Basic Use of the RItools Package *

Jake Bowers and Ben Hansen

June 12, 2006, Draft Version 26

This document describes how to use `RItools` combined with `optmatch` to first balance, and then estimate effects. For background and inspiration see (Rosenbaum, 2002b). For other uses of this set of tools see (Bowers and Hansen, 2005b,a).

The substantive question we pursue here is whether the change in rules at the United States Food and Drug Administration (FDA) led to more safety problems with drugs approved after the rule change. The relevant feature of the rule change, from the perspective of drug safety, is that, for the first time in 60 years the FDA was given deadlines for approval of drugs, and drug companies were charged fees to aid the FDA in this task. Thus, some observers have worried that increased speed combined with a regulator funded by the regulated would cause drugs that ought not to be approved, to be approved nonetheless.

```
> load("fdapdufa.rda")
```

In the language of causal inference, the "treatment" in this case is the change in regulatory regime, which, as far as we know, took place on Sept 1, 1992. New drugs submitted for approval between Sept 1, 1988 and Aug 31, 1992 are considered to be the "control" group. New drugs submitted between Sept 1, 1992 and Sept 1, 1996 are the "treatment" group. The outcomes are the number of drugs withdrawn from the market for safety reasons. In the control group there were 5 withdrawals by the year 200?4?, and in the treatment group there were 7 withdrawals by that same year.

```
> table(fdapdufa$pdufaF, fdapdufa$anywithdrawF)
```

```
             AnyWithdraw NoWithdraw
  PostDiscont           7        114
  PreDiscont            5         93
```

Checking Balance

We have a long list of covariates that we expect might have something to do with the safety of the drugs after approval and/or the speed of drug approval. We will look for balance on those covariates within two strata at first: The FDA has two different sets of deadline: one for drugs

*Draft Version 26

it called "priority" (which are supposed to be rushed through the system as fast as possible) and other for drugs it calls "standard".

The main tool currently in the `RItools` package is called `xBalance`. It uses a formula interface to assess differences between groups defined by the response (in this case on the treatment variable, "pdufaZ") in terms of the covariates (in this case, the list of covariates includes variables such as "media" [coverage about the primary indication (i.e. disease treated by the drug) for the drug in the newspaper and television], "medline1safetytotal" [number of mentions of the disease in the context of safety or danger in medline articles], etc.. The main output of `xBalance` so far is a list of tests of the bivariate relationship between the covariate listed and, in this case, the treatment variable, conditional on a stratification variable. The covariates are listed connected by "+" and, if the argument `chisquare.test` is `TRUE`, then balance on their linear combination is tested. If such a test is not desired, the variables still must be listed connected with "+" signs even though the test will only return the set of bivariate tests.

The tests themselves are randomization tests in the style presented by Rosenbaum (2002b), especially his chapter 2, §2.4.4. More specifically, the current version of this function uses normal approximations to the exact distributions for test statistics that are sum statistics of the form $t(\mathbf{Z}, \mathbf{r}) = \mathbf{Z}^T q$ such as the Mantel-Haenszel and the Fisher test.[1] For more details on these test statistics see Hansen (2006).

```
> balpsformula <- formula(pdufaZ ~ media + I(prevgenxA/1000) +
+     prevgenxANA + dthrtgenA + dthrtgenANA + I(hhospdisc/1e+05) +
+     orderent + fsubmitsA + fsubmitsANA + I(medline1total/1000) +
+     I(medline3total/1000) + I(medline1safetytotal/10000) +
+     I(medline3safetytotal/1000) + factor(discodeA))
> balonpriority <- xBalance(balpsformula, ~priorityF,
+     data = fdapdufa, chisquare.test = TRUE)
> print(methods(class = "newbal"))

[1] plot.newbal  print.newbal

> options(width = 80)
> print(round(balonpriority, 2))

                          pre.difference pre.sig post.difference post.sig
media                               0.09                   0.10
I(prevgenxA/1000)                   0.16                   0.15
prevgenxANA                         0.05                   0.04
dthrtgenA                          -0.21                  -0.19
dthrtgenANA                         0.28     *             0.26      .
I(hhospdisc/1e+05)                 -0.10                  -0.11
orderent                           -0.03                  -0.06
fsubmitsA                           0.03                   0.05
```

[1] We plan to add an option `slow=TRUE,FALSE` to the package which would call `fisher.test` or `mantelhaen.test` in the case where an analyst worries about our normal approximations; or perhaps allow direct simulation or sampling from the exact distribution for more complex test statistics.

fsubmitsANA	-0.31	*	-0.34	*
I(medline1total/1000)	-0.17		-0.16	
I(medline3total/1000)	-0.26	.	-0.25	.
I(medline1safetytotal/10000)	-0.27	*	-0.25	.
I(medline3safetytotal/1000)	-0.23	.	-0.21	
factor(discodeA)1600	0.21		0.20	
factor(discodeA)2300	-0.09		-0.11	
factor(discodeA)2500	-0.03		-0.03	
factor(discodeA)3100	0.08		0.08	
factor(discodeA)3230	-0.10		-0.11	
factor(discodeA)3300	-0.02		-0.01	
factor(discodeA)3500	-0.35	*	-0.32	*
factor(discodeA)3700	-0.17		-0.15	
factor(discodeA)3800	0.17		0.18	
factor(discodeA)4050	0.17		0.17	
factor(discodeA)4100	-0.21		-0.21	
factor(discodeA)4140	-0.02		-0.02	
factor(discodeA)4400	-0.15		-0.16	
factor(discodeA)5200	-0.02		-0.02	
factor(discodeA)5260	-0.21		-0.22	.
factor(discodeA)5400	-0.03		-0.03	
factor(discodeA)5500	0.04		0.02	
factor(discodeA)5610	0.11		0.11	
factor(discodeA)6100	-0.02		-0.02	
factor(discodeA)6140	0.10		0.12	
factor(discodeA)6200	0.17		0.18	
factor(discodeA)6400	-0.34	*	-0.35	*
factor(discodeA)6500	-0.21		-0.22	.
factor(discodeA)6640	-0.02		-0.03	
factor(discodeA)7500	-0.21		-0.21	
factor(discodeA)10100	-0.02		-0.02	
factor(discodeA)10400	-0.03		-0.03	
factor(discodeA)10800	-0.21		-0.20	
factor(discodeA)10820	-0.21		-0.22	.
factor(discodeA)10900	0.05		0.04	
factor(discodeA)11600	-0.10		-0.10	
factor(discodeA)11700	-0.21		-0.22	.
factor(discodeA)12300	0.17		0.18	
factor(discodeA)13000	-0.21		-0.21	
factor(discodeA)13100	0.17		0.17	
factor(discodeA)13120	-0.02		-0.03	
factor(discodeA)80200	0.21		0.21	
factor(discodeA)80300	-0.02		-0.01	
factor(discodeA)80700	0.21		0.20	
factor(discodeA)82200	0.28	*	0.29	*
factor(discodeA)85300	-0.04		-0.04	
factor(discodeA)88888	-0.06		-0.08	

```
> options(width = 60)
```

This suggests that merely stratifying the variables by priority/standard has little to do with their being submitted before the regime change. It also suggests that drugs for certain diseases are more likely to be submitted before rather than after this change (see the "pre.sig" and "post.sig" columns and the associated "Significance" markings).

Matching

In order to balance these covariates, we use `fullmatch` from the `optmatch` package. This involves (1) creating a score on which to match (in our case this score is a propensity score [cite], but using the `brlr` command instead of `glm` because of the large number of diseases and relatively small number of drugs), (2) creating a list of distance matrices, (3) creating a caliper (of 3 sd) by setting distances of greater than 4 to Inf, and (4) calling `fullmatch`. (The functions included here are not as good as the ones in the now current version of the optmatch package). We created the score elsewhere and have attached it to the data.frame under the name "ps4yr".

```
> absDist <- function(trtvar, data, scalarname,
+     cal = Inf) {
+     sclr <- data[names(trtvar), scalarname]
+     names(sclr) <- names(trtvar)
+     dist <- abs(outer(sclr[trtvar], sclr[!trtvar],
+         "-"))
+     dist/(dist <= cal)
+ }
> psdistlist <- makedist(pdufaT ~ priorityF, data = fdapdufa,
+     fn = absDist, scalarname = "ps4yr", cal = 3)
> thefm <- fullmatch(psdistlist)
> table(fdapdufa[names(thefm), "pdufaT"], thefm)

       thefm
        Pr.1 Pr.10 Pr.13 Pr.2 Pr.21 Pr.3 Pr.30 Pr.31 Pr.33
  FALSE    1     2     1    5     1    1     2     6    15
  TRUE    13     1     5    1     3    2     1     1     1
       thefm
        Pr.4 Pr.5 Pr.6 St.01 St.02 St.1 St.10 St.14 St.2
  FALSE    1    1    1     1     1    1     1     1    1
  TRUE     9    1    1     0     0   21     1     4   15
       thefm
        St.21 St.24 St.27 St.28 St.37 St.39 St.40 St.46
  FALSE     1     1     1     1     1     1     1    36
  TRUE      1     1     1     2     1     1     1     1
       thefm
        St.5 St.53 St.59 St.6 St.7 St.71 St.75 St.77 St.8
  FALSE    1     1     3    1    1     1     1     1    1
```

```
  TRUE    14     1     1    4    2     1     1     1    1
      thefm
       St.9
  FALSE    1
  TRUE     6
```

```
> good <- names(thefm)[matched(thefm)]
> table(matched(thefm))
```

```
FALSE  TRUE
    2   217
```

We can see that the matching routine excluded two drugs from the control group, and it also created a variety of sized matched sets. Now, we call `xBalance` to check balance given the matching and demonstrate the print method for the `newbal` class of object. The balance improves a lot!

```
> thefmbal <- xBalance(balpsformula, ~thefm, data = fdapdufa,
+     chisquare.test = TRUE)
> pchisq(attr(thefmbal, "post.chisquare"), df = attr(thefmbal,
+     "post.df"), lower = FALSE)
```

```
[1] 1
```

```
> options(width = 80)
> print(round(thefmbal, 2))
```

```
                              pre.difference pre.sig post.difference post.sig
media                                   0.08                    0.08
I(prevgenxA/1000)                       0.17                    0.08
prevgenxANA                             0.04                    0.09
dthrtgenA                              -0.19                   -0.01
dthrtgenANA                             0.26     .              0.03
I(hhospdisc/1e+05)                     -0.10                   -0.03
orderent                               -0.02                    0.11
fsubmitsA                               0.03                   -0.14
fsubmitsANA                            -0.29     *              0.12
I(medline1total/1000)                  -0.17                   -0.04
I(medline3total/1000)                  -0.21                    0.03
I(medline1safetytotal/10000)           -0.24     .             -0.02
I(medline3safetytotal/1000)            -0.19                    0.06
factor(discodeA)1600                    0.21                    0.06
factor(discodeA)2300                   -0.10                    0.08
factor(discodeA)2500                   -0.03                   -0.07
factor(discodeA)3100                    0.07                   -0.01
factor(discodeA)3230                   -0.11                   -0.12
```

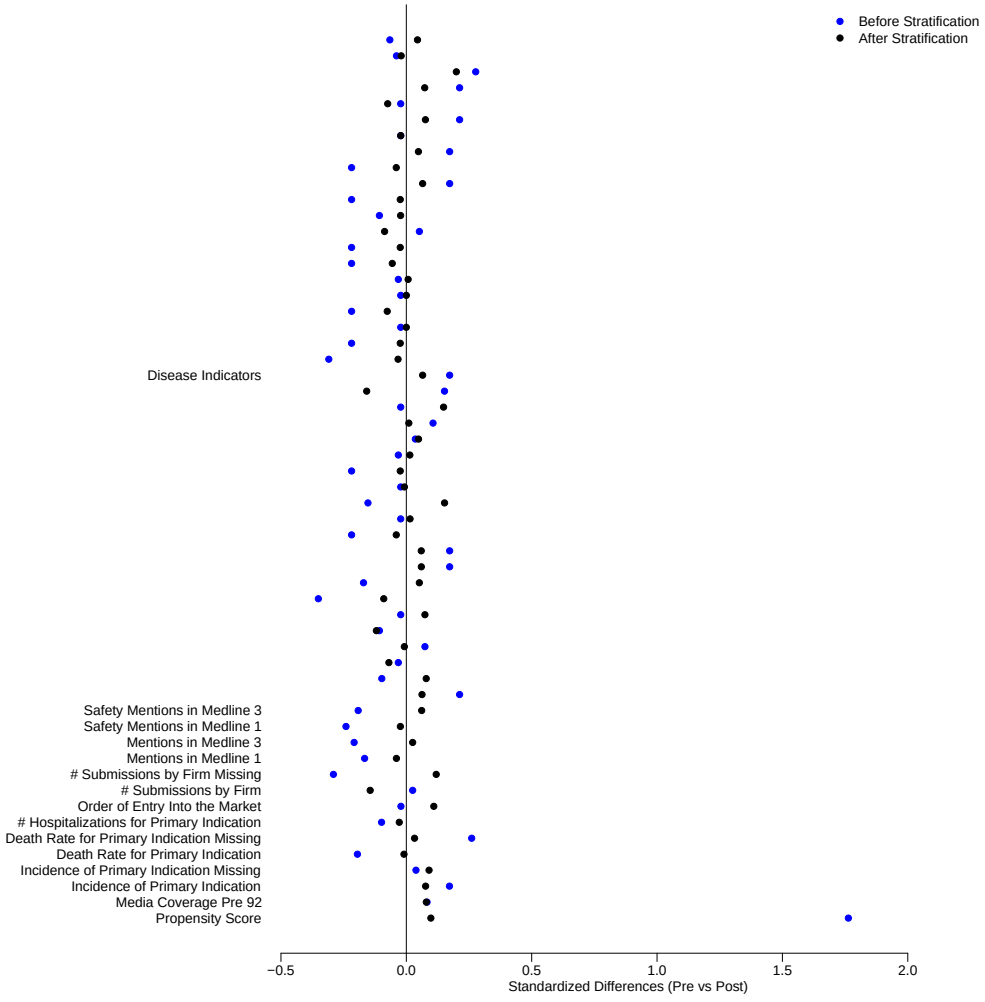
```
factor(discodeA)3300                -0.02                 0.07
factor(discodeA)3500                -0.35     *          -0.09
factor(discodeA)3700                -0.17                 0.05
factor(discodeA)3800                 0.17                 0.06
factor(discodeA)4050                 0.17                 0.06
factor(discodeA)4100                -0.22                -0.04
factor(discodeA)4140                -0.02                 0.01
factor(discodeA)4400                -0.15                 0.15
factor(discodeA)5200                -0.02                -0.01
factor(discodeA)5260                -0.22                -0.02
factor(discodeA)5400                -0.03                 0.01
factor(discodeA)5500                 0.04                 0.05
factor(discodeA)5610                 0.11                 0.01
factor(discodeA)6100                -0.02                 0.15
factor(discodeA)6140                 0.15                -0.16
factor(discodeA)6200                 0.17                 0.07
factor(discodeA)6400                -0.31     *          -0.03
factor(discodeA)6500                -0.22                -0.02
factor(discodeA)6640                -0.02                 0.00
factor(discodeA)7500                -0.22                -0.08
factor(discodeA)10100               -0.02                 0.00
factor(discodeA)10400               -0.03                 0.01
factor(discodeA)10800               -0.22                -0.06
factor(discodeA)10820               -0.22                -0.02
factor(discodeA)10900                0.05                -0.09
factor(discodeA)11600               -0.11                -0.02
factor(discodeA)11700               -0.22                -0.02
factor(discodeA)12300                0.17                 0.07
factor(discodeA)13000               -0.22                -0.04
factor(discodeA)13100                0.17                 0.05
factor(discodeA)13120               -0.02                -0.02
factor(discodeA)80200                0.21                 0.08
factor(discodeA)80300               -0.02                -0.07
factor(discodeA)80700                0.21                 0.07
factor(discodeA)82200                0.28     *           0.20
factor(discodeA)85300               -0.04                -0.02
factor(discodeA)88888               -0.07                 0.04
```

```
> options(width = 60)
```

We can also show this information in a graph (using our proposed plot method). I've added the propensity score to the list of covariates here.

```
> balpsformula <- update(balpsformula, . ~ ps4yr +
+     .)
> thefmbal <- xBalance(balpsformula, ~thefm, data = fdapdufa,
+     chisquare.test = TRUE)
```

```
> somelabs <- c("Propensity Score", "Media Coverage Pre 92",
+     "Incidence of Primary Indication", "Incidence of Primary Indication Missing",
+     "Death Rate for Primary Indication", "Death Rate for Primary Indication Missing",
+     "# Hospitalizations for Primary Indication",
+     "Order of Entry Into the Market", "# Submissions by Firm",
+     "# Submissions by Firm Missing", "Mentions in Medline 1",
+     "Mentions in Medline 3", "Safety Mentions in Medline 1",
+     "Safety Mentions in Medline 3", rep("", 20),
+     "Disease Indicators", rep("", 21))
> par(mar = c(3, 17, 0, 0), mgp = c(1.5, 0.5, 0),
+     oma = c(0, 0, 0, 0))
> plot(thefmbal, thelab = somelabs, cex.axis = 1)
```



Estimating Effects

So now that we've achieved some balance by the combination of choosing a narrow window around a relatively deterministic discontinuity combined with full matching, we can estimate some effects.

In this case our outcome is binary, and since we'd like to stay in the potential outcomes causal framework [cite] we would like to use the attributable effects estimand (Rosenbaum, 2002a, 2001, See, for example) and Rosenbaum (2002b, Chapter 5).

We can produce this estimate using `rdz` or calling `mantelhean.test` directly — since we are not testing that many covariates at once.[2]

```
> tc <- table(pdufa = fdapdufa[good, "pdufaF"],
+     withdraw = fdapdufa[good, "anywithdraw"],
+     match = thefm[good, drop = TRUE], exclude = NULL)
> dimnames(tc)

$pdufa
[1] "PostDiscont" "PreDiscont"

$withdraw
[1] "0" "1"

$match
 [1] "Pr.1"  "Pr.10" "Pr.13" "Pr.2"  "Pr.21" "Pr.3"  "Pr.30"
 [8] "Pr.31" "Pr.33" "Pr.4"  "Pr.5"  "Pr.6"  "St.1"  "St.10"
[15] "St.14" "St.2"  "St.21" "St.24" "St.27" "St.28" "St.37"
[22] "St.39" "St.40" "St.46" "St.5"  "St.53" "St.59" "St.6"
[29] "St.7"  "St.71" "St.75" "St.77" "St.8"  "St.9"

> tc[1, , ]

        match
withdraw Pr.1 Pr.10 Pr.13 Pr.2 Pr.21 Pr.3 Pr.30 Pr.31 Pr.33
       0   13     1     5    1     1    2     1     1     1
       1    0     0     0    0     2    0     0     0     0
        match
withdraw Pr.4 Pr.5 Pr.6 St.1 St.10 St.14 St.2 St.21 St.24
       0    8    1    1   20     1     4   14     1     0
       1    1    0    0    1     0     0    1     0     1
        match
withdraw St.27 St.28 St.37 St.39 St.40 St.46 St.5 St.53
       0     1     2     1     1     1     1   13     1
       1     0     0     0     0     0     0    1     0
        match
withdraw St.59 St.6 St.7 St.71 St.75 St.77 St.8 St.9
       0     1    4    2     1     1     1    1    6
       1     0    0    0     0     0     0    0    0
```

[2] We also have done some preliminary work to use `xBalance` for this task since we need to (1) indicate which units are eligible for attribution of effects and (2) adjust the responses of those units appropriately to reflect a range of null hypotheses about treatment effects.

```
> sum(tc[, "1", ])
```

```
[1] 12
```

This $2 \times 2 \times S$ table format is necessary to use `rdz` or `mantelhaen.test`. Now, although there are other ways to estimate attributable effects (i.e. shortcuts using the notion of "asymptotic separability" introduced by Rosenbaum (2001, 2002a)— seeBowers and Hansen (2005b,a)), we are going to list all of the possible ways to attribute between 0 and 7 withdrawals to the specific pattern of withdrawals among drugs submitted and approved after Sept 1, 1992 actually observed in our data.

The way we do this listing does not scale well to larger number of attributions, and it is idiosyncratic to a situation where the only eligible units are in 6 sets. Any ideas about how to generalize this would be greatly appreciated!

```
> sum(0:7 * c(1, 36, 35, 34, 33, 32, 31, 30))
```

```
[1] 896
```

```
> deltas <- expand.grid(0:7, 0:7, 0:7, 0:7, 0:7,
+     0:7)
> deltassum <- rowSums(deltas)
> deltas <- deltas[deltassum <= 7, ]
> clevs <- dimnames(tc)[[3]][tc["PostDiscont", "1",
+     ] > 0]
> names(deltas) <- clevs
> junk <- matrix(tc["PostDiscont", "1", clevs],
+     nrow = nrow(deltas), ncol = ncol(deltas),
+     byrow = TRUE)
> gooddeltas <- rowSums((deltas <= junk))
> table(gooddeltas)
```

```
gooddeltas
  3   4   5   6
 80 740 800  96
```

```
> table(unlist(deltas[gooddeltas == 6, ]))
```

```
  0   1   2
272 272  32
```

```
> deltas <- deltas[gooddeltas == 6, ]
> table(unlist(deltas))
```

```
  0   1   2
272 272  32
```

So, although there are many thousands of ways to conceivably attribute between 0 and 7 withdrawals to 6 sets, in fact, given our fixed outcomes, there are only 96 different attribution scenarios to test (We call an attribution scenario $\boldsymbol{\delta}$ since it is a vector indicating which units can have their responses adjusted or not.)

Now we will use the function `rdz` to test each of those 96 scenarios — each one is an atomic hypothesis. This function requires a $2 \times 2 \times S$ table of treatment by responses by strata and also a $2 \times 2 \times S \times \#A$ array containing the different attribution scenarios.

```
> theAs <- rowSums(deltas)
> myattrib.arr <- array(0, dim = c(2, 2, 34, nrow(deltas)),
+     dimnames = list(0:1, 0:1, dimnames(tc)[[3]],
+         1:nrow(deltas)))
> myattrib.arr[1, 1, , ] <- tc["PostDiscont", "0",
+     ]
> myattrib.arr[2, 1, , ] <- tc["PostDiscont", "1",
+     ]
> myattrib.arr[2, 1, clevs, ] <- myattrib.arr[2,
+     1, clevs, ] - t(deltas)
> myattrib.arr[2, 2, clevs, ] <- t(deltas)
> thezs <- rdz(tc, myattrib.arr)
> aes1 <- data.frame(A = theAs, Z = thezs)
> aes1$p <- pnorm(abs(aes1$Z), lower = FALSE) *
+     2
> tapply(aes1$p, aes1$A, range)

$`0`
[1] 0.2699236 0.2699236

$`1`
[1] 0.2794225 0.4930960

$`2`
[1] 0.2934774 0.6871424

$`3`
[1] 0.3095547 0.9175236

$`4`
[1] 0.3370392 0.9975618

$`5`
[1] 0.4901884 0.9457969

$`6`
[1] 0.6554437 0.9425480
```

```
$`7`
[1] 0.6308128 0.6308128
```

So, we cannot reject any of the null hypotheses posed here: it is probable that no drugs were withdrawn because of the change in law, but it is more probable than some were.

References

Bowers, J. and Hansen, B. B. (2005a), "Attributing Effects to A Cluster Randomized Get-Out-The-Vote Campaign: An Application of Randomization Inference Using Full Matching." Presented at annual meeting of the Political Methodology Section of the American Political Science Association.

— (2005b), "Attributing Effects to a Get-Out-The-Vote Campaign Using Full Matching and Randomization Inference," Prepared for presentation at the Annual Meeting of the Midwestern Political Science Association.

Hansen, B. B. (2006), "Appraising Covariate Balance after Assignment to Treatment by Groups," Tech. Rep. 436, University of Michigan, Statistics Department.

Rosenbaum, P. R. (2001), "Effects Attributable to Treatment: Inference in experiments and observational studies with a discrete pivot," *Biometrika*, 88, 219–231.

— (2002a), "Attributing Effects to Treatment in Matched Observational Studies," *Journal of the American Statistical Association*, 97, 183–192.

— (2002b), *Observational Studies*, Springer.