Decomposition methods in the social sciences

Bamberg Graduate School of Social Sciences, June 7–8, 2018

Ben Jann

University of Bern, Institut of Sociology

Reweighting and RIF regression

Beyond the mean

- The discussed Oaxaca-Blinder procedures and their extensions to non-linear models focus on the decomposition of differences in the expected value (mean) of an outcome variable.
- In many cases, however, on is interested in other distributional statistics, say the Gini coefficient or a the D9/D1 quantile ratio, or even in whole distributions (density curves, Lorenz curves).
- The basic setup is the same; an estimate of $F_{Y^g|G\neq g}$ is needed to be able to compute a decomposition such as

$$\begin{aligned}\Delta^\nu &= \nu\big(F_{Y|G=0}\big) - \nu\big(F_{Y|G=1}\big)\\ &= \big\{\nu\big(F_{Y|G=0}\big) - \nu\big(F_{Y^0|G=1}\big)\big\} + \big\{\nu\big(F_{Y^0|G=1}\big) - \nu\big(F_{Y|G=1}\big)\big\}\\ &= \Delta^\nu_X + \Delta^\nu_S\end{aligned}$$

where

$$F_{Y^g|G\neq g}(y) = \int F_{Y|X,G=g}(y|x) f_{X|G\neq g}(x)\, dx$$

Beyond the mean

- Several approaches have been proposed in the literature:
  - Estimating $F_{Y^g|G\neq g}$ by reweighting (DiNardo et al. 1996).
  - Imputing values for Y^g in group $G \neq g$
    - based on regression residuals (Juhn et al. 1993)
    - based on quantile regression (Machado and Mata 2005, Melly 2005, 2006)
  - Estimating $F_{Y^g|G\neq g}$ by distribution regression (Chernozhukov et al. 2013)
  - Estimating $\nu(F_{Y^g|G\neq g})$ via recentered influence function regression (Firpo et al. 2007, 2009)
- Today, we will only look at reweighting and RIF regression.

Contents

Basic procedure

- DiNardo, Fortin, and Lemieux (DFL) (1996) proposed a simple reweighting procedure to obtain an estimate of $F_{Y^g|G\neq g}$ or any functional $\nu()$ of $F_{Y^g|G\neq g}$.
- Let $F_{Y|X^g}$ stand for $F_{Y|X,G=g}$ and F_{X^g} for $F_{X|G=1}$. Multiplying

$$F_{Y^0|G=1}(y) = \int F_{Y|X^0}(y|x)\, dF_{X^1}(x)$$

by $dF_{X^0}/\, dF_{X^0}$ leads to

$$\begin{aligned} F_{Y^0|G=1}(y) &= \int F_{Y|X^0}(y|x) \frac{dF_{X^1}(x)}{dF_{X^0}(x)}\, dF_{X^0}(x) \\ &= \int F_{Y|X^0}(y|x)\Psi(x)\, dF_{X^0}(x) \end{aligned}$$

where

$$\Psi(x) = \frac{dF_{X^1}(x)}{dF_{X^0}(x)} = \frac{\Pr(x|G=1)}{\Pr(x|G=0)}$$

Basic procedure

- Based on Bayes' rule $\Pr(A|B) = \Pr(B|A)\Pr(A)/\Pr(B)$ we can rewrite $\Pr(X|G=g)$ as

$$\Pr(X|G=g) = \frac{\Pr(G=g|X)\Pr(X)}{\Pr(G=g)}$$

such that

$$\Psi(X) = \frac{\Pr(X|G=1)}{\Pr(X|G=0)} = \frac{\Pr(G=1|X)\Pr(x)/\Pr(G=1)}{\Pr(G=0|X)\Pr(X)/\Pr(G=0)}$$
$$= \frac{\Pr(G=1|X)/\Pr(G=1)}{\Pr(G=0|X)/\Pr(G=0)}$$

- $\Psi(X)$ is easy to estimate.
- An estimate for $\Pr(G=1) = 1 - \Pr(G=0)$ is simply the proportion of group 1 in the sample.
- $\Pr(G=1|X) = 1 - \Pr(G=0|X)$, the "propensity score", can be estimated by regressing G on X using logit or similar.

Basic procedure

- As soon as we have $\widehat{\Psi}(X)$, the counterfactual distribution $F_{Y^0|G=1}$, or any functional of the distribution, can be estimated from the $G = 0$ sample by weighting the observations by $\widehat{\Psi}(X)$.
- In this way we can easily get
  - a counterfactual kernel density estimate
  - an estimate of the counterfactual mean
  - an estimate of the counterfactual variance
  - estimates of counterfactual quantiles
  - an estimate of the counterfactual D9/D1 ratio
  - an estimate of the counterfactual Gini
  - ...
- A commands called `dfl` exists for Stata, but is limited to comparing kernel density estimates.
- In practice, therefore, one has to compute $\widehat{\Psi}(X)$ and the resulting decomposition manually (which fairly easy to do).

How to estimate the weights

- As said, the propensity score $\Pr(G = 1|X)$ can be estimated by regressing G on X using logit or probit or similar.
- The model specification should be fairly flexible to capture possible non-linearities and interaction effects. If data permits, you can also try nonparametric estimators such as `npregress` (official Stata) or `krls` (Hainmueller and Hazlett 2014).
- Furthermore, note that $\frac{\Pr(G=0)}{\Pr(G=1)}$ in $\Psi(X)$ does not depend on X. It is the same for all observations and can be omitted from the weights.
- This also clarifies that weighting by $\Psi(X)$ is equivalent to inverse probability weighting (IPW) known in the causal inference literature.
- That is, you can also obtain the weights by other causal inference procedures such as matching (e.g. `kmatch`, Jann 2017) or entropy balancing (`ebalance`, Hainmueller 2012).
  - Also see Ñopo (2008) for a decomposition procedure based on matching.

