Machine learning methods for intergenerational elasticity estimation

Winter School on Inequality and Social Welfare Theory
Canazei - 2019

Paolo Brunori

University of Florence & University of Bari

Machine learning: an other "revolution" for empirical economics?

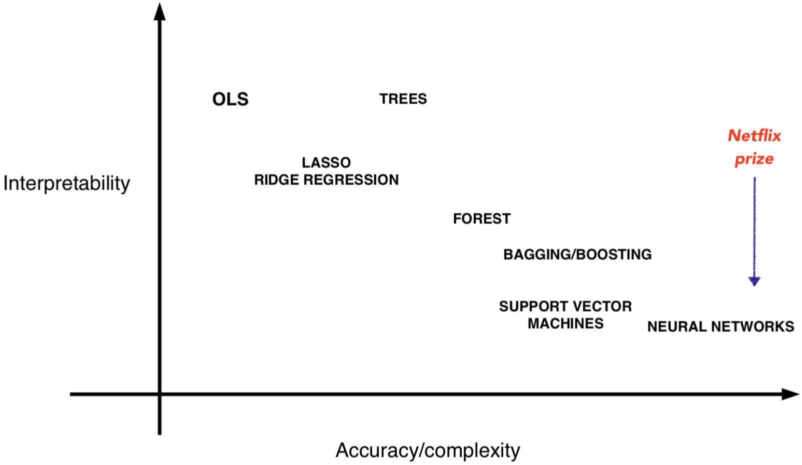
big data

data-driven

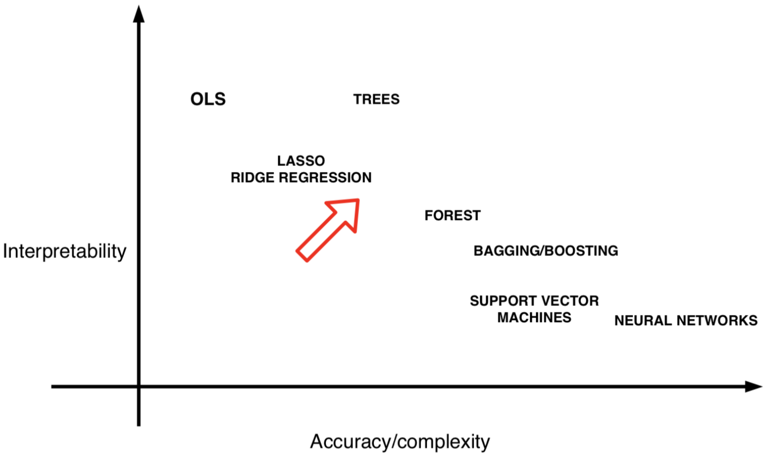
predictive performance

(above all) necessary condition: large N

Regularized regression



Regularized regression



A specific example: intergenerational elasticity of earnings

This lecture is based on the paper:
Estimating intergenerational income mobility on two samples: sensitivity to model selection, joint with Francesco Bloise and Patrizio Piraino

The paper uses machine learning to evaluate income elasticity in South Africa.

Intergenerational elasticity of earnings

$$y_i^c = \beta_0 + \beta y_i^p + \epsilon_i \quad (1)$$

y_i^c is the logarithm of the child's permanent income

y_i^p is the logarithm of the parent's permanent income

β is the intergenerational elasticity of income (IGE)

Two-Sample Two-Stage Least Squares (TSTSLS)

Björklund and Jäntti (1997) two samples

main sample: information on adult income and their parents' socio-economic characteristics

2,587 working male residents in South Africa (National Income Dynamics Study, 2008-2012).

auxiliary sample: earlier information about income and socio-economic characteristics

1,355 working males (Project for Statistics on Living Standards and Development, 1994).

TSTSLS: first step

$$y_i^{ps} = \gamma z_i^{ps} + \theta_i \tag{2}$$

where y_i^{ps} is the income of the pseudo-parents.

the vector $\hat{\gamma}$ is estimated minimizing the sum of squared residuals.

TSTSLS: second step

$\hat{y}_i^p$ is the predicted income of individual i in the main sample based on coefficients estimated in (2)

$$y_i^c = \beta_0 + \beta\left(\hat{\gamma} z_i^p\right) + \omega_{it} \tag{3}$$

where z_i^p are characteristics of the real fathers.

and $\hat{\beta}_{TSTSLS}$ is IGE.

