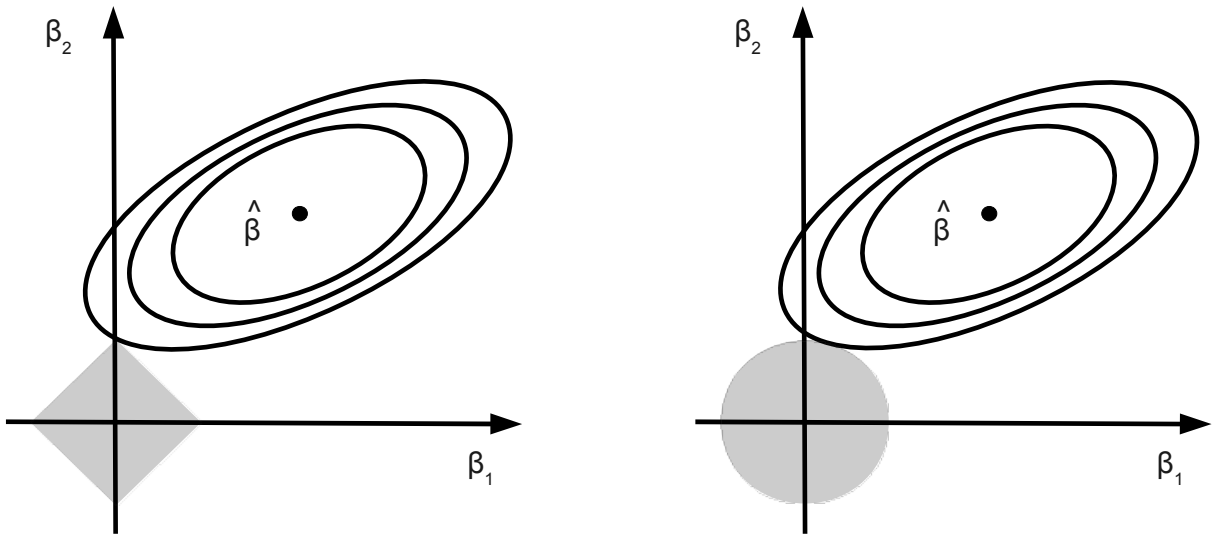
Recap

Lasso:

$$\hat{\beta}(\lambda) = \text{argmin}_{\beta}(\|Y - X\beta\|_2^2/n + \lambda\|\beta\|_1)$$

- sparse estimator



- convex optimization

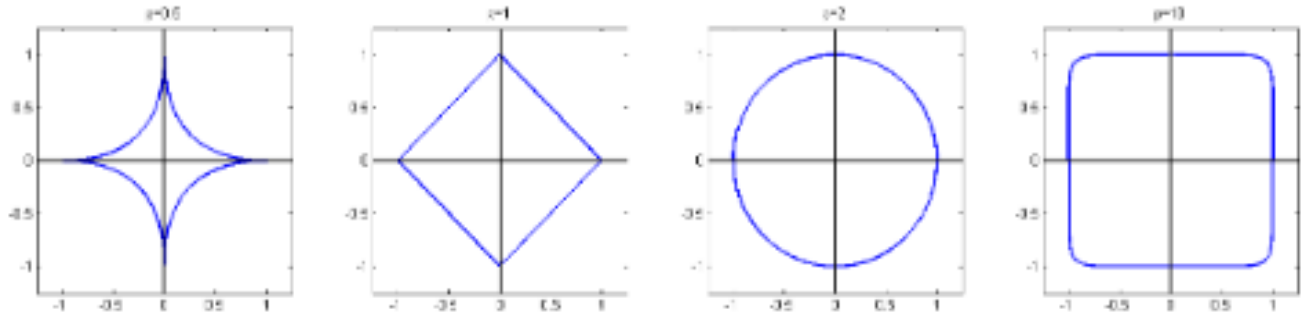


Figure 1: Unit circles for several Minkowski-p-norms $\|\mathbf{x}\|_p$: from left to right $p = 0.5$, $p = 1$ (Manhatten), $p = 2$ (Euclidean), $p = 10$.