Limitations

- If the sample is small, flexible estimation of the propensity score will not be possible and the performance of the reweighting procedure may be poor.
- A related problem is that in small samples common support problems are likely (observations for which the estimated propensity score is close to zero or one); this can make the estimates unreliably (large variance in the weights).
- The effect of the weights is that they balance X between the groups, i.e. the distribution of X in one group is adjusted to the distribution of X in the other group. If the groups are very different with respect to X, this is hard to achieve. One consequence is again that the weights will have a large variance (making estimates imprecise). Furthermore, the desired balancing of X may be very poor in such cases.
- It is thus always a good idea to check the balancing, like you would do in a matching analysis.

Example analysis

```
. use gsoep29, clear
(BCPGEN: Nov 12, 2013 17:15:52-251 DBV29)
. // selection
. generate age = 2012 - bcgeburt
. keep if inrange(age, 25, 55)
(10,780 observations deleted)
. // compute gross wages and ln(wage)
. generate wage = labgro12 / (bctatzeit * 4.3) if labgro12>0 & bctatzeit>0
(1,936 missing values generated)
. generate lnwage = ln(wage)
(1,936 missing values generated)
. // X variables
. generate schooling = bcbilzeit if bcbilzeit>0
(318 missing values generated)
. generate ft_experience = expft12 if expft12>=0
(15 missing values generated)
. generate ft_experience2 = expft12^2 if expft12>=0
(15 missing values generated)
. generate public = oeffd12==1 if oeffd12>0
(2,274 missing values generated)
. // summarize
. summarize wage lnwage schooling ft_experience ft_experience2 public
```

Variable	Obs	Mean	Std. Dev.	Min	Max
wage	8,090	16.26903	15.21083	.3624283	914.7287
lnwage	8,090	2.615219	.5944705	-1.014929	6.818627
schooling	9,708	12.76118	2.73677	7	18
ft_experie~e	10,011	13.41052	10.03473	0	39
ft_experie~2	10,011	280.5277	324.8873	0	1521
public	7,752	.2582559	.4377037	0	1

```
. marktouse touse lnwage schooling ft_experience ft_experience2 public
(7388 observations marked)
```

Example analysis

```
. sum lnwage if public==0 & touse, detail
                              lnwage
-------------------------------------------------------------
      Percentiles      Smallest
 1%     .8439701      -1.014929
 5%     1.621675      -.4470141
10%     1.881213      -.3600028       Obs               5,476
25%     2.230264      -.2954642       Sum of Wgt.       5,476
50%     2.594971                      Mean             2.5881
                        Largest       Std. Dev.      .6079005
75%     2.964234       5.225996
90%     3.304779        5.42561       Variance        .369543
95%      3.50323       5.660272       Skewness      -.2808259
99%     4.038553       6.818627       Kurtosis       5.460504
. local prAVG  = r(mean)
. local prD9D1 = r(p90)-r(p10)
. local prD9D5 = r(p90)-r(p50)
. local prD5D1 = r(p50)-r(p10)
. local prVar  = r(Var)
. display exp(`prAVG')
13.304465
. display exp(`prD9D1')
4.1518992
. display exp(`prD9D5')
2.033601
. display exp(`prD5D1')
2.0416489
. display `prVar'
.36954301
```

Example analysis

```
. sum lnwage if public==1 & touse, detail
                              lnwage
-------------------------------------------------------------
      Percentiles      Smallest
 1%     1.472579      -.6376345
 5%     1.989102      -.3600028
10%     2.219793      -.1856493       Obs               1,912
25%     2.548718       .1803817       Sum of Wgt.       1,912
50%      2.78988                      Mean           2.749522
                        Largest       Std. Dev.      .4519296
75%     3.018445       4.062846
90%      3.23959       4.067734       Variance       .2042404
95%     3.394793       4.370919       Skewness        -1.1418
99%     3.685552       5.287405       Kurtosis       9.000465
. local puAVG  = r(mean)
. local puD9D1 = r(p90)-r(p10)
. local puD9D5 = r(p90)-r(p50)
. local puD5D1 = r(p50)-r(p10)
. local puVar  = r(Var)
. display exp(`puAVG')
15.635162
. display exp(`puD9D1')
2.7726314
. display exp(`puD9D5')
1.5678569
. display exp(`puD5D1')
1.7684212
. display `puVar'
.20424037
```

Example analysis

```
. display exp(`prAVG'  - `puAVG' )
.85093231
. display exp(`prD9D1' - `puD9D1')
1.497458
. display exp(`prD9D5' - `puD9D5')
1.2970578
. display exp(`prD5D1' - `puD5D1')
1.1545037
. display `prVar' - `puVar'
.16530265
```

Example analysis

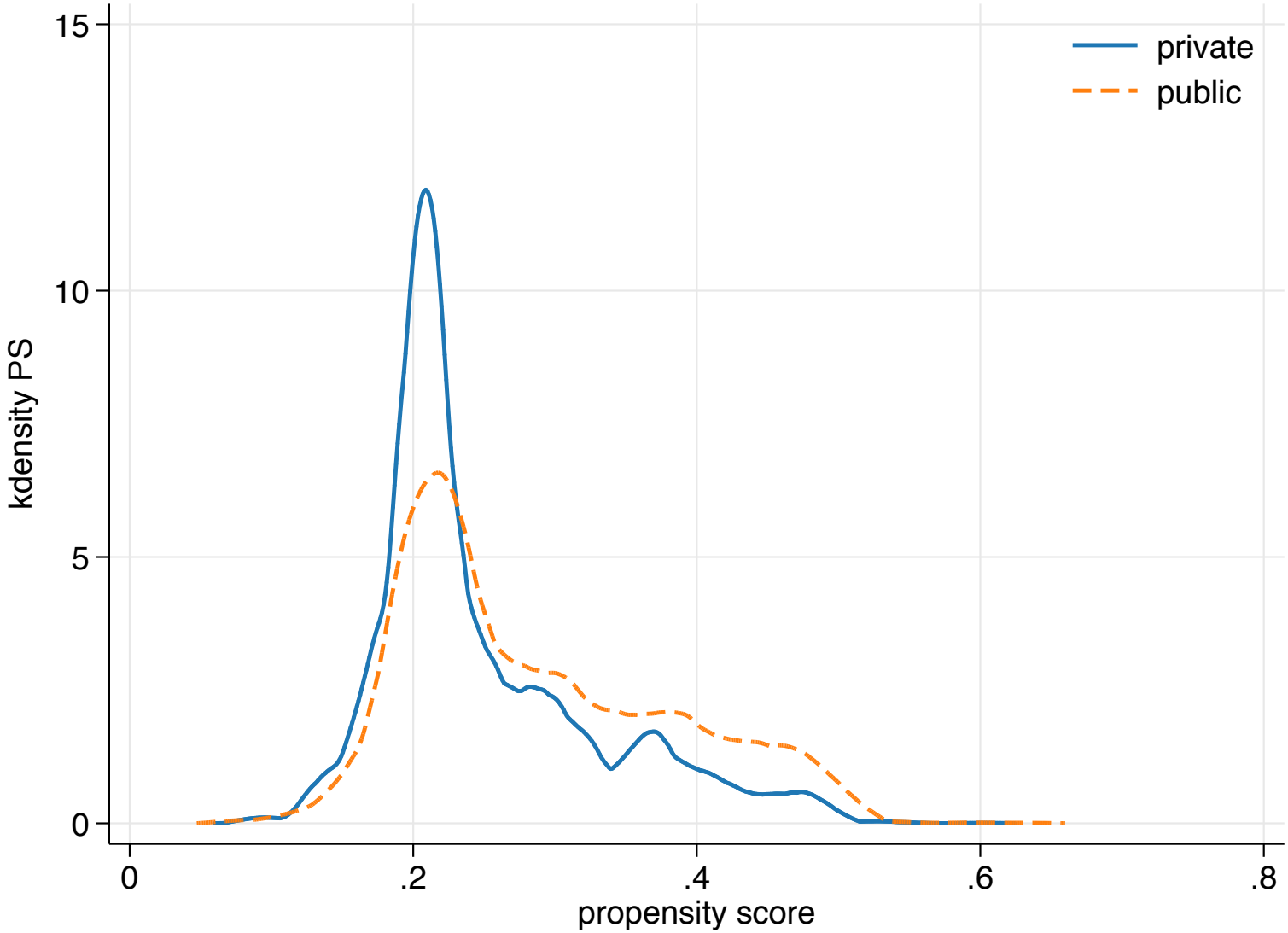
```
. logit public c.schooling##c.ft_experience##c.ft_experience2 if touse, vsquish
Iteration 0:   log likelihood = -4224.4285
Iteration 1:   log likelihood = -4096.5329
Iteration 2:   log likelihood = -4094.8721
Iteration 3:   log likelihood = -4094.8719
Logistic regression                               Number of obs     =      7,388
                                                  LR chi2(7)        =     259.11
                                                  Prob > chi2       =     0.0000
Log likelihood = -4094.8719                       Pseudo R2         =     0.0307
```

public	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
schooling	.2096272	.0276666	7.58	0.000	.1554017	.2638528
ft_experience	.1109149	.1066695	1.04	0.298	-.0981534	.3199833
c.schooling# c.ft_experience	-.009321	.0077856	-1.20	0.231	-.0245806	.0059385
ft_experience2	-.0016162	.0075363	-0.21	0.830	-.016387	.0131547
c.schooling# c.ft_experience2	.0001383	.0005697	0.24	0.808	-.0009782	.0012548
c.ft_experience# c.ft_experience2	-.0000484	.0001506	-0.32	0.748	-.0003436	.0002468
c.schooling# c.ft_experience# c.ft_experience2	4.62e-06	.0000117	0.39	0.694	-.0000184	.0000276
_cons	-3.801943	.3974551	-9.57	0.000	-4.58094	-3.022945

```
. predict PS if e(sample), pr
(2,638 missing values generated)
```

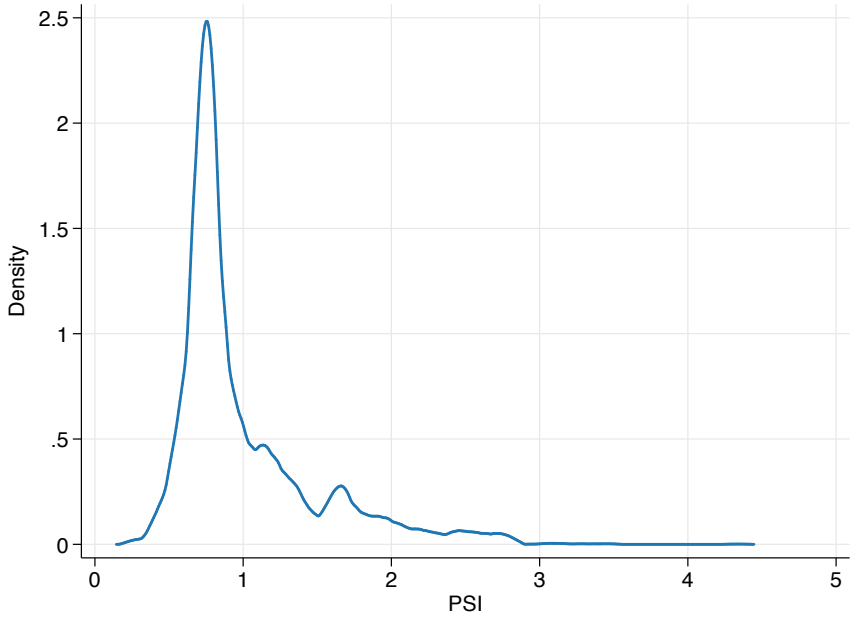
Example analysis

```
. quietly two (kdens PS if public==0) (kdens PS if public==1), ///
>     xti("propensity score") legend(order(1 "private" 2 "public"))
```



Example analysis

```
. summarize public if touse
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
      public |      7,388    .2587981    .4380041          0          1
. local P_public = r(mean)
. generate PSI = (PS / `P_public') / ((1-PS) / (1 - `P_public')) if public==0 & touse
(4,550 missing values generated)
. replace PSI = 1 if public==1 & touse
(1,912 real changes made)
. summarize PSI if public==0 & touse
    Variable |        Obs        Mean    Std. Dev.       Min        Max
-------------+---------------------------------------------------------
         PSI |      5,476    .9999512    .4742042   .2555806   4.334548
. kdens PSI if public==0 & touse
(bandwidth = .11136615)
```



Example analysis

```
. // Raw mean differences in covariates
. tabstat PS schooling ft_experience ft_experience2 if touse, by(public) nototal ///
>     stat(mean var p10 p50 p90) columns(statistics)
Summary for variables: PS schooling ft_experience ft_experience2
     by categories of: public
  public |      mean  variance       p10       p50       p90
---------+----------------------------------------------------
       0 |    .249332  .0060701  .1765754  .2183861  .3716415
         |   12.63705  6.828291      10.5      11.5        18
         |   14.94805  98.56863         2        14        29
         |   321.9947    113321         4       196       841
---------+----------------------------------------------------
       1 |    .285909  .0086856  .1928914  .2541142  .4321675
         |   13.77641  8.355398      10.5        13        18
         |   14.52866  101.5415         2      13.3        29
         |   312.5704  112618.3         4    176.89       841
----------------------------------------------------------------

. // Mean differens in weighted sample
. tabstat PS schooling ft_experience ft_experience2 [aw=PSI] if touse, by(public) nototal
> ///
>     stat(mean var p10 p50 p90) columns(statistics)
Summary for variables: PS schooling ft_experience ft_experience2
     by categories of: public
  public |      mean  variance       p10       p50       p90
---------+----------------------------------------------------
       0 |   .2858741  .0087147  .1937293  .2543135  .4287887
         |    13.7805  8.523594      10.5        13        18
         |   14.55628  101.4321       1.9      13.2        29
         |   313.2988  112554.4      3.61    174.24       841
---------+----------------------------------------------------
       1 |    .285909  .0086856  .1928914  .2541142  .4321675
         |   13.77641  8.355398      10.5        13        18
         |   14.52866  101.5415         2      13.3        29
         |   312.5704  112618.3         4    176.89       841
----------------------------------------------------------------
```

Example analysis

```
. sum lnwage [aw=PSI] if public==0 & touse, detail
                              lnwage
-------------------------------------------------------------
      Percentiles      Smallest
 1%     .9273517      -1.014929
 5%     1.704171      -.4470141
10%     1.942582      -.3600028       Obs               5,476
25%     2.317276      -.2954642       Sum of Wgt.  5,475.7329
50%      2.69457                      Mean           2.684965
                        Largest       Std. Dev.      .6211304
75%     3.071832       5.225996
90%     3.409304        5.42561       Variance        .385803
95%     3.616559       5.660272       Skewness      -.2256962
99%     4.245167       6.818627       Kurtosis       5.072671
. local cAVG  = r(mean)
. local cD9D1 = r(p90)-r(p10)
. local cD9D5 = r(p90)-r(p50)
. local cD5D1 = r(p50)-r(p10)
. local cVar  = r(Var)
. display exp(`cAVG')
14.657694
. display exp(`cD9D1')
4.3349997
. display exp(`cD9D5')
2.0436427
. display exp(`cD5D1')
2.1212122
. display `c_Var'
```

Example analysis

```
. foreach s in AVG D9D1 D9D5 D5D1 Var {
  2.     display %6s "`s': " "total difference = " %9.0g `pr`s'' - `pu`s'' ///
>         "    explained = " %9.0g `pr`s'' - `c`s''
  3. }
 AVG: total difference = -.1614227    explained = -.0968657
D9D1: total difference =   .403769    explained = -.0431557
D9D5: total difference =  .2600985    explained = -.0049257
D5D1: total difference =  .1436706    explained = -.0382299
 Var: total difference =  .1653026    explained =   -.01626

.
. oaxaca lnwage schooling ft_experience ft_experience2, by(public) weight(1) nodetail
Blinder-Oaxaca decomposition                    Number of obs     =      7,388
                                                  Model           =     linear
Group 1: public = 0                               N of obs 1      =       5476
Group 2: public = 1                               N of obs 2      =       1912
```

