Propensity Scores with R Tutorial

Lucy D'Agostino McGowan

November 26, 2015

Install R Packages

For this tutorial, you will need Frank Harrell's Regression Modeling Strategies package, Matt Shotwell's package to read in SAS datasets, and the MatchIt package for matching. If this has not yet been installed, run `install.packages("rms")`, `install.packages("sas7bdat")`, and `install.packages("MatchIt")`. Otherwise, run the following,

```
require(rms)
require(sas7bdat)
require(MatchIt)
```

Read in your data

There are many different ways to read in data. Here, I will demonstrate how to read in .csv files and SAS data files. Here `dat` is what I am naming the dataset in `R`. The file path for my data is "/Users/lucymcgowan/Documents/Consulting/Edwards/data.csv". I will first use the `read.csv()` function,

```
dat<-read.csv("/Users/lucymcgowan/Documents/Consulting/Edwards/data.csv")
```

If the data was a SAS dataset, you can import it as follows,

```
dat<-read.sas7bdat("/Users/lucymcgowan/Documents/Consulting/Edwards/data.sas7bdat")
```

Run descriptive Statistics

To look at descriptives, we can use Harrell's `describe()` function,

```
describe(dat)
```

dat
8 Variables 1000 Observations

id

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
1000	0	1000	1	500.5	50.95	100.90	250.75	500.50	750.25	900.10	950.05

```
lowest :    1    2    3    4    5, highest:  996  997  998  999 1000
```

age

n	missing	unique	Info	Mean	.05	.10	.25	.50	.75	.90	.95
793	207	56	1	30.37	13.6	17.0	24.0	30.0	37.0	44.0	47.0

```
lowest :  4  5  6  7  8, highest: 55 56 60 62 65
```

sex

n	missing	unique
1000	0	2

```
0 (485, 48%), 1 (515, 52%)
```

---

dx_diabetes

n	missing	unique	Info	Mean
938	62	2	0.51	1.215

```
1 (736, 78%), 2 (202, 22%)
```

---

dx_chf

n	missing	unique
1000	0	2

```
0 (787, 79%), 1 (213, 21%)
```

---

smoking

n	missing	unique
1000	0	2

```
0 (890, 89%), 1 (110, 11%)
```

---

race

n	missing	unique
1000	0	2

```
0 (190, 19%), 1 (810, 81%)
```

---

treat

n	missing	unique
1000	0	2

```
0 (533, 53%), 1 (467, 47%)
```

Looking at this, we see that age and diabetes diagnosis both have missing values. Let's look more at that! We can use Harrell's `naclus` and `naplot` functions to look at the fraction missing in each variable,

```
n<-naclus(dat)
a<-naplot(n, which=('na per var'))
```

