

# A Vignette for `iWeigReg` - version 1.0

Zhiqiang Tan<sup>1</sup>

February 2013

## 1 Introduction

The R package `iWeigReg` – version 1.0 can be used for two main tasks:

- to estimate the mean of an outcome in the presence of missing data,
- to estimate the average treatment effect in causal inference.

There are 4 functions provided for the first task:

- `mn.lik`: the non-calibrated (or non-doubly robust) likelihood estimator in Tan (2006),
- `mn.clik`: the calibrated (or doubly robust) likelihood estimator in Tan (2010),
- `mn.reg`: the non-calibrated (or non-doubly robust) regression estimator,
- `mn.creg`: the calibrated (or doubly robust) regression estimator in Tan (2006).

In parallel, there are also 4 functions for the second task, `ate.lik`, `ate.clik`, `ate.reg`, and `ate.creg`. Currently, the treatment is assumed to be binary (i.e., untreated or treated). Extensions to multi-valued treatments will be incorporated in later versions.

In general, the function recommended to use is the calibrated (or doubly robust) likelihood estimator, `mn.clik` or `ate.clik`, which is a two-step procedure with the first step corresponding to the non-calibrated (or non-doubly robust) likelihood estimator. The calibrated (or doubly robust) regression estimator, `mn.creg` or `ate.creg`, is a close relative to the calibrated likelihood estimator, but may sometimes yield an estimate lying outside the sample range, for example, outside the unit interval  $(0, 1)$  for estimating the mean of a binary outcome.

---

<sup>1</sup>Department of Statistics, Rutgers University. Address: 110 Frelinghuysen Road, Piscataway, NJ 08854. E-mail: [ztan@stat.rutgers.edu](mailto:ztan@stat.rutgers.edu).

The package also provides two functions, `mn.HT` and `ate.HT`, for the Horvitz-Thompson estimator, i.e., the unaugmented inverse probability weighted estimator. These functions can be used for balance checking.

## 2 An example

We illustrate the use of the package for causal inference on a simulated dataset according to Kang & Schafer (2007). The use of the package is similar in the missing-data setup. The dataset, `KS.data`, is included as part of the package.

```
> library(iWeigReg)
> data(KS.data)
> attach(KS.data)
```

The following shows the first 3 rows of the dataset:

```
> KS.data[1:3,]
      y tr      z1      z2      z3      z4      x1
1 239.1378 0 1.2629543 -0.2868516 0.444345820 -0.5380638 1.8803861
2 235.9742 1 -0.3262334 1.8411069 0.011929380 0.4965946 0.8494921
3 223.3349 0 1.3297993 -0.1567643 -0.009280045 -1.4419661 1.9442954
      x2      x3      x4
1 -0.06324094 0.2411617 0.9192097
2 1.069393334 0.2158319 1.2474323
3 -0.03279393 0.2154673 0.8465168
```

For the setup of causal inference, suppose that `y` gives the observed outcome, `tr` the treatment indicator, `(z1, z2, z3, z4)` the covariates leading to correct models, and `(x1, x2, x3, x4)` the covariates leading to misspecified models. The true value of the average treatment effect is 0.

```
> n=1000
> z=cbind(z1,z2,z3,z4)
> x=cbind(x1,x2,x3,x4)
```

## 2.1 Fitting models

Suppose that a misspecified propensity score model is fitted. (A logistic regression model is *assumed* to be used, although other types of regression models may be allowed in later versions of the package.) The model matrix is recorded as `X`, and the fitted propensity scores are recorded in `ppi.hat`.

```
> ppi.glm <- glm(tr~x, family=binomial(link=logit))
> X <- model.matrix(ppi.glm)
> ppi.hat <- ppi.glm$fitted
```

Suppose that a correct outcome regression model is fitted, separately in the treated group (`tr==1`) and the untreated group (`tr==0`). The fitted outcome regression functions are recorded as `eta1.hat` and `eta0.hat` respectively.

```
> y.fam <- gaussian(link=identity)
> eta1.glm <- glm(y ~ z, subset=tr==1,
+               family=y.fam, control=glm.control(maxit=1000))
> eta1.hat <- predict.glm(eta1.glm,
+                       newdata=data.frame(z=z), type="response")
> eta0.glm <- glm(y ~ z, subset=tr==0,
+               family=y.fam, control=glm.control(maxit=1000))
> eta0.hat <- predict.glm(eta0.glm,
+                       newdata=data.frame(z=z), type="response")
```

## 2.2 Checking balance

If the propensity score model is correctly specified, then the treated group is expected to be similar to (or matched with) the untreated group in terms of the covariates, after inverse probability weighting. In fact, the weighted distributions of the covariates within each treatment group are expected to be similar to the unweighted distributions in the overall sample. This balancing property provides an informative way to check propensity score models as suggested in Tan (2006).

