Machine Learning in International Trade Research - Evaluating the Impact of Trade Agreements

Holger Breinlich
University of Surrey

Valentina Corradi
University of Surrey

Nadia Rocha
World Bank

Michele Ruta
World Bank

João Santos Silva
University of Surrey

Thomas Zylkin
University of Richmond

EIIT - UC San Diego

October 16th, 2022

Question - which PTA provisions matter for trade?

- Preferential trade agreements contain a diverse array of provisions beyond zeroing out tariffs
  - Competition policy, patent protection, financial regulations, environmental and labor standards, so much more
  - NAFTA, EU, MERCOSUR, ECOWAS all very different agreements with very different sets of provisions

Question - which PTA provisions matter for trade?

- Preferential trade agreements contain a diverse array of provisions beyond zeroing out tariffs
  - Competition policy, patent protection, financial regulations, environmental and labor standards, so much more
  - NAFTA, EU, MERCOSUR, ECOWAS all very different agreements with very different sets of provisions
- Ideally, we would like to use information on content to project likely effects of future and recent agreements
  - UK-Japan agreement: just signed in September 2020
  - UK-EU post-Brexit agreement: how important is the “level playing field” that was pushed for by EU?
  - UK-US agreement: ???

Question - which PTA provisions matter for trade?

- Preferential trade agreements contain a diverse array of provisions beyond zeroing out tariffs
  - Competition policy, patent protection, financial regulations, environmental and labor standards, so much more
  - NAFTA, EU, MERCOSUR, ECOWAS all very different agreements with very different sets of provisions
- Ideally, we would like to use information on content to project likely effects of future and recent agreements
  - UK-Japan agreement: just signed in September 2020
  - UK-EU post-Brexit agreement: how important is the “level playing field” that was pushed for by EU?
  - UK-US agreement: ???
- **Methodological challenges**: measuring which provisions matter for trade (and how much) is not a straightforward problem, for two reasons:
  (i) Estimation challenges associated with modeling trade data (zeroes, “multilateral resistance”, heteroskedasticity)
  (ii) Large number of provisions, high correlation $\implies$ “overfitting” problems

This paper

Methodology in a nutshell

We combine "lasso" methods with structural gravity in order to learn key provisions in PTAs, reduce overfitting in predicted PTA effects.

Key concepts:

- "*variable selection*": choosing the most relevant subset of a large number of variables
- "*lasso*": penalized regression technique that reduces overfitting and performs selection by shrinking coefficients toward zero
- "*overfitting*": estimates mainly reflect noise in the data, leading to unreliable estimates and predictions
- "*bootstrap aggregation*" (*or* "*bagging*"): using the average of results based on resampled data in order to reduce overfitting

This paper

Methodology in a nutshell

We combine "lasso" methods with structural gravity in order to learn key provisions in PTAs, reduce overfitting in predicted PTA effects.

Contributions

- Extend computational approach of Correia, Guimaraes and Zylkin (2020) for PPML-lasso with high-dimensional fixed effects
  - R package: penppml
- Adapt "plugin" lasso of Belloni, Chernozhukov, Hansen, and Kozbur (2016)
  - allows for panel data, clustering, heteroskedasticity
  - very strict; good for prediction but *under*-selects
- New methods for variable selection based on BCHK plugin lasso
  - "*bootstrap lasso*": bootstrap and aggregate ("bag") the selected variables
  - "*iceberg lasso*": regress selected variables on unselected variables

This paper

Methodology in a nutshell

We combine "lasso" methods with state-of-the-art "gravity models" used in trade in order to learn key provisions in PTAs, reduce overfitting in predicted PTA effects.

Contributions

- PPML-lasso with high-dimensional fixed effects
- BCHK plugin lasso: good for prediction but under-selects
- new methods for variable selection: bootstrap lasso and iceberg lasso

Findings

- Plugin lasso method finds PTA effects on trade well approximated by a simple model that depends on only 7 out of 305 provision variables
- The selected provisions create more predictability in areas of anti-dumping, technical barriers to trade, competition policy, and trade facilitation
- Bootstrap lasso and iceberg lasso paint a more nuanced picture but support same general conclusions

This paper

Other findings

We do a further application where we compute heterogeneous estimates for individual PTAs based on their provision contents:

- Plugin lasso, bootstrap lasso produce reasonable estimates.
- Other methods (PPML, cross-validation, iceberg lasso) seem to overfit.

Simulation results

Simulations show new methods (iceberg lasso and bootstrap lasso) outperform traditional cross-validation lasso for both variable selection and prediction.

Related Literature

Modeling effects of FTAs using provisions data

- "Depth" measures based on counts of provisions: Kohl, Brakman, and Garretsen (2016), Mattoo, Mulabdic and Ruta (2017), Falvey & Foster-McGregor (2018)
- "Breadth" measures based on min. coverage of each core area: Falvey & Foster-McGregor (2022)
- Focus on specific provisions:
  - Dhingra, Freeman, and Mavroedi (2018) combine services, investment, & competition into a single variable
  - Prusa, Teh, and Zhu (2022) show PTAs with anti-dumping rules reduce intra-PTA anti-dumping filings
- Using machine learning: Regmi and Baier (2021) use unsupervised learning (textual analysis + clustering) to categorize PTAs into 4-5 clusters

Related Literature

Modeling effects of FTAs using provisions data

Kohl, Brakman, and Garretsen (2016), Mattoo, Mulabdic and Ruta (2017), Falvey & Foster-McGregor (2022), Dhingra, Freeman, and Mavroedi (2018), Prusa, Teh, and Zhu (2022), Regmi and Baier (2021)

Variable selection using Lasso-based methods

Tibsharini (1996), Zhao and Yu (2006), Hastie, Tibshirani, and Friedman (2009), Belloni, Chernozhukov, and Hansen (2014), Belloni, Chernozhukov, Hansen, and Kozbur (2016)

Three-way gravity models for empirical trade policy analysis

Baier and Bergstrand (2007), Weidner and Zylkin (2021), Yotov, Larch, Monteiro, and Piermartini (2016), Baier, Yotov, and Zylkin (2019)

Question - which PTA provisions matter for trade?

- Free trade agreements contain a diverse array of provisions beyond zeroing out tariffs
  - Competition policy, patent protection, financial regulations, environmental and labor standards, so much more
  - NAFTA, EU, MERCOSUR, ECOWAS all very different agreements with very different sets of provisions

- Ideally, we would like to use information on content to project likely effects of future and recent agreements
  - UK-Japan agreement: just signed in September
  - UK-EU post-Brexit agreement: how important is the "level playing field" pushed for by EU?
  - UK-US agreement: ???

- **Methodological challenges**: measuring which provisions matter for trade (and how much) is not a straightforward problem, for two reasons:
  (i) Estimation challenges associated with modeling trade data (zeroes, multilateral resistance, etc)
  (i) Large number of provisions to consider creates an "overfitting" problem

Challenge #1: Modeling trade flows using three-way gravity

Standard "three-way gravity" model for estimating PTA effects:

$$y_{ijt} = \exp\left(\alpha_{it} + \gamma_{jt} + \eta_{ij} + \beta \mathbf{PTA}_{ijt}\right)\omega_{ijt}. \tag{1}$$

- $\mathbf{PTA}_{ijt}$: a set of (time-varying) dummies for the presence of a bilateral trade agreement.
- δ_{it} and ψ_{jt}: *exporter-time* and *importer-time* FEs to account for country-specific & GE factors
- η_{ij}: *time-invariant* bilateral FE to absorb ex ante trade frictions
- PPML leads to consistent estimates (Santos Silva & Tenreyro, 2006; Weidner & Zylkin, 2021)

Baseline objective: estimate β, the "average partial effect" of signing a PTA.

Challenge #1: Modeling trade flows

exporter ("i")	importer ("j")	year ("t")	trade ("y_{ijt}")	FTA?
AUS	JPN	2002	$13.5bn	0
AUS	USA	2002	5.9 bn	0
JPN	AUS	2002	11.6 bn	**0**
JPN	USA	2002	123.3bn	0
USA	AUS	2002	11.2 bn	**0**
USA	JPN	2002	43.0 bn	0
AUS	JPN	2007	34.5 bn	0
AUS	USA	2007	8.5 bn	**1**
JPN	AUS	2007	16.6 bn	0
JPN	USA	2007	132.4 bn	0
USA	AUS	2007	19.3 bn	**1**
USA	JPN	2007	59.0 bn	0

Three-way gravity model for estimating PTA effects:

$$y_{ijt} = \exp(\alpha_{it} + \gamma_{jt} + \eta_{ij} + \beta FTA_{ijt})\omega_{ijt}$$

This example:
"Did the 2005 U.S-Aus. FTA increase trade?", using 3 countries and 2 years

(real data set has 200+ countries, 14 years)

Challenge #1: Modeling trade flows

exporter ("i")	importer ("j")	year ("t")	trade ("y_{ijt}")	FTA?
AUS	JPN	2002	$13.5bn	0
AUS	USA	2002	5.9 bn	0
JPN	AUS	2002	11.6 bn	**0**
JPN	USA	2002	123.3bn	0
USA	AUS	2002	11.2 bn	**0**
USA	JPN	2002	43.0 bn	0
AUS	JPN	2007	34.5 bn	0
AUS	USA	2007	8.5 bn	**1**
JPN	AUS	2007	16.6 bn	0
JPN	USA	2007	132.4 bn	0
USA	AUS	2007	19.3 bn	**1**
USA	JPN	2007	59.0 bn	0

Three-way gravity model for estimating PTA effects:

$$y_{ijt} = \exp(\alpha_{it} + \gamma_{jt} + \eta_{ij} + \beta FTA_{ijt})\omega_{ijt}$$

To underline the sources of complexity

- Nonlinearity
- Three-way high-dimensional fixed effects

Nonetheless, recent computational advances have made this model simple to estimate w/ PPML.

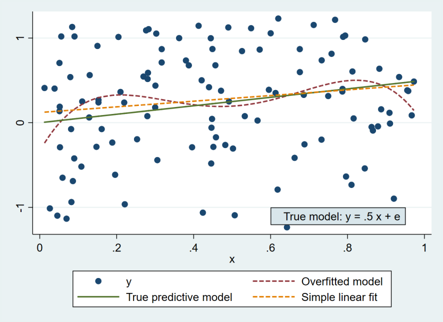
Challenge #1: Modeling trade flows

However, because we work with highly detailed data on the underlying provisions included in FTAs, our data set instead looks like this:

exporter ("i")	importer ("j")	year ("t")	trade ("y_{ijt}")	prov. 1?	prov. 2?	prov. 3?	prov. 4?	prov. 5?	prov. 6?	...	prov. 305?
AUS	JPN	2002	$13.5bn	0	0	0	0	0	0	...	0
AUS	USA	2002	5.9 bn	0	0	0	0	0	0	...	0
JPN	AUS	2002	11.6 bn	0	0	0	0	0	0	...	0
JPN	USA	2002	123.3bn	0	0	0	0	0	0	...	0
USA	AUS	2002	11.2 bn	0	0	0	0	0	0	...	0
USA	JPN	2002	43.0 bn	0	0	0	0	0	0	...	0
AUS	JPN	2007	34.5 bn	0	0	0	0	0	0	...	0
AUS	USA	2007	8.5 bn	1	0	1	0	1	0	...	1
JPN	AUS	2007	16.6 bn	0	0	0	0	0	0	...	0
JPN	USA	2007	13.2 bn	0	0	0	0	0	0	...	0
USA	AUS	2007	19.3 bn	1	0	1	0	1	0	...	1
USA	JPN	2007	59.0 bn	0	0	0	0	0	0	...	0
...	...	...	...	...	...	...	...	...	...	...	...

Thus, on top of the estimation challenges that are unique to trade data, we also need to be concerned about multicollinearity and overfitting.

Challenge #2: The overfitting problem



“Overfitting”

When you add more predictors, you get better at "explaining" the data, but you may only be getting better at explaining the random noise in the data.

You may actually be getting **worse** at explaining what’s really going on.

- True model: $y = 0.5x +$ random noise
- Simple linear fit: x is the only predictor
- Overfitted model: “quartic fit” that adds x^2, x^3, and x^4 as predictors

Data

Trade data

from UN COMTRADE, 1964-2016 (every 4 years), 196,978 observations

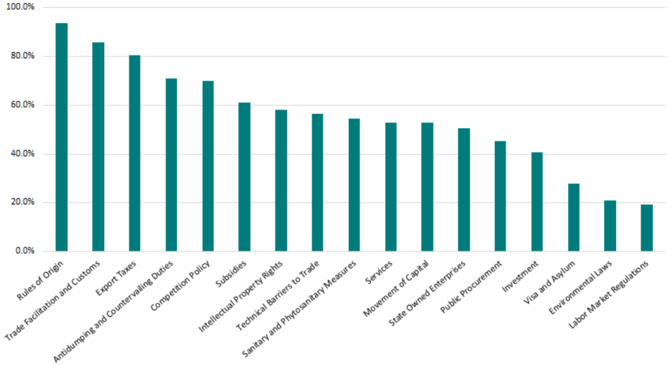
FTA provisions

from Handbook of Deep Trade Ageements (Mattoo, Rocha and Ruta 2020), using 305 “essential” provisions only.

Policy area	**No. of provisions**	**No. of Essential Provisions**
Anti-dumping and Countervailing Duties	53	31
Competition Policy	35	14
Environmental Laws	48	27
Export taxes	46	23
Intellectual Property Rights	120	67
Investment	57	15
Labor Market Regulations	18	12
Movement of Capital	94	8
Public Procurement	100	5
Rules of Origin	38	19
Sanitary and Phytosanitary	59	24
Services	64	21
State-Owned Enterprises	53	13
Subsidies	36	13
Technical Barriers to Trade	34	19
Trade Facilitation and Customs	52	11
Visa and Asylum	30	3
Total	**937**	**305**

Data

Figure 1: Share of PTAs that cover selected policy areas



Note: Figure shows the share of PTAs that cover a policy area.
Source: Mattoo, Rocha and Ruta (2020).

Some caveats:

- One thing we don't have data on: tariff preferences
- No data on most agreements that are no longer in effect; we drop these observations.

Data

Table: Coverage of essential provisions by policy area

	Share of agreements covering:		
Policy Area	0 to 25%	25% to 75%	over 75%
Anti-dumping and Countervailing Duties	99%	1%	0%
Competition Policy	48%	47%	5%
Environmental Laws	88%	12%	0%
Export Taxes	41%	59%	0%
Intellectual Property Rights	76%	23%	1%
Investment	6%	64%	30%
Labor Market Regulations	68%	17%	15%
Movement of Capital	44%	42%	13%
Public Procurement	53%	40%	7%
Rules of Origin	7%	93%	0%
Sanitary and Phytosanitary Measures	87%	13%	0%
Services	6%	62%	33%
State-Owned Enterprises	45%	54%	1%
Subsidies	59%	41%	0%
Technical Barriers to Trade	93%	7%	0%
Trade Facilitation and Customs	21%	78%	0%
Visa and Asylum	27%	70%	3%

Note: Coverage ratio refers to the share of essential provisions for a policy area contained in a given agreement relative to the maximum number of essential provisions in that policy area.

Method: PPML Lasso w/ 3-way HDFEs

Obtain coefficients for each provision ($\beta \equiv \beta_1 ... \beta_p$) using

$$\min \underbrace{\sum_{i,j,t} -y_{ijt} \ln \mu_{ijt} + \sum_{i,j,t} \mu_{ijt}}_{\text{standard PPML loss term}} + \underbrace{\sum_{k=1}^{p} \widehat{\phi}_k \lambda \, |\beta_k|}_{\text{Lasso penalty term}}$$

where

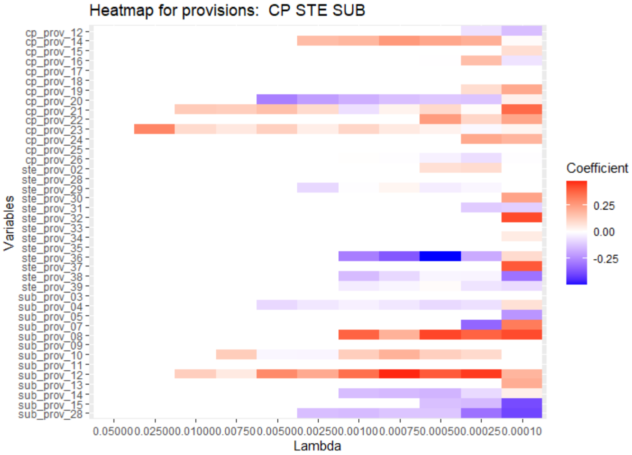
$$\mu_{ijt} = e^{x_{ijt}\beta + \alpha_{it} + \gamma_{jt} + \eta_{ij}}$$

Intuition:

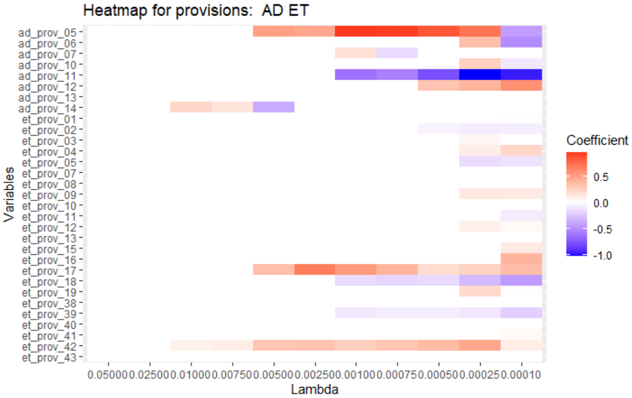
- when λ is large, there is a larger penalty for having a non-zero β-coefficient causing coefficients for many provisions to be zeroed out.
- As we make λ smaller, penalty becomes less strict and more variables are "selected".

details on computation

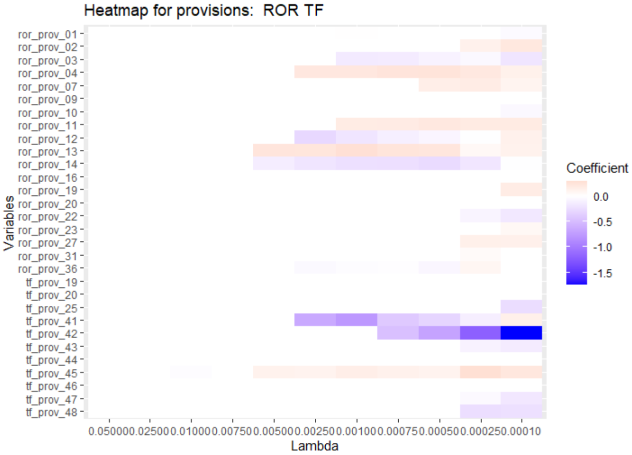
Regularization path (1/7): Comp Policy, State Aid, Subsidies only



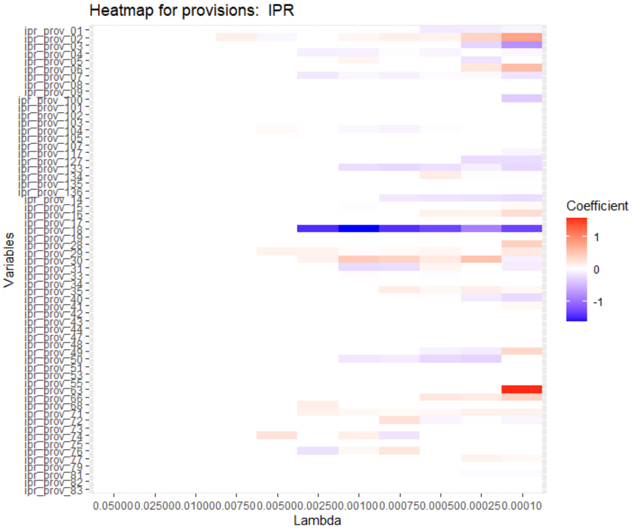
Regularization path (2/7): AD and export tax provisions only



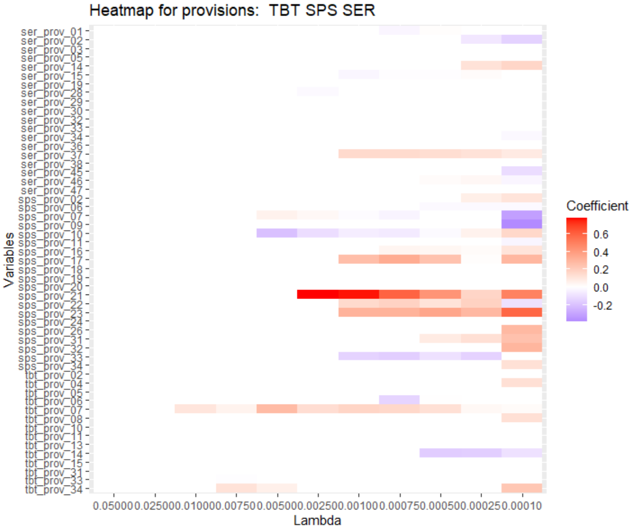
Regularization path (3/7): Rules of Origin, Customs processing only



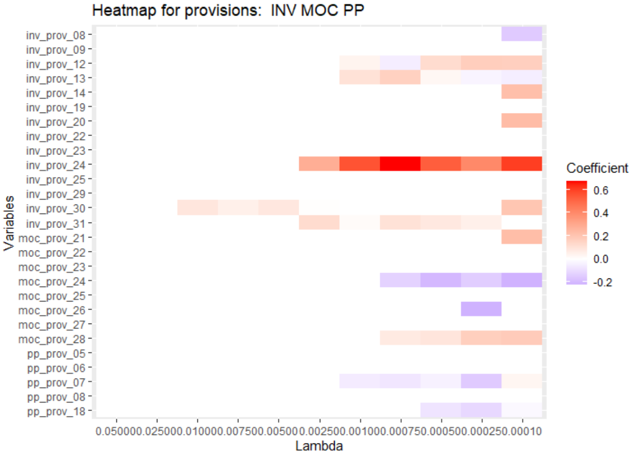
Regularization path (4/7) (IPR)



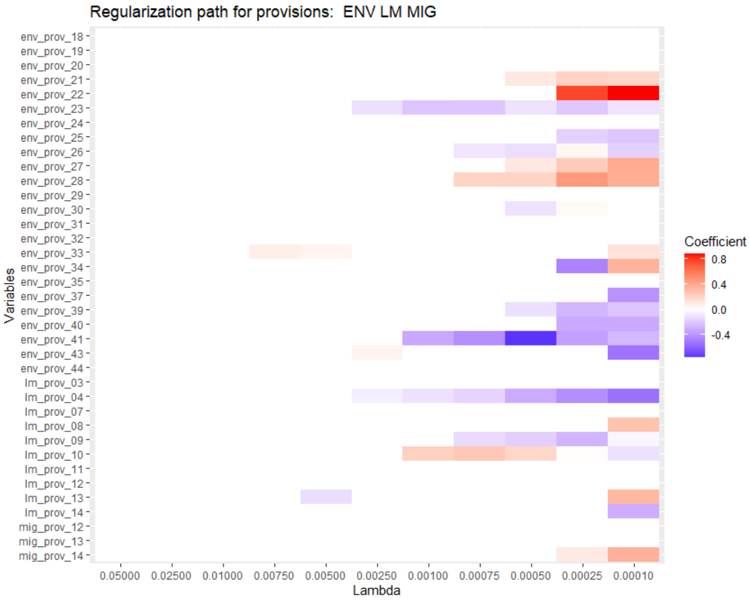
Regularization path (5/7) (TBT, SPS, Services)



Regularization path (6/7) (Inv., Capital, Public Procurement)



Regularization path (7/7) (Env., Labor, Migration)



Implementing the Lasso

How to choose the "right" values for penalty terms λ, ϕ_k??

$$\min \sum_{i,j,t} -y_{ijt} \ln \mu_{ijt} + \sum_{i,j,t} \mu_{ijt} + \sum_{k=1}^{p} \widehat{\phi}_k \lambda \left|\beta_k\right|$$

Implementing the Lasso

How to choose the "right" values for penalty terms λ, ϕ_k??

$$\min \sum_{i,j,t} -y_{ijt} \ln \mu_{ijt} + \sum_{i,j,t} \mu_{ijt} + \sum_{k=1}^{p} \widehat{\phi}_k \lambda \left|\beta_k\right|$$

1. Cross-validation
   - Provision-specific penalty weights ϕ_k set to 1
   - Drop agreements and try to predict their effects out-of-sample
   - Choose λ that minimizes prediction error (turns out to be $\lambda = 0.0025$)
   - Known to be too lenient (errs on side of selecting too many features)

Implementing the Lasso

How to choose the "right" values for penalty terms λ, ϕ_k??

$$\min \sum_{i,j,t} -y_{ijt} \ln \mu_{ijt} + \sum_{i,j,t} \mu_{ijt} + \sum_{k=1}^{p} \widehat{\phi}_k \lambda \left|\beta_k\right|$$

1. Cross-validation
   - Provision-specific penalty weights ϕ_k set to 1
   - Drop agreements and try to predict their effects out-of-sample
   - Choose λ that minimizes prediction error (turns out to be $\lambda = 0.0025$)
   - Known to be too lenient (errs on side of selecting too many features)

2. Theory-driven "plug-in" method (Belloni, Chernozhukov, Hansen, and Kozbur 2016)
   - Stricter than CV (selects fewer variables)
   - λ chosen so only variables with a statistically large effect on model fit are selected.
   - Heteroskedasticity and error-clustering increase likelihood that variables could be mistakenly selected; ϕ_k-weights constructed to address this
   - Despite parsimony, turns out to be superior to CV for prediction!

more details

Implementing the Lasso

Extensions of the plug-in lasso

These extensions build on plug-in lasso while relaxing some of its strictness in order to increase likelihood of selecting correct causal provisions.

3. Two-step "iceberg" lasso
    - Regress selected provisions on all other provisions in a second step using another set of plug-in lasso regressions
    - Idea is initial plug-in lasso may only give us the "tip of the iceberg" when there is high collinearity
    - Over-selects by design but outperforms CV in simulations

Implementing the Lasso

Extensions of the plug-in lasso

These extensions build on plug-in lasso while relaxing some of its strictness in order to increase likelihood of selecting correct causal provisions.

3. Two-step "iceberg" lasso
4. Bootstrap lasso
   - Repeatedly run the plug-in lasso using bootstrap resampling
   - Record provisions selected in more than 5% of bootstrap trials as being "selected"
   - Again, idea is to correct for over-strictness of the original plug-in method while still leveraging its ability to mimic the DGP
   - Takes advantage of "bootstrap aggregation" ("bagging") principle from machine learning

For sampling: treat pairs that join the same agreements as being in the same cluster; treat pairs as clusters otherwise. Re-sample by cluster. We use $B = 250$ bootstraps.

Caveats

- By construction, not all of the provisions selected by bootstrap lasso or iceberg lasso can be said to have causal effects.
- Conversely, plugin lasso under-selects by design, leaving out relevant variables
  - OVB by construction
  - obviously complicates interpretation of coefficient estimates
- In general, we need to be very humble about potential causal interpretations of our results
  - requires taking the three-way gravity model to be an appropriate representation of the determinants of trade.

Simulations: overview

Model for simulations:

$$y = \exp\left(1 + \beta x_1 + z + \sigma\varepsilon\right)$$

- x_1 only true causal variable
- z unpenalized regressor whose coefficient is not penalized (stands in for fixed effects)
- x_1, z, ε are independent $N(0,1)$ draws
- $\beta = 0.2$, $\sigma = 0.3$

Remaining variables $x_2, \ldots x_p$ are introduced to create a **selection problem**:

- The first κ variables $x_1, \ldots x_\kappa$ are equi-correlated with correlation ρ
- set $n = 250, 1000, 4000$; $p = 5\left\lceil\sqrt{n}\right\rceil$ (corresponds to 80,160, or 320)
- To complicate selection, vary $\kappa \in \{5, 10, 20\}$, $\rho \in \{0.75, 0.90, 0.99\}$.

Simulations: The Selection Problem

Table: Percentage of times correct regressor is selected

		$\rho = 0.75$			$\rho = 0.90$			$\rho = 0.99$		
n		$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$
250	CV Lasso	100.0	99.7	99.3	96.6	91.8	85.5	52.2	37.7	23.4
	Adaptive Lasso	99.7	99.4	97.9	93.9	87.4	80.4	45.3	29.4	17.7
	Plug-in Lasso	91.6	89.9	88.1	80.6	72.1	63.7	41.1	26.8	16.9
	Bootstrap Lasso	100.0	**100.0**	**99.8**	96.6	**98.4**	**96.7**	**90.4**	**79.2**	**64.2**
	Iceberg Lasso	95.7	95.9	95.2	95.9	**95.8**	**93.0**	**95.3**	**93.4**	**80.1**
1000	CV Lasso	100.0	100.0	100.0	100.0	100.0	99.9	81.0	69.8	56.4
	Adaptive Lasso	100.0	100.0	100.0	100.0	99.7	99.7	68.3	54.8	40.8
	Plug-in Lasso	99.8	99.8	99.7	99.2	98.4	98.4	71.4	55.0	41.4
	Bootstrap Lasso	100.0	100.0	100.0	100.0	100.0	**100.0**	**98.0**	**93.7**	**87.1**
	Iceberg Lasso	100.0	100.0	100.0	100.0	100.0	**100.0**	**100.0**	**100.0**	**98.8**
4000	CV Lasso	100.0	100.0	100.0	100.0	100.0	100.0	99.0	97.8	94.9
	Adaptive Lasso	100.0	100.0	100.0	100.0	100.0	100.0	91.9	86.0	79.1
	Plug-in Lasso	100.0	100.0	100.0	100.0	100.0	100.0	98.0	93.9	88.1
	Bootstrap Lasso	100.0	100.0	100.0	100.0	100.0	100.0	**100.0**	**99.9**	**99.8**
	Iceberg Lasso	100.0	100.0	100.0	100.0	100.0	100.0	**100.0**	**100.0**	**100.0**

Simulations: The Selection Problem

Table: Avg. number of regressors selected

		$\rho = 0.75$			$\rho = 0.90$			$\rho = 0.99$		
n		$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$
250	CV Lasso	8.65	8.55	8.74	8.87	8.66	8.64	8.52	8.22	7.93
	Adaptive Lasso	7.22	7.21	7.05	7.34	7.21	7.05	6.99	6.72	6.26
	Plug-in Lasso	1.26	1.52	1.89	1.45	1.73	2.06	1.23	1.33	1.41
	Bootstrap Lasso	11.11	12.81	15.27	11.31	13.25	15.66	11.27	12.77	14.03
	Iceberg Lasso	4.80	9.14	15.97	4.81	9.43	17.00	4.78	9.32	15.65
1000	CV Lasso	9.43	9.59	10.05	9.76	10.10	10.69	9.92	10.11	10.51
	Adaptive Lasso	3.93	4.19	4.49	4.71	5.22	5.85	5.37	5.97	6.22
	Plug-in Lasso	1.31	1.54	1.88	1.63	2.02	2.57	1.75	2.02	2.34
	Bootstrap Lasso	8.88	10.89	13.91	9.26	11.67	15.23	9.36	11.85	14.81
	Iceberg Lasso	5.01	10.00	19.22	5.00	10.01	19.69	5.01	10.01	19.72
4000	CV Lasso	10.46	10.85	11.24	10.78	11.28	11.88	11.18	12.06	12.63
	Adaptive Lasso	1.00	1.00	1.00	1.03	1.03	1.03	1.18	1.30	1.70
	Plug-in Lasso	1.23	1.43	1.68	1.53	1.96	2.42	2.00	2.60	3.18
	Bootstrap Lasso	7.86	9.91	13.03	8.44	11.04	14.94	8.93	11.94	16.27
	Iceberg Lasso	5.00	10.00	19.99	5.00	10.00	20.00	5.01	10.00	20.00

Simulations: The Prediction Problem

Table: MSE for out-of-sample predictions

		$\rho = 0.75$			$\rho = 0.90$			$\rho = 0.99$		
n		$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$	$k = 5$	$k = 10$	$k = 20$
250	CV Lasso	6.85	6.83	6.86	6.87	6.88	6.88	6.83	6.83	6.80
	Adaptive Lasso	7.27	7.23	7.22	7.29	7.26	7.24	7.17	7.18	7.08
	Plug-in Lasso	**6.57**	**6.53**	**6.66**	**6.59**	6.63	6.71	**6.53**	**6.52**	**6.52**
	Bootstrap Lasso	6.66	6.60	**6.66**	6.64	**6.62**	**6.66**	6.57	6.53	6.53
	Iceberg Lasso	6.71	6.83	7.21	6.71	6.84	7.25	6.72	6.85	7.23
	All regressors	10.98	10.98	10.98	10.98	10.98	10.98	10.98	10.98	10.98
	Oracle	6.39	6.39	6.39	6.39	6.39	6.39	6.39	6.39	6.39
1000	CV Lasso	6.34	6.35	6.35	6.34	6.34	6.35	6.33	6.32	6.34
	Adaptive Lasso	6.34	6.31	6.30	6.35	6.39	6.40	6.39	6.41	6.47
	Plug-in Lasso	**6.19**	6.19	6.22	**6.18**	6.19	6.22	**6.16**	6.17	6.20
	Bootstrap Lasso	**6.19**	**6.18**	**6.21**	**6.18**	**6.18**	**6.21**	**6.16**	**6.16**	**6.18**
	Iceberg Lasso	6.22	6.31	6.48	6.22	6.31	6.47	6.22	6.31	6.48
	All regressors	8.44	8.44	8.44	8.44	8.44	8.44	8.44	8.44	8.44
	Oracle	6.19	6.19	6.19	6.19	6.19	6.19	6.19	6.19	6.19
4000	CV Lasso	6.37	6.37	6.37	6.36	6.37	6.38	6.37	6.38	6.38
	Adaptive Lasso	**6.34**	**6.34**	**6.34**	**6.33**	**6.33**	**6.34**	6.34	**6.34**	**6.34**
	Plug-in Lasso	**6.34**	6.35	6.36	6.34	6.35	6.35	**6.33**	**6.34**	6.35
	Bootstrap Lasso	6.35	6.36	6.37	6.36	6.37	6.37	6.36	6.37	6.38
	Iceberg Lasso	**6.34**	6.35	6.43	6.34	6.35	6.43	6.34	6.35	6.43
	All regressors	7.39	7.39	7.39	7.39	7.39	7.39	7.39	7.39	7.39
	Oracle	6.34	6.34	6.34	6.34	6.34	6.34	6.34	6.34	6.34

Simulations: Takeaways

1. Traditional CV-based lasso not reliable for either selection or prediction in finite samples
2. Plugin lasso performs well at minimizing RMSE of predictions, under-selects by design.
3. Iceberg lasso and bootstrap lasso over-select by design, but more likely than CV-lasso to select correct regressors.
4. Bootstrap lasso performs best at prediction in moderate samples; relative performance improves with more regressors.

Plug-in Lasso Results

	Dep. variable: Bilateral Trade Flows (1964-2016, every 4 years)			
	PPML	Lasso	PPML Post-lasso	PPML
	(1)	(2)	(3)	(4)
FTA	0.131			
	(0.044)***			
AD14. Anti-dumping – Material Injury				
CP23. Competition Policy – Transparency / Coordination				
TBT provisions:				
TBT2 / TBT29. Mutual Recognition†				
TBT7. Technical Reg's: use International Standards				
TBT8. Conformity Assessment: Mutual Recognition				
TBT33. Standards: use Regional Standards				
Trade Facilitation:				
TF45. Issuance of Proof of Origin				

Gravity estimates are obtained using Poisson Pseudo-maximum Likelihood with exporter-time, importer-time, and exporter-importer FEs. The number of observations is 316,317. Columns labelled "PPML post-lasso" report PPML coefficients for all variables selected by a plug-in lasso method in a prior step. All other columns report further experiments using PPML. PPML cluster-robust standard errors, reported in parentheses, are clustered so that pairs belonging to the same agreement are treated as belonging to the same cluster. * $p < 0.10$, ** $p < .05$, *** $p < .01$. †TBT2 is perfectly collinear with TBT29. TBT2 refers to mutual recognition of technical regulations, whereas TBT29 refers to mutual recognition of standards.

Plug-in Lasso Results

	Dep. variable: Bilateral Trade Flows (1964-2016, every 4 years)			
	PPML	Lasso	PPML Post-lasso	PPML
	(1)	(2)	(3)	(4)
FTA	0.131			-0.008
	(0.044)***			(0.062)
AD14. Anti-dumping – Material Injury		0.329	0.349	0.347
			(0.117)***	0.119)***
CP23. Competition Policy – Transparency / Coordination		0.002	0.118	0.118
			(0.077)	(0.078)
TBT provisions:				
TBT2 / TBT29. Mutual Recognition†		0.142	0.184	0.182
			(0.142)	(0.144)
TBT7. Technical Reg's: use International Standards		0.016	0.032	0.034
			(0.078)	(0.080)
TBT8. Conformity Assessment: Mutual Recognition		0.028	0.123	0.124
			(0.099)	0.099
TBT33. Standards: use Regional Standards		0.109	0.113	0.116
			(0.061)*	(0.064)*
Trade Facilitation:				
TF45. Issuance of Proof of Origin		0.000	0.089	0.095
			(0.032)***	(0.053)*

Gravity estimates are obtained using Poisson Pseudo-maximum Likelihood with exporter-time, importer-time, and exporter-importer FEs. The number of observations is 316,317. Columns labelled "PPML post-lasso" report PPML coefficients for all variables selected by a plug-in lasso method in a prior step. All other columns report further experiments using PPML. PPML cluster-robust standard errors, reported in parentheses, are clustered so that pairs belonging to the same agreement are treated as belonging to the same cluster. * $p < 0.10$, ** $p < .05$, *** $p < .01$. †TBT2 is perfectly collinear with TBT29. TBT2 refers to mutual recognition of technical regulations, whereas TBT29 refers to mutual recognition of standards.

"Iceberg Lasso" Results

"Iceberg Lasso": perform a further lasso of each selected provision on every non-selected provision to see if we may only be getting the "tip of the iceberg"

AD14	**CP23**	**TBT02/29**	**TBT07**	**TBT33**	**TF45**
AD06 (**0.98**)	AD06 (0.40)	AD06 (-0.07)	AD06 (0.51)	AD11 (-0.05)	AD06 (0.16)
AD08 (**0.98**)	AD08 (0.40)	AD08 (-0.07)	AD08 (0.51)	ENV44 (-0.02)	AD08 (0.16)
ENV42 (**0.98**)	CP22 (**0.80**)	CP14 (0.61)	ENV42 (0.51)	MOC26 (-0.10)	AD11 (0.08)
	CP24 (**0.89**)	CP21 (0.77)	ENV44 (0.08)	PP08 (0.05)	CP15 (0.71)
	ENV41 (-0.06)	CP22 (**0.80**)	SPS21 (0.16)	SUB07 (0.07)	ENV19 (0.40)
	ENV42 (0.40)	ENV22 (-0.01)	SUB07 (0.10)	TBT05 (0.61)	ENV27 (0.50)
	PP08 (0.05)	ENV42 (-0.07)	TBT15 (0.68)	TBT06 (**0.98**)	ENV42 (0.16)
	SPS24 (-0.05)	ENV44 (-0.01)	TBT34 (**0.93**)	TBT15 (0.69)	MOC26 (0.16)
	STE31 (0.54)	SPS11 (-0.00)		TBT32 (0.61)	STE37 (0.06)
	TBT10 (-0.01)	STE32 (0.66)		TBT34 (0.53)	SUB07 (0.03)
	TF42 (0.65)	SUB09 (0.78)		TF42 (0.64)	SUB10 (0.28)
	TF43 (-0.04)	SUB10 (**0.90**)			TF44 (**0.98**)
	TF44 (0.38)	TF42 (**0.98**)			

Raw correlations shown in parentheses.

Bootstrap Lasso results

Table: Bootstrap Lasso results: largest average coefficients and selection frequencies

Provisions with largest average coefficients		Provisions selected most frequently	
AD14	0.079	AD14	0.372
CP23	0.065	CP23	0.320
CP22	0.063	TBT07	0.308
AD05	0.055	SPS06	0.228
TBT07	0.054	TBT08	0.208
TBT02	0.048	SUB12	0.184
TBT08	0.038	TBT02	0.168
SUB12	0.030	TBT33	0.160
TBT34	0.029	CP22	0.156
SPS06	0.028	TBT34	0.152
TF42	0.027	TBT06	0.148
TBT33	0.023	AD05	0.140
TF41	0.023	CP21	0.124
TBT06	0.021	TF45	0.116
CP21	0.020	ENV33	0.116

Uses cluster-bootstrap resampling with 250 replications.

Bootstrap Lasso results

Table: Bootstrap Lasso results: Summarizing results by Provision category

	Number of provisions selected more than 5% of the time	Number of provisions selected more than 1% of the time	Sum of average coefficients across categories
Anti-dumping	3	5	0.171
Competition Policy	3	5	0.151
Environment	1	5	0.017
Export Taxes	2	5	0.049
Investment	0	2	0.020
IPR	0	5	0.019
Labor Markets	0	0	0.000
Migration	1	1	0.012
Movement of Capital	1	2	0.023
Public Procurement	0	1	0.013
Rules of Origin	1	4	0.021
Services	0	1	0.004
SPS	1	10	0.062
State aid	2	2	0.011
Subsidies	5	7	0.076
TBTs	8	13	0.237
Trade Facilitation	2	5	0.064
Total	30	74	0.951

Categories in which provisions were most likely to be selected and the total of the average coefficients of each provision within each category.

Lasso results: Takeaways

- Plugin lasso gives us a very parsimonious model as expected
- Bootstrap lasso and iceberg lasso don't always select exact same provisions, but results broadly similar
- Trade-promoting effects concentrated in TBTs, anti-dumping, competition policy, trade facilitation, & subsidies
  - bootstrap lasso ranking of trade-promoting categories comports with intuition
- Bootstrap lasso results suggest we should not place too much confidence in the selection of any one provision.

Application: Heterogeneous PTA Effects

As a simple application, we use our methods to estimate heterogeneity in the effects of PTAs on trade based on provision variables.

This is a setting where overfitting is a known problem:

- Kohl (2014), Baier, Yotov, and Zylkin (2019)
- With roughly the same number of provisions and agreements, unpenalized estimates of individual PTA effects likely to reflect significant noise

Table: Summarizing Estimates of Heterogeneous PTA Effects

	PPML	CV	Plug-in	Iceberg	Bootstrap
Min	-81.2%	-50.4%	0.0%	-62.8%	0.0%
Max	>1e6%	387.0%	144.4%	284.9%	101.0%
Mean	328774.6%	32.1%	13.8%	17.2%	12.5%
Median	26.4%	14.4%	9.3%	6.7%	7.2%
Stdev.	300514.7%	63.0%	20.7%	42.4%	15.3%
Correlations					
PPML	1	0.146	-0.054	0.233	0.041
CV	0.146	1	0.391	0.550	0.513
Plug-in	-0.054	0.391	1	0.507	0.925
Iceberg	0.233	0.550	0.507	1	0.679
Bootstrap	0.041	0.513	0.925	0.679	1
Estimated partial effects for selected PTAs					
EU	104.9%	105.4%	87.1%	101.6%	64.2%
EEA	80.4%	90.5%	9.3%	94.4%	18.3%
Eurasian Econ. Union	-21.8%	71.8%	144.4%	38.5%	101.0%
NAFTA	77.9%	77.5%	79.9%	81.5%	52.9%
MERCOSUR	145.5%	115.9%	42.1%	76.2%	39.6%
ECOWAS	469.6%	379.2%	9.3%	23.3%	19.4%
ASEAN	1.8%	-9.4%	0.0%	0.0%	3.3%

This table summarizes estimated partial effects for individual PTAs produced by the different methods we consider. The column labelled “PPML” refers to an unpenalized PPML regression with all 305 provision variables. The other columns refer to variants of the lasso discussed in Section 3.

Conclusion

Summary

- We combine Lasso with 3-way PPML estimator used in trade policy analysis, apply to rich data on FTA provisions
- Plug-in lasso isolates 7 provisions that promote more predictability in the areas of anti-dumping, competition policy, and technical barriers to trade.
- These provisions in turn tend to be entangled with other provisions whose role may be obscured by collinearity.
- Introduce bootstrap lasso and iceberg lasso as new methods for variable selection
- Plug-in, bootstrap methods show promise for prediction

Future work and extensions

- predicting effects of prospective agreements
- explore using bootstrap lasso to estimate prediction uncertainty
- complementarity / substitutability between provision configurations

More details on plugin lasso

Intuition

Variable $x_{ijt,k}$ is selected if the absolute value of the estimated score for β_k is "statistically large" when evaluated near $\beta_k = 0$.

Estimated score for $\widehat{\beta}_k$:

$$\underbrace{\frac{1}{n}\sum_{i,j,t}\left(y_{ijt} - \mu_{ijt}\right)\widetilde{x}_{ijt,k}}_{\text{FE-PPML score}} - \underbrace{\frac{1}{n}\sum_{i,j,t}\widehat{\phi}_k \lambda sign(\widehat{\beta}_k)}_{\text{from penalty term}}$$

$\widehat{\phi}_k$ is an estimate of the dispersion of the score:

$$\widehat{\phi}_k^2 = \frac{1}{n}\sum_{i,j}\left(\sum_t \widetilde{x}_{ijt,k}\widehat{\varepsilon}_{ijt}\right)^2.$$

Compute in the same way you would clustered standard errors.

λ is set so that the estimated score for $\widehat{\beta}_k$ must be large as compared to its standard deviation in order for $x_{ijt,k}$ to be selected.

back

Computation: HDFE-IRLS

Re-write the penalized minimization using weighted least squares

$$\min_{\beta}\left[\frac{1}{2n}\sum_{i,j,t}\mu_{ijt}\left(z_{ijt}-\alpha_{it}-\gamma_{jt}-\eta_{ij}-x'_{ijt}\beta\right)^{2}+\frac{1}{n}\sum_{k=1}^{p}\widehat{\phi}_{k}\lambda\left|\beta_{k}\right|\right]$$

where

$$z_{ijt}=\frac{y_{ijt}-\mu_{ijt}}{\mu_{ijt}}+\log\mu_{ijt}.$$

Convenient to further re-write by sweeping out the fixed effects

$$\min_{\beta}\left[\frac{1}{2n}\sum_{i,j,t}\mu_{ijt}\left(\widetilde{z}_{ijt}-\widetilde{x}'_{ijt}\beta\right)^{2}+\frac{1}{n}\sum_{k=1}^{p}\widehat{\phi}_{k}\lambda\left|\beta_{k}\right|\right]$$

where $\widetilde{z}_{ijt}$ and $\widetilde{x}_{ijt}$ are partialed-out versions of z_{ijt} and x_{ijt} using same approach as Correia, Guimaraes, and Zylkin ("ppmlhdfe" in Stata)

Iterate on β, μ, z until convergence. back