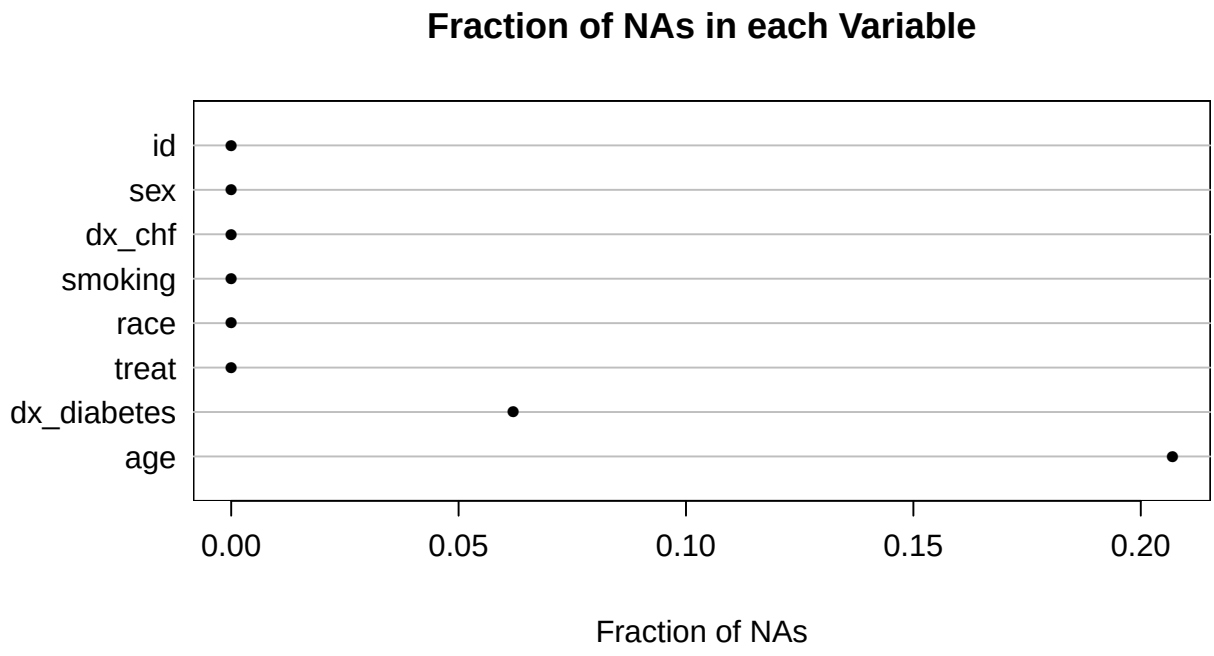

Figure 1: This plot shows us the fraction of missing for each variable. We see that diabetes has 10 percent missing and age has about 20 percent missing.

Multiple Imputation

To perform multiple imputation, we will use Harrell's `aregImpute` function. This will use predictive mean matching by default. Because the variable with the largest missingness has 20% missing, we will perform 20 imputations. To impute all variables with one line of code, we will put everything on the right side of the equation, separated by +. The continuous covariates are fit with restricted cubic splines. The `nk` option lets us set the number of knots. I will set it to the default, 4. I will name my imputation object `dat.imp`. We can use this later to perform the propensity score analysis

```
set.seed(91690)
dat.imp <- aregImpute(~age + sex + dx_diabetes + dx_chf + smoking + race + treat,
    n.impute = 20, nk = 4, data = dat, pr = F)
```

Propensity Scores

To generate the propensity scores, we will fit a logistic regression. To do this, we will use Harrell's `lrm()` function. In order to incorporate the multiple imputations, we will use Harrell's `fit.mult.impute()`. I am going to fit the continuous covariate (age) as a restricted cubic spline with 3 knots with the `rcs()` function. Here is the code,

```
fit <- fit.mult.impute(treat ~ rcs(age, 3) + sex + race + smoking + dx_diabetes +
    dx_chf, fitter = lrm, data = dat, xtrans = dat.imp, pr = F)
```

Notice that we incorporated the imputation object with the `xtrans` option, and we set the `fitter` to `lrm`, to invoke a logistic regression model. Lets look at that model. I am going to print it with the latex function, so it looks pretty,

```
print(fit,latex=T)
```

Logistic Regression Model

```
fit.mult.impute(formula = treat ~ rcs(age, 3) + sex + race +
    smoking + dx_diabetes + dx_chf, fitter = lrm, xtrans = dat.imp,
    data = dat, pr = F)
```

		Model Likelihood Ratio Test		Discrimination Indexes		Rank Discrim. Indexes	
Obs	1000	LR χ^2	228.24	R^2	0.272	C	0.763
0	533	d.f.	7	g	1.302	D_{xy}	0.527
1	467	$\Pr(>\chi^2) < 0.0001$		g_r	3.677	γ	0.529
$\max\lvert\frac{\partial \log L}{\partial \beta}\rvert$	9×10^{-12}			g_p	0.261	τ_a	0.262
				Brier	0.197		

	Coef	S.E.	Wald Z	$\Pr(>\lvert Z\rvert)$
Intercept	1.7572	0.5111	3.44	0.0006
age	0.0345	0.0183	1.89	0.0591
age'	-0.0132	0.0214	-0.62	0.5367
sex=1	0.9222	0.1465	6.29	< 0.0001
race=1	-1.4571	0.1976	-7.37	< 0.0001
smoking=1	0.8377	0.2349	3.57	0.0004
dx_diabetes	-1.7304	0.2043	-8.47	< 0.0001
dx_chf=1	-0.8738	0.1820	-4.80	< 0.0001

Now let's extract the propensity scores using the `predict()` function.

```
dat$p<-predict(fit)
```

Let's look at the distribution of propensity scores for the treatment and control group using the `hist()` and `plot()` functions,

```
p1<-hist(dat$p[dat$treat==1])
p2<-hist(dat$p[dat$treat==0])
```

```
plot(p1, col = rgb(0, 0, 1, 1/4), ylim = c(0, 150), xlim = c(-4, 4), main = "Propensity Score Distributio
    xlab = "Propensity Score (logit)")
plot(p2, col = rgb(1, 0, 0, 1/4), add = T)
```