The function `ate.HT` can be used to compute the differences between the weighted treatment-specific means and the unweighted overall means of the covariates, i.e.,

$$\frac{1}{n} \sum_{i=1}^n \frac{R_i Z_i}{\hat{\pi}_i} - \frac{1}{n} \sum_{i=1}^n Z_i,$$
$$\frac{1}{n} \sum_{i=1}^n \frac{(1 - R_i) Z_i}{1 - \hat{\pi}_i} - \frac{1}{n} \sum_{i=1}^n Z_i,$$

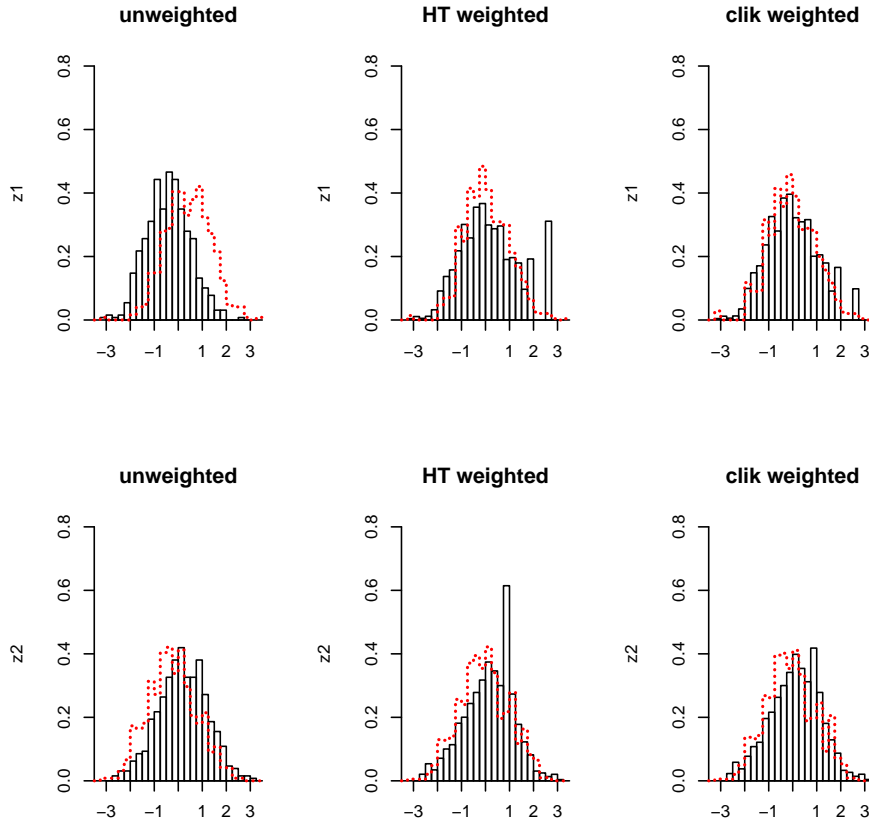
where  $R_i$  are treatment indicators,  $\hat{\pi}_i$  the fitted propensity scores, and  $Z_i$  the vector of covariates for which balance is to be checked. Typically, the covariates in  $Z_i$  are those entered in the outcome regression models.

```
> out.HT <- ate.HT(z, tr, ppi.hat, X, bal=TRUE)
> out.HT$mu
              mu1          mu0
z1  0.19538500  0.06452627
z2  0.20168063 -0.01549701
z3 -0.20481183  0.13499424
z4  0.02092013  0.05784173
```

The statistical significance of the differences can be assessed by the  $z$ -statistics as follows. There are a relatively large number of  $z$ -values exceeding 2 in absolute value, which (correctly) indicates that the propensity score model used is misspecified.

```
> out.HT$mu/ sqrt(out.HT$v)
```

Figure 1: Unweighted and weighted histograms (black: treated group, red: untreated group), with a misspecified propensity score model and a correct outcome regression model.



	mu1	mu0
z1	1.0634099	2.8036871
z2	2.8913106	-0.9982941
z3	-2.6421169	3.8847479
z4	0.7401349	2.7844917