TSTSLS: bias

$\hat{\beta}_{TSTSLS}$ is biased:

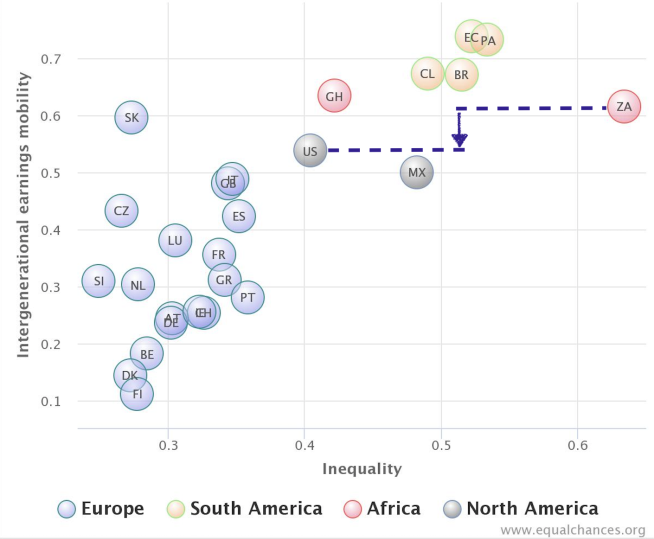
$$P\lim \beta_{TSTSLS} = \beta + \frac{cov(\hat{y}_i^p, y_i^c)}{R^2 var(y_i^p)} - \frac{cov(y_i^p, y_i^c)}{var(y_i^p)} \tag{4}$$

$\hat{\beta}_{TSTSLS}$ in South Africa

Intergenerational elasticity of income in South Africa

Let's have a look at the code...

Is it a large change?



Sensitivity to model specification

β_{TSTSLS} with a linear model is 0.62, adding interactions becomes 0.54.

$$P \lim \beta_{TSTSLS} = \beta + \left[\frac{cov(\hat{y}_i^p, y_i^c)}{R^2 var(y_i^p)} - \frac{cov(y_i^p, y_i^c)}{var(y_i^p)} \right] \quad (5)$$

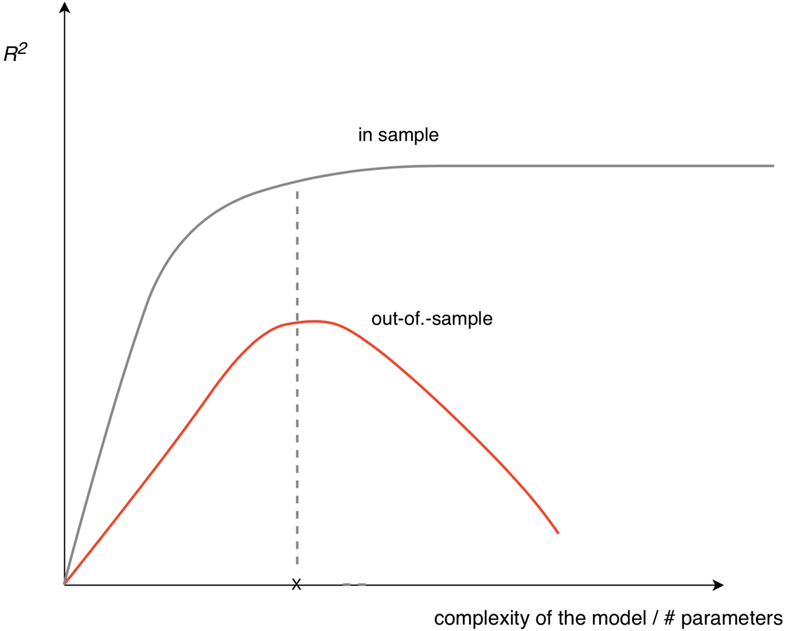
R^2 and $cov(\hat{y}_i^p, y_i^c)$ should go up (wrong assumption!).

Model selection

R^2 monotonically increase with the number of regressors <u>in sample</u>

but we are interested in maximizing out-of-sample R^2 to minimize the bias.

MSE out-of-sample



Model selection

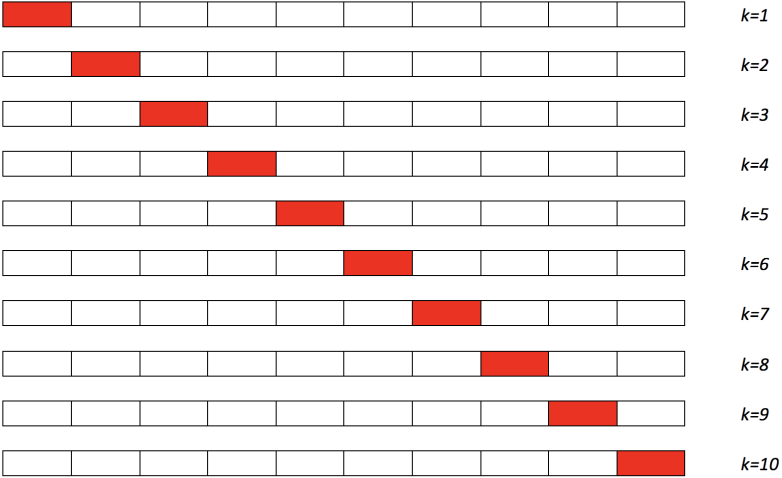
R^2 monotonically increase with the number of regressors
in sample

but we are interested in maximizing out-of-sample R^2!

Cross validation, equivalently, minimizes Mean Squared Error (MSE):

$$(1 - R^2) = n \frac{MSE}{\sum_{i=1}^{n} (y_i - \bar{y})^2}$$

k-fold cross validation



10-fold Cross Validation

Cross-validation in South Africa

We evaluate out-of-sample MSE for first stage regression in South Africa

Let's have a look at the code...

Model selection

First stage out-of-sample MSE is 1.01 for the model without interactions and is 0.98 including interactions.

There is not reason to stop here. Two options:

- estimate MSE for all possible models (feasible in this case)
- a more general and smoother approach: regularization of linear models

Regression regularization

- OLS search for the parameters that minimize MSE in sample
- shrinking methods search for parameters that minimize MSE out-of-sample
- general approach: penalize models with many parameters and models with large coefficients

Ridge regression

Ridge regression shrinks regression coefficients by imposing a penalty on their size:

$$\hat{\beta}_{RIDGE} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \tag{6}$$

Ridge regression

Ridge regression shrinks regression coefficients by imposing a penalty on their size:

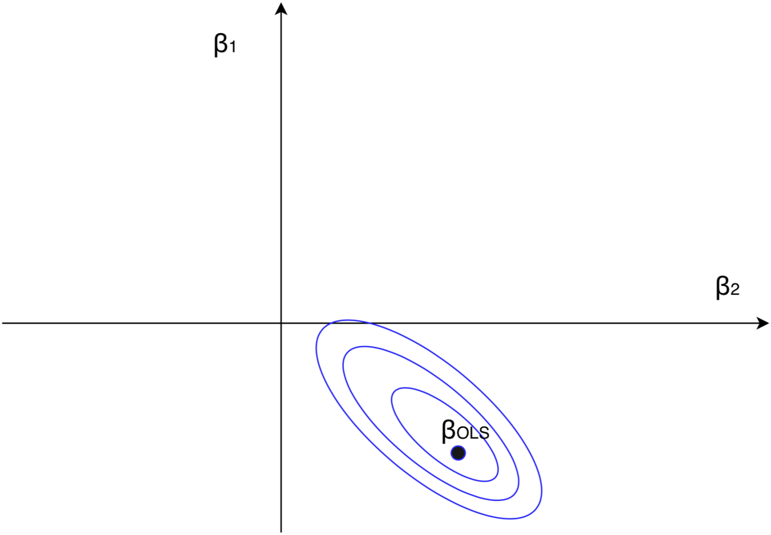
$$\hat{\beta}_{RIDGE} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} \beta_j^2 \right\} \tag{7}$$

This is equivalent to:

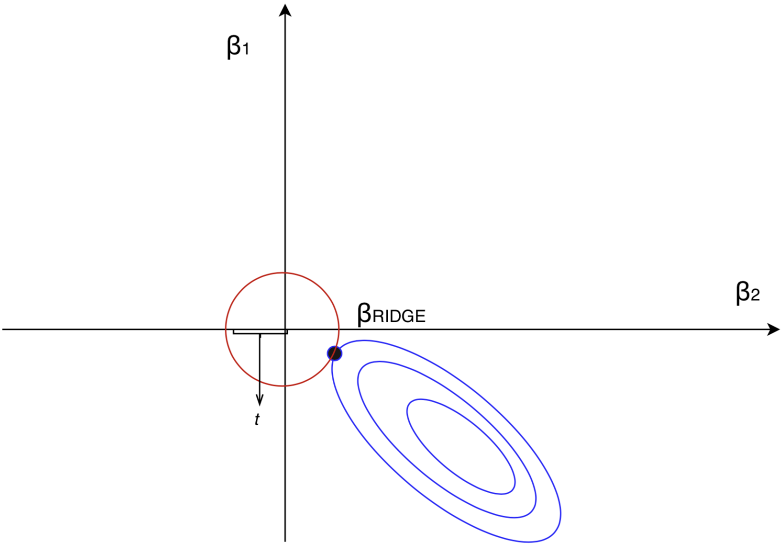
$$\hat{\beta}_{RIDGE} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j \right)^2 \right\}$$

$$\text{subject to } \sum_{j=1}^{p} \beta_j^2 \leq t$$

OLS



Ridge regression



Regression regression

- contrary to other parsimony criteria (BIC, AIC) λ is not predetermined

- ridge regression is *tuned* searching for λ that produces lowest out-of-sample MSE by cross-validation

Least absolute shrinkage and selection operator (Lasso)

Lasso performs both variables selection and shringag by imposing a penalty on their absolute size:

$$\hat{\beta}_{LASSO} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j}\beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\} \tag{8}$$

Lasso

Lasso shrinks regression coefficients by imposing a penalty on their absolute size:

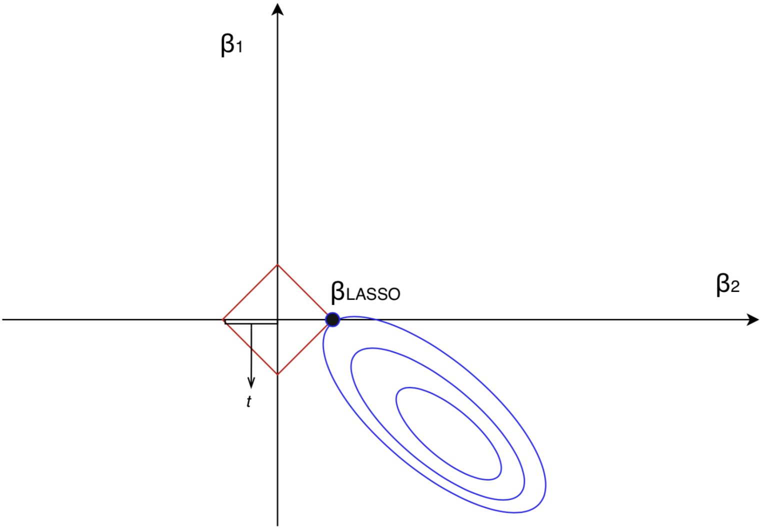
$$\hat{\beta}_{LASSO} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j} \beta_j \right)^2 + \lambda \sum_{j=1}^{p} |\beta_j| \right\} \tag{9}$$

This is equivalent to:

$$\hat{\beta}_{LASSO} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j} \beta_j \right)^2 \right\}$$

$$\text{subject to } \sum_{j=1}^{p} |\beta_j| \leq t$$

Lasso



Lasso

- Lasso is also *tuned* searching for λ that produces lowest out-of-sample MSE by cross-validation
- The non linearity of the constraint forces some coefficient to be exactly zero (a variables selection alogirthm)
- Zou and Hastie (2005) have proposed a to use a weighted average of the two methods: *elastic net*

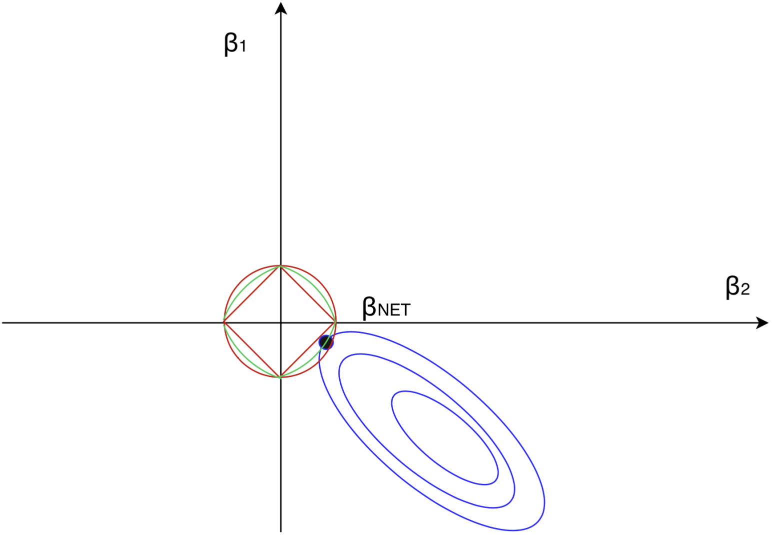
Elastic net

Elastic net is a weighted average of Lasso and ridge algorithm:

$$\hat{\beta}_{NET} = \underset{\beta}{\operatorname{argmin}} \left\{ \frac{1}{2} \sum_{i=1}^{n} \left(y_i - \beta_0 - \sum_{j=1}^{p} x_{i,j} \beta_j \right)^2 \right\} \tag{10}$$

$$\text{subject to}: (1-\alpha) \sum_{j=1}^{p} |\beta_j| + \alpha \sum_{j=1}^{p} \beta_j^2 \leq t$$

Elastic net



Elastic net

- Tuning the elastic net implies searching for the couple α and β that minimize MSE
- when $\alpha = 0$ we are back to ridge regression
- when $\alpha = 1$ we are using a Lasso

Regularization of the first stage regression for income in South Africa

- We estimate first stage regression of the model with interaction using elastic net

Let's have a look at the code...

β_{TSTSLS} sensitivity

- $\beta_{\text{base}} = 0.63$
- $\beta_{\text{full model}} = 0.54$
- $\beta_{\text{net}} = 0.58$

Are these large differences?

Will using this criterion affect similarly all estimates?

Can we use other ML algorithms to further reduce MSE?