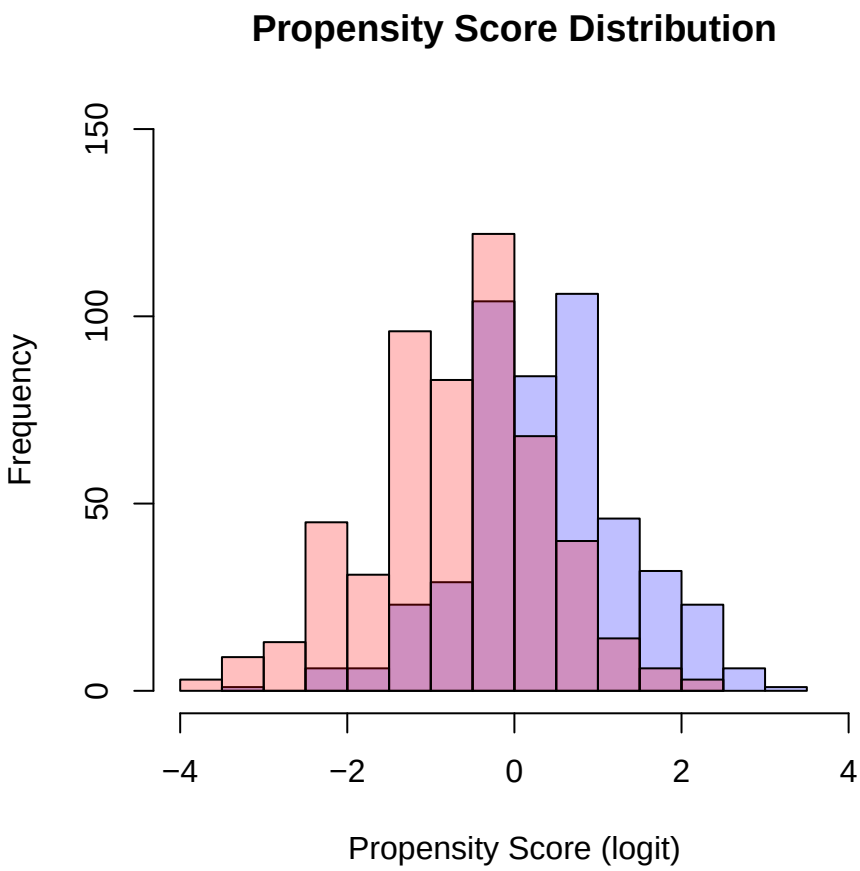

Figure 2: The control group propensity scores are shown in red, and the treatment group in black

Matching!

Now let's match them with a caliper of $.2 \times SD$,

```
prop <- data.frame(id = dat$id, treat = dat$treat, p = dat$p)
match <- matchit(treat ~ p, data = prop, method = "nearest", caliper = 0.2)
summary(match)

##
## Call:
## matchit(formula = treat ~ p, data = prop, method = "nearest",
##     caliper = 0.2)
##
## Summary of balance for all data:
##          Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean
## distance        0.5820        0.3663     0.2068    0.2157  0.2284   0.2165
## p               0.3754       -0.6713     1.0557    1.0466  1.0052   1.0525
##          eQQ Max
## distance  0.2798
## p         1.5028
##
```

```
##
## Summary of balance for matched data:
##          Means Treated Means Control SD Control Mean Diff eQQ Med eQQ Mean
## distance        0.5086        0.4829     0.1790    0.0257  0.0223   0.0263
## p               0.0327       -0.0884     0.8136    0.1211  0.0977   0.1248
##          eQQ Max
## distance  0.0463
## p         0.6973
##
## Percent Balance Improvement:
##          Mean Diff. eQQ Med eQQ Mean eQQ Max
## distance    88.0781 90.2465  87.8549 83.4662
## p           88.4336 90.2814  88.1422 53.5993
##
## Sample sizes:
##           Control Treated
## All           533     467
## Matched       312     312
## Unmatched     221     155
## Discarded       0       0
```

Let's get a dataset of only the matched pairs

```
m.dat <- match.data(match)
m.dat <- dat[dat$id %in% m.dat$id, ]
```

Assess Balance

Now let's look at the tables before and after matching

```
vars <- Cs(age, race, sex, smoking, dx_diabetes, dx_chf)
summByDx <- summaryM(as.formula(paste(paste(vars, collapse = "+"), "~ treat")),
    data = dat, overall = T)
summByDx2 <- summaryM(as.formula(paste(paste(vars, collapse = "+"), "~ treat")),
    data = m.dat, overall = T)
latex(summByDx, file = "", center = "centering", what = "%", where = "H", caption = "Pre-Matching Descrip
```

Table 1: Pre-Matching Descriptive Statistics. $_a\ b\ _c$ represent the lower quartile a, the median b, and the upper quartile c for continuous variables. N is the number of non–missing values. Numbers after percents are frequencies.

	N	0	1	Combined
		$N = 533$	$N = 467$	$N = 1000$
age	793	$_{23}$ 29 $_{36}$	$_{24}$ 32 $_{38}$	$_{24}$ 30 $_{37}$
race	1000	90% (478)	71% (332)	81% (810)
sex	1000	43% (231)	61% (284)	52% (515)
smoking	1000	9% (47)	13% (63)	11% (110)
dx_diabetes : 2	938	33% (162)	9% (40)	22% (202)
dx_chf	1000	27% (145)	15% (68)	21% (213)

```
latex(summByDx2, file = "", center = "centering", what = "%", where = "H", caption = "Post-Matching Descr
```

Table 2: Post-Matching Descriptive Statistics. $_a\ b\ _c$ represent the lower quartile a, the median b, and the upper quartile c for continuous variables. N is the number of non–missing values. Numbers after percents are frequencies.

	N	0	1	Combined
		$N = 312$	$N = 312$	$N = 624$
age	492	$_{24}$ 31 $_{38}$	$_{24}$ 32 $_{38}$	$_{24}$ 31 $_{38}$
race	624	86% (268)	82% (257)	84% (525)
sex	624	50% (157)	53% (166)	52% (323)
smoking	624	12% (36)	12% (38)	12% (74)
dx_diabetes : 2	588	12% (35)	13% (38)	12% (73)
dx_chf	624	18% (56)	18% (55)	18% (111)

We can also look at the propensity scores post matching,

```
p1<-hist(m.dat$p[dat$treat==1])
p2<-hist(m.dat$p[dat$treat==0])
```

```
plot(p1, col = rgb(0, 0, 1, 1/4), ylim = c(0, 150), xlim = c(-4, 4), main = "Post-Matching Propensity Sco
    xlab = "Propensity Score (logit)")
plot(p2, col = rgb(1, 0, 0, 1/4), add = T)
```

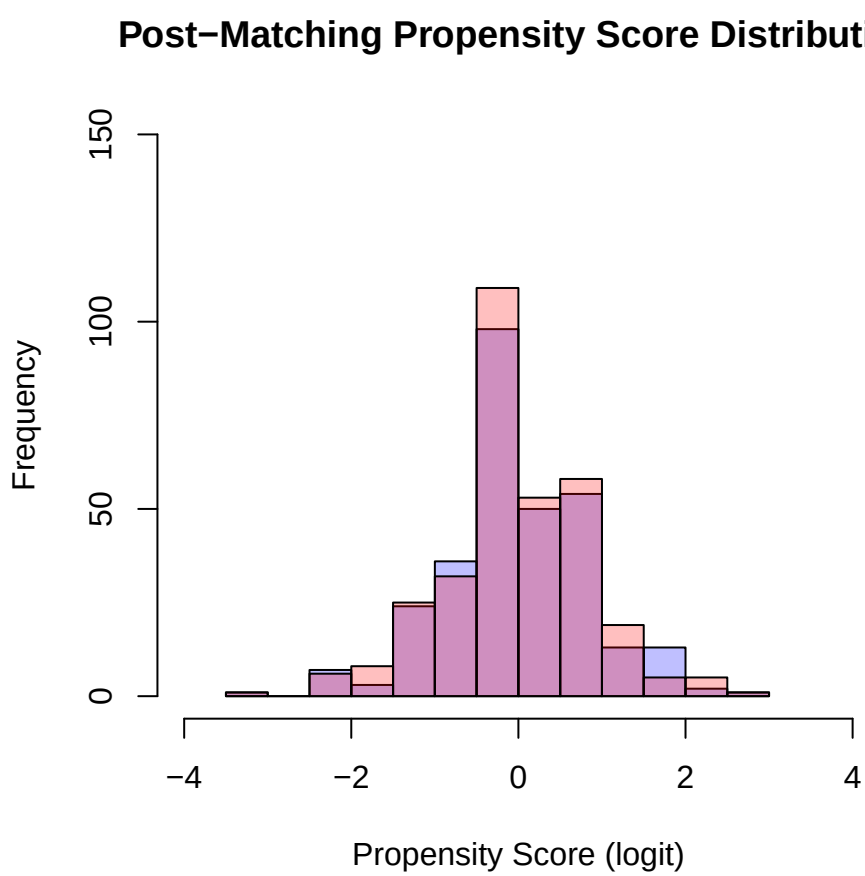

Figure 3: The control group propensity scores are shown in red, and the treatment group in black