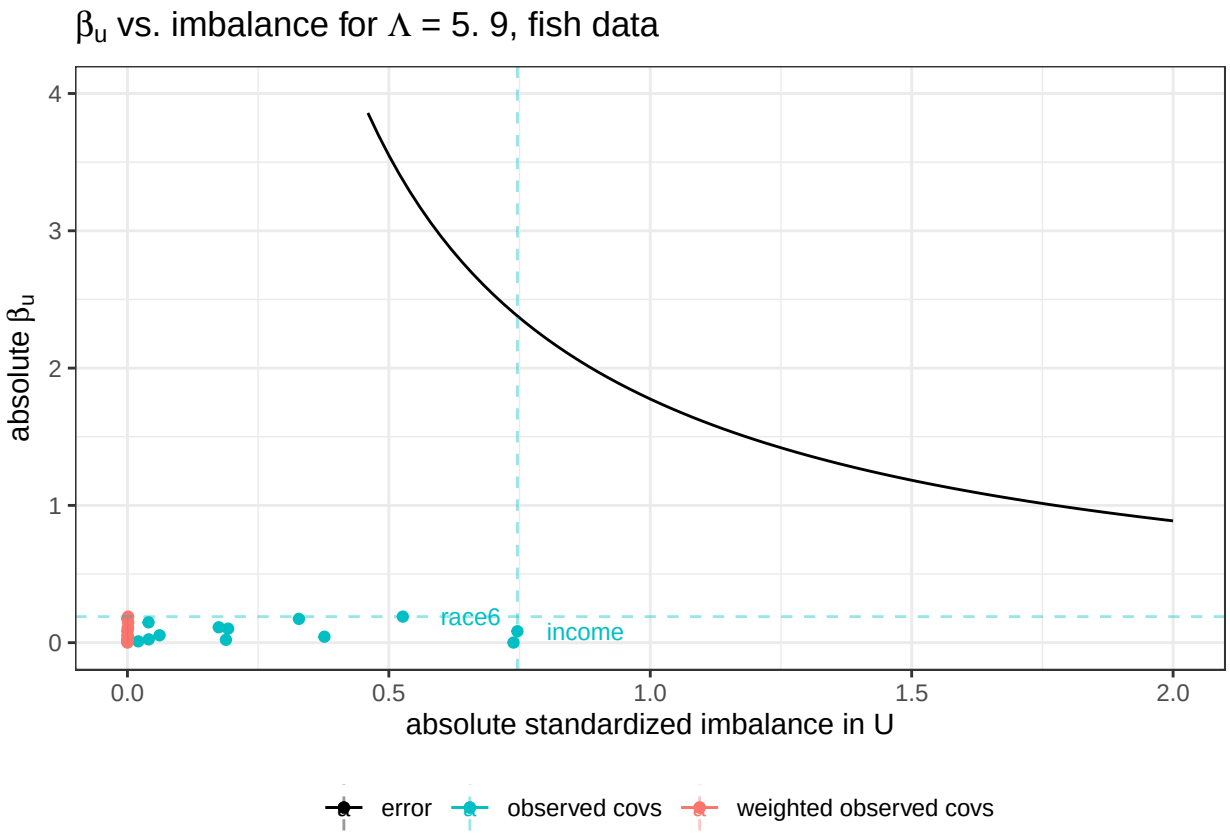
Sensitivity Analysis for Balancing Weights

dansoriano

2021-01-04

Amplification

Amplification of bias = imbalance in $U \times \beta_u$



- **bound:**
  - if estimated ATT is positive, bound $= \left(\sup_{h \in \mathcal{H}(\Lambda)} \hat{\mu}_0^{(h)}\right) - \hat{\mu}_0$
  - if estimated ATT is negative, bound $= \left(\inf_{h \in \mathcal{H}(\Lambda)} \hat{\mu}_0^{(h)}\right) - \hat{\mu}_0$
- We consider $U \in [0, 1]$, so we transform each observed covariate as follows:
  - Make min = 0: subtract min value of covariate