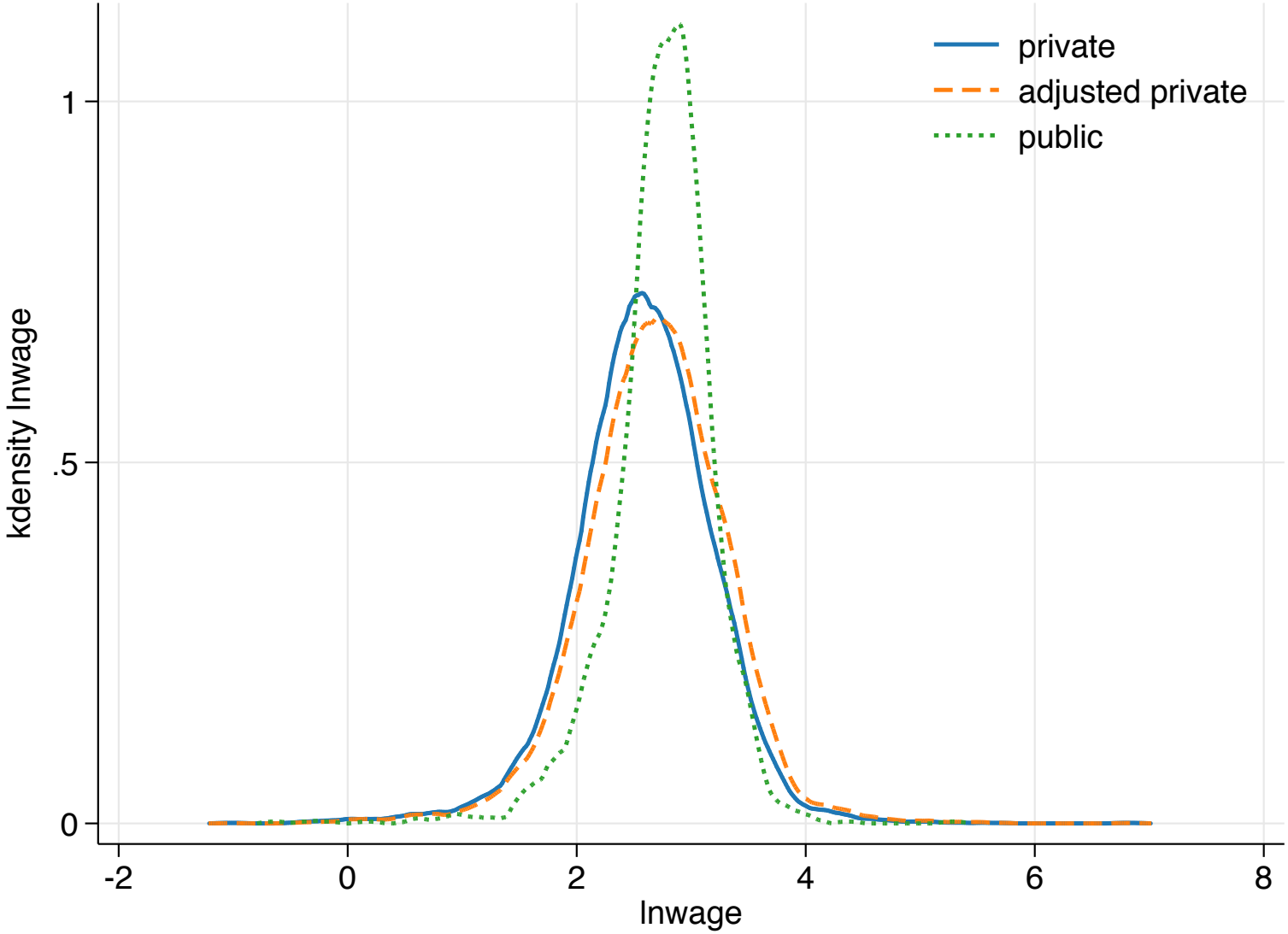
lnwage	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
overall						
group_1	2.5881	.0082165	314.99	0.000	2.571996	2.604204
group_2	2.749522	.0103413	265.88	0.000	2.729254	2.769791
difference	-.1614227	.0132081	-12.22	0.000	-.18731	-.1355354
explained	-.0982116	.0089312	-11.00	0.000	-.1157164	-.0807068
unexplained	-.0632111	.0118564	-5.33	0.000	-.0864493	-.0399729

Example analysis

```
. gen s = schooling
(318 missing values generated)

. gen e = ft_experience
(15 missing values generated)

. gen se = s*e
(322 missing values generated)

. gen ee = e*e
(15 missing values generated)

. gen see = s*e*e
(322 missing values generated)

. oaxaca lnwage s e se ee see, by(public) weight(1) nodetail

Blinder-Oaxaca decomposition                    Number of obs     =      7,388
                                                  Model           =     linear
Group 1: public = 0                               N of obs 1      =       5476
Group 2: public = 1                               N of obs 2      =       1912
```

lnwage	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
overall						
group_1	2.5881	.0082176	314.94	0.000	2.571993	2.604206
group_2	2.749522	.0103451	265.78	0.000	2.729246	2.769798
difference	-.1614227	.0132118	-12.22	0.000	-.1873173	-.1355281
explained	-.0946359	.0090254	-10.49	0.000	-.1123254	-.0769464
unexplained	-.0667868	.0118623	-5.63	0.000	-.0900365	-.0435372

```
. quietly two (kdens lnwage if public==0 & touse) ///
>             (kdens lnwage [aw=PSI] if public==0 & touse) ///
>             (kdens lnwage if public==1 & touse) ///
>     , legend(order(1 "private" 2 "adjusted private" 3 "public")) xti(lnwage)
```



Detailed decomposition

- For binary covariates, a detailed decomposition of the contribution to the quantity effect can be obtained as follows.
- Let X_1 be a binary and X_2 be the vector of all other covariates. A counterfactual distribution of Y in group 0, where the conditional distribution of X_1 given the other covariate is changed to the conditional distribution of X_1 in group 1, can be written as

$$\begin{aligned} F_{Y^0|X_1^1}(y) &= \int\int F_{Y|X^0}(y|X_1,X_2)\, dF_{X^1}(X_1|X_2)\, dF_{X^0}(X_2) \\ &= \int\int F_{Y|X^0}(y|X_1,X_2)\Psi_1(X_1,X_2)\, dF_{X^0}(X_1|X_2)\, dF_{X^0}(X_2) \\ &= \int\int F_{Y|X^0}(y|X_1,X_2)\Psi_1(X_1,X_2)\, dF_{X^0}(X_1,X_2) \end{aligned}$$

where

$$\Psi_1(X_1,X_2) = \frac{dF_{X^1}(X_1|X_2)}{dF_{X^0}(X_1|X_2)} = X_1\frac{\Pr^1(X_1=1|X_2)}{\Pr^0(X_1=1|X_2)} + (1-X_1)\frac{\Pr^1(X_1=0|X_2)}{\Pr^0(X_1=0|X_2)}$$

Detailed decomposition

- To compute $\Psi_1(X_1, X_2)$, regress X_1 on X_2 in separately group 0 and in group 1 using logistic regression or similar. Then replace $\Pr^0(X_1 = 1|X_2)$, $\Pr^0(X_1 = 0|X_2)$, $\Pr^1(X_1 = 1|X_2)$ and $\Pr^1(X_1 = 0|X_2)$ by predictions from these models.
- A similar approach can also be used to determine the contribution of a binary covariate to the structure component (see Fortin et al. 2011).
- For continuous covariates, things are less clear. One approach followed in the literature is to compute a series of reweighting decompositions where the covariates are introduced one after the other. The problem with this approach is that results will be path dependent.
- A better approach is, for each covariate, to compute the contribution of the covariate while controlling for all other covariates.

Detailed decomposition

- Let $X_{\bar{k}}$ be all covariates except X_k. Based on a similar derivation as above, Fortin et al. (2001) suggest using reweighting factor

$$\Psi_{X_k|X_{\bar{k}}}(X_{\bar{k}}) = \frac{\Psi(X)}{\Psi(X_{\bar{k}})}$$

  where $\Psi(X_{\bar{k}})$ is computed in the same way as the overall reweighting factor $\Psi(X)$, only that variable X_k is omitted from the logit model.
- Using this reweighting factor we can get the counterfactual distribution of Y in group 0, if the conditional distribution of X_k given the other covariates is changed to the conditional distribution of X_k in group 1.
- That procedure is as follows:
  1. Compute $\Psi(X)$ using all covariates.
  2. For each k, compute $\Psi(X_{\bar{k}})$.
  3. For each k, compute the counterfactual statistic using weights $\Psi(X)/\Psi(X_{\bar{k}})$ and compare the result to the unweighted statistic. The difference is the contribution of X_k to the composition effect.
- Note that the single contributions do not add up to the total

Example analysis

```
. // schooling
. drop PS
. quietly logit public c.ft_experience##c.ft_experience2 if touse, vsquish
. predict PS if e(sample), pr
(2,638 missing values generated)
. generate PSI_schooling = (PS / `P_public') / ((1-PS) / (1 - `P_public')) if public==0 &
> touse
(4,550 missing values generated)
. quietly sum lnwage [aw=PSI/PSI_schooling] if public==0 & touse, detail
. local  cAVGx = r(mean)
. local cD9D1x = r(p90)-r(p10)
. local cD9D5x = r(p90)-r(p50)
. local cD5D1x = r(p50)-r(p10)
. local  cVarx = r(Var)
. foreach s in AVG D9D1 D9D5 D5D1 Var {
  2.     display %6s "`s': " "explained by schooling = " %9.0g `pr`s'' - `c`s'x'
  3. }
 AVG: explained by schooling = -.1058771
D9D1: explained by schooling =  -.037792
D9D5: explained by schooling = -.0082781
D5D1: explained by schooling = -.0295138
 Var: explained by schooling = -.0123014
```

Example analysis

```
. // experience
. drop PS
. quietly logit public schooling if touse, vsquish
. predict PS if e(sample), pr
(2,638 missing values generated)
. generate PSI_experience = (PS / `P_public') / ((1-PS) / (1 - `P_public')) if public==0 &
>  touse
(4,550 missing values generated)
. quietly sum lnwage [aw=PSI/PSI_experience] if public==0 & touse, detail
. local  cAVGx = r(mean)
. local cD9D1x = r(p90)-r(p10)
. local cD9D5x = r(p90)-r(p50)
. local cD5D1x = r(p50)-r(p10)
. local  cVarx = r(Var)
. foreach s in AVG D9D1 D9D5 D5D1 Var {
  2.     display %6s "`s': " "explained by experience = " %9.0g `pr`s'' - `c`s'x'
  3. }
 AVG: explained by experience = -.0051177
D9D1: explained by experience =  .0166521
D9D5: explained by experience =    .01406
D5D1: explained by experience =  .0025921
 Var: explained by experience =  .0081723
```

Example analysis

```
. oaxaca lnwage schooling (experience: ft_experience ft_experience2), by(public) weight(1)
Blinder-Oaxaca decomposition                    Number of obs     =      7,388
                                                  Model           =     linear
Group 1: public = 0                               N of obs 1      =       5476
Group 2: public = 1                               N of obs 2      =       1912
```

lnwage	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
overall						
group_1	2.5881	.0082165	314.99	0.000	2.571996	2.604204
group_2	2.749522	.0103413	265.88	0.000	2.729254	2.769791
difference	-.1614227	.0132081	-12.22	0.000	-.18731	-.1355354
explained	-.0982116	.0089312	-11.00	0.000	-.1157164	-.0807068
unexplained	-.0632111	.0118564	-5.33	0.000	-.0864493	-.0399729
explained						
schooling	-.1100746	.0078852	-13.96	0.000	-.1255293	-.0946199
experience	.011863	.0059652	1.99	0.047	.0001714	.0235545
unexplained						
schooling	.3844041	.0573983	6.70	0.000	.2719055	.4969026
experience	.0522172	.0253538	2.06	0.039	.0025247	.1019097
_cons	-.4998323	.0645484	-7.74	0.000	-.6263449	-.3733197

```
experience: ft_experience ft_experience2
```

Exercise 7

- Rerun the above analysis (without the detailed decomposition to save time), but use a different counterfactual (distribution of public sector wages if X is adjusted to the private sector).
- Furthermore, wrap the analysis into a program and apply the bootstrap to compute standard errors and confidence intervals.

Influence functions

- A very nice approach to compute Oaxaca-Blinder type decompositions for almost any distributional statistic of interest is based on influence functions.
- An influence functions is a function that quantifies how a target statistic changes in response to small changes in the data. That is, for each value y, the influence function $\mathrm{IF}(y; \nu, F_Y)$ provides an approximation of how the functional $\nu(F_Y)$ changes if a small probability mass is added at point y.
- Influence functions are used in robust statistics to describe the robustness properties of various statistic (a robust statistic has a bounded influence function).
- There is also a close connection to the sampling variance of a statistic. The asymptotic sampling variance of a statistic is equal to the sampling variance of the mean of the influence function. Therefore, influence functions provide an easy way to estimate standard errors for many statistics (e.g. inequality measured).

RIF regression

- For example, the influence function of quantile Q_p is simply

$$\mathrm{IF}(y; Q_p, F_Y) = \frac{p - I(y \leq Q_p)}{f_Y(Q_p)}$$

- Influence functions are centered around zero (that is, have an expected value of zero). To center an influence function around the statistic of interest, we can simply add the statistic to the influence function. This is called a recentered influence function

$$\mathrm{RIF}(y; \nu, F_Y) = \nu(F_Y) + \mathrm{IF}(y; \nu, F_Y)$$

- The idea now is to model the conditional expectation of $\mathrm{RIF}(y; \nu, F_Y)$ using regression models, e.g. using a linear model

$$\mathrm{E}(\mathrm{RIF}(Y; \nu, F_Y)|X) = X\gamma$$

- Coefficient γ thus provides an approximation of how $\nu(F_Y)$ reacts to changes in X.

RIF regression decomposition

- In practice, taking the example of a quantile, we would first compute the sample quantile $\widehat{Q}_p$ and then use kernel density estimation to get $\widehat{f}(\widehat{Q}_p)$, the density of Y at point $\widehat{Q}_p$.
- $\text{RIF}(Y_i; Q_p, F_Y)$ is then computed for each observation by plugging these estimates in to the above formula.
- Finally, we regress $\text{RIF}(Y_i; Q_p, F_Y)$ on X to get an estimate of γ.
- Using the coefficients from RIF regression in two groups, we can perform an Oaxaca-Blinder type decomposition for Q_p. For example:

$$\hat{\Delta}^{Q_p} = \hat{\Delta}_X^{Q_p} + \hat{\Delta}_S^{Q_p} = (\bar{X}^0 - \bar{X}^1)\hat{\gamma}^0 + \bar{X}^1(\hat{\gamma}^0 - \hat{\gamma}^1)$$

- A similar procedure can be followed for any other statistic $\nu(F_Y)$. All you have to know is the influence function, which is usually easy to find in the statistical literature.

RIF regression decomposition

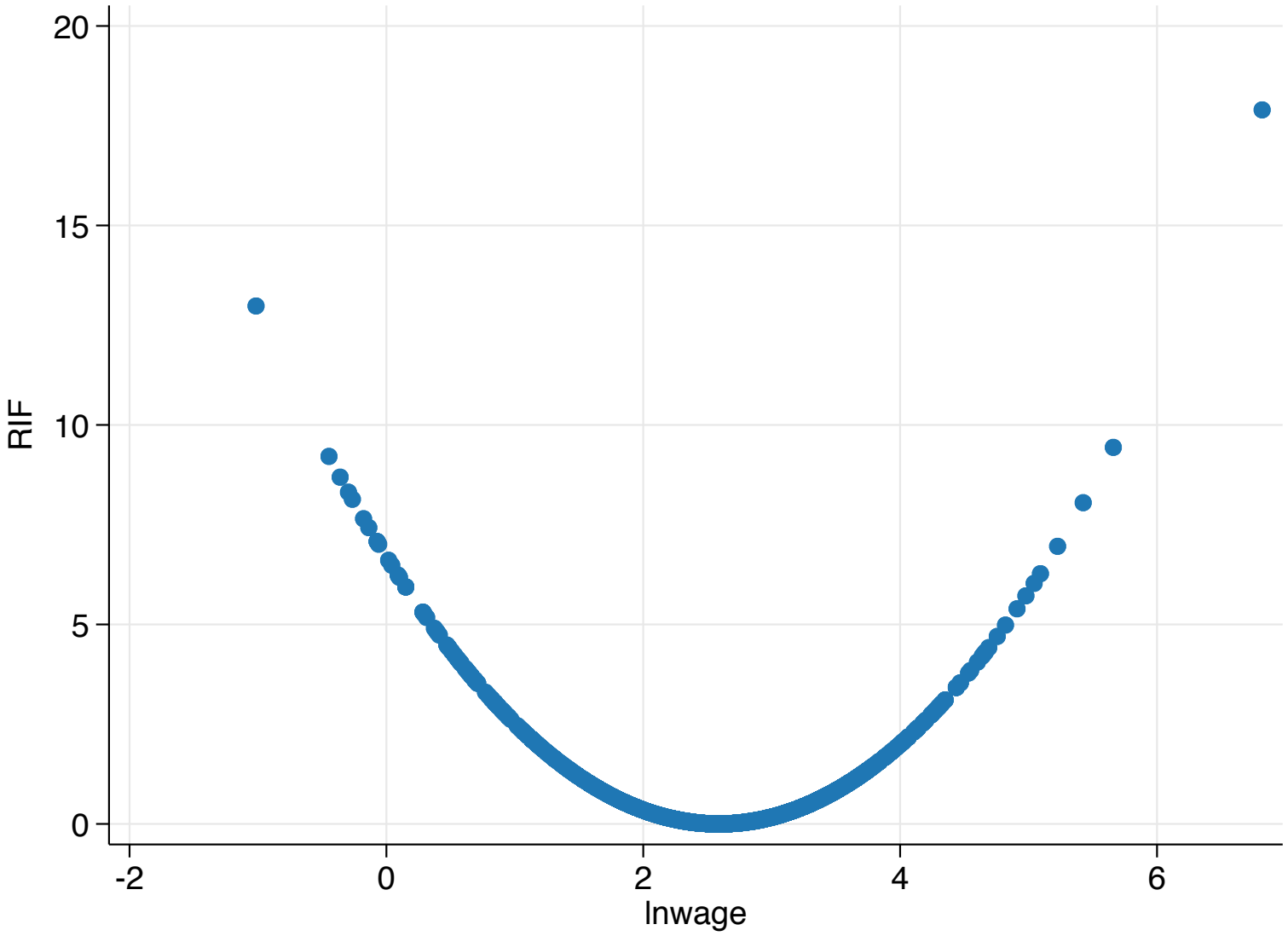
- Command `rifreg` provides RIF regression for quantiles, the Gini coefficient, and the variance. It can be obtained from https://economics.ubc.ca/faculty-and-staff/nicole-fortin/.
- The RIF variables stored by `rifreg` can then be used in `oaxaca`.

Example analysis

```
. rifreg lnwage schooling ft_experience ft_experience2 if public==0 & touse, variance reta
> in(RIF)
(4,550 missing values generated)
      Source |       SS       df       MS              Number of obs =    5476
-------------+------------------------------           F(  3,  5472) =   21.01
       Model |  37.9668705     3  12.6556235           Prob > F      =  0.0000
    Residual |  3296.44132  5472  .602419832           R-squared     =  0.0114
-------------+------------------------------           Adj R-squared =  0.0108
       Total |  3334.40819  5475  .609024327           Root MSE      =  .77616

------------------------------------------------------------------------------
           RIF |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     schooling |   .0226022   .0040773     5.54   0.000      .014609    .0305954
 ft_experience |   -.014324   .0038439    -3.73   0.000    -.0218596   -.0067885
ft_experience2 |   .0002986   .0001136     2.63   0.009     .0000758    .0005214
         _cons |    .201826   .0589913     3.42   0.001     .0861797    .3174723
------------------------------------------------------------------------------

. regress RIF schooling ft_experience ft_experience2, noheader
------------------------------------------------------------------------------
           RIF |      Coef.   Std. Err.      t    P>|t|     [95% Conf. Interval]
-------------+----------------------------------------------------------------
     schooling |   .0226022   .0040773     5.54   0.000      .014609    .0305954
 ft_experience |   -.014324   .0038439    -3.73   0.000    -.0218596   -.0067885
ft_experience2 |   .0002986   .0001136     2.63   0.009     .0000758    .0005214
         _cons |    .201826   .0589913     3.42   0.001     .0861797    .3174723
------------------------------------------------------------------------------
```

Example analysis

```
. scatter RIF lnwage
. drop RIF
```



Example analysis

```
. quietly rifreg lnwage if public==0 & touse, variance retain(RIFprivate)
. quietly rifreg lnwage if public==1 & touse, variance retain(RIFpublic)
. gen RIF = cond(public==1, RIFpublic, RIFprivate)
(2,638 missing values generated)
. oaxaca RIF schooling (experience: ft_experience ft_experience2), by(public) weight(1)
Blinder-Oaxaca decomposition                    Number of obs     =      7,388
                                                  Model           =     linear
Group 1: public = 0                               N of obs 1      =       5476
Group 2: public = 1                               N of obs 2      =       1912
```

RIF	Coef.	Std. Err.	z	P>|z|	[95% Conf.	Interval]
overall						
group_1	.3694755	.0105488	35.03	0.000	.3488003	.3901508
group_2	.2041335	.0132183	15.44	0.000	.1782262	.2300409
difference	.165342	.0169115	9.78	0.000	.132196	.198488
explained	-.0289454	.0052513	-5.51	0.000	-.0392378	-.0186531
unexplained	.1942874	.0174656	11.12	0.000	.1600555	.2285194
explained						
schooling	-.025752	.0049448	-5.21	0.000	-.0354436	-.0160605
experience	-.0031934	.0015953	-2.00	0.045	-.0063201	-.0000667
unexplained						
schooling	.34344	.0852943	4.03	0.000	.1762663	.5106137
experience	.0831629	.0376505	2.21	0.027	.0093693	.1569565
_cons	-.2323155	.0959429	-2.42	0.015	-.4203601	-.0442708

```
experience: ft_experience ft_experience2
```

Limitations

- Effects on the RIF of statistics such as inequality measures are likely to be highly nonlinear. Interaction effects are also likely.
- It is therefore important to use a flexible model specification.
- In case of interaction effects this again limits the usefulness of the procedure for obtaining detailed decompositions.

References

- Chernozhukov, Victor, Iván Fernández-Val, Blaise Melly (2013). Inference on Counterfactual Distributions. Econometrica 81(6):2205–2268.
- DiNardo, John E., Nicole Fortin, Thomas Lemieux (1996). Labour Market Institutions and the Distribution of Wages, 1973-1992: A Semiparametric Approach. Econometrica 64(5):1001–1046.
- Firpo, Sergio, Nicole Fortin, Thomas Lemieux (2007). Decomposing Wage Distributions using Recentered Influence Function Regressions. Working paper.
- Firpo, Sergio, Nicole M. Fortin, Thomas Lemieux (2009). Unconditional Quantile Regressions. Econometrica 77:953–973.
- Hainmueller, Jens (2012). Entropy Balancing: A Multivariate Reweighting Method to Produce Balanced Samples in Observational Studies. Political Analysis 20(1):25–46.
- Hainmueller, Jens, Chad Hazlett (2014). Kernel Regularized Least Squares: Reducing Misspecification Bias with a Flexible and Interpretable Machine Learning Approach. Political Analysis 22(2):143–168.

References

- Jann, Ben (2017). kmatch: Stata module for multivariate-distance and propensity-score matching. Available from http://ideas.repec.org/c/boc/bocode/s458346.html.
- Juhn, Chinhui, Kevin M. Murphy, Brooks Pierce (1993). Wage Inequality and the Rise in Returns to Skill. Journal of Political Economy 101(3):410–442.
- Machado, José A. F., José Mata (2005). Counterfactual decomposition of changes in wage distributions using quantile regression. Journal of Applied Econometrics 20(4):445–465.
- Melly, Blaise (2005). Decomposition of differences in distribution using quantile regression. Labour Economics 12(4):577–590.
- Melly, Blaise, 2006. Estimation of counterfactual distributions using quantile regression. University of St. Gallen, Discussion Paper.
- Ñopo, Hugo (2008). Matching as a Tool to Decompose Wage Gaps. The Review of Economics and Statistics 90:290–299.