Figure from Lange, Zühlke, Holz, Villmann (2014)

convex: ℓ_p-norm with $p \geq 1$
sparse: ℓ_p-norm with $p \leq 1$ (need "edges" in the ball)
$\Longrightarrow$ $p = 1$(*Lasso*) for sparse and convex estimator

Orthonormal design: explicit solution

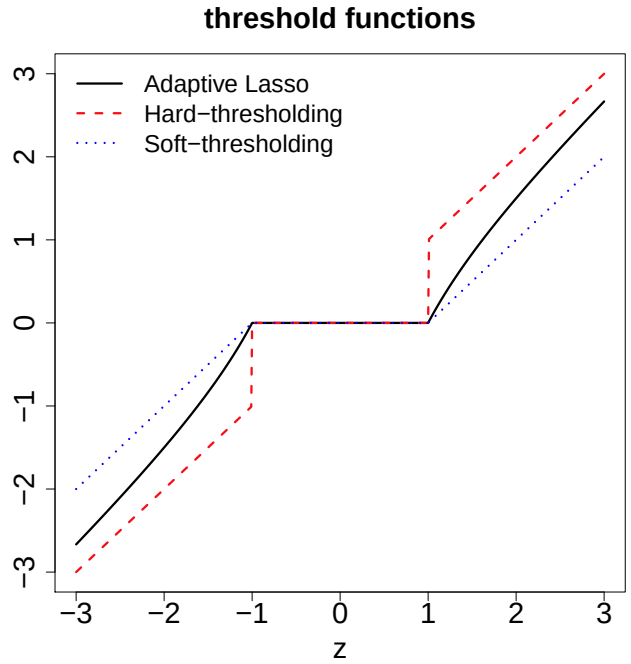
$X^T X/n = I_{p\times p}$ (implying that $p \leq n$)

Lasso = soft-thresholding of ordinary least squares

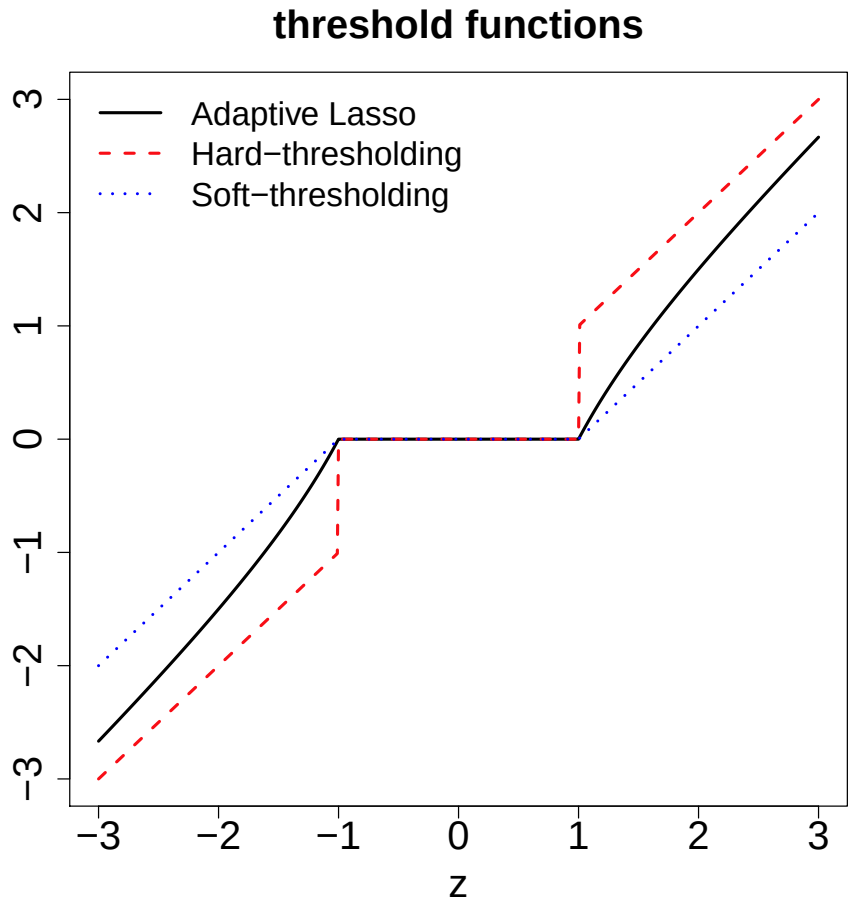
Proposition 1

Assume orthogonal design $X^T X/n = I$. Then,

$$\hat{\beta}_j(\lambda) = g_{\lambda/2}(Z_j), \quad Z_j = (X^T Y)_j/n = \hat{\beta}_{\mathrm{OLS},j},$$
$$g_\lambda(z) = \mathrm{sign}(z)(|z| - \lambda)_+$$

