Normalization for Class-Imbalanced Binary Features in Regularized Regression

Johan Larsson

larssonjohan.com, @jolars@fediscience.org

Department of Statistics, Lund University

April 10, 2024

Presentation

- PhD student at Lund University (supervised by Jonas Wallin). As of September, post doc at Copenhagen University.
- Work so far: mostly computational optimization and algorithms for speeding up sparse regression.

This Talk

Topic

Normalization (scaling) of binary features in regularized regression

This Talk

Topic

Normalization (scaling) of binary features in regularized regression

Problem

The elastic net (combination of lasso and ridge)

This Talk

Topic

Normalization (scaling) of binary features in regularized regