

Introducing RobustiPy:

The next generation of model uncertainty software

Daniel Valdenegro, Jiani Yan, Duiyi Dai, and **Charlie Rahal**

Keynote Talk at International Conference on Social Computing

December 11, 2025



RESEARCH ARTICLE | SOCIAL SCIENCES | 

Observing many researchers using the same data and hypothesis reveals a hidden universe of uncertainty

Nate Breznau  , Elke Mark Rinke , Alexander Wuttke ,  +162, and Tomasz Żółtak  [Authors Info & Affiliations](#)

Edited by Douglas Massey, Princeton University, Princeton, NJ; received March 6, 2022; accepted August 22, 2022

October 28, 2022 | 119 (44) e2203150119 | <https://doi.org/10.1073/pnas.2203150119>

THIS ARTICLE HAS BEEN UPDATED

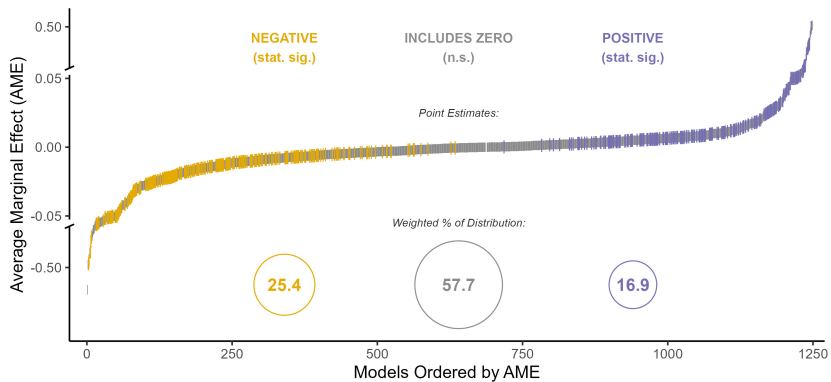
THIS ARTICLE HAS BEEN CORRECTED +

VIEW RELATED CONTENT +

 65,932 | 58



PDF/EPUB



Types of Model Uncertainty

- ◇ **Parameter Uncertainty:**
 - ▶ Variability due to estimation from limited data.
- ◇ **Structural Uncertainty:**
 - ▶ Inadequate model to capture system complexities fully.
 - ▶ Simplifications that may oversimplify or miss critical aspects.
- ◇ **Data Uncertainty:**
 - ▶ Errors in data collection or measurement.
 - ▶ Outliers affecting model stability or accuracy.
- ◇ **Model Formulation Uncertainty:**
 - ▶ Choice of mathematical model structure.
 - ▶ Assumptions about relationships between variables.

Some types of uncertainty can be handled by researchers, some can't.

Let's consider a simple example

- ◇ Imagine we want to know the effect of one variable x_1 , on an output variable y .
- ◇ Example: x_1 is the effect of inequality on y , state-level crime rate.
- ◇ That is to say: our 'estimand' is β_1 in the following regression:

$$y_i = \alpha + \beta_1 x_{i,1} + \varepsilon_i \quad (1)$$

OLS Regression Results

```

=====
Dep. Variable:          R      R-squared:          0.032
Model:                  OLS    Adj. R-squared:       0.011
Method:                 Least Squares    F-statistic:       2.010
Date:                  Sun, 12 Oct 2025    Prob (F-statistic): 0.163
Time:                  05:04:45    Log-Likelihood:    -237.21
No. Observations:      47    AIC:              478.4
Df Residuals:          45    BIC:              482.1
Df Model:              1
Covariance Type:       HCL1
=====

```

```

=====
              coef      std err          z      P>|z|      [0.025      0.975]
-----
Intercept    124.1773     26.789      4.635     0.000      71.672     176.682
Inequality   -0.1736      0.122     -1.418     0.156     -0.413      0.066
=====

```

```

=====
Omnibus:          8.646    Durbin-Watson:       2.461
Prob(Omnibus):    0.013    Jarque-Bera (JB):     7.831
Skew:             0.948    Prob(JB):             0.0199
Kurtosis:         3.636    Cond. No.             993.
=====

```

But don't we also need to correct for things like age?

OLS Regression Results

```

=====
Dep. Variable:          R      R-squared:          0.033
Model:                  OLS    Adj. R-squared:      -0.011
Method:                 Least Squares    F-statistic:      0.9856
Date:                   Sun, 12 Oct 2025    Prob (F-statistic): 0.381
Time:                   05:05:32    Log-Likelihood:   -237.19
No. Observations:      47    AIC:              480.4
Df Residuals:          44    BIC:              485.9
Df Model:               2
Covariance Type:       HC1
=====

```

```

=====
              coef      std err          z      P>|z|      [0.025      0.975]
-----
Intercept    111.2518     46.834      2.375     0.018     19.459    203.044
Inequality   -0.1997       0.172     -1.160     0.246     -0.537     0.138
Age           0.1299       0.449      0.289     0.773     -0.751     1.011
=====

```

```

=====
Omnibus:          8.799    Durbin-Watson:          2.457
Prob(Omnibus):    0.012    Jarque-Bera (JB):        8.002
Skew:             0.956    Prob(JB):                0.0183
Kurtosis:         3.655    Cond. No.:               2.79e+03
=====

```

What about if we build out a fuller model, in order to help this estimation satisfy the Gauss-Markov properties?

OLS Regression Results

```

=====
Dep. Variable:          R      R-squared:          0.522
Model:                  OLS    Adj. R-squared:       0.436
Method:                 Least Squares    F-statistic:       7.276
Date:                  Sun, 12 Oct 2025    Prob (F-statistic): 1.39e-05
Time:                  05:06:00    Log-Likelihood:    -220.64
No. Observations:      47    AIC:              457.3
Df Residuals:          39    BIC:              472.1
Df Model:              7
Covariance Type:       HC1
=====

```

```

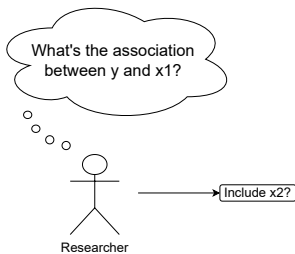
=====
              coef      std err          z      P>|z|      [0.025      0.975]
-----
Intercept    -764.2379    223.962     -3.412     0.001    -1203.196    -325.280
Inequality    0.8445      0.264      3.196     0.001      0.327      1.363
Age           0.9750      0.578      1.688     0.091     -0.157      2.107
Ed            0.9510      0.793      1.199     0.231     -0.604      2.506
Unemployment  -0.0593      0.302     -0.196     0.844     -0.652      0.533
Males         0.2211      0.241      0.918     0.359     -0.251      0.693
N             0.2714      0.117      2.326     0.020      0.043      0.500
Wealth        0.4449      0.125      3.559     0.000      0.200      0.690
=====

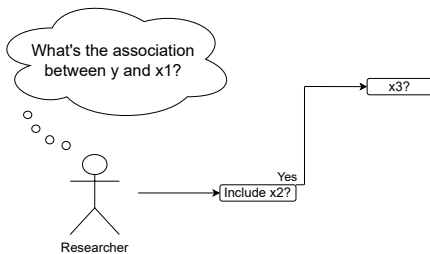
```

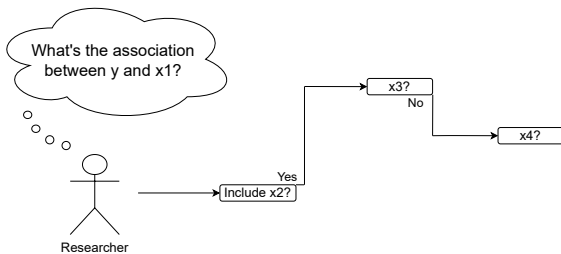
```

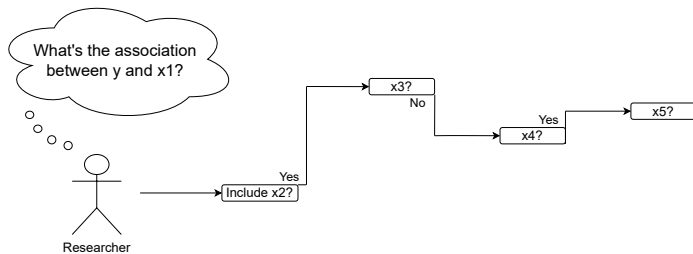
=====
Omnibus:          11.181    Durbin-Watson:       1.794
Prob(Omnibus):    0.004    Jarque-Bera (JB):    10.897
Skew:             1.061    Prob(JB):            0.00430
Kurtosis:         4.031    Cond. No.            5.08e+04
=====

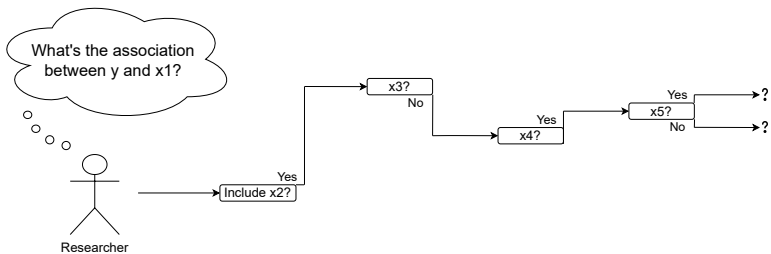
```

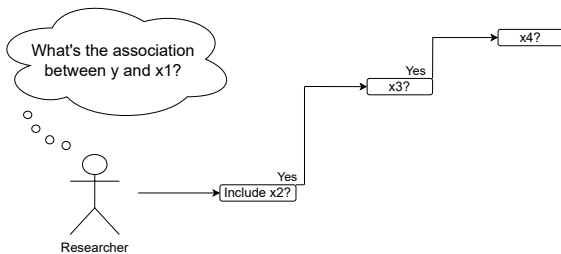


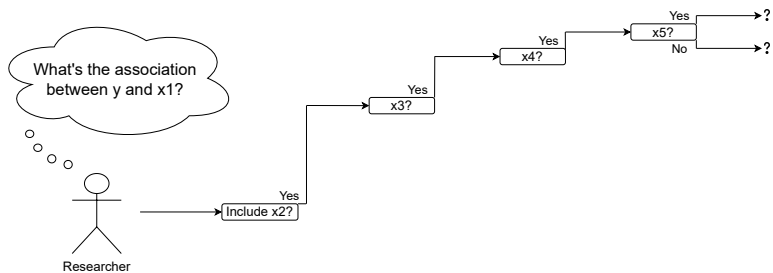


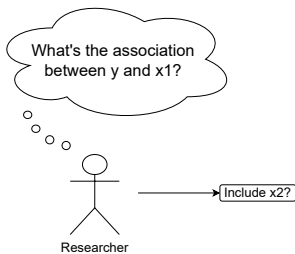


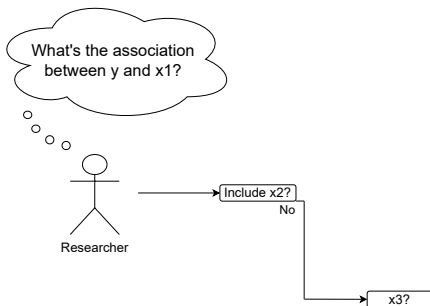


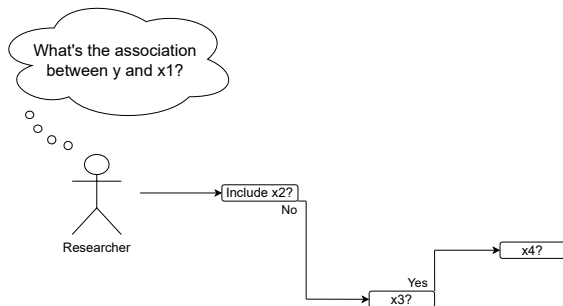


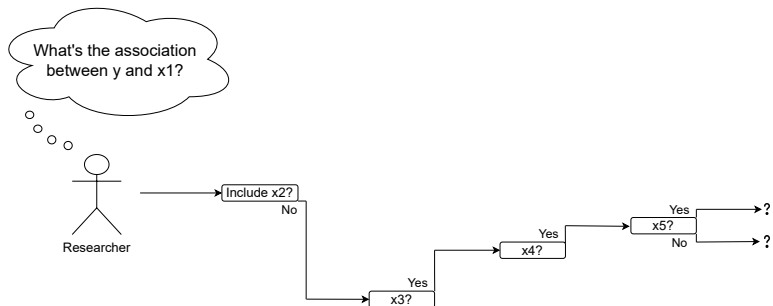


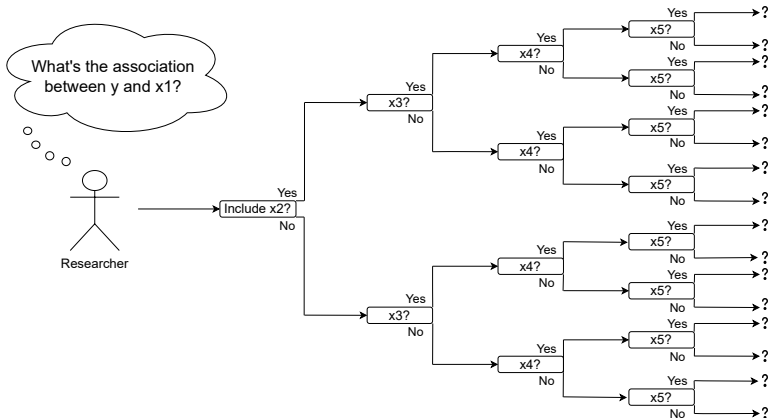












**This type of problem has enormous implications for p-hacking,
HARKing, and the validity of the scientific record.**

Introducing the ‘Feasible Model Space’

- ◇ From this simple choice of four auxiliary control variables, we have 16 possible models.
 - ▶ In general, this is 2^{n+1} .
 - ▶ The ‘+1’ comes from models with or without a constant.
 - ▶ With a choice of eight possible controls: $2^9 = 512$ models.
 - ▶ This can quickly get large.
- ◇ In an age of high performance computing: no problem!
- ◇ The main challenge: systematically reporting all these results.

Introducing the ‘Feasible Model Space’

- ◇ What’s critical here is that we consider only a ‘feasible model space’.
- ◇ We should **not** be including in this space noisy variables.
- ◇ This is where the role Social ‘Theory’ becomes critical.
- ◇ Systematic inclusion/omission allows holistic tests of theory.
- ◇ This does not prohibit us from having *one* preferred model.
 - ▶ This could be the one that is classically built in isolation.
 - ▶ But, now we can compare it with all others.

nature human behaviour

View all journals

Q Search

Log in

Explore content ▾

About the journal ▾

Publish with us ▾

[nature](#) > [nature human behaviour](#) > [resources](#) > article

Resource | Published: 27 July 2020

Specification curve analysis

[Uri Simonsohn](#) , [Joseph P. Simmons](#) & [Leif D. Nelson](#)[Nature Human Behaviour](#) **4**, 1208–1214 (2020) | [Cite this article](#)**8225** Accesses | **237** Citations | **68** Altmetric | [Metrics](#) A [Publisher Correction](#) to this article was published on 09 October 2020 This article has been [updated](#)

Sociological Methods & Research

Impact Factor: **6.5** / 5-Year Impact Factor: **7.1**Available access | Research article | First published online October 23, 2015 | [Request permissions](#)

Model Uncertainty and Robustness: A Computational Framework for Multimodel Analysis

[Cristobal Young](#) and [Katherine Holsteen](#) [View all authors and affiliations](#)[Volume 46, Issue 1](#) | <https://doi.org/10.1177/0049124115610347>

Contents



PDF/EPUB



Cite



Share options



Information, rights and permissions



Metrics

Abstract

Model uncertainty is pervasive in social science. A key question is how robust empirical results are to sensible changes in model specification. We present a new approach and applied statistical software for computational multimodel analysis. Our approach proceeds in two steps: First, we estimate the modeling distribution of estimates across all combinations of possible controls as well as specified functional form issues, variable definitions, standard error calculations, and estimation commands. This allows analysts to present their core, preferred estimate in the context of a distribution of plausible estimates. Second, we develop a model influence analysis showing how each model ingredient affects the coefficient of interest. This shows which model assumptions, if any, are critical to obtaining an empirical result. We demonstrate the architecture and interpretation of multimodel analysis using data on the union wage premium, gender dynamics in mortgage lending, and tax flight migration among U.S. states. These illustrate how initial results can be strongly robust to alternative model specifications or remarkably dependent on a knife-edge specification.

Limitations of Existing Tools

This is great!

- ◇ Not all researchers are, nor need to be, expert programmers.

However, these existing tools are limited.

- ◇ They don't include any real indication of model fit.
- ◇ They simply visualise distributions of the estimand.
- ◇ There is no out-of-sample validity.
- ◇ They place a uniform likelihood over all models.

We set ourselves the task:

Can we create a reliable tool to conduct this/adjacent analysis?



[README](#)
[Code of conduct](#)
[Contributing](#)
[GPL-3.0 license](#)



RobustiPy

Purpose [Research](#)
 Python [3.11](#)
 License [GNU3.0](#)
 DOI [10.5281/zenodo.15700698](#)

Welcome to the home of **RobustiPy**, a Python library for the creation of a more robust and stable model space. RobustiPy does a large number of things, included but not limited to: high dimensional visualisation, Bayesian Model Averaging, bootstrapped resampling, (in- and)out-of-sample model evaluation, model selection via Information Criterion, explainable AI (via [SHAP](#)), and joint inference tests (as per [Simonsohn et al. 2019](#)).

Full documentation is available on [Read the Docs](#). The first release is indexed onto Zenodo [here](#).

RobustiPy performs Multiversal/Specification Curve Analysis which attempts to compute most or all reasonable specifications of a statistical model, understanding a specification as a single attempt to create an estimand of interest, whether through a particular choice of covariates, hyperparameters, data cleaning decisions, and so forth.

Introducing RobustiPy: An efficient next generation multiversal library with model selection, averaging, resampling, and explainable AI

Daniel Valdenegro Ibarra^{1,3}, Jiani Yan^{1,2,3,4}, Duiyi Dai^{1,3}, and Charles Rahal^{1,3,5}

¹Leverhulme Centre for Demographic Science, University of Oxford

²Department of Sociology, University of Oxford

³ESRC Centre for Care, University of Oxford

⁴Wolfson College, University of Oxford

⁵Nuffield College, University of Oxford

October 8, 2025

Abstract

Scientific inference is often undermined by the vast but rarely explored “multiverse” of defensible modelling choices, which can generate results as variable as the phenomena under study. We introduce **RobustiPy**, an open-source Python library that systematizes multiverse analysis and model-uncertainty quantification at scale. **RobustiPy** unifies bootstrap-based inference, combinatorial specification search, model selection and averaging, joint-inference routines, and explainable AI methods within a modular, reproducible framework. Beyond exhaustive specification curves, it supports rigorous out-of-sample validation and quantifies the marginal contribution of each covariate. We demonstrate its utility across five simulation designs and ten empirical case studies spanning economics, sociology, psychology, and medicine—including a re-analysis of widely cited findings with documented discrepancies. Benchmarking on ~672 million simulated regressions shows that **RobustiPy** delivers state-of-the-art computational efficiency while expanding transparency in empirical research. By standardizing and accelerating robustness analysis, **RobustiPy** transforms how researchers interrogate sensitivity across the analytical multiverse, offering a practical foundation for more reproducible and interpretable computational science.

Keywords: *Model Uncertainty, Statistical Software, Open Science*

RobustiPy: The formal problem

Consider the key association between variables Y and X , where a set of covariates $\mathbf{Z}$ can influence the relationship between the former as follows:

$$Y = F(X, \mathbf{Z}) + \varepsilon. \quad (2)$$

Let's now observe that:

- ◇ Usually Y and X are imprecisely defined latent variables.
- ◇ Likewise, the set of covariates $\mathbf{Z}$ can also be composed of imprecisely defined variables from which the number of them can be unknown and/or non-finite.
- ◇ Finally, $F()$ is an unknown data generating function.

RobustiPy: The formal problem

Let's define the set of *reasonable* operationalisations of Y , X , Z and $F()$ as $\overleftrightarrow{Y}$, $\overleftrightarrow{X}$, $\overleftrightarrow{Z}$ and $\overleftrightarrow{F()}$. Then, we have:

$$\overleftrightarrow{Y}_{k_Y} = \overleftrightarrow{F}_{k_f}(\overleftrightarrow{X}_{k_X}, \overleftrightarrow{Z}_{k_Z}) + \varepsilon. \quad (3)$$

- ◇ Eq. 3 corresponds to a single possible *specification* of the Π set of all possible specifications.
- ◇ The total number of specifications can then be calculated as 2^N where N is $n_{\overleftrightarrow{Y}} + n_{\overleftrightarrow{X}} + n_{\overleftrightarrow{Z}} + n_{\overleftrightarrow{F()}}$.
- ◇ For example, an analysis with two functional forms, two target variables, two predictors and five covariates will yield a specification space Π of 2^{10} or 1024 possible specifications.

RobustiPy: The formal problem

We can also split the computation based on the operationalised variables in Eq. 3 by creating a ‘reasonable specification space’ for each: $\Pi_{\overleftrightarrow{Y}}$, $\Pi_{\overleftrightarrow{F()}}$, $\Pi_{\overleftrightarrow{X}}$, $\Pi_{\overleftrightarrow{Z}}$. The whole specification space can be obtained again as follows:

$$\Pi = \Pi_{\overleftrightarrow{Y}} \times \Pi_{\overleftrightarrow{F()}} \times \Pi_{\overleftrightarrow{X}} \times \Pi_{\overleftrightarrow{Z}} \quad (4)$$

- ◇ Any random and independently selected sample π from Π would lead to a reasonable approximation of $Y = F(X, Z)$.
- ◇ The main problem with current research practice is that the sample of Π reported by the researchers is not random.

The goal of RobustiPy is to automate this process to make it as easy as possible!

Functionality and Examples

- ◇ RobustiPy undertakes five novel types of common research design.
- ◇ It comes with:
 - ▶ Five simulated examples
 - ▶ Ten empirical examples
 - ▶ Full time profiling
- ◇ It gives you back:
 - ▶ Static but comprehensive visualisations
 - ▶ Vast summary printouts
 - ▶ Output results for custom analysis

Vanilla Computation

- ◇ Baseline case: single outcome Y , single predictor X , and varying controls Z .
- ◇ Data generating process:

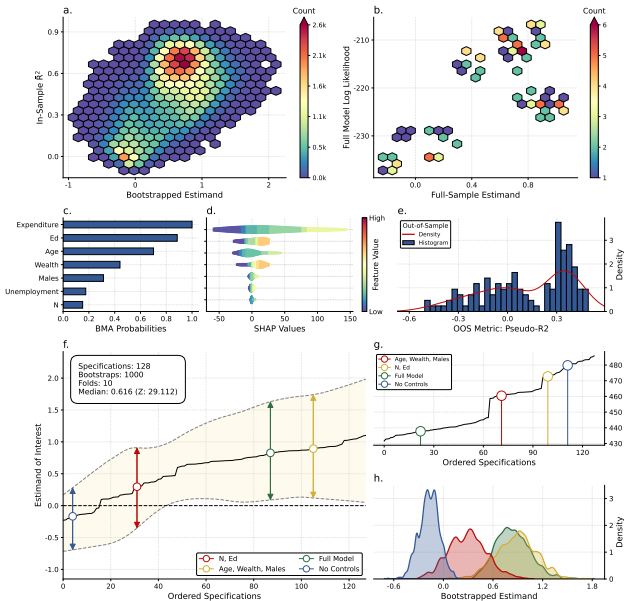
$$Y = \alpha + \beta X + \gamma Z + \varepsilon$$

- ◇ Specification space reduces to $\Pi_{\overleftarrow{Z}}$ only.
- ◇ Estimation: OLS with k -fold CV and bootstrap resampling.
- ◇ Outputs: distribution of $\hat{\beta}$ across $2^{|Z|}$ models, with null vs full model.

```

=====
1. Model Summary
=====
Model: OLS Robust
Inference Tests: Yes
Dependent variable: R
Independent variable: Inequality
Number of possible controls: 7
...
=====
2. Model Robustness Metrics
=====
2.1 Inference Metrics
=====
Median  $\beta$  (specs, no resampling): 0.6976 (null-calibrated p: 0)
Min Adj-R2: -0.0310, Specs: ['Age', 'Unemployment']
...
Share  $\beta < 0$  & significant (specs, no resampling): 0.0000
Share  $\beta < 0$  & significant (bootstraps  $\times$  specs): 0.0163 | null-calibrated p: 0.001998
Stouffer's Z = 29.1123 (two-sided p = 2.51e-186)
=====
2.2 In-Sample Metrics (Full Sample)
=====
Min AIC: 507.5432, Specs: ['Unemployment', 'Expenditure', 'N', 'Wealth', 'Males', 'Age', 'Ed']
Max Adj-R2: 0.6922, Specs: ['Expenditure', 'Wealth', 'Males', 'Age', 'Ed']
Min Adj-R2: -0.0310, Specs: ['Age', 'Unemployment']
=====
2.3 Out-Of-Sample Metrics (pseudo-r2 averaged across folds)
=====
Max Average: 0.4925, Specs: ['Age', 'Unemployment', 'Expenditure', 'Ed']
Min Average: -0.5057, Specs: ['Age', 'Unemployment', 'Ed', 'Males']
Mean Average: 0.12
Median Average: 0.2026

```



Importantly, note the opportunity to highlight individual specifications!

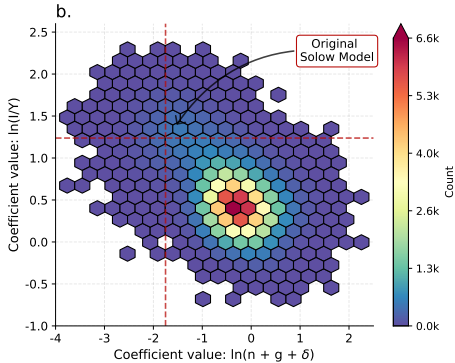
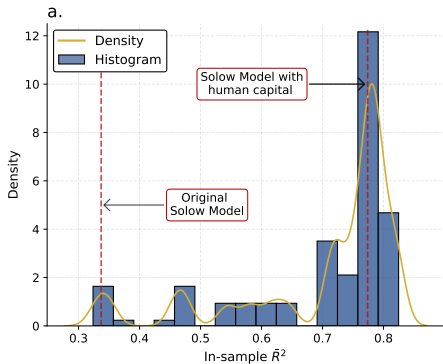
This allows us to consider where classical models live in the permissible model space.

Array of Never-Changing Variables

- ◇ Some predictors are fixed (always included in $\overleftrightarrow{X}$).
- ◇ Reduces computational cost by shrinking the specification space.
- ◇ Empirical motivation: theoretically indispensable covariates.
- ◇ Specification space:

$$\Pi = \Pi_{\overleftrightarrow{Z}} \quad \text{with fixed } X_f$$

- ◇ Results: interpretable estimates with reduced variance and runtime.



Fixed Effects OLS

- ◇ Designed for panel/longitudinal data with grouping variable g .
- ◇ Within-transformation removes group-specific heterogeneity:

$$Y_{it} - \bar{Y}_i = \beta(X_{it} - \bar{X}_i) + \varepsilon_{it}$$

- ◇ Implemented by demeaning variables by g .
- ◇ $\Pi_{\overleftrightarrow{Z}}$ still varied, but estimation is within-groups.
- ◇ Useful for causal inference in repeated-measures data.

Binary Dependents

- ◇ Logistic regression estimator for binary $Y \in \{0, 1\}$.
- ◇ Model:

$$\Pr(Y = 1 \mid X, Z) = \frac{1}{1 + \exp(-(\beta X + \gamma Z))}$$

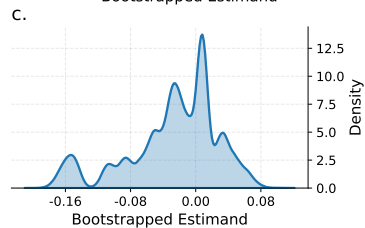
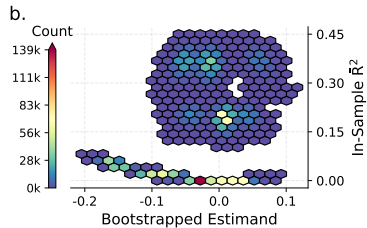
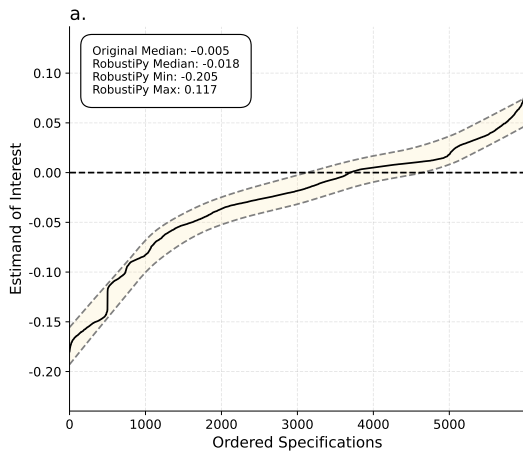
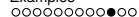
- ◇ $\Pi_{\overleftarrow{Z}}$ explored as before, with log-likelihood fit.
- ◇ Supports odds ratio interpretation of $\hat{\beta}$.
- ◇ Alternative: linear probability model with OLSRobust.

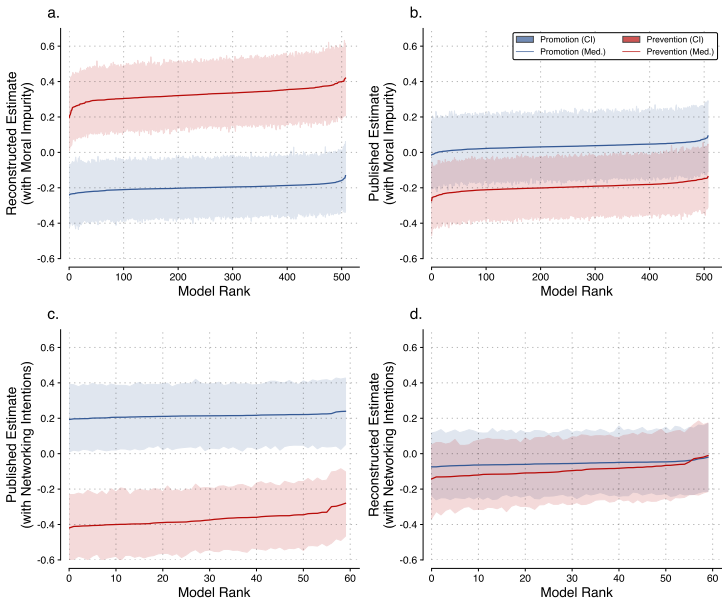
Multiple Dependents

- ◇ Allows multiple operationalisations of Y .
- ◇ Construct composites by averaging z-scored outcomes:

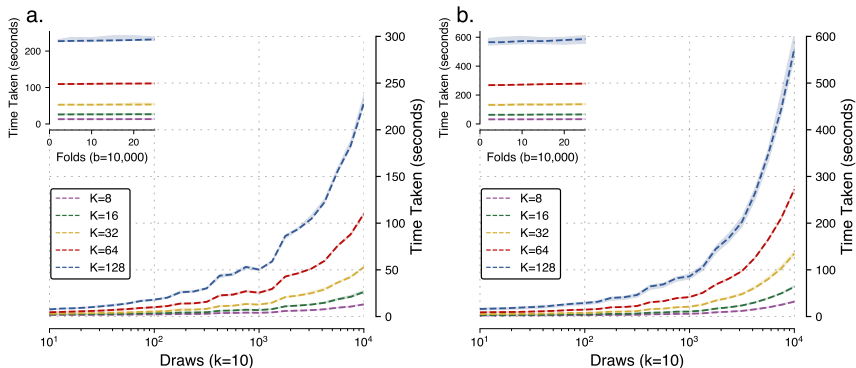
$$Y_{i,S}^{\text{comp}} = \frac{1}{|S|} \sum_{j \in S} \tilde{Y}_i^{(j)}$$

- ◇ Specification space enlarges: $\Pi_{\overleftarrow{Y}} \times \Pi_{\overleftarrow{Z}}$.
- ◇ Provides robustness to measurement uncertainty in the outcome.
- ◇ Outputs: specification curves across all dependent definitions/composites.





Time Profiling



Conclusion

- ◇ Model uncertainty: measurement/functional form/covariate choices.
 - ▶ A feasible specification space makes assumptions explicit.
- ◇ RobustiPy automates exploration of Π across $\overleftrightarrow{Y}, \overleftrightarrow{X}, \overleftrightarrow{Z}, \overleftrightarrow{F()}$.
 - ▶ CV, bootstraps, spec curves, averaging, selection, etc.
- ◇ Spotlight preferred models or undertake joint inference.
- ◇ Workflow encourages transparent, theory-led reporting.
 - ▶ It reduces room for p-hacking or HARKing.
- ◇ Open-source and ready to use: github.com/RobustiPy.
 - ▶ Try the notebooks and tell us what to add next!
 - ▶ We're specifically looking for help with more estimators!

Sincere thanks to my **wonderful** co-authors!

