Unlocking the power of big data: The importance of measurement error in machine assisted content analysis

Nathan TeBlunthuis

nathan.teblunthuis@northwestern.edu

https://teblunthuis.cc

File failed to load: https://mathjax.rstudio.com/latest/extensions/MathZoom.js

Machine assistent content analysis (MACA)

Machine assisted content analysis (MACA) uses machine learning for scientific measurement.

Content analysis: Statistical analysis of variables measured by human labeling ("coding") of content. This might be simple categorical labels, or maybe more advanced annotations.

Machine assisted content analysis (MACA) uses machine learning for scientific measurement.

Content analysis: Statistical analysis of variables measured by human labeling ("coding") of content. This might be simple categorical labels, or maybe more advanced annotations.

Downside: Human labeling is *a lot* of work.

Machine assisted content analysis (MACA) uses machine learning for scientific measurement.

Content analysis: Statistical analysis of variables measured by human labeling ("coding") of content. This might be simple categorical labels, or maybe more advanced annotations.

Downside: Human labeling is *a lot* of work.

Machine assisted content analysis: Use a *predictive algorithm* (often trained on human-made labels) to measure variables for use in a downstream *primary analysis*.

Machine assisted content analysis (MACA) uses machine learning for scientific measurement.

Content analysis: Statistical analysis of variables measured by human labeling ("coding") of content. This might be simple categorical labels, or maybe more advanced annotations.

Downside: Human labeling is *a lot* of work.

Machine assisted content analysis: Use a *predictive algorithm* (often trained on human-made labels) to measure variables for use in a downstream *primary analysis.*

Downside: Algorithms can be *biased* and *inaccurate* in ways that could invalidate the statistical analysis.

How can MACA go wrong?

Algorithms can be *biased* and *error prone* (*noisy*).

How can MACA go wrong?

Algorithms can be *biased* and *error prone* (*noisy*).

Predictor bias is a potentially difficult problem that requires causal inference methods. I'll focus on *noise* for now.

How can MACA go wrong?

Algorithms can be *biased* and *error prone* (*noisy*).

Predictor bias is a potentially difficult problem that requires causal inference methods. I'll focus on *noise* for now.

Noise in the predictive model introduces bias in the primary analysis.

How can MACA go wrong?

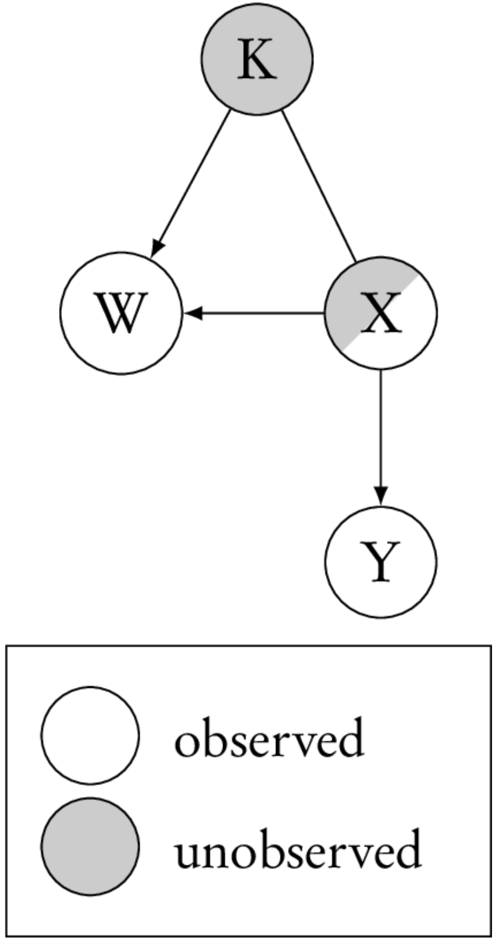
Algorithms can be *biased* and *error prone* (*noisy*).

Predictor bias is a potentially difficult problem that requires causal inference methods. I'll focus on *noise* for now.

Noise in the predictive model introduces bias in the primary analysis.

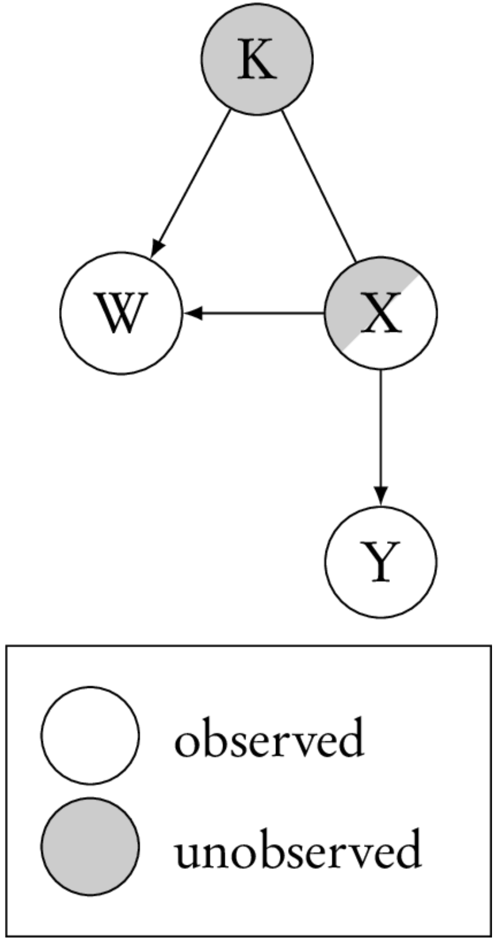
We can reduce and sometimes even *eliminate* this bias introduced by noise.

Example 1: An unbiased, but noisy classifier



x is *partly observed* because we have *validation data* x^*.

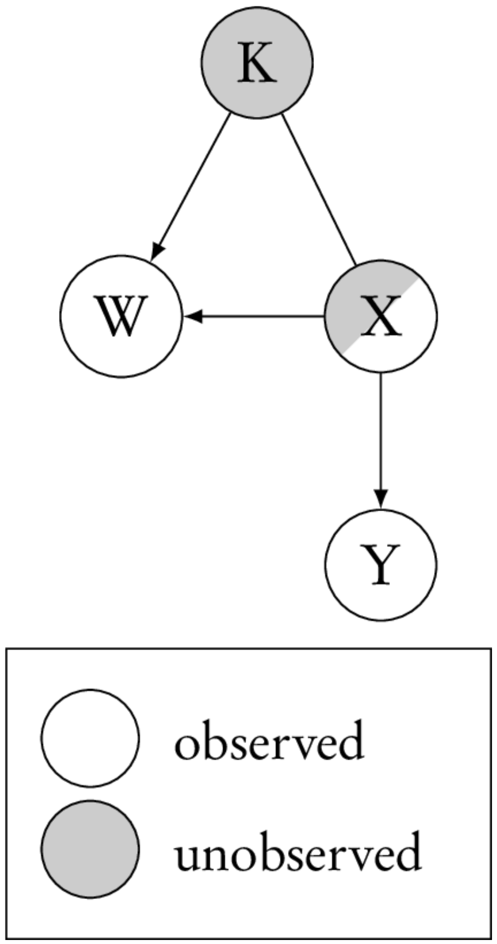
Example 1: An unbiased, but noisy classifier



\(x\) is *partly observed* because we have *validation data* \(x^*\).