For graphical comparisons, the function `histw` can be used to generate weighted histograms (and also unweighted histograms) of the covariates for the two treatment groups, as illustrated in Figure 1. To save space, the following code corresponds only

to the first two plots in the first row of Figure 1. There are appreciable differences in the HT-weighted histograms between the two treatment groups (see the right tail of  $z_1$  and the values of  $z_2$  near 1), although these differences are not as substantial as those in the unweighted histograms.

```
> gp1 <- tr==1
> gp0 <- tr==0
> par(mfrow=c(2,3))
> look <- z1
> histw(look[gp1], rep(1,sum(gp1)), xaxis=seq(-3.5,3.5,.25),
+       xmin=-3.5, xmax=3.5, ymax=.8)
> histw(look[gp0], rep(1,sum(gp0)), xaxis=seq(-3.5,3.5,.25),
+       xmin=-3.5, xmax=3.5, ymax=.8, bar=0, add=TRUE, col="red")
> title(main="unweighted", ylab="z1")
> histw(look[gp1], 1/ppi.hat[gp1], xaxis=seq(-3.5,3.5,.25),
+       xmin=-3.5, xmax=3.5, ymax=.8)
> histw(look[gp0], 1/(1-ppi.hat[gp0]), xaxis=seq(-3.5,3.5,.25),
+       xmin=-3.5, xmax=3.5, ymax=.8, bar=0, add=TRUE, col="red")
> title(main="HT weighted", ylab="z1")
```

### 2.3 Applying the calibrated likelihood method

If balance checking suggests possible model misspecification, then it is usually advisable to revise the propensity score model by, for example, introducing additional terms. This process of model building and checking needs to be carefully done in real data analysis. Nevertheless, we proceed to apply the calibrated likelihood estimator based on the given propensity score model and outcome regression model.

```
> out.clik <- ate.clik(y, tr, ppi.hat,
+                    g0=cbind(1,eta0.hat),g1=cbind(1,eta1.hat))
```

The weighted histograms based on `out.clik` are also shown in Figure 1. To save space, the following code corresponds to the third plot in the first row of Figure 1. Remarkably, the differences between the weighted treatment-specific histograms based on `out.clik` appear to be considerably reduced, compared with the HT-weighted histograms. Therefore, the calibrated likelihood method can be seen to (partially) correct for misspecification of the propensity score model. This effect can persist even if misspecified outcome regression models are used (see Section 2.4).

```
> look <- z1
> histw(look[gp1], 1/out.clik$w[gp1,1], xaxis=seq(-3.5,3.5,.25),
+       xmin=-3.5, xmax=3.5, ymax=.8)
> histw(look[gp0], 1/out.clik$w[gp0,2], xaxis=seq(-3.5,3.5,.25),
+       xmin=-3.5, xmax=3.5, ymax=.8, bar=0, add=TRUE, col="red")
> title(main="clik weighted", ylab="z1")
```

The following code gives the estimated treatment means and their standard errors. The true values of the two treatment means are both 210.

```
> out.clik$mu

  treat 1  treat 0
210.1775 210.2444

> sqrt(out.clik$v)

  treat 1  treat 0
1.122953 1.121321
```

The next code gives the estimated average treatment effect and its standard error. The true value of the average treatment effect is 0.

```
> out.clik$diff
```

```
[1] -0.0669473
```

```
> sqrt(out.clik$v.diff)
```

```
[1] 0.0760788
```

The vectors of calibration variables, `g0` and `g1`, can be specified in other ways as shown in the following example. Double robustness is achieved provided that the fitted outcome regression function for treatment 0 and 1 is contained in the linear span generated by the columns of `g0` and `g1` respectively.

```
> out.clik2 <- ate.clik(y, tr, ppi.hat,  
+                       g0=cbind(1,z),g1=cbind(1,z))
```

By the calibrated likelihood method, the weighted treatment-specific means of each calibration variable are equal to the overall unweighted mean.

```
> apply(z, 2, mean)
```

```
      z1      z2      z3      z4  
-0.015829573 -0.024786443  0.068142732  0.002485961
```

```
> apply(z[gp1,]/out.clik2$w[gp1,1], 2, sum)/n
```

```
      z1      z2      z3      z4  
-0.015829567 -0.024786450  0.068142733  0.002485964
```

```
> apply(z[gp0,]/out.clik2$w[gp0,2], 2, sum)/n
```

```
      z1      z2      z3      z4  
-0.015829725 -0.024786396  0.068142489  0.002485977
```



## 2.4 Additional illustration

Now suppose that a misspecified outcome regression model is fitted.

