Lasso-type estimators for variable selection

Y. Grandvalet & S. Canu

Heudiasyc — PSI

CNRS/ Univ. de Compiègne — INSA de Rouen

France

http://www.hds.utc.fr/~grandval

Motivations

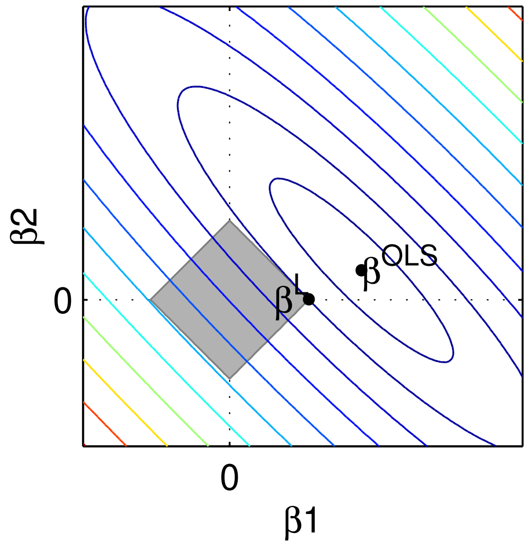
- Subset selection provides sparse solution, but it is unstable: slight changes in the learning set cause important changes in the selected subset;
- Ridge regression does not provide sparse solutions, but is is stable.

Possible improvements

- Stabilize subset selection (bagging);
- **Sparsify penalization procedures**

Lasso (Tibshirani 1994)

$$\widehat{\boldsymbol{\beta}} = \underset{\boldsymbol{\beta}}{\text{Argmin}} \frac{1}{n} \sum_{i=1}^{n} \Big(\sum_{j=1}^{d} \beta_j x_{ij} - y_i \Big)^2 \text{ subject to } \sum_{j=1}^{d} |\beta_j| \leq c$$

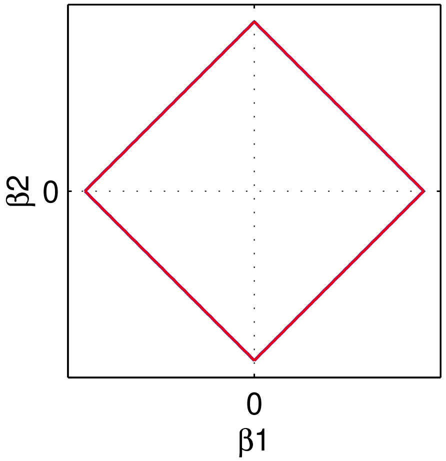
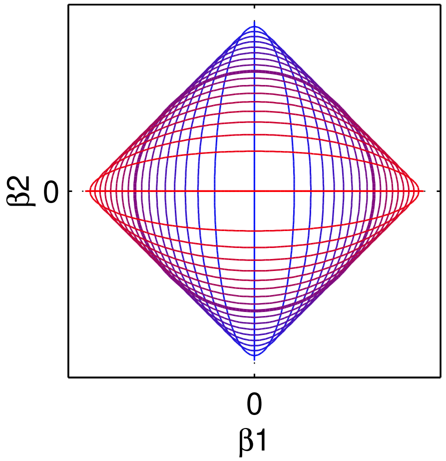


Vertices encourage sparsity

Convexity ensures the unicity of the solution and its continuity w.r.t. data perturbation, hence the algorithm is stable

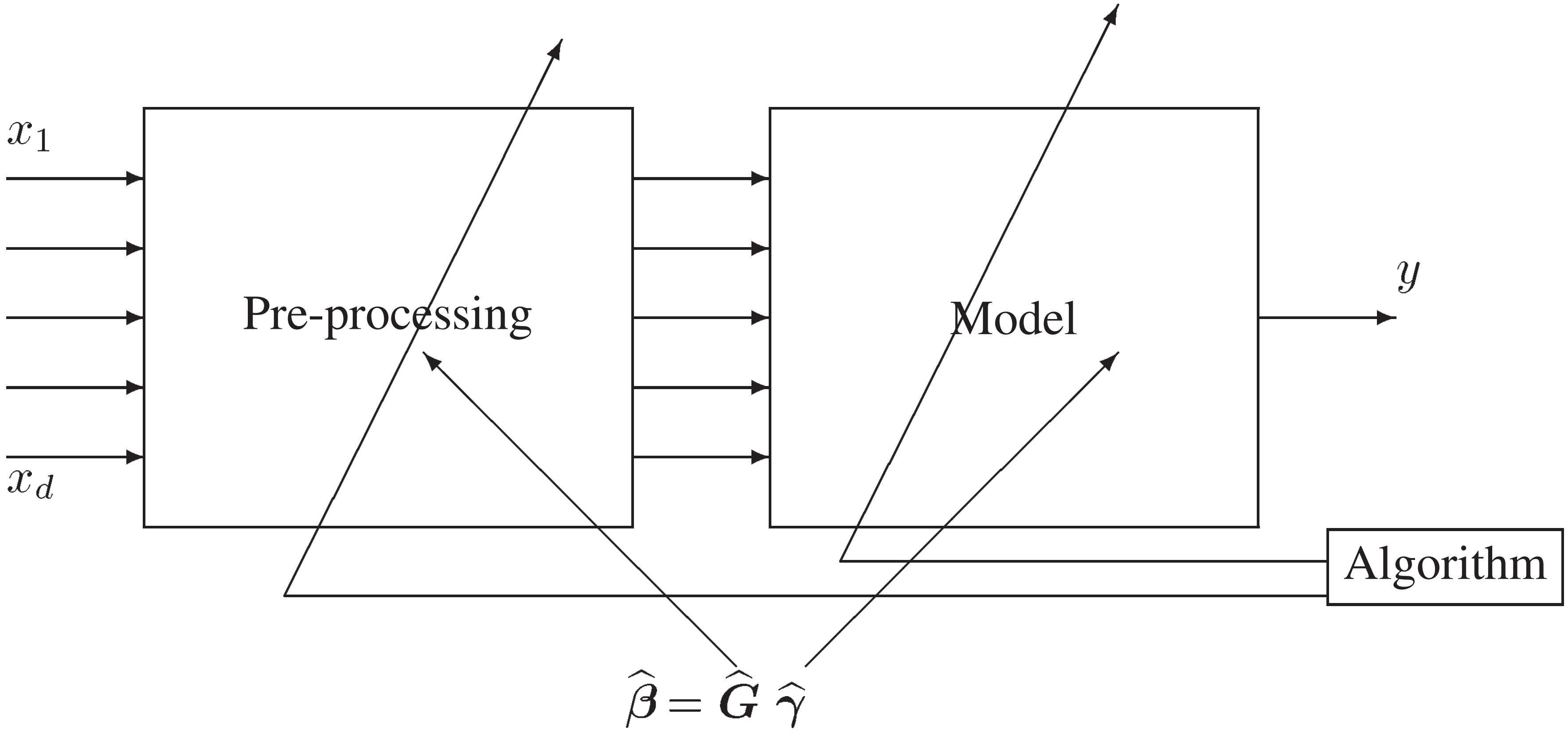
The lasso is equivalent to an adaptive ridge procedure

Admissible region

Lasso	Adaptive Ridge
$\displaystyle\sum_{j=1}^{d} \lvert\beta_j\rvert = c$	$\displaystyle\sum_{j=1}^{d} \frac{\beta_j^2}{b_j^2} = \frac{c^2}{d}, \; \frac{1}{d}\sum_{j=1}^{d} b_j^2 = 1$
	

Anisotropic ridge is equivalent to ridge regression on scaled data

An integrated pre-processing view of lasso



Integrated pre-processing view of lasso

- Optimize the "pre-processor"

$$\widehat{\boldsymbol{G}} = \underset{\boldsymbol{G}}{\text{Argmin}} \frac{1}{n} \sum_{i=1}^{n} \Big(\sum_{j=1}^{d} g_{jj} \gamma_j x_{ij} - y_i \Big)^2 \text{ subject to } \sum_{j=1}^{d} g_{jj}^2 = d$$

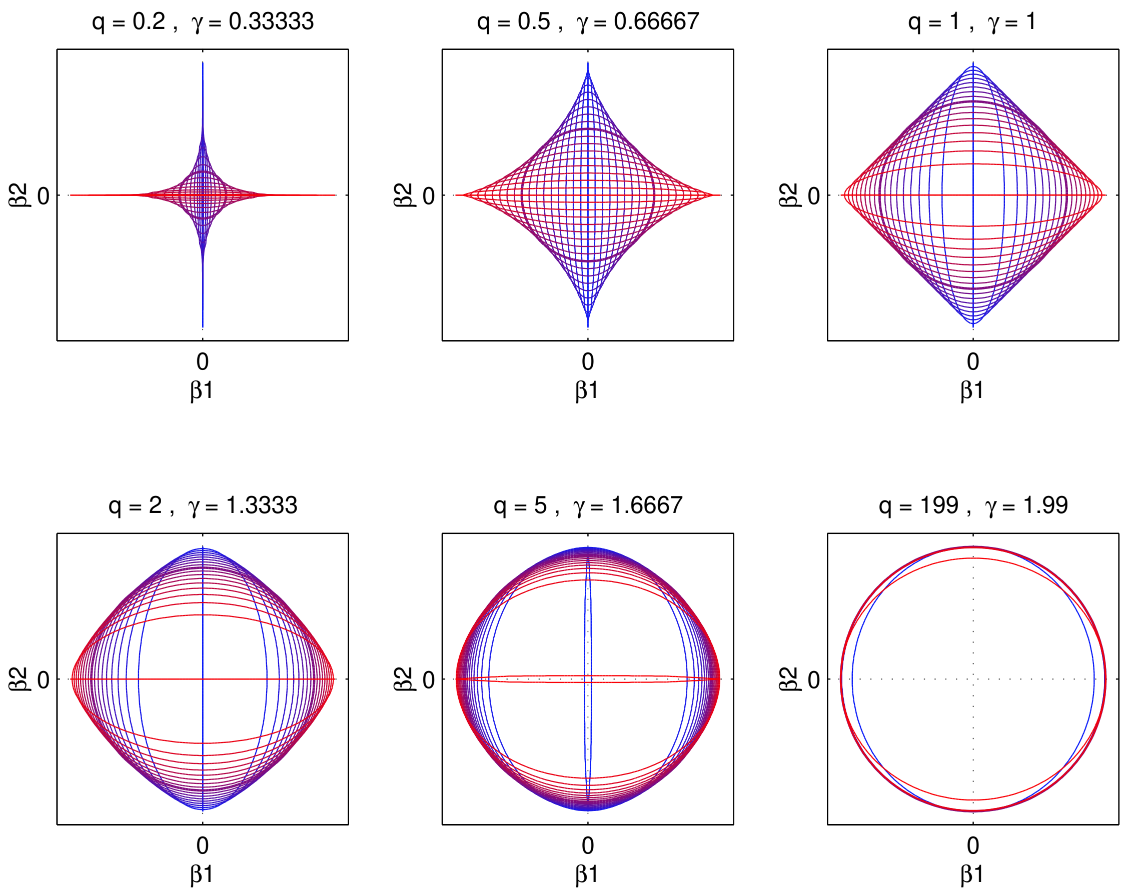
- Optimize the "model": ridge regression on scaled inputs

$$\widehat{\boldsymbol{\gamma}} = \underset{\boldsymbol{\gamma}}{\text{Argmin}} \frac{1}{n} \sum_{i=1}^{n} \Big(\sum_{j=1}^{d} g_{jj} \gamma_j x_{ij} - y_i \Big)^2 \text{ subject to } \sum_{j=1}^{d} \boldsymbol{\gamma}_j^2 \leq c$$

Solutions

$$|\widehat{g}_{jj}| \propto |\widehat{\gamma}_j| \propto \sqrt{|\widehat{\beta}_j|}$$

Sparser solutions: Bridge with admissible set $\sum |\beta_j|^q \leq c$



. . . for $q < 1$solutions should be more sparse, but there are local optima, and the estimate is discontinuous w.r.t. data perturbation

Chaining lasso: a unique and continuous solution to Bridge

- solve the lasso problem in the original feature space $\rightarrow \widehat{\boldsymbol{\beta}}^{(0)}$
- scale input space by $\boldsymbol{G}$, $g_{jj} \propto \sqrt{|\widehat{\boldsymbol{\beta}}_j^{(0)}|}$
- solve the lasso problem in the scaled input space $\rightarrow \widehat{\boldsymbol{\beta}}^{(1)}$,
  then $\boldsymbol{G}\widehat{\boldsymbol{\beta}}^{(1)}$ is a solution with constraints on $||\widehat{\boldsymbol{\beta}}||_{1/2}$ in the original input space.

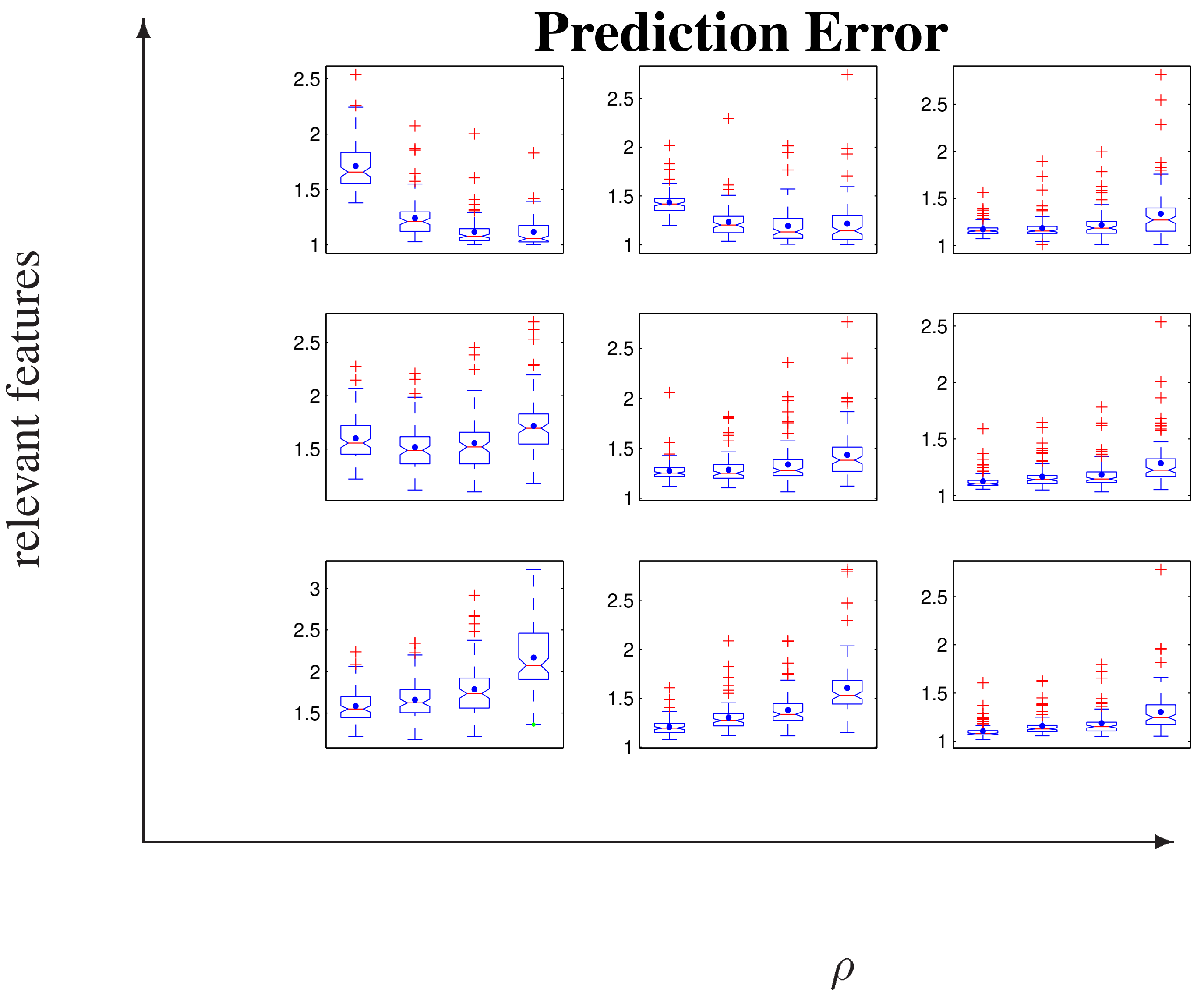
If your patient enough iterate $k-1$ times to get a solution with

$$||\widehat{\boldsymbol{\beta}}||_{1/k} \leq c$$

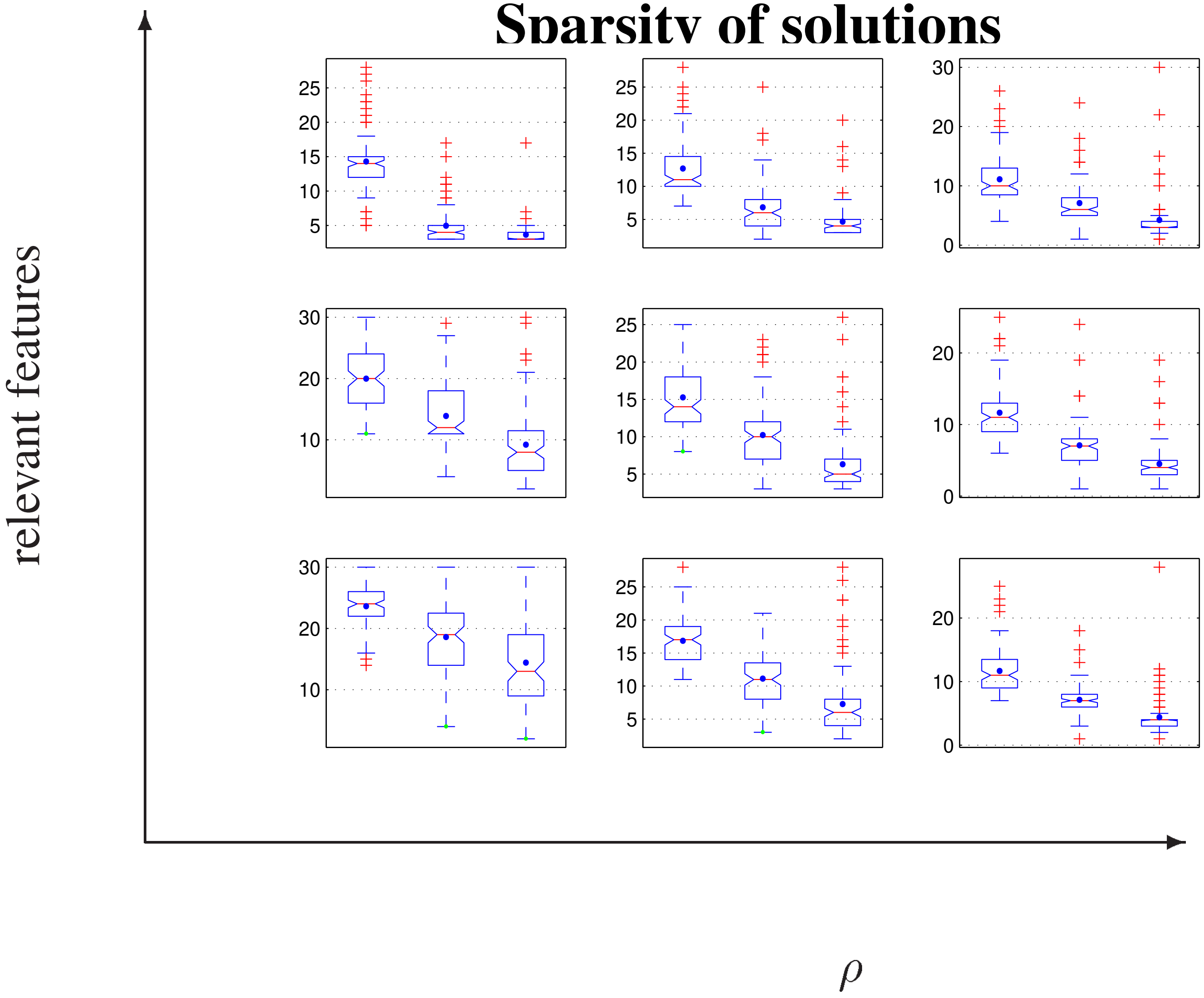
The solution will still be unique and continuous.

Regression example (Breiman, 1996)

- 3, 15 or 27 relevant features out of 30
- correlation of input features k and k $\rho^{|k-l|}$ $\rho = 0.2, 0.8, 0.95$
- 60 learning cases
- 100 learning sets per setup