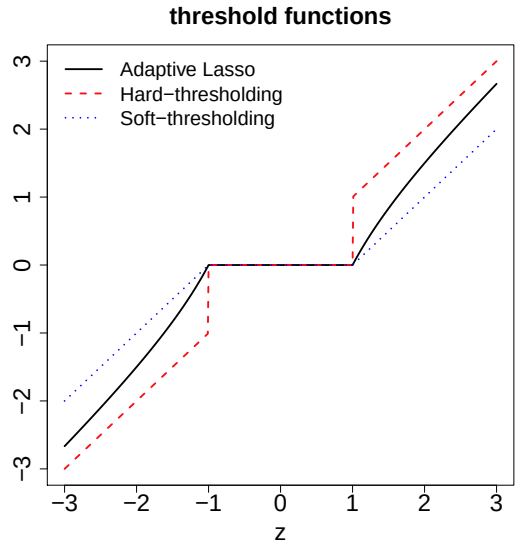


Orthonormal design: explicit solution



$\rightsquigarrow$ Lasso (blue dashed line) exhibits bias!

(Note that OLS is unbiased)



Hard-thresholding:

Proposition 2

Assume orthonormal design $X^T X/n = I$. Then,

$$\hat{\beta}_{\ell_0}(\lambda) = \text{argmin}_\beta \left(\|Y - X\beta\|_2^2/n + \lambda \|\beta\|_0 \right)$$

equals hard-thresholding with threshold value $\sqrt{\lambda}$, that is

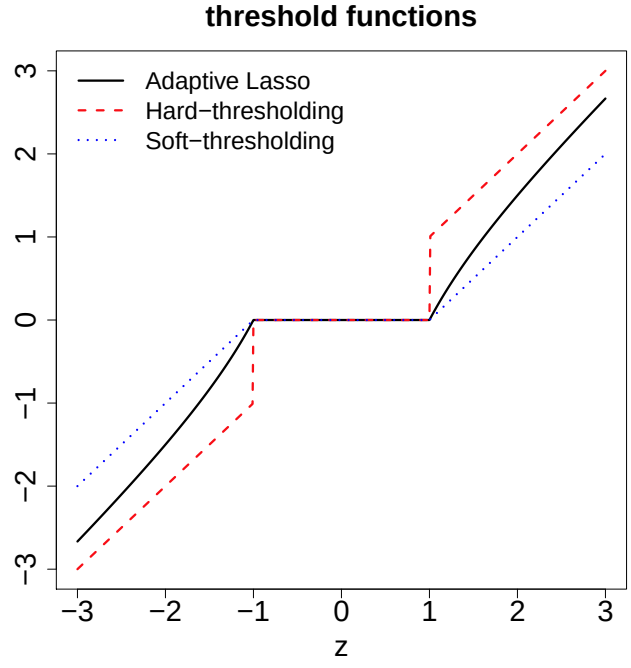
$$\hat{\beta}_{\ell_0;j}(\lambda) = Z_j I(|Z_j| > \sqrt{\lambda}), \; Z = X^T Y/n$$

for example: AIC, BIC are ℓ_0-norm penalized regression

$\rightsquigarrow$ non-convex difficult optimization (but can use recent progress on mixed-integer programming to deal with $p \leq 500$)

- hard-thresholding exhibits less bias than Lasso (but still $\mathbb{E}[\hat{\beta}_{\ell_0}] \neq \beta^0$)

  but it is hard to compute
- adaptive Lasso (black line in the plot) has also less bias than Lasso but can be computed with two Lasso fits, see later