\(k\) are the *features* used by the *predictive model* \(g(k)\).

Example 1: An unbiased, but noisy classifier

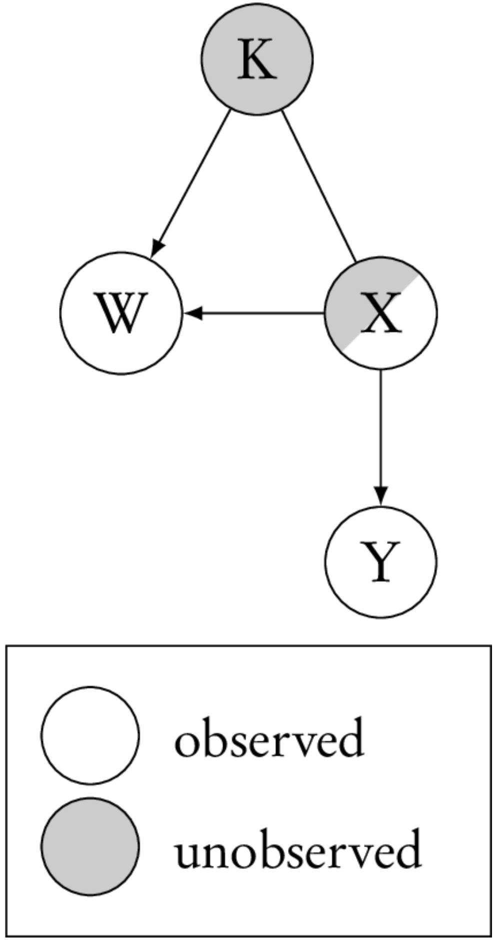


x is *partly observed* because we have *validation data* x^*.

k are the *features* used by the *predictive model* $g(k)$.

The predictions w are a *proxy variable* $g(k) = \hat{x} = w$.

Example 1: An unbiased, but noisy classifier



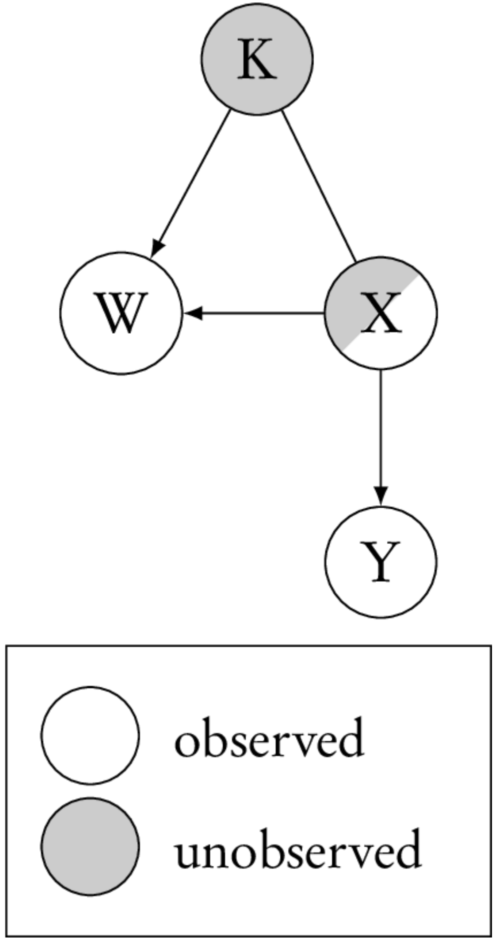
x is *partly observed* because we have *validation data* x^*.

k are the *features* used by the *predictive model* $g(k)$.

The predictions w are a *proxy variable* $g(k) = \hat{x} = w$.

$x = w + \xi$ because the predictive model makes errors.

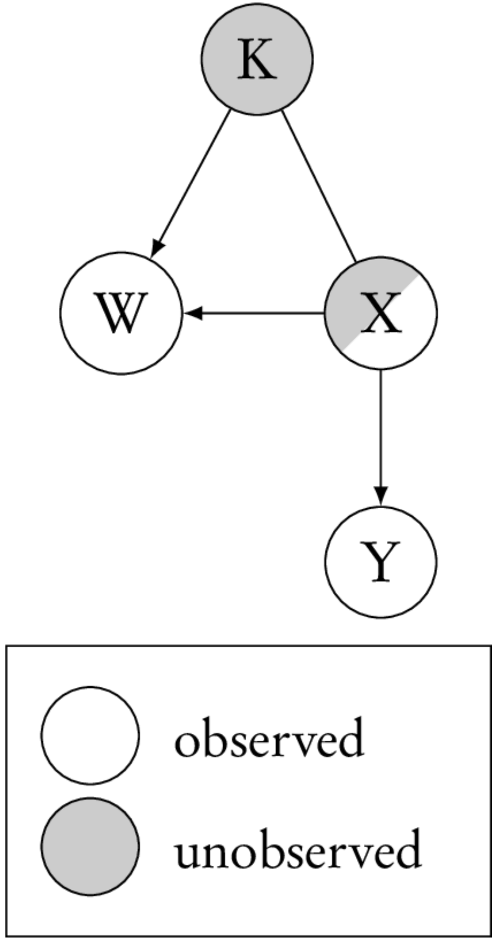
Noise in a *covariate* creates *attenuation bias*.



We want to estimate, $y = Bx + \varepsilon$, but we estimate $y = Bw + \varepsilon$ instead.

$x = w + \xi$ because the predictive model makes errors.

Noise in a *covariate* creates *attenuation bias*.

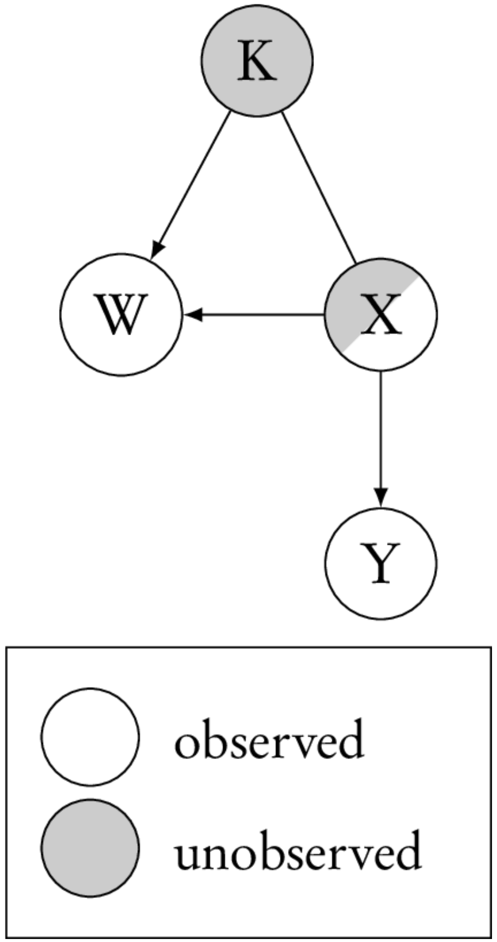


We want to estimate, $y = Bx + \varepsilon$, but we estimate $y = Bw + \varepsilon$ instead.

$x = w + \xi$ because the predictive model makes errors.

Assume $g(k)$ is *unbiased* so $E(\xi)=0$. Also assume error is *nondifferential* so $E(\xi y)=0$:

Noise in a *covariate* creates *attenuation bias*.



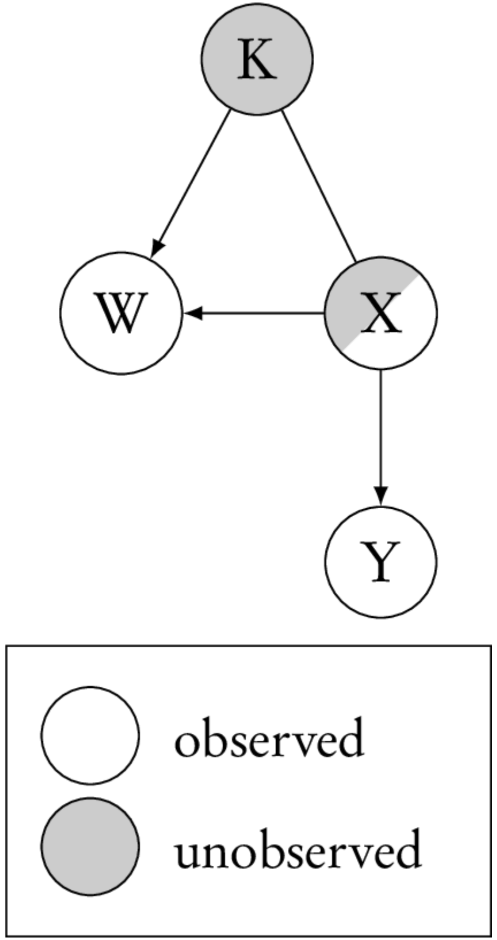
We want to estimate, $y = Bx + \varepsilon$, but we estimate $y = Bw + \varepsilon$ instead.

$x = w + \xi$ because the predictive model makes errors.

Assume $g(k)$ is *unbiased* so $E(\xi)=0$. Also assume error is *nondifferential* so $E(\xi y)=0$:

$$\widehat{B_w}^{ols}=\frac{\sum^n_{j=j}{(x_j + \xi_j - \overline{(x + \xi)})}(y_j - \bar{y})}{\sum_{j=1}^n{(x_j + \xi_j - \overline{(x+\xi)})^2}} = \frac{\sum^n_{j=j}{(x_j - \bar{x})(y_j - \bar{y})}}{\sum_{j=1}^n{(x_j + \xi_j - \bar{x}){^2}}}$$

Noise in a *covariate* creates *attenuation bias*.



We want to estimate, $y = Bx + \varepsilon$, but we estimate $y = Bw + \varepsilon$ instead.

$x = w + \xi$ because the predictive model makes errors.

Assume $g(k)$ is *unbiased* so $E(\xi)=0$. Also assume error is *nondifferential* so $E(\xi y)=0$:

$$\widehat{B_w}^{ols}=\frac{\sum^n_{j=j}{(x_j + \xi_j - \overline{(x + \xi)})}(y_j - \bar{y})}{\sum_{j=1}^n{(x_j + \xi_j - \overline{(x+\xi)})^2}} = \frac{\sum^n_{j=j}{(x_j - \bar{x})(y_j - \bar{y})}}{\sum_{j=1}^n{(x_j + \color{red}{\xi_j} - \bar{x})\color{red}{^2}}}$$

In this scenario, it's clear that $\widehat{B_w}^{ols} < B_x$.

Beyond attenuation bias

Measurement error can theaten validity because:

- Attenuation bias *spreads* (e.g., to marginal effects as illustrated later).

Beyond attenuation bias

Measurement error can theaten validity because:

- Attenuation bias *spreads* (e.g., to marginal effects as illustrated later).
- Measurement error can be *differential*— not distributed evenly and possible correlated with x, y, or ε.

Beyond attenuation bias

Measurement error can theaten validity because:

- Attenuation bias *spreads* (e.g., to marginal effects as illustrated later).
- Measurement error can be *differential*— not distributed evenly and possible correlated with x, y, or ε.
- *Bias can be away from 0* in GLMs and nonlinear models or if measurement error is differential.

Beyond attenuation bias

Measurement error can theaten validity because:

- Attenuation bias *spreads* (e.g., to marginal effects as illustrated later).
- Measurement error can be *differential*— not distributed evenly and possible correlated with x, y, or ε.
- *Bias can be away from 0* in GLMs and nonlinear models or if measurement error is differential.
- *Confounding* if the *predictive model is biased* introducing a correlation the measurement error and the residuals $(E[\xi\varepsilon]=0)$.

Correcting measurement error

There's a vast literature in statistics on measurement error. Mostly about noise you'd find in sensors. Lots of ideas. No magic bullets.

Correcting measurement error

There's a vast literature in statistics on measurement error. Mostly about noise you'd find in sensors. Lots of ideas. No magic bullets.

I'm going to briefly cover 3 different approaches: *multiple imputation*, *regression calibration* and *2SLS+GMM*.

Correcting measurement error

There's a vast literature in statistics on measurement error. Mostly about noise you'd find in sensors. Lots of ideas. No magic bullets.

I'm going to briefly cover 3 different approaches: *multiple imputation*, *regression calibration* and *2SLS+GMM*.

These all depend on *validation data*. I'm going to ignore where this comes from, but assume it's a random sample of the hypothesis testing dataset.

Correcting measurement error

There's a vast literature in statistics on measurement error. Mostly about noise you'd find in sensors. Lots of ideas. No magic bullets.

I'm going to briefly cover 3 different approaches: *multiple imputation*, *regression calibration* and *2SLS+GMM*.

These all depend on *validation data*. I'm going to ignore where this comes from, but assume it's a random sample of the hypothesis testing dataset.

You can *and should* use it to improve your statistical estimates.

Multiple Imputation (MI) treats Measurement Error as a Missing Data Problem

1. Use validation data to estimate $f(x|w,y)$, a probabilistic model of x.

Multiple Imputation (MI) treats Measurement Error as a Missing Data Problem

1. Use validation data to estimate $f(x|w,y)$, a probabilistic model of x.
2. *Sample* m datasets from $\widehat{f(x|w,y)}$.

Multiple Imputation (MI) treats Measurement Error as a Missing Data Problem

1. Use validation data to estimate $f(x|w,y)$, a probabilistic model of x.
2. *Sample* m datasets from $\widehat{f(x|w,y)}$.
3. Run your analysis on each of the m datasets.

Multiple Imputation (MI) treats Measurement Error as a Missing Data Problem

1. Use validation data to estimate $f(x|w,y)$, a probabilistic model of x.
2. *Sample* m datasets from $\widehat{f(x|w,y)}$.
3. Run your analysis on each of the m datasets.
4. Average the results from the m analyses using Rubin's rules.

Multiple Imputation (MI) treats Measurement Error as a Missing Data Problem

1. Use validation data to estimate $f(x|w,y)$, a probabilistic model of x.
2. *Sample* m datasets from $\widehat{f(x|w,y)}$.
3. Run your analysis on each of the m datasets.
4. Average the results from the m analyses using Rubin's rules.

Advantages: *Very flexible!* Sometimes can work if the predictor $g(k)$ is biased. Good R packages (**`{Amelia}`**, `{mi}`, `{mice}`, `{brms}`).

Multiple Imputation (MI) treats Measurement Error as a Missing Data Problem

1. Use validation data to estimate $f(x|w,y)$, a probabilistic model of x.
2. *Sample* m datasets from $\widehat{f(x|w,y)}$.
3. Run your analysis on each of the m datasets.
4. Average the results from the m analyses using Rubin's rules.

Advantages: *Very flexible!* Sometimes can work if the predictor $g(k)$ is biased. Good R packages (**`{Amelia}`**, `{mi}`, `{mice}`, `{brms}`).

Disadvantages: Results depend on quality of $\widehat{f(x|w,y)}$; May require more validation data, computationally expensive, statistically inefficient and doesn't seem to benefit much from larger datasets.

Regression calibration directly adjusts for attenuation bias.

1. Use validation data to estimate the errors \(\hat{\xi}\).

Regression calibration directly adjusts for attenuation bias.

1. Use validation data to estimate the errors $\hat{\xi}$.
2. Use $\hat{\xi}$ to correct the OLS estimate.

Regression calibration directly adjusts for attenuation bias.

1. Use validation data to estimate the errors \(\hat{\xi}\).
2. Use \(\hat{\xi}\) to correct the OLS estimate.
3. Correct the standard errors using MLE or bootstrapping.

Regression calibration directly adjusts for attenuation bias.

1. Use validation data to estimate the errors $\hat{\xi}$.
2. Use $\hat{\xi}$ to correct the OLS estimate.
3. Correct the standard errors using MLE or bootstrapping.

Advantages: Simple, fast.

Regression calibration directly adjusts for attenuation bias.

1. Use validation data to estimate the errors $\hat{\xi}$.
2. Use $\hat{\xi}$ to correct the OLS estimate.
3. Correct the standard errors using MLE or bootstrapping.

Advantages: Simple, fast.

Disadvantages: Limited to OLS models. Requires an unbiased predictor $g(k)$. R support (`{mecor}` R package) is pretty new.

2SLS+GMM is designed for this specific problem

PA

Machine Learning Predictions as Regression Covariates

Christian Fong[1] and Matthew Tyler[2]

[1] Assistant Professor, Department of Political Science, University of Michigan, Ann Arbor, MI, USA. Email: cjfong@umich.edu
[2] Ph.D. Candidate, Department of Political Science, Stanford University, Stanford, CA, USA. Email: mdtyler@stanford.edu

Abstract

In text, images, merged surveys, voter files, and elsewhere, data sets are often missing important covariates, either because they are latent features of observations (such as sentiment in text) or because they are not collected (such as race in voter files). One promising approach for coping with this missing data is to find the true values of the missing covariates for a subset of the observations and then train a machine learning algorithm to predict the values of those covariates for the rest. However, plugging in these predictions without regard for prediction error renders regression analyses biased, inconsistent, and overconfident. We characterize the severity of the problem posed by prediction error, describe a procedure to avoid these inconsistencies under comparatively general assumptions, and demonstrate the performance of our estimators through simulations and a study of hostile political dialogue on the Internet. We provide software implementing our approach.

Keywords: machine learning, classification, inference, instrumental variables

Regression calibration with a trick.

1. Estimate $x = w + \xi$ to obtain $\hat{x}$. (First-stage LS).

2SLS+GMM is designed for this specific problem

Machine Learning Predictions as Regression Covariates

Christian Fong[1] and Matthew Tyler[2]

[1]Assistant Professor, Department of Political Science, University of Michigan, Ann Arbor, MI, USA. Email: cjfong@umich.edu
[2]Ph.D. Candidate, Department of Political Science, Stanford University, Stanford, CA, USA. Email: mdtyler@stanford.edu

Abstract
In text, images, merged surveys, voter files, and elsewhere, data sets are often missing important covariates, either because they are latent features of observations (such as sentiment in text) or because they are not collected (such as race in voter files). One promising approach for coping with this missing data is to find the true values of the missing covariates for a subset of the observations and then train a machine learning algorithm to predict the values of those covariates for the rest. However, plugging in these predictions without regard for prediction error renders regression analyses biased, inconsistent, and overconfident. We characterize the severity of the problem posed by prediction error, describe a procedure to avoid these inconsistencies under comparatively general assumptions, and demonstrate the performance of our estimators through simulations and a study of hostile political dialogue on the Internet. We provide software implementing our approach.

Keywords: machine learning, classification, inference, instrumental variables

Regression calibration with a trick.

1. Estimate $x = w + \xi$ to obtain $\hat{x}$. (First-stage LS).
2. Estimate $y = B^{2sls}\hat{x} + \varepsilon^{2sls}$. (Second-stage LS / regression calibration).

2SLS+GMM is designed for this specific problem

PA

Machine Learning Predictions as Regression Covariates

Christian Fong[1] and Matthew Tyler[2]

[1] Assistant Professor, Department of Political Science, University of Michigan, Ann Arbor, MI, USA. Email: cjfong@umich.edu
[2] Ph.D. Candidate, Department of Political Science, Stanford University, Stanford, CA, USA. Email: mdtyler@stanford.edu

Abstract
In text, images, merged surveys, voter files, and elsewhere, data sets are often missing important covariates, either because they are latent features of observations (such as sentiment in text) or because they are not collected (such as race in voter files). One promising approach for coping with this missing data is to find the true values of the missing covariates for a subset of the observations and then train a machine learning algorithm to predict the values of those covariates for the rest. However, plugging in these predictions without regard for prediction error renders regression analyses biased, inconsistent, and overconfident. We characterize the severity of the problem posed by prediction error, describe a procedure to avoid these inconsistencies under comparatively general assumptions, and demonstrate the performance of our estimators through simulations and a study of hostile political dialogue on the Internet. We provide software implementing our approach.

Keywords: machine learning, classification, inference, instrumental variables

Regression calibration with a trick.

1. Estimate $x = w + \xi$ to obtain $\hat{x}$. (First-stage LS).
2. Estimate $y = B^{2sls}\hat{x} + \varepsilon^{2sls}$. (Second-stage LS / regression calibration).
3. Estimate $y = B^{val}x^* + \varepsilon^{val}$. (Validation dataset model).

2SLS+GMM is designed for this specific problem

PA

Machine Learning Predictions as Regression Covariates

Christian Fong[1] and Matthew Tyler[2]

[1] *Assistant Professor, Department of Political Science, University of Michigan, Ann Arbor, MI, USA. Email: cjfong@umich.edu*
[2] *Ph.D. Candidate, Department of Political Science, Stanford University, Stanford, CA, USA. Email: mdtyler@stanford.edu*

Abstract

In text, images, merged surveys, voter files, and elsewhere, data sets are often missing important covariates, either because they are latent features of observations (such as sentiment in text) or because they are not collected (such as race in voter files). One promising approach for coping with this missing data is to find the true values of the missing covariates for a subset of the observations and then train a machine learning algorithm to predict the values of those covariates for the rest. However, plugging in these predictions without regard for prediction error renders regression analyses biased, inconsistent, and overconfident. We characterize the severity of the problem posed by prediction error, describe a procedure to avoid these inconsistencies under comparatively general assumptions, and demonstrate the performance of our estimators through simulations and a study of hostile political dialogue on the Internet. We provide software implementing our approach.

Keywords: machine learning, classification, inference, instrumental variables

Regression calibration with a trick.

1. Estimate $x = w + \xi$ to obtain $\hat{x}$. (First-stage LS).
2. Estimate $y = B^{2sls}\hat{x} + \varepsilon^{2sls}$. (Second-stage LS / regression calibration).
3. Estimate $y = B^{val}x^* + \varepsilon^{val}$. (Validation dataset model).
4. Combine B^{val} and B^{2sls} using the generalized method of moments (GMM).

2SLS+GMM is designed for this specific problem

PA

Machine Learning Predictions as Regression Covariates

Christian Fong[1] and Matthew Tyler[2]

[1]Assistant Professor, Department of Political Science, University of Michigan, Ann Arbor, MI, USA. Email: cjfong@umich.edu
[2]Ph.D. Candidate, Department of Political Science, Stanford University, Stanford, CA, USA. Email: mdtyler@stanford.edu

Abstract

In text, images, merged surveys, voter files, and elsewhere, data sets are often missing important covariates, either because they are latent features of observations (such as sentiment in text) or because they are not collected (such as race in voter files). One promising approach for coping with this missing data is to find the true values of the missing covariates for a subset of the observations and then train a machine learning algorithm to predict the values of those covariates for the rest. However, plugging in these predictions without regard for prediction error renders regression analyses biased, inconsistent, and overconfident. We characterize the severity of the problem posed by prediction error, describe a procedure to avoid these inconsistencies under comparatively general assumptions, and demonstrate the performance of our estimators through simulations and a study of hostile political dialogue on the Internet. We provide software implementing our approach.

Keywords: machine learning, classification, inference, instrumental variables

Regression calibration with a trick.

1. Estimate $x = w + \xi$ to obtain $\hat{x}$. (First-stage LS).
2. Estimate $y = B^{2sls}\hat{x} + \varepsilon^{2sls}$. (Second-stage LS / regression calibration).
3. Estimate $y = B^{val}x^* + \varepsilon^{val}$. (Validation dataset model).
4. Combine B^{val} and B^{2sls} using the generalized method of moments (GMM).

Advantages: Accurate. Sometimes robust if biased predictor $g(k)$ is biased. In theory, flexible to any models that can be fit using GMM.

2SLS+GMM is designed for this specific problem

PA

Machine Learning Predictions as Regression Covariates

Christian Fong[1] and Matthew Tyler[2]

[1]Assistant Professor, Department of Political Science, University of Michigan, Ann Arbor, MI, USA. Email: cjfong@umich.edu
[2]Ph.D. Candidate, Department of Political Science, Stanford University, Stanford, CA, USA. Email: mdtyler@stanford.edu

Abstract
In text, images, merged surveys, voter files, and elsewhere, data sets are often missing important covariates, either because they are latent features of observations (such as sentiment in text) or because they are not collected (such as race in voter files). One promising approach for coping with this missing data is to find the true values of the missing covariates for a subset of the observations and then train a machine learning algorithm to predict the values of those covariates for the rest. However, plugging in these predictions without regard for prediction error renders regression analyses biased, inconsistent, and overconfident. We characterize the severity of the problem posed by prediction error, describe a procedure to avoid these inconsistencies under comparatively general assumptions, and demonstrate the performance of our estimators through simulations and a study of hostile political dialogue on the Internet. We provide software implementing our approach.

Keywords: machine learning, classification, inference, instrumental variables

Regression calibration with a trick.

1. Estimate $x = w + \xi$ to obtain $\hat{x}$. (First-stage LS).
2. Estimate $y = B^{2sls}\hat{x} + \varepsilon^{2sls}$. (Second-stage LS / regression calibration).
3. Estimate $y = B^{val}x^* + \varepsilon^{val}$. (Validation dataset model).
4. Combine B^{val} and B^{2sls} using the generalized method of moments (GMM).

Advantages: Accurate. Sometimes robust if biased predictor $g(k)$ is biased. In theory, flexible to any models that can be fit using GMM.

Disadvantages: Implementation (`{predictionError}`) is new. API is cumbersome and only supports linear models. Not robust if $E(w\varepsilon) \ne 0$. GMM may be unfamiliar to audiences.

Testing attention bias correction

I've run simulations to test these approaches in several scenarios.

The model is not very good: about 70% accurate.

Most plausible scenario:

y is continuous and normal-ish.

Testing attention bias correction

I've run simulations to test these approaches in several scenarios.

The model is not very good: about 70% accurate.

Most plausible scenario:

y is continuous and normal-ish.

x is binary (human labels) $P(x)=0.5$.

Testing attention bias correction

I've run simulations to test these approaches in several scenarios.

The model is not very good: about 70% accurate.

Most plausible scenario:

y is continuous and normal-ish.

x is binary (human labels) $P(x)=0.5$.

w is the *continuous predictor* (e.g., probability) output of $f(x)$ (not binary predictions).

Testing attention bias correction

I've run simulations to test these approaches in several scenarios.

The model is not very good: about 70% accurate.

Most plausible scenario:

y is continuous and normal-ish.

x is binary (human labels) $P(x)=0.5$.

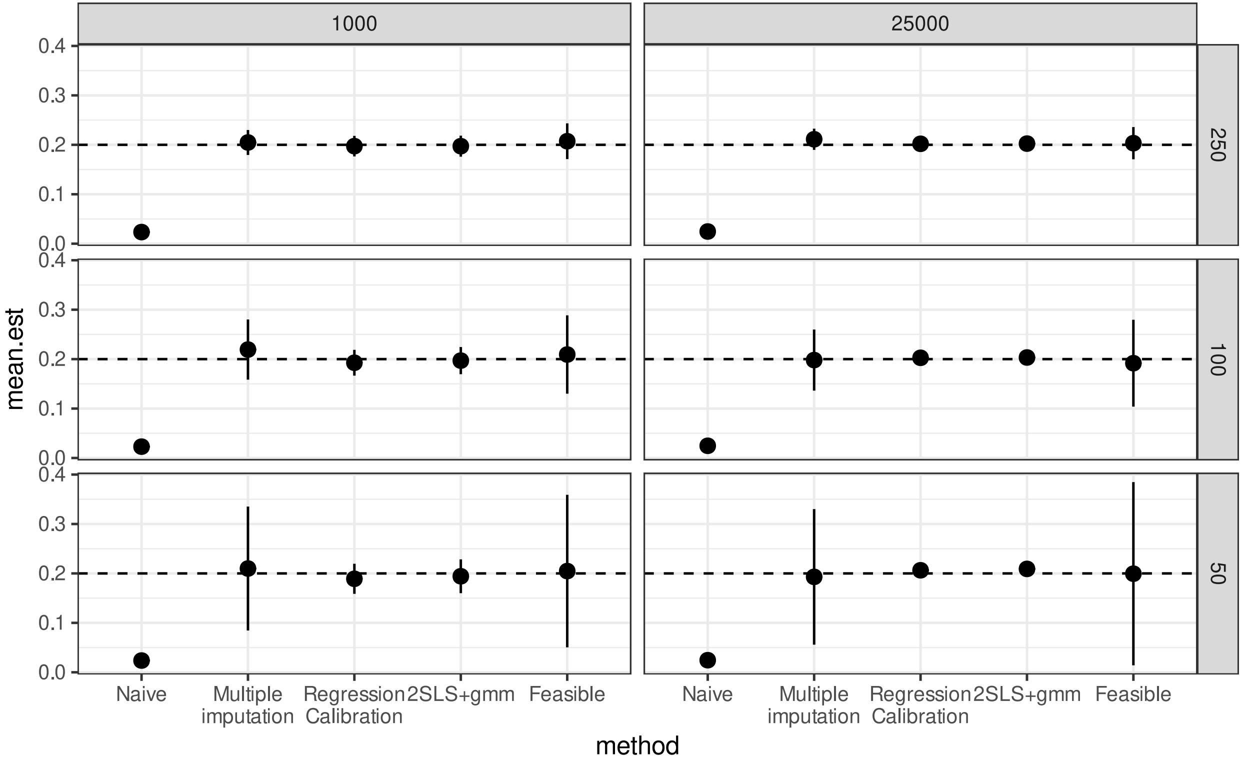
w is the *continuous predictor* (e.g., probability) output of $f(x)$ (not binary predictions).

if w is binary, most methods struggle, but regression calibration and 2SLS+GMM can do okay.

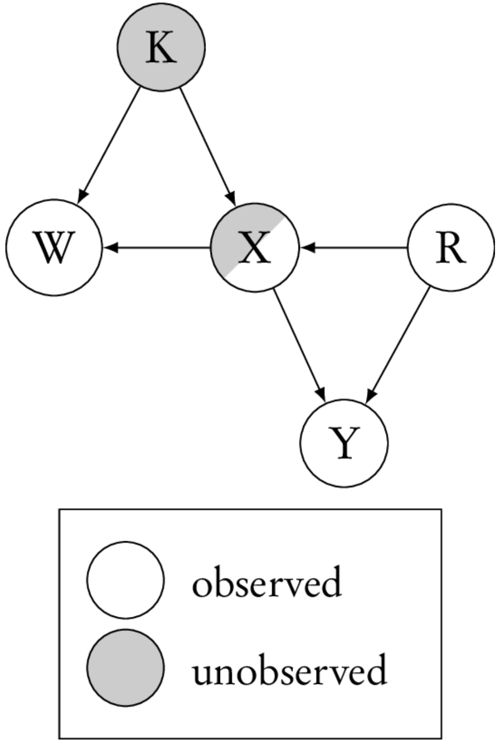
Example 1: estimator of the effect of x

All methods work in this scenario

Multiple imputation is inefficient.



What about bias?



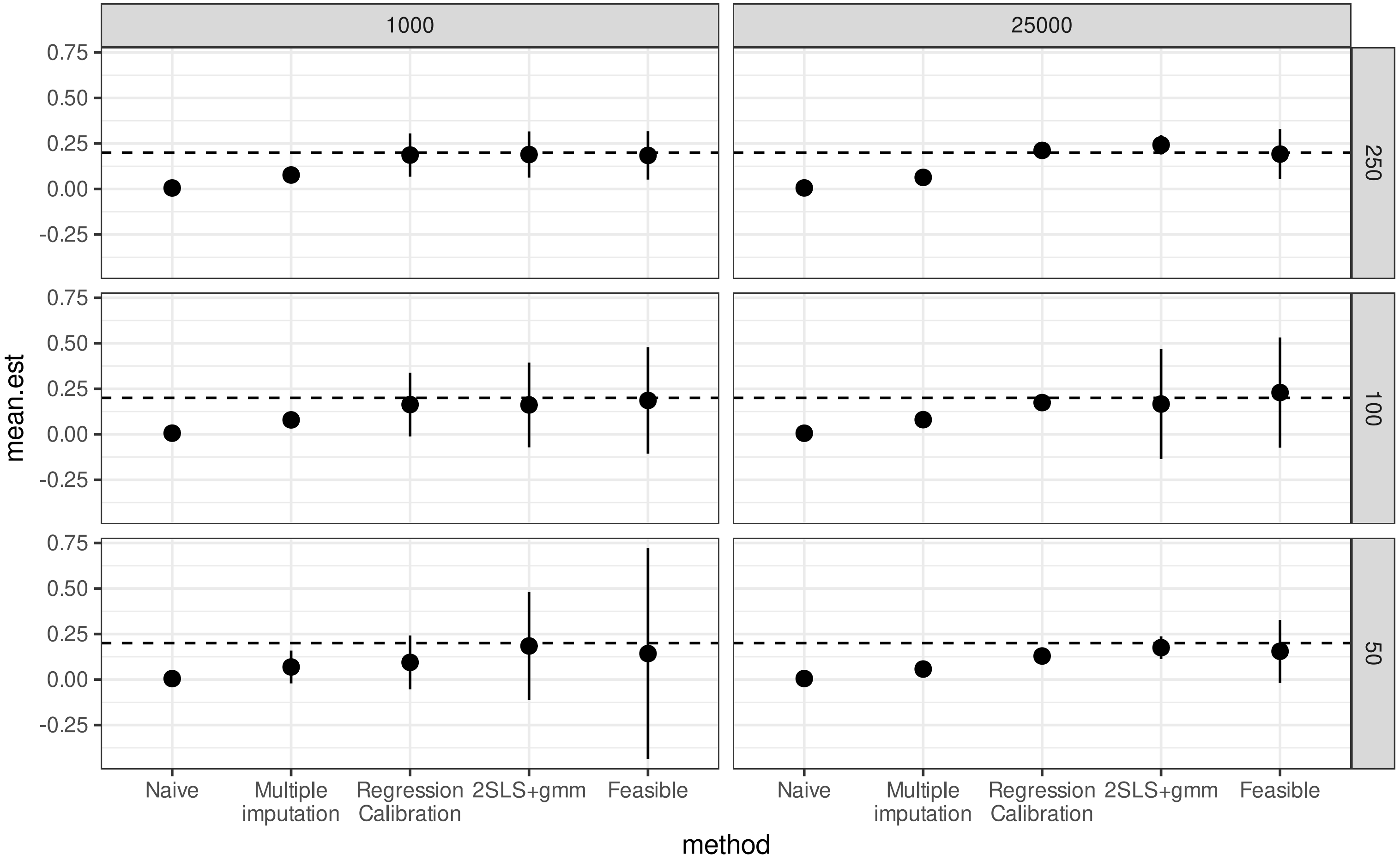
A few notes on this scenario.

$B_x = 0.2$, $B_g=-0.2$ and $sd(\varepsilon)=3$. So the signal-to-noise ratio is high.

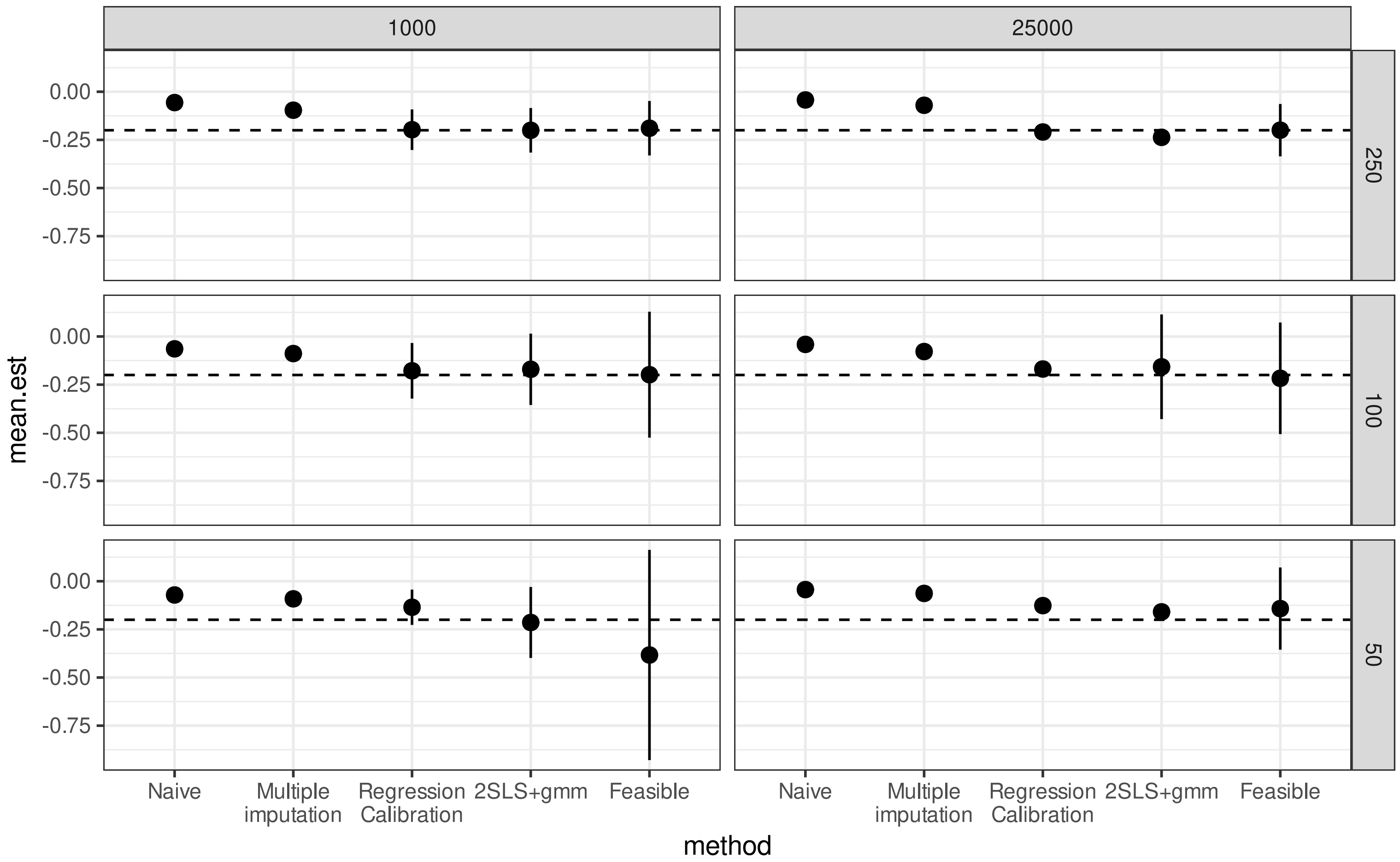
r can be concieved of as a missing feature in the predictive model $g(k)$ that is also correlated with y.

For example r might be the *race* of a commentor, x could be *racial harassment*, y whether the commentor gets banned and k only has textual features but human coders can see user profiles to know r.

Example 2: Estimates of the effect of x



Example 2: Estimates of the effect of r



Takeaways from example 2

Bias in the predictive model creates bias in hypothesis tests.

Takeaways from example 2

Bias in the predictive model creates bias in hypothesis tests.

Bias can be corrected *in this case*.

Takeaways from example 2

Bias in the predictive model creates bias in hypothesis tests.

Bias can be corrected *in this case*.

The next scenario has bias that's more tricky.

Takeaways from example 2

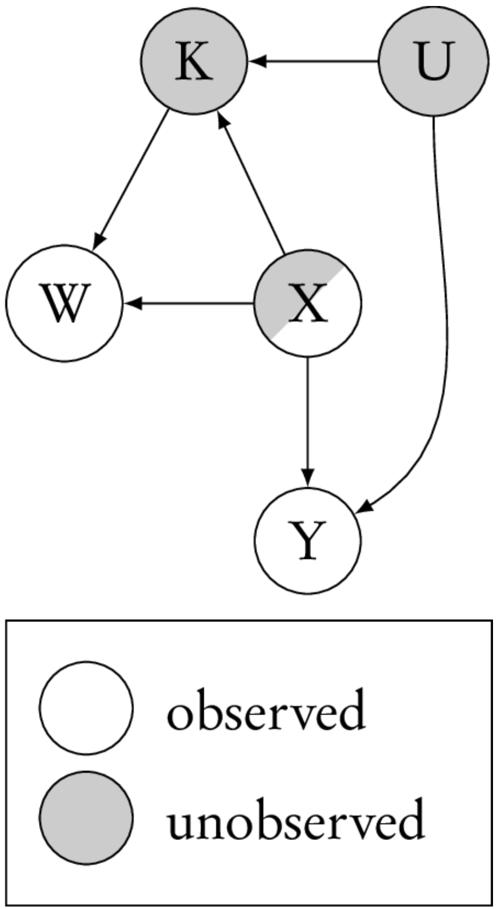
Bias in the predictive model creates bias in hypothesis tests.

Bias can be corrected *in this case*.

The next scenario has bias that's more tricky.

Multiple imputation helps, but doesn't fully correct the bias.

When will GMM+2SLS fail?

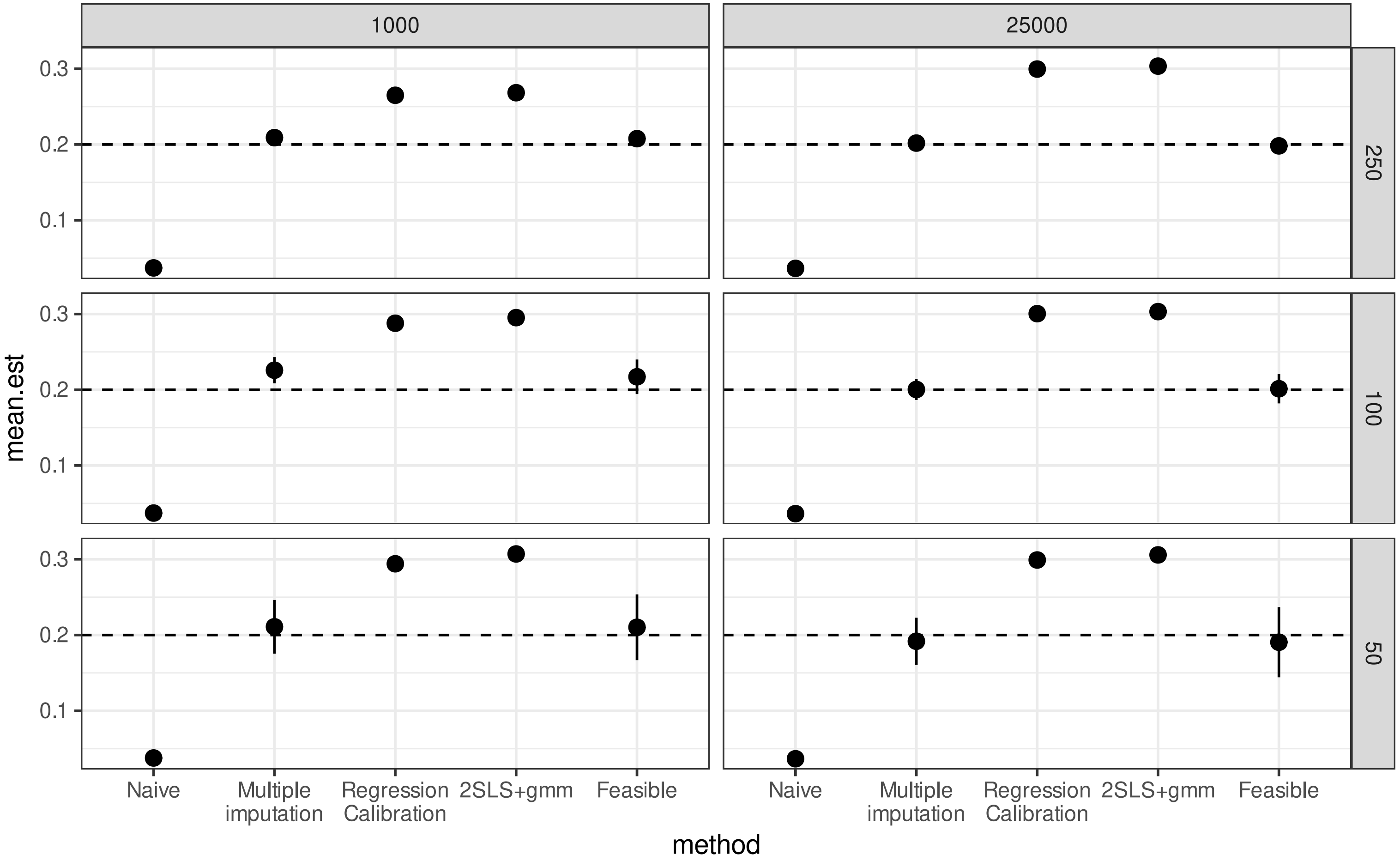


The catch with GMM:

Exclusion restriction: $E[w \varepsilon] = 0$.

The restriction is violated if a variable U causes both K and Y and X causes K (not visa-versa).

Example 3: Estimates of the effect of x



Takaways

- Attenuation bias can be a big problem with noisy predictors—leading to small and biased estimates.
- For more general hypothesis tests or if the predictor is biased, measurement error can lead to false discovery.
- It's fixable with validation data—you may not need that much and you should already be getting it.
- This means it can be okay poor predictors for hypothesis testing.
- The ecosystem is underdeveloped, but a lot of methods have been researched.
- Take advantage of machine learning + big data and get precise estimates when the signal-to-noise ratio is high!

Future work: Noise in the *outcome*

I've been focusing on noise in *covariates*. What if the predictive algorithm is used to measure the *outcome* y?

Future work: Noise in the *outcome*

I've been focusing on noise in *covariates*. What if the predictive algorithm is used to measure the *outcome* y?

This isn't a problem in the simplest case (linear regression with homoskedastic errors). Noise in y is projected into the error term.

Future work: Noise in the *outcome*

I've been focusing on noise in *covariates*. What if the predictive algorithm is used to measure the *outcome* y?

This isn't a problem in the simplest case (linear regression with homoskedastic errors). Noise in y is projected into the error term.

Noise in the outcome is still a problem if errors are heteroskedastic and for GLMs / non-linear regression (e.g., logistic regression).

Future work: Noise in the *outcome*

I've been focusing on noise in *covariates*. What if the predictive algorithm is used to measure the *outcome* y?

This isn't a problem in the simplest case (linear regression with homoskedastic errors). Noise in y is projected into the error term.

Noise in the outcome is still a problem if errors are heteroskedastic and for GLMs / non-linear regression (e.g., logistic regression).

Multiple imputation (in theory) could help here. The other method's aren't designed for this case.

Future work: Noise in the *outcome*

I've been focusing on noise in *covariates*. What if the predictive algorithm is used to measure the *outcome* y?

This isn't a problem in the simplest case (linear regression with homoskedastic errors). Noise in y is projected into the error term.

Noise in the outcome is still a problem if errors are heteroskedastic and for GLMs / non-linear regression (e.g., logistic regression).

Multiple imputation (in theory) could help here. The other method's aren't designed for this case.

Solving this problem could be an important methodological contribution with a very broad impact.

Questions?

Links to slides:html pdf

Link to a messy git repository:

nathan.teblunthuis@northwestern.edu

@groceryheist

https://communitydata.science