  - Make max = 1: divide by max of shifted covariate

- **max β obs:** max absolute value of coefficients of transformed covariates from OLS of Y on transformed covariates for control units.
- **max imbal obs:** max absolute value of difference in means of transformed covariates before weighting between treatment and control.
- **top β obs:** coefficient and imbalance for specified number of observed covariates sorted by descending coefficient value

covariate	coefficient	imbalance	post-weighting imbalance
race6	0.190	0.527	0.001
income	0.083	0.746	0.000
age	0.173	0.328	0.000
race3	0.102	0.193	0.001
race4	0.112	0.175	0.001
race1	0.043	0.377	0.001
race2	0.148	0.040	0.001
smoking.now	0.021	0.188	0.000
income.missing	0.054	0.062	0.000
smoking.everTRUE	0.024	0.041	0.000
education	0.001	0.738	0.000
gender	0.010	0.021	0.000

Confidence and point estimate intervals

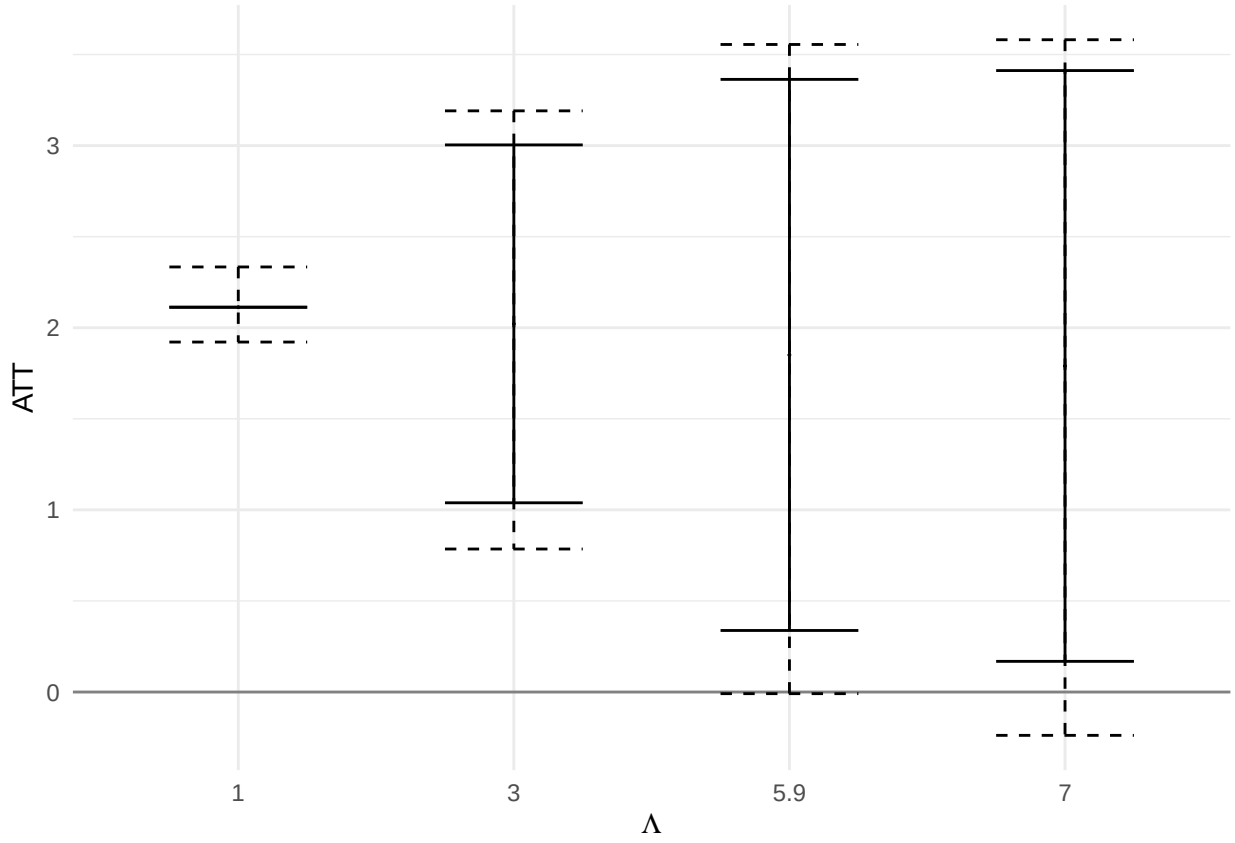
```
## Creating linear term vector...

## Creating quadratic term matrix...

## Creating constraint matrix...

##  x Sum to one constraint

##  x Upper and lower bounds

##  (SKIPPING) Enforce exact global balance

##  x Fit weights to data

##  x Constrain treated weights to be zero

##  x Combining constraints

## -----------------------------------------------------------------
##            OSQP v0.6.0  -  Operator Splitting QP Solver
##               (c) Bartolomeo Stellato,  Goran Banjac
##         University of Oxford  -  Stanford University 2019
## -----------------------------------------------------------------
## problem:  variables n = 1119, constraints m = 2227
```

```
##           nnz(P) + nnz(A) = 9866
## settings: linear system solver = qdldl,
##           eps_abs = 1.0e-05, eps_rel = 1.0e-05,
##           eps_prim_inf = 1.0e-04, eps_dual_inf = 1.0e-04,
##           rho = 1.00e-01 (adaptive),
##           sigma = 1.00e-06, alpha = 1.60, max_iter = 4000
##           check_termination: on (interval 25),
##           scaling: on, scaled_termination: off
##           warm start: on, polish: off, time_limit: off
##
## iter   objective    pri res    dua res    rho        time
##    1  -8.0792e+07   2.34e+02   4.62e+10   1.00e-01   2.83e-03s
##  200  -1.1189e+08   4.25e-04   1.64e+02   1.51e-02   1.06e-02s
##  400  -1.1189e+08   1.06e-04   6.58e-01   2.35e-04   1.84e-02s
##  550  -1.1189e+08   7.52e-06   6.63e-02   3.68e-05   2.38e-02s
##
## status:               solved
## number of iterations: 550
## optimal objective:    -111892871.0672
## run time:             2.39e-02s
## optimal rho estimate: 2.94e-05

## Reordering weights...

## Creating linear term vector...

## Creating quadratic term matrix...

## Creating constraint matrix...

##  x Sum to one constraint

##  x Upper and lower bounds

##  (SKIPPING) Enforce exact global balance

##  x Fit weights to data

##  x Constrain treated weights to be zero

##  x Combining constraints

## -----------------------------------------------------------------
##            OSQP v0.6.0  -  Operator Splitting QP Solver
##               (c) Bartolomeo Stellato,  Goran Banjac
##         University of Oxford  -  Stanford University 2019
## -----------------------------------------------------------------
## problem:  variables n = 1119, constraints m = 2227
##           nnz(P) + nnz(A) = 9866
## settings: linear system solver = qdldl,
##           eps_abs = 1.0e-05, eps_rel = 1.0e-05,
```

```
##           eps_prim_inf = 1.0e-04, eps_dual_inf = 1.0e-04,
##           rho = 1.00e-01 (adaptive),
##           sigma = 1.00e-06, alpha = 1.60, max_iter = 4000
##           check_termination: on (interval 25),
##           scaling: on, scaled_termination: off
##           warm start: on, polish: off, time_limit: off
##
## iter   objective    pri res    dua res    rho        time
##    1  -8.0792e+07   2.34e+02   4.62e+10   1.00e-01   1.95e-03s
##  200  -1.1189e+08   2.88e-03   2.05e+01   3.37e-03   9.97e-03s
##  400  -1.1189e+08   1.05e-04   5.87e-01   1.27e-04   1.76e-02s
##  600  -1.1189e+08   1.22e-06   1.21e-01   1.27e-04   2.49e-02s
##  625  -1.1189e+08   9.74e-07   9.92e-02   1.27e-04   2.58e-02s
##
## status:               solved
## number of iterations: 625
## optimal objective:    -111892871.0554
## run time:             2.58e-02s
## optimal rho estimate: 2.99e-05

## Reordering weights...

## Creating linear term vector...

## Creating quadratic term matrix...

## Creating constraint matrix...

##  x Sum to one constraint

##  x Upper and lower bounds

##  (SKIPPING) Enforce exact global balance

##  x Fit weights to data

##  x Constrain treated weights to be zero

##  x Combining constraints

## -----------------------------------------------------------------
##           OSQP v0.6.0  -  Operator Splitting QP Solver
##              (c) Bartolomeo Stellato,  Goran Banjac
##        University of Oxford  -  Stanford University 2019
## -----------------------------------------------------------------
## problem:  variables n = 1119, constraints m = 2227
##           nnz(P) + nnz(A) = 9866
## settings: linear system solver = qdldl,
##           eps_abs = 1.0e-05, eps_rel = 1.0e-05,
##           eps_prim_inf = 1.0e-04, eps_dual_inf = 1.0e-04,
##           rho = 1.00e-01 (adaptive),
```

```
##           sigma = 1.00e-06, alpha = 1.60, max_iter = 4000
##           check_termination: on (interval 25),
##           scaling: on, scaled_termination: off
##           warm start: on, polish: off, time_limit: off
##
## iter   objective    pri res    dua res    rho        time
##    1  -8.0792e+07   2.34e+02   4.62e+10   1.00e-01   1.75e-03s
##  200  -1.1189e+08   2.88e-03   2.05e+01   3.37e-03   9.35e-03s
##  400  -1.1189e+08   1.05e-04   5.87e-01   1.27e-04   1.68e-02s
##  600  -1.1189e+08   1.22e-06   1.21e-01   1.27e-04   2.52e-02s
##  625  -1.1189e+08   9.74e-07   9.92e-02   1.27e-04   2.63e-02s
##
## status:               solved
## number of iterations: 625
## optimal objective:    -111892871.0554
## run time:             2.63e-02s
## optimal rho estimate: 2.99e-05

## Reordering weights...

## Creating linear term vector...

## Creating quadratic term matrix...

## Creating constraint matrix...

##  x Sum to one constraint

##  x Upper and lower bounds

##  (SKIPPING) Enforce exact global balance

##  x Fit weights to data

##  x Constrain treated weights to be zero

##  x Combining constraints

## -----------------------------------------------------------------
##            OSQP v0.6.0  -  Operator Splitting QP Solver
##               (c) Bartolomeo Stellato,  Goran Banjac
##         University of Oxford  -  Stanford University 2019
## -----------------------------------------------------------------
## problem:  variables n = 1119, constraints m = 2227
##           nnz(P) + nnz(A) = 9866
## settings: linear system solver = qdldl,
##           eps_abs = 1.0e-05, eps_rel = 1.0e-05,
##           eps_prim_inf = 1.0e-04, eps_dual_inf = 1.0e-04,
##           rho = 1.00e-01 (adaptive),
##           sigma = 1.00e-06, alpha = 1.60, max_iter = 4000
##           check_termination: on (interval 25),
```

```
##           scaling: on, scaled_termination: off
##           warm start: on, polish: off, time_limit: off
##
## iter   objective    pri res    dua res    rho        time
##    1  -8.0792e+07   2.34e+02   4.62e+10   1.00e-01   1.76e-03s
##  200  -1.1189e+08   2.88e-03   2.05e+01   3.37e-03   8.92e-03s
##  400  -1.1189e+08   1.05e-04   5.87e-01   1.27e-04   1.69e-02s
##  600  -1.1189e+08   1.22e-06   1.21e-01   1.27e-04   2.51e-02s
##  625  -1.1189e+08   9.74e-07   9.92e-02   1.27e-04   2.60e-02s
##
## status:               solved
## number of iterations: 625
## optimal objective:    -111892871.0554
## run time:             2.61e-02s
## optimal rho estimate: 2.99e-05

## Reordering weights...
```



Lambda	point estimate	95% confidence interval
1	[2.11, 2.11]	[1.92, 2.33]
3	[1.04, 3]	[0.79, 3.19]
5.9	[0.34, 3.36]	[-0.01, 3.56]
7	[0.17, 3.41]	[-0.24, 3.58]