Proof of Proposition 1:
see visualizer

II.4. Prediction with the Lasso

goal: estimation of the regression function

$$f(x) = \mathbb{E}[Y|X = x] = \sum_{j=1}^{p} \beta_j^0 x_j = (\beta^0)^T x$$

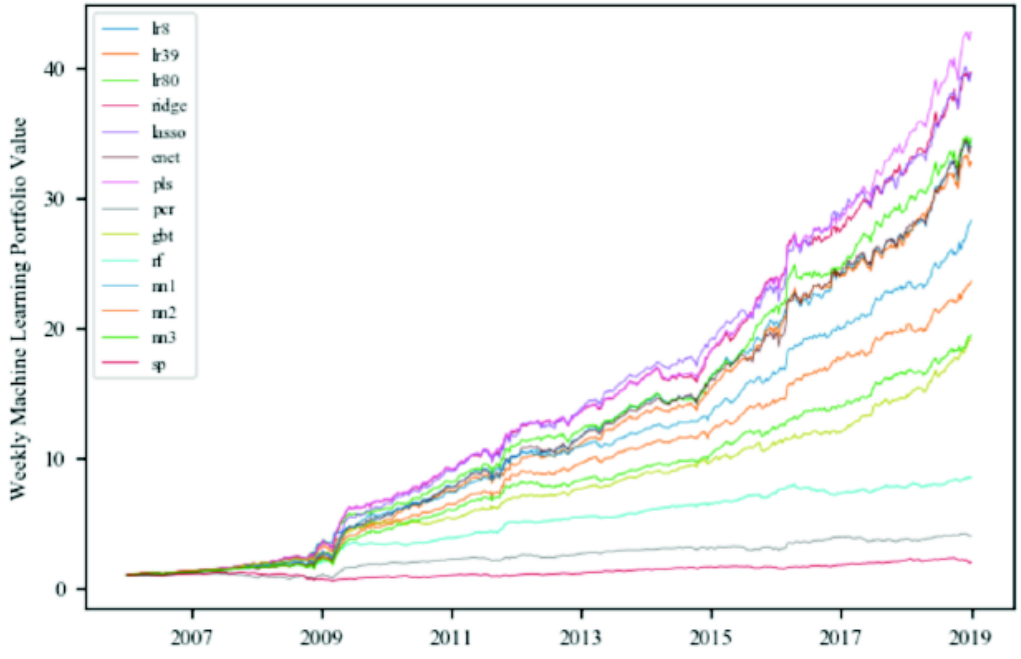
II.4.1. Practical aspects

use $\hat{f}(x) = \hat{\beta}(\lambda)^T x$
and choose λ via cross-validation

often quite a powerful prediction machine

Fig. 6. Cumulative Performance for Weekly Machine Learning Portfolios

This figure depicts the cumulative weekly returns for equal-weighted long-short machine learning portfolios for 13 machine learning models. 'lr80' denotes the linear regression model with all 80 variables. 'lr39' denotes the linear regression model with variables that have strong predictability for weekly returns. 'lr8' denotes the linear regression model with the eight strongest predictor variables. 'ridge' denotes the ridge regression model. 'lasso' denotes the lasso regression model. 'enet' denotes the elastic net regression model. 'pls' denotes the partial least squares regression model. 'pcr' denotes the principal component regression model. 'gbt' denotes the gradient boosting regression tree model. 'rf' denotes the random forest regression model. 'nn1' denotes the neural network model with one hidden layer. 'sp' denotes the benchmark S&P 500 index. The accumulation period is from January 2006 to December 2018, and the initial investment is set as 1 at the start of the accumulation period.



taken from MSc thesis Jiawen Le (September 2019)

also e.g. in connection with deep neural networks: the prediction from the last layer to Y is based on regularized linear models

- last layer features are $\phi(X_i) \in \mathbb{R}^d$
- Lasso:

$$\hat{\beta}(\lambda) = \| Y - (\phi(X_1)^T, \ldots, \phi(X_n)^T)^T \|_2^2 / n + \|\lambda\|_1$$

- prediction $\hat{f}(x) = \hat{\beta}(\lambda)^T \phi(x)$

How to measure prediction quality?

CV test error

but from a theory point of view, we also look at

$$\|X(\hat{\beta} - \beta^0)\|_2^2/n$$

for fixed design:

$$\mathbb{E}[\|X(\hat{\beta} - \beta^0)\|_2^2/n] = \mathbb{E}\|Y - X\hat{\beta}\|_2^2/n + \sigma_\varepsilon^2$$

that is: expected value of in-sample point prediction accuracy