Module 1: Hospital Pricing and Selection on Observables

Part 3: Hospital Prices and Penalties

Ian McCarthy | Emory University
Econ 470 & HLTH 470

Penalized hospitals

```r
hcris.data <- read_rds(here("data/HCRIS_Data.rds"))

hcris.data <- hcris.data %>%
  mutate( discount_factor = 1-tot_discounts/tot_charges,
          price_num = (ip_charges + icu_charges + ancillary_charges)*discount_factor - tot_mcare_payment,
          price_denom = tot_discharges - mcare_discharges,
          price = price_num/price_denom)

final.hcris <- hcris.data %>% ungroup() %>%
  filter(price_denom>100, !is.na(price_denom),
         price_num>0, !is.na(price_num),
         price<100000,
         beds>30, year==2012) %>%
  mutate( hvbp_payment = ifelse(is.na(hvbp_payment),0,hvbp_payment),
          hrrp_payment = ifelse(is.na(hrrp_payment),0,abs(hrrp_payment)),
    penalty = (hvbp_payment-hrrp_payment<0))
```

Summary stats

Always important to look at your data before doing any formal analysis. Ask yourself a few questions:

1. Are the magnitudes reasonable?

2. Are there lots of missing values?

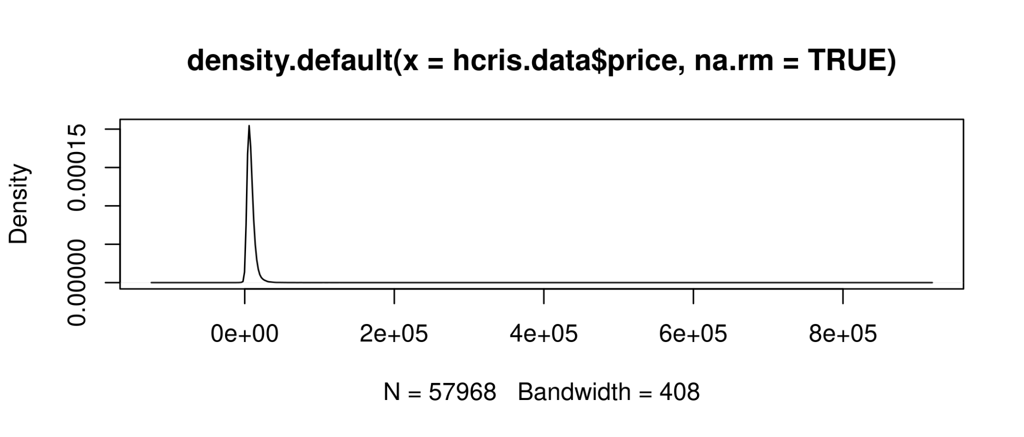
3. Are there clear examples of misreporting?

Summary stats

```r
summary(hcris.data$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
## -123697    4783    7113     Inf   10230     Inf   63662
```

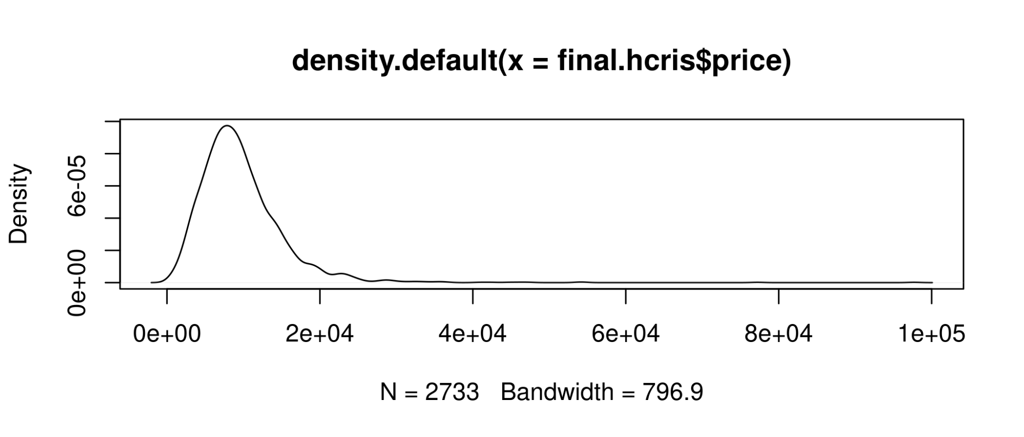
```r
plot(density(hcris.data$price, na.rm=TRUE))
```




```r
summary(final.hcris$price)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max. 
##   340.8  6129.9  8705.4  9646.9 11905.4 97688.8
```

```r
plot(density(final.hcris$price))
```



Dealing with problems

We've adopted a very brute force way to deal with outlier prices. Other approaches include:

1. Investigate very closely the hospitals with extreme values
2. Winsorize at certain thresholds (replace extreme values with pre-determined thresholds)
3. Impute prices for extreme hospitals

Differences among penalized hospitals

- Mean price among penalized hospitals: 9,896.31
- Mean price among non-penalized hospitals: 9,560.41
- Mean difference: 335.9

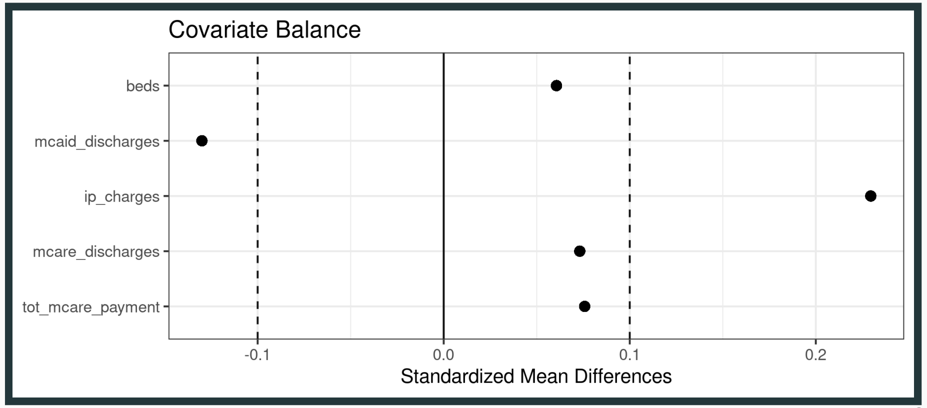
Comparison of hospitals

Are penalized hospitals sufficiently similar to non-penalized hospitals?

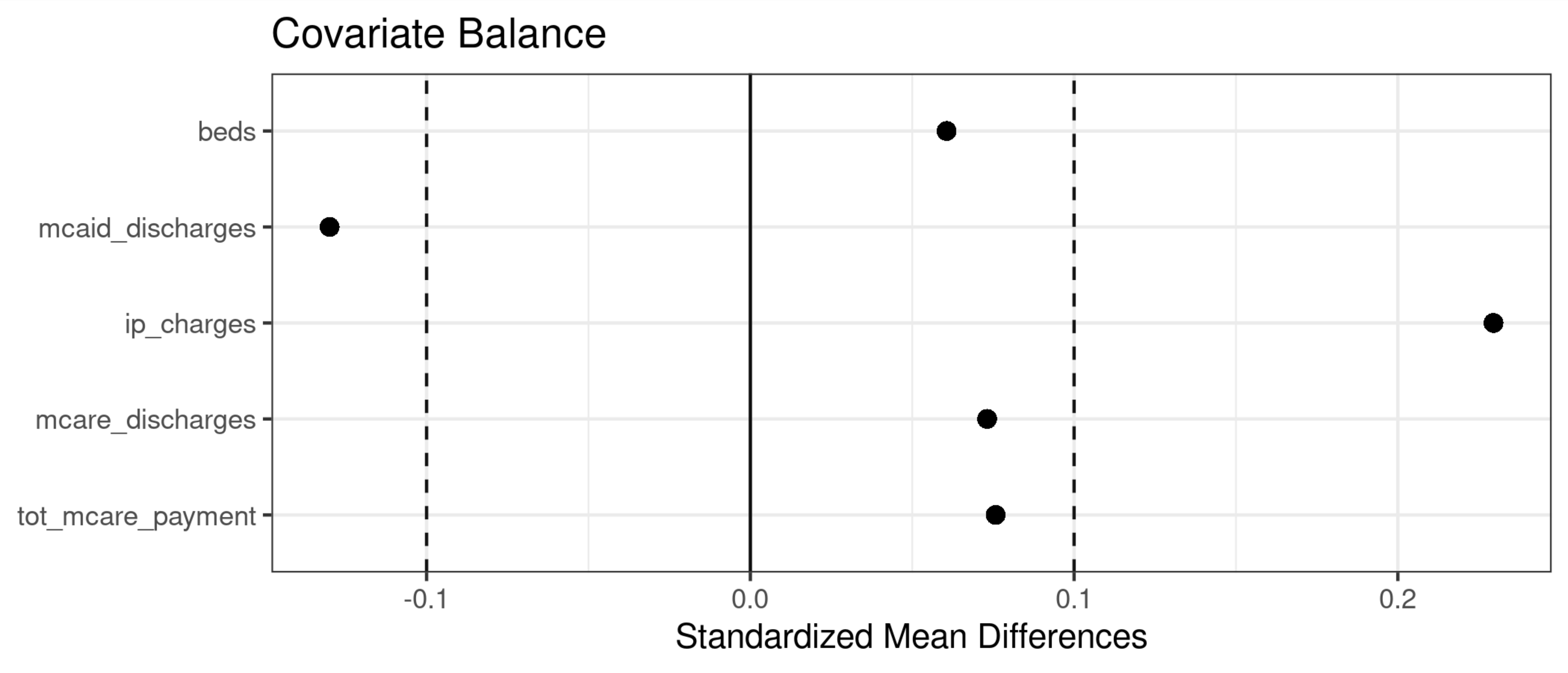
Let's look at covariate balance using a love plot, part of the `library(cobalt)` package.

Love plots without adjustment

```
love.plot(bal.tab(lp.covs,treat=lp.vars$penalty), colors="black", shapes="circle", threshold=0.1) +
  theme_bw() + theme(legend.position="none")
```



Love plots without adjustment



Using matching to improve balance

Some things to think about:

- exact versus nearest neighbor
- with or without ties (and how to break ties)
- measure of distance

1. Exact Matching

```r
m.exact <- Matching::Match(Y=lp.vars$price,
                           Tr=lp.vars$penalty,
                           X=lp.covs,
                           M=1,
                           exact=TRUE)
print(m.exact)
```

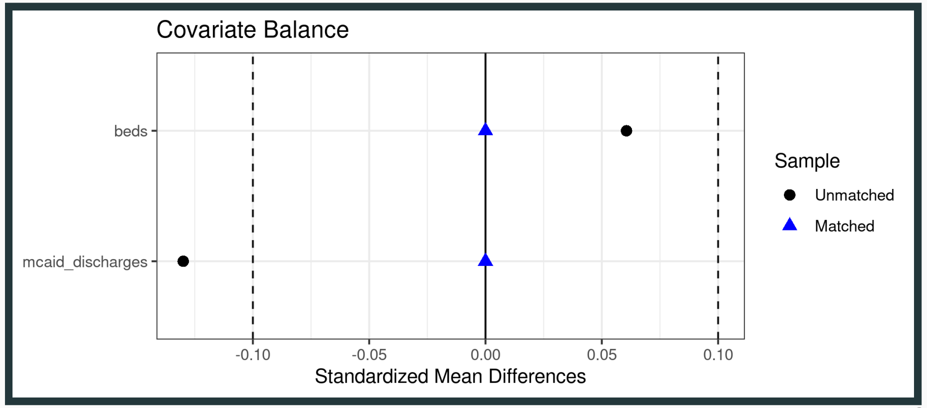
```
## [1] NA
## attr(,"class")
## [1] "Match"
```

1. Exact Matching (on a subset)

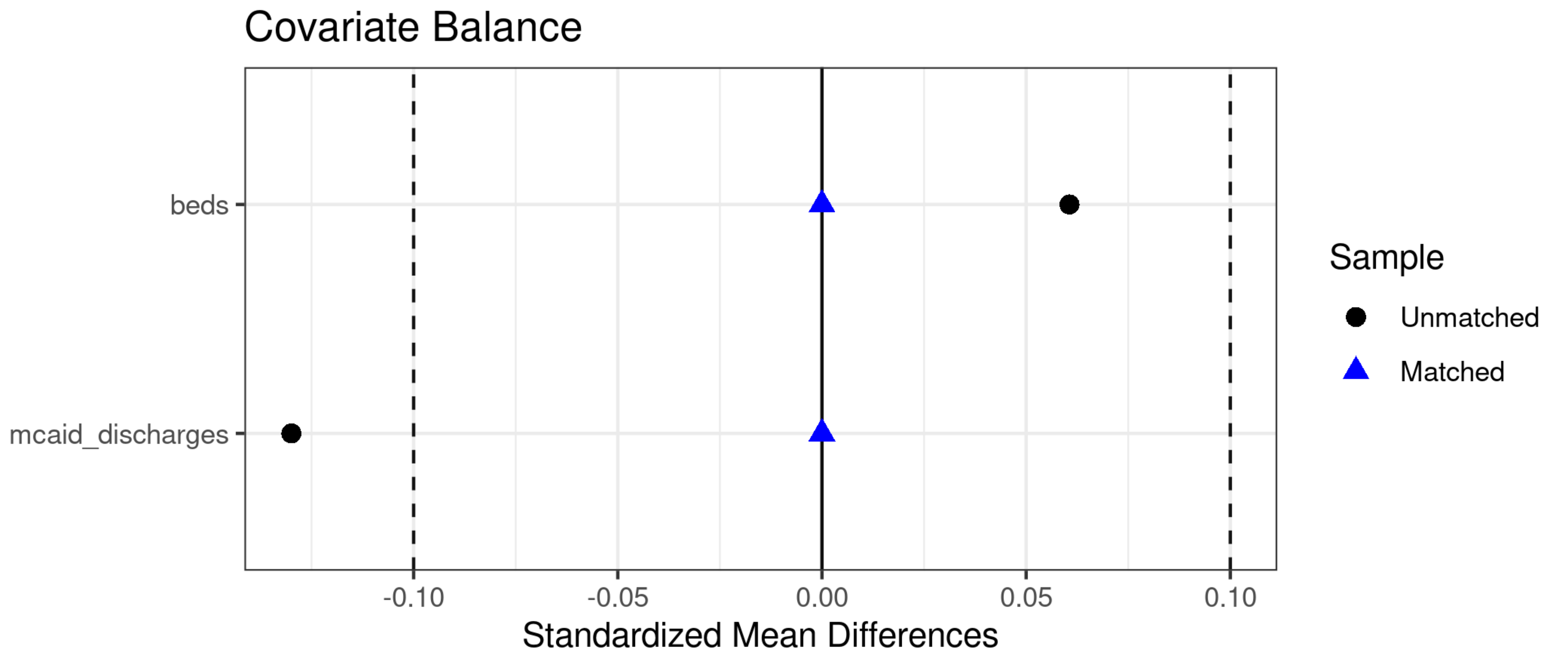
```
lp.covs2 ← lp.covs %>% select(beds, mcaid_discharges)
m.exact ← Matching::Match(Y=lp.vars$price,
                           Tr=lp.vars$penalty,
                           X=lp.covs2,
                           M=1,
                           exact=TRUE,
                           estimand="ATE")
```

1. Exact Matching (on a subset)

```r
love.plot(bal.tab(m.exact, covs = lp.covs2, treat = lp.vars$penalty),
          threshold=0.1,
          grid=FALSE, sample.names=c("Unmatched", "Matched"),
          position="top", shapes=c("circle","triangle"),
          colors=c("black","blue")) +
  theme_bw()
```



1. Exact Matching (on a subset)

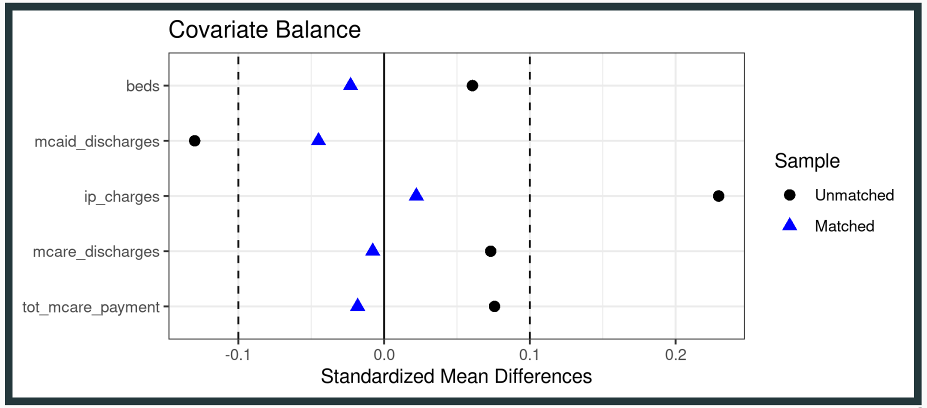


2. Nearest neighbor matching (inverse variance)

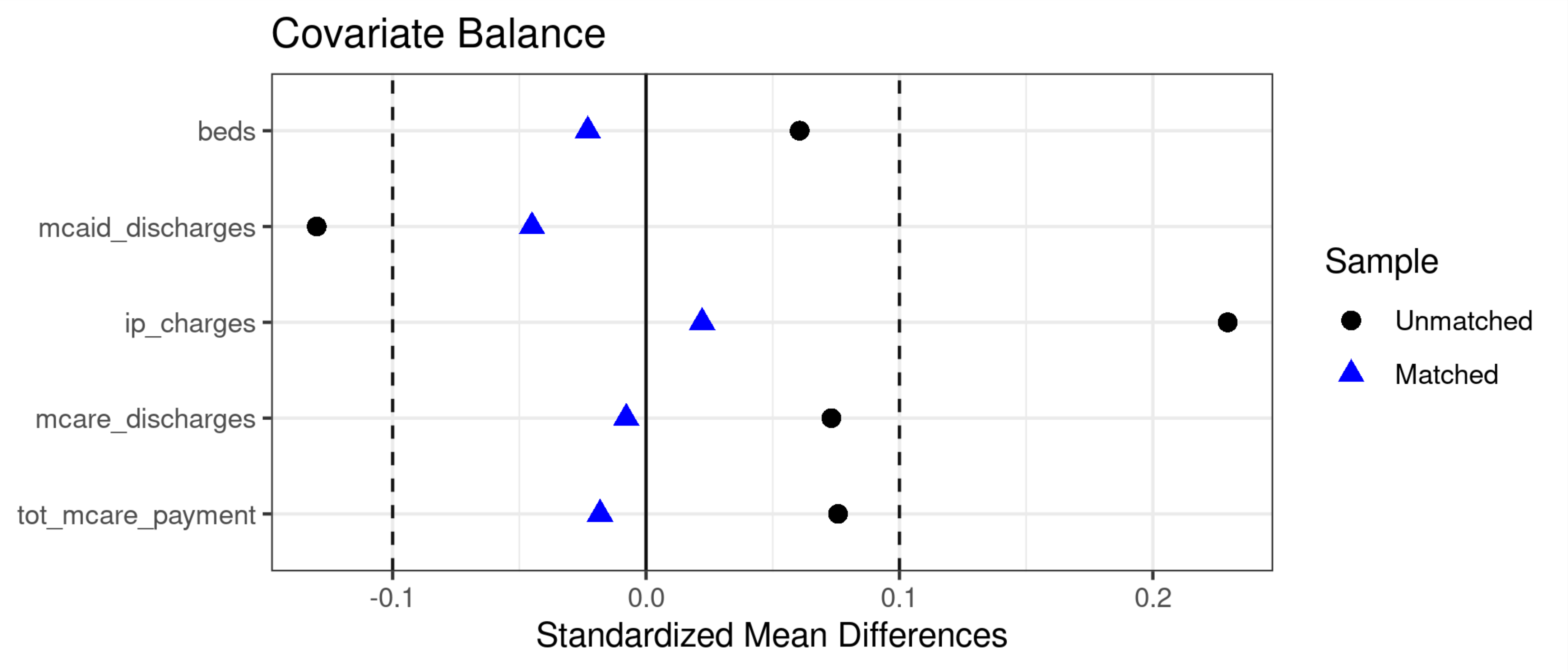
```
m.nn.var <- Matching::Match(Y=lp.vars$price,
                            Tr=lp.vars$penalty,
                            X=lp.covs,
                            M=4,
                            Weight=1,
                            estimand="ATE")

v.name=data.frame(new=c("Beds","Medicaid Discharges", "Inaptient Charges",
                   "Medicare Discharges", "Medicare Payments"))
```

2. Nearest neighbor matching (inverse variance)

```
love.plot(bal.tab(m.nn.var, covs = lp.covs, treat = lp.vars$penalty),
          threshold=0.1,
          var.names=v.name,
          grid=FALSE, sample.names=c("Unmatched", "Matched"),
          position="top", shapes=c("circle","triangle"),
          colors=c("black","blue")) +
  theme_bw()
```



2. Nearest neighbor matching (inverse variance)

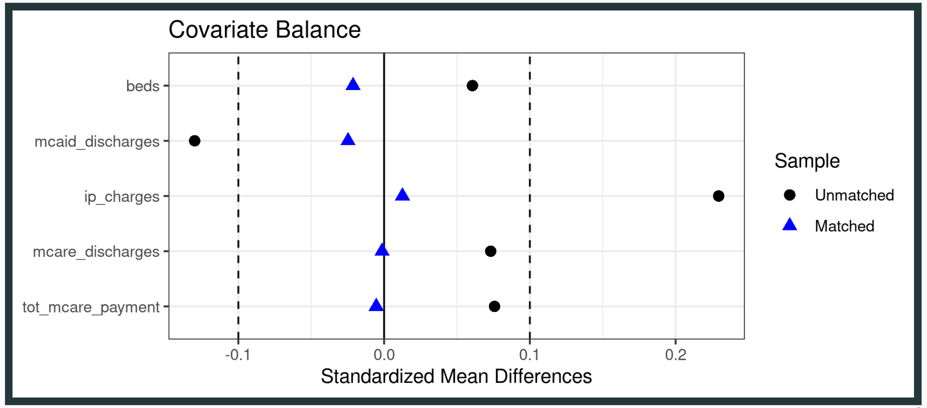


2. Nearest neighbor matching (inverse variance)

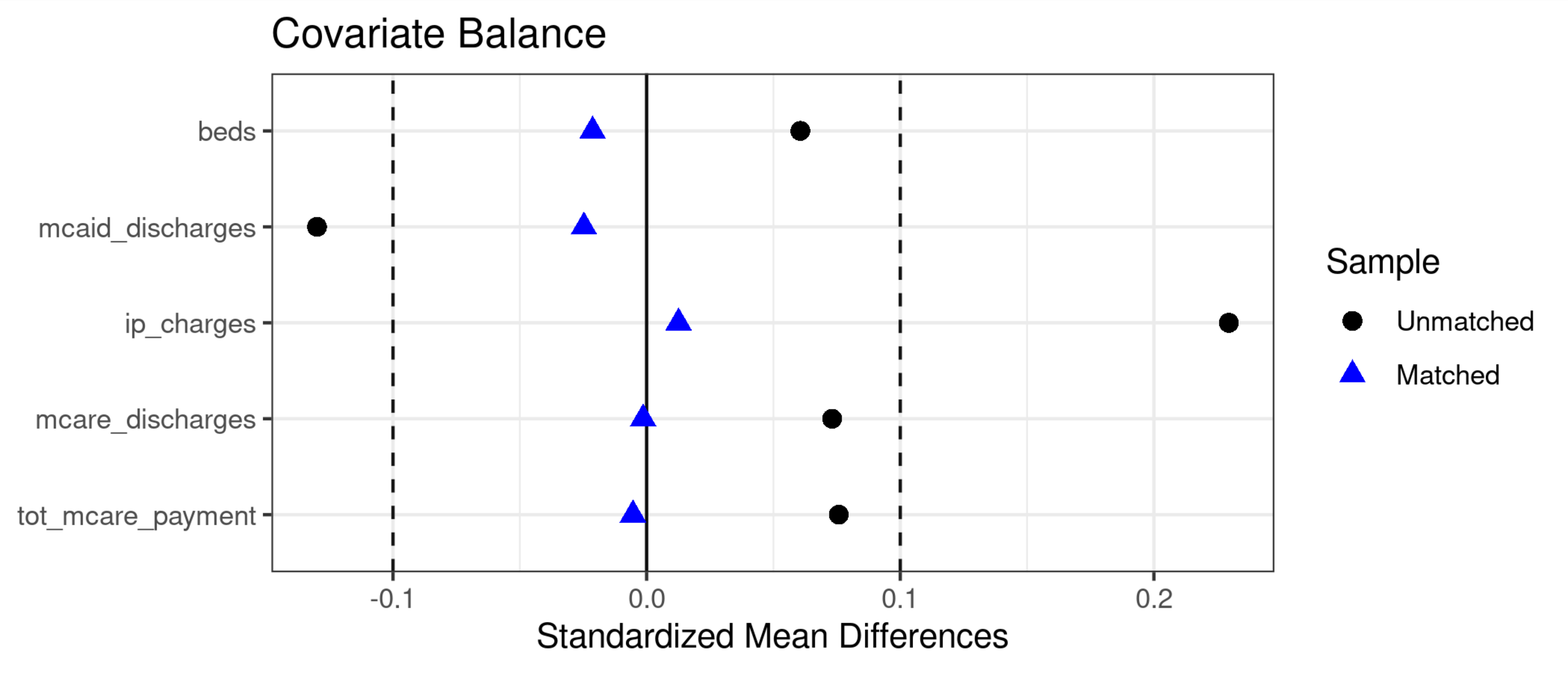
```r
m.nn.var2 <- Matching::Match(Y=lp.vars$price,
                             Tr=lp.vars$penalty,
                             X=lp.covs,
                             M=1,
                             Weight=1,
                             estimand="ATE")
```

2. Nearest neighbor matching (inverse variance)

```
love.plot(bal.tab(m.nn.var2, covs = lp.covs, treat = lp.vars$penalty),
          threshold=0.1,
          var.names=v.name,
          grid=FALSE, sample.names=c("Unmatched", "Matched"),
          position="top", shapes=c("circle","triangle"),
          colors=c("black","blue")) +
  theme_bw()
```



2. Nearest neighbor matching (inverse variance)

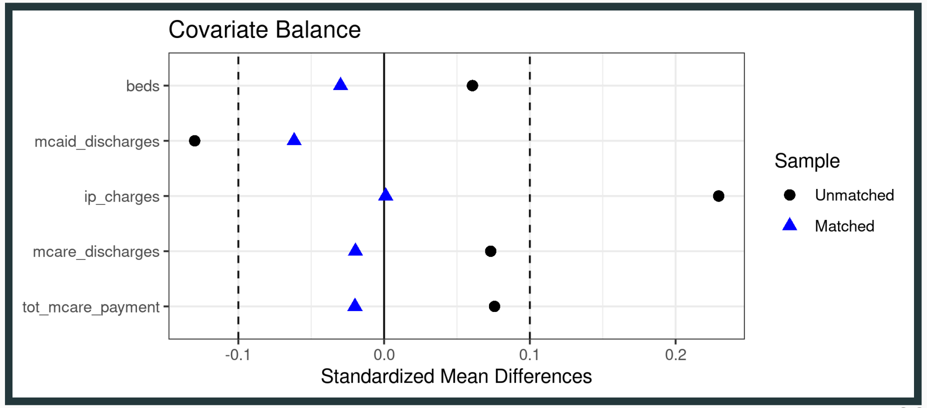


2. Nearest neighbor matching (Mahalanobis)

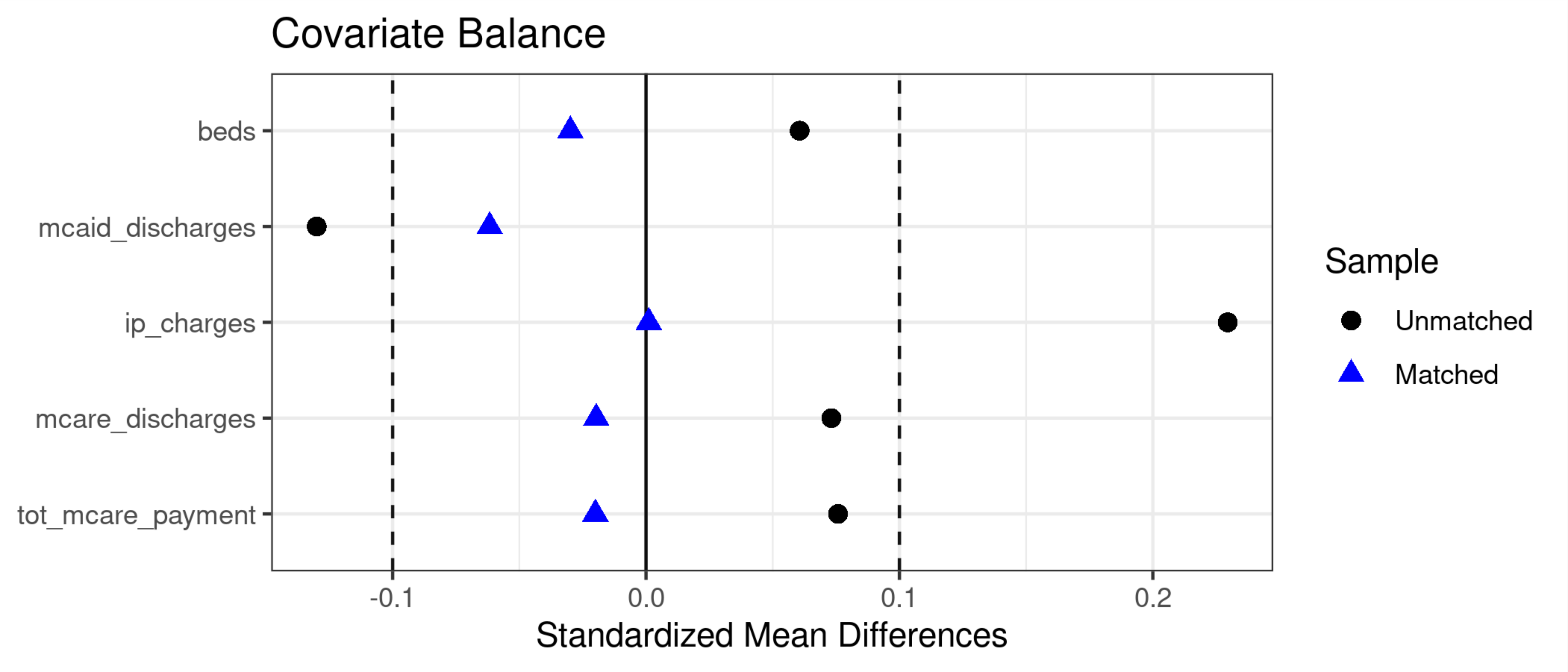
```r
m.nn.md <- Matching::Match(Y=lp.vars$price,
                           Tr=lp.vars$penalty,
                           X=lp.covs,
                           M=1,
                           Weight=2,
                           estimand="ATE")
```

2. Nearest neighbor matching (Mahalanobis)

```
love.plot(bal.tab(m.nn.md, covs = lp.covs, treat = lp.vars$penalty),
          threshold=0.1,
          var.names=v.name,
          grid=FALSE, sample.names=c("Unmatched", "Matched"),
          position="top", shapes=c("circle","triangle"),
          colors=c("black","blue")) +
  theme_bw()
```



2. Nearest neighbor matching (Mahalanobis)

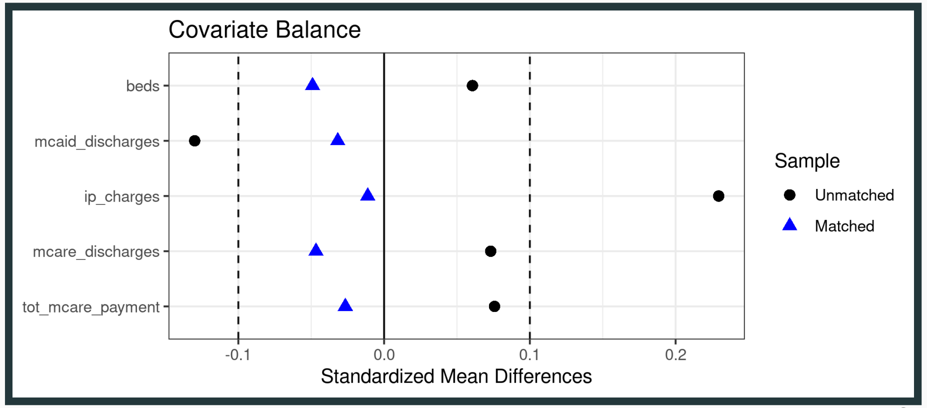


2. Nearest neighbor matching (propensity score)

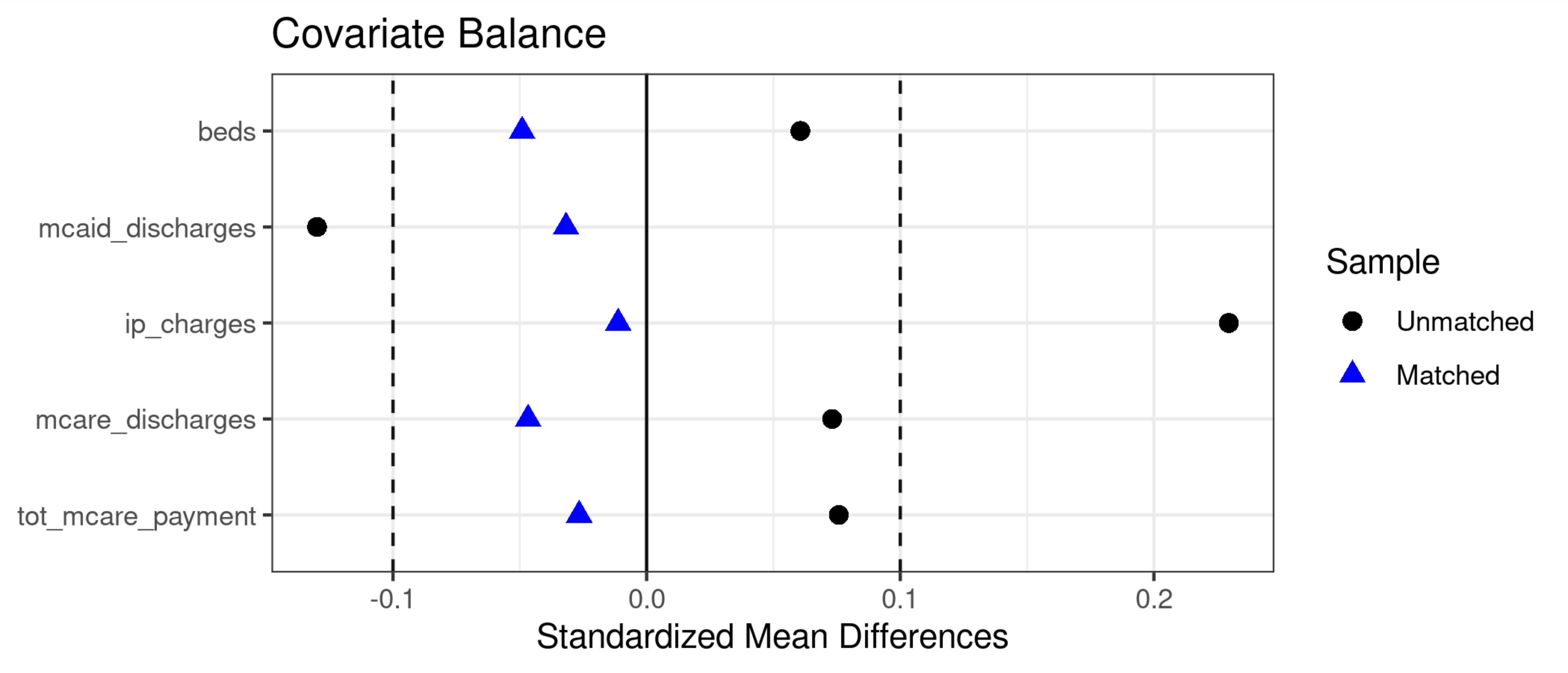
```r
logit.model <- glm(penalty ~ beds + mcaid_discharges + ip_charges + mcare_discharges +
            tot_mcare_payment, family=binomial, data=lp.vars)
ps <- fitted(logit.model)
m.nn.ps <- Matching::Match(Y=lp.vars$price,
                           Tr=lp.vars$penalty,
                           X=ps,
                           M=1,
                           estimand="ATE")
```

2. Nearest neighbor matching (propensity score)

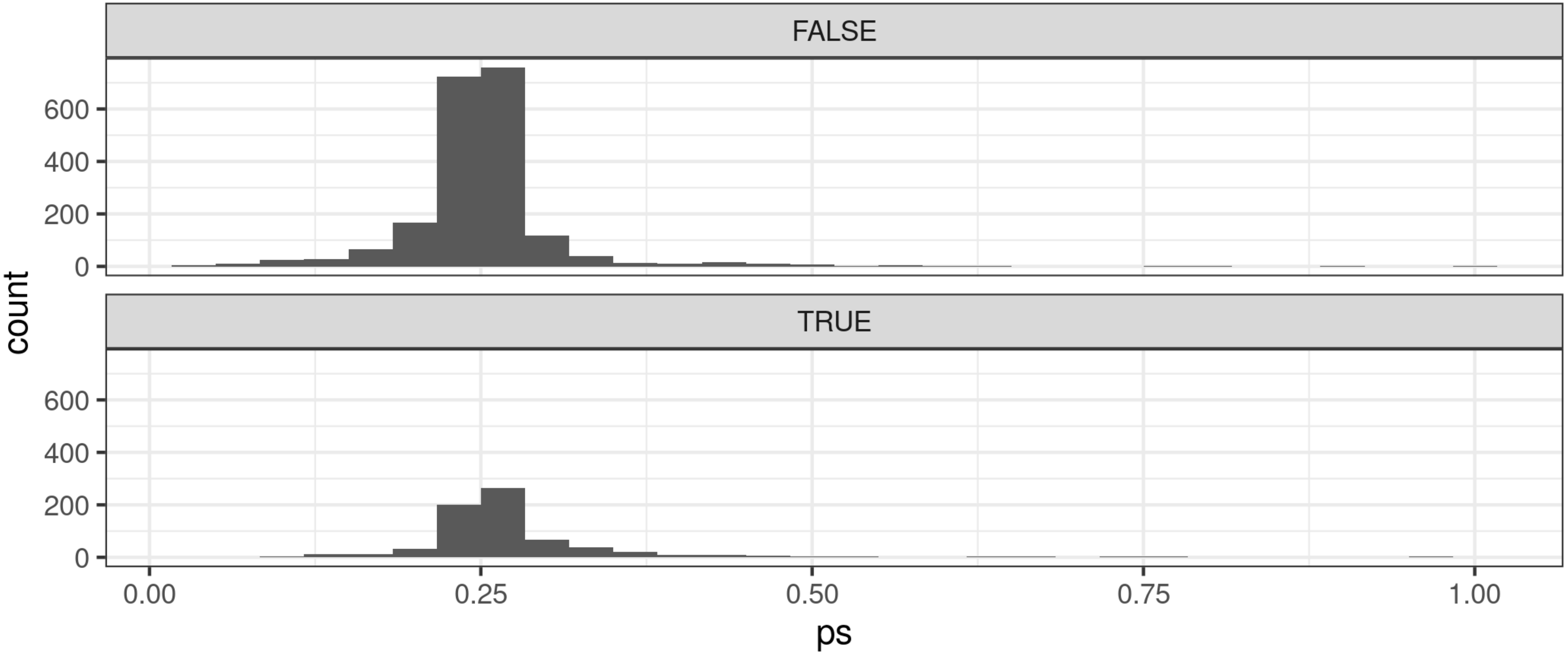
```
love.plot(bal.tab(m.nn.ps, covs = lp.covs, treat = lp.vars$penalty),
          threshold=0.1,
          var.names=v.name,
          grid=FALSE, sample.names=c("Unmatched", "Matched"),
          position="top", shapes=c("circle","triangle"),
          colors=c("black","blue")) +
  theme_bw()
```



2. Nearest neighbor matching (propensity score)



3. Weighting


FALSE
TRUE
count
600
400
200
0
600
400
200
0
0.00
0.25
0.50
0.75
1.00
ps

Results: Exact matching

```
##
## Estimate...  1777.6
## AI SE......  34.725
## T-stat.....  51.191
## p.val......  < 2.22e-16
##
## Original number of observations..............  2707
## Original number of treated obs...............  698
## Matched number of observations...............  12
## Matched number of observations  (unweighted).  12
##
## Number of obs dropped by 'exact' or 'caliper'  2695
```

Results: Nearest neighbor

- Inverse variance

```
##
## Estimate...  -526.95
## AI SE......  223.06
## T-stat.....  -2.3623
## p.val......  0.01816
##
## Original number of observations..............  2707
## Original number of treated obs...............  698
## Matched number of observations...............  2707
## Matched number of observations  (unweighted).  2711
```

Results: Nearest neighbor

- Mahalanobis

```
##
## Estimate...  -492.82
## AI SE......  223.55
## T-stat.....  -2.2046
## p.val......  0.027485
##
## Original number of observations..............  2707
## Original number of treated obs...............  698
## Matched number of observations...............  2707
## Matched number of observations  (unweighted).  2708
```

Results: Nearest neighbor

- Propensity score

```
##
## Estimate...  -201.03
## AI SE......  275.76
## T-stat.....  -0.72898
## p.val......  0.46601
##
## Original number of observations..............  2707
## Original number of treated obs...............  698
## Matched number of observations...............  2707
## Matched number of observations  (unweighted).  14795
```

Results: IPW weighting

```
lp.vars <- lp.vars %>%
  mutate(ipw = case_when(
    penalty==1 ~ 1/ps,
    penalty==0 ~ 1/(1-ps),
    TRUE ~ NA_real_
  ))
mean.t1 <- lp.vars %>% filter(penalty==1) %>%
  select(price, ipw) %>% summarize(mean_p=weighted.mean(price,w=ipw))
mean.t0 <- lp.vars %>% filter(penalty==0) %>%
  select(price, ipw) %>% summarize(mean_p=weighted.mean(price,w=ipw))
mean.t1$mean_p - mean.t0$mean_p
```

```
## [1] -196.8922
```

Results: IPW weighting with regression

```
ipw.reg <- lm(price ~ penalty, data=lp.vars, weights=ipw)
summary(ipw.reg)
```

```
##
## Call:
## lm(formula = price ~ penalty, data = lp.vars, weights = ipw)
##
## Weighted Residuals:
##    Min     1Q Median     3Q    Max
## -18691  -4802  -1422   2651  94137
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)   9876.4      147.8  66.808   <2e-16 ***
## penaltyTRUE   -196.9      211.2  -0.932    0.351
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7829 on 2705 degrees of freedom
## Multiple R-squared:  0.0003211,    Adjusted R-squared:  -4.85e-05
## F-statistic: 0.8688 on 1 and 2705 DF,  p-value: 0.3514
```

Results: Regression

```r
reg1.dat <- lp.vars %>% filter(penalty==1, complete.cases(.))
reg1 <- lm(price ~ beds+ mcaid_discharges + ip_charges + mcare_discharges +
            tot_mcare_payment, data=reg1.dat)

reg0.dat <- lp.vars %>% filter(penalty==0, complete.cases(.))
reg0 <- lm(price ~ beds + mcaid_discharges + ip_charges + mcare_discharges +
            tot_mcare_payment, data=reg0.dat)
pred1 <- predict(reg1,new=lp.vars)
pred0 <- predict(reg0,new=lp.vars)
mean(pred1-pred0)
```

```
## [1] -5.845761
```

Results: Regression in one step

```
reg.dat <- lp.vars %>% ungroup() %>% filter(complete.cases(.)) %>%
  mutate(beds_diff = penalty*(beds - mean(beds)),
         mcaid_diff = penalty*(mcaid_discharges - mean(mcaid_discharges)),
         ip_diff = penalty*(ip_charges - mean(ip_charges)),
         mcare_diff = penalty*(mcare_discharges - mean(mcare_discharges)),
         mpay_diff = penalty*(tot_mcare_payment - mean(tot_mcare_payment)))
reg <- lm(price ~ penalty + beds + mcaid_discharges + ip_charges + mcare_discharges + tot_mcare_payment +
            beds_diff + mcaid_diff + ip_diff + mcare_diff + mpay_diff,
          data=reg.dat)
```

Results: Regression in one step

```
##
## Call:
## lm(formula = price ~ penalty + beds + mcaid_discharges + ip_charges +
##     mcare_discharges + tot_mcare_payment + beds_diff + mcaid_diff +
##     ip_diff + mcare_diff + mpay_diff, data = reg.dat)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -38175  -2900   -597   2105  67409
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)        8.466e+03  1.711e+02  49.482  < 2e-16 ***
## penaltyTRUE       -5.846e+00  2.124e+02  -0.028  0.97804
## beds               1.107e+00  1.421e+00   0.779  0.43618
## mcaid_discharges  -4.714e-01  7.296e-02  -6.462 1.23e-10 ***
## ip_charges         6.426e-06  1.285e-06   5.002 6.04e-07 ***
## mcare_discharges  -8.122e-01  9.257e-02  -8.774  < 2e-16 ***
## tot_mcare_payment  9.502e-05  6.858e-06  13.857  < 2e-16 ***
## beds_diff          2.517e+00  2.986e+00   0.843  0.39931
## mcaid_diff         1.058e-01  1.570e-01   0.674  0.50050
## ip_diff           -4.534e-06  2.027e-06  -2.237  0.02539 *
## mcare_diff         4.806e-01  1.809e-01   2.657  0.00793 **
## mpay_diff         -5.452e-05  1.321e-05  -4.128 3.78e-05 ***
## ---
```

Summary of ATEs

1. Exact matching: 1777.63
2. NN matching, inverse variance: -526.95
3. NN matching, mahalanobis: -492.82
4. NN matching, pscore: -201.03
5. Inverse pscore weighting: -196.89
6. IPW regression: -196.89
7. Regression: -5.85
8. Regression 1-step: -5.85

Summary of ATEs

Why such large differences between linear (unweighted) regression and other approaches?

Problem is due to common support. Without weighting, the treated group looks very different than the control group, and standard OLS (without weights) doesn't do anything to account for this.

So what have we learned?

Key assumptions for causal inference

1. Selection on observables
2. Common support

These become more nuanced but the intuition is the same in almost all questions of causal inference.

Causal effect assuming selection on observables

If we assume selection on observables holds, then we only need to condition on the relevant covariates to identify a causal effect. But we still need to ensure common support...

1. Matching
2. Reweighting
3. Regression