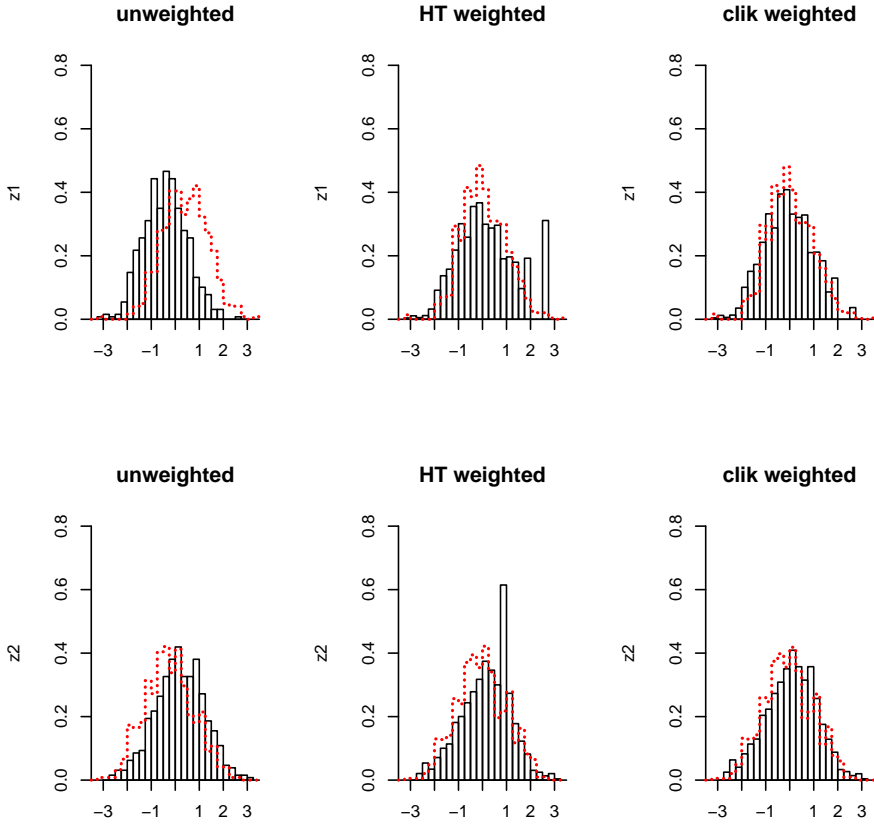
```
> eta1.glm <- glm(y ~ x, subset=tr==1,
+               family=y.fam, control=glm.control(maxit=1000))
> eta1.hat <- predict.glm(eta1.glm,
+                       newdata=data.frame(x=x), type="response")
> eta0.glm <- glm(y ~ x, subset=tr==0,
+               family=y.fam, control=glm.control(maxit=1000))
> eta0.hat <- predict.glm(eta0.glm,
+                       newdata=data.frame(x=x), type="response")
```

The calibrated likelihood method is applied next.

```
> out.clik <- ate.clik(y, tr, ppi.hat,
+                   g0=cbind(1,eta0.hat),g1=cbind(1,eta1.hat))
```

Figure 2 shows the weighted treatment-specific histograms of  $(z_1, z_2)$  based on the new results, `out.clik`, as the third plot in each row. The first two plots included in each row are the same as in Figure 1. There appears to be much smaller differences in the weighted treatment-specific histograms based on the calibrated likelihood method than those in the HT-weighted histograms, even though both the propensity score model and the outcome regression model are misspecified.

Figure 2: Unweighted and weighted histograms (black: treated group, red: untreated group), with a misspecified propensity score model and a misspecified outcome regression model.



## REFERENCES

- Kang, J.D.Y. and Schafer, J.L. (2007) “Demystifying double robustness: A comparison of alternative strategies for estimating a population mean from incomplete data,” *Statistical Science*, 22, 523-539.
- Tan, Z. (2006) “A distributional approach for causal inference using propensity scores,” *Journal of the American Statistical Association*, 101, 1619–1637.
- Tan, Z. (2010) “Bounded, efficient, and doubly robust estimation with inverse weighting,” *Biometrika*, 97, 661–682.