Prediction Error
relevant features
ρ

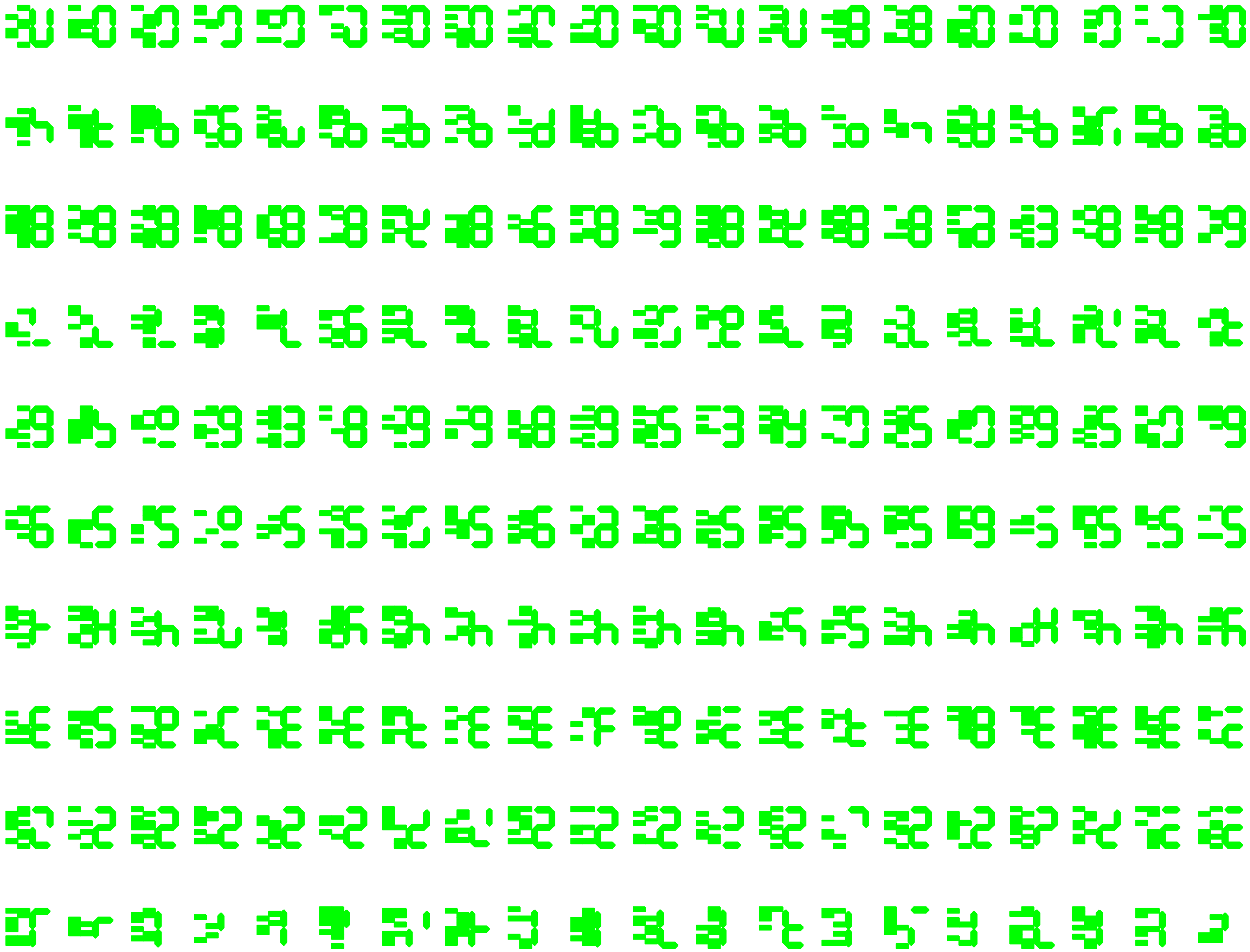


Sparsity of solutions
relevant features
ρ

Noisy LED display (Breiman *et al*, 1984)

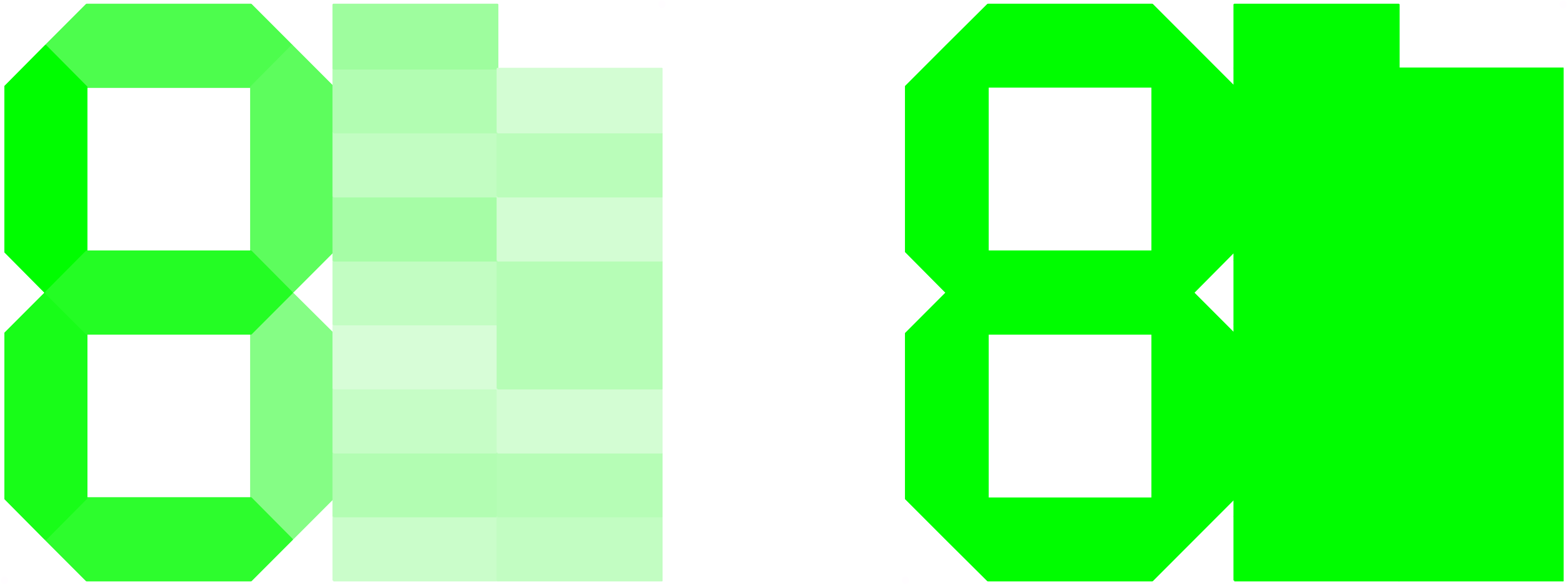
Goal : recognize the digit on a noisy LED display:

- 7 relevant features (binary LED display swithed with $p = 0.1$)
- 17 irrelevant features (Bernoulli distributed, $p = 0.5$)
- 10 classes (0–9)
- Bayes error 26%
- 200 learning cases
- logistic regression: $\widehat{P}(Y = l|\boldsymbol{x}) = \dfrac{\exp(\boldsymbol{x}^T\boldsymbol{\beta}_l)}{\sum_{k=1}^{K} \exp(\boldsymbol{x}^T\boldsymbol{\beta}_k)}$



ℓ_2 **penalization:** $\sum_{j,k,l} (\beta_{kj} - \beta_{lj})^2$

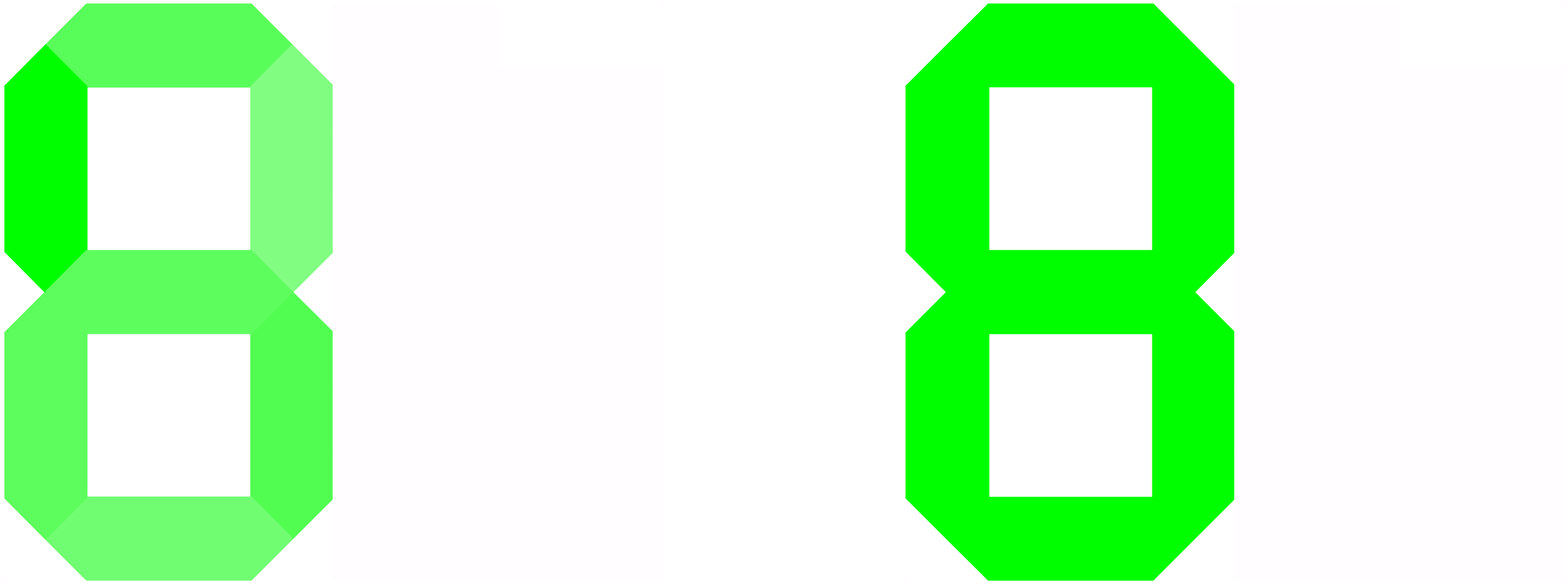
Test error: 34.7%



All inputs seem somewhat relevant

subset selection

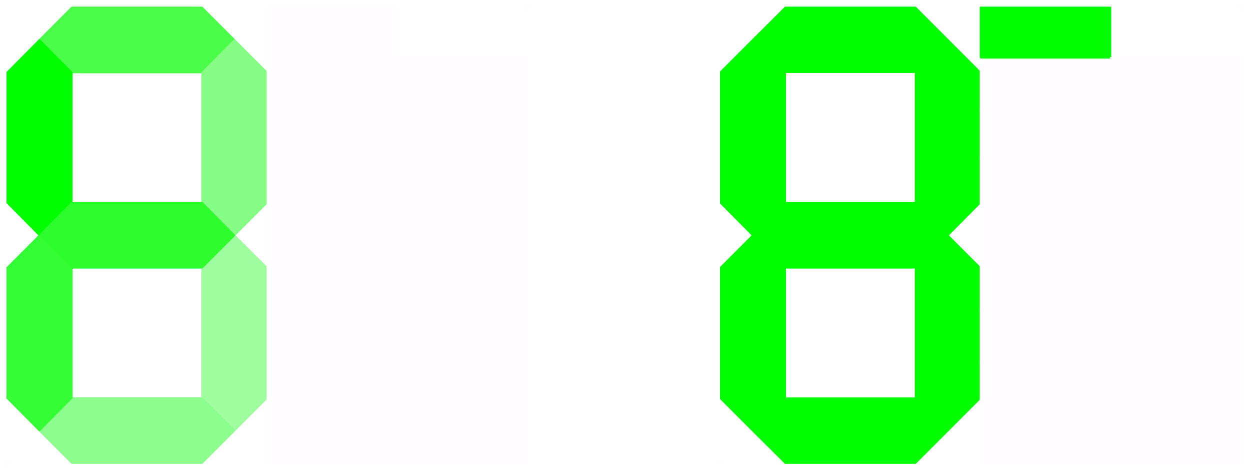
Test error: 28.9%



All really relevant inputs are detected

ℓ_1 **penalization of differences:** $\sum_{j,k,l} |\beta_{kj} - \beta_{lj}|$

Test error: 26.6%



All really relevant inputs detected, one irrelevant input kept . . .

but it yields sparse pairwise classification

subset selection $\qquad$ ℓ_1 penalization

Generalized ℓ_1 penalization:

ℓ_2 **norm of difference vectors** $\sum_j \sqrt{\sum_{k,l} (\beta_{kj} - \beta_{lj})^2}$

Test error: 27.4%

All really relevant inputs are detected,
sparsity over all classes but not for pairwise classification

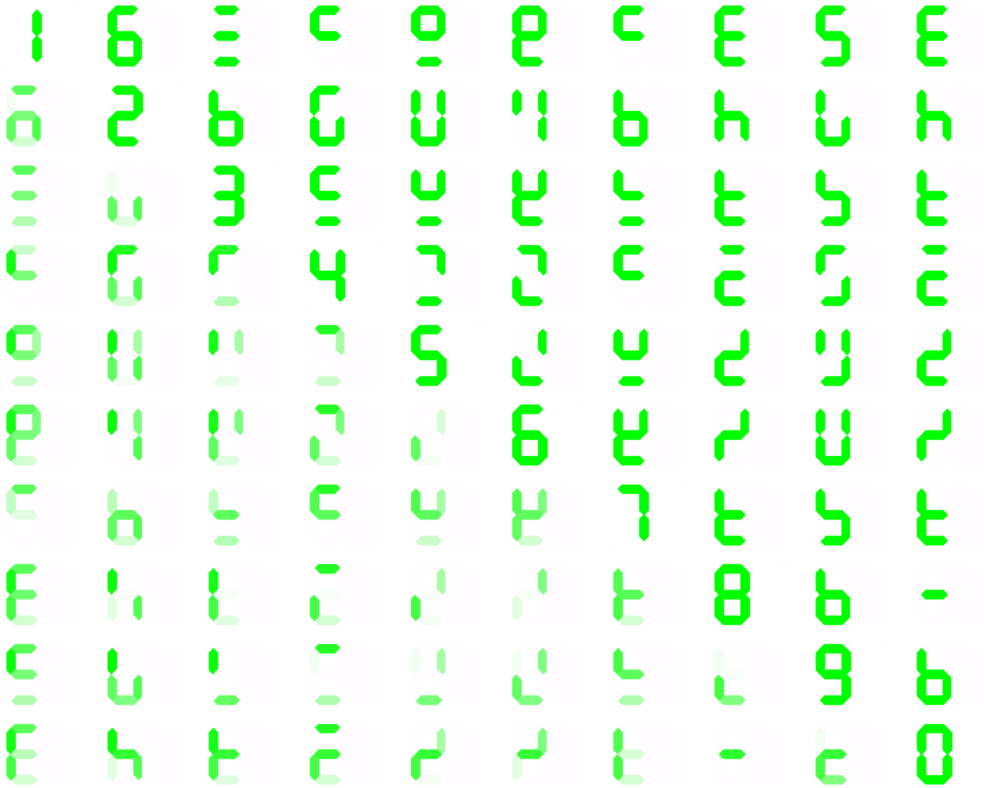
Chaining to get sparsity at both levels

- Scale the input variables by multiplying x_j by $\sqrt{\sum_{k,l}(\beta_{kj} - \beta_{lj})^2}$
- run the algorithm penalizing the ℓ_1 penalization of differences

The previous "adaptive pre-processing" sets the present input scale

Results

Test error: 26.5%



All really relevant inputs are detected, sparsity over all classes and for pairwise classification