings for generating the coarsenings. Also, we're going to allow CEM to prune both treated and control observations.
Consult the documentation for more details about how to use the `cem()` function.

5A

Use CEM to prune the **observational** dataset using the default coarsening settings.

How many treated and control observations. observations does your matched dataset contain after you prune with CEM?

There are 69 treated observations in the matched dataset and 79 controls.

```
\\
cem.match <- cem(treatment="treat",
                 data=lalonde.psid,
                 drop=c("re78", "pscore"))

cem.matched.data <- lalonde.psid[cem.match$matched,]

### Number of treateds
sum(cem.matched.data$treat == 1)

### Number of controls
sum(cem.matched.data$treat == 0)
```

5B

Estimate the treatment effect by using a difference-in-means estimate from your CEM-matched data using the `att()` function. Report the point estimate of the treatment effect.

The estimated effect using CEM is -1829.24

```
estim <- att(cem.match, re78 ~ treat, data = lalonde.psid)
summary(estim)
```

5C

Let's assess the improvement in balance from the CEM match in 5A. Take your matched dataset and calculate the weighted absolute standardized difference in means across all of your ten covariates (using the observation weights provided by `cem()`).**Hint:** Use the `weighted.mean()` function to calculate a weighted average.

What is the average weighted absolute standardized difference in means? How does this compare to the propensity score and Mahalanobis distance matches?

The average is 0.06002835, which is close to the Mahalanobis imbalance, but note that we are also changing the quantity of interest by dropping treated units, so our estimates may not be directly comparable.

```
cem.matched.data <- lalonde.psid[cem.match$matched,]
cem.matched.data$w <- cem.match$w[cem.match$matched]
abs.std.diff.cem <- abs(apply(cem.matched.data , 2, function(x
    ) (weighted.mean(x[cem.matched.data$treat==1], cem.matched.
    data$w[cem.matched.data$treat==1]) - weighted.mean(x[cem.
    matched.data$treat==0], cem.matched.data$w[cem.matched.data
    $treat==0]))/sd(x)))
```

```
#### Drop "treat" and "re78" and any other variables

abs.std.diff.cem <- abs.std.diff.cem[c(1:8, 10, 11)]

mean(abs.std.diff.cem)
```

R Code

Please submit all your code for this assignment as a .R file. Your code should be clean, commented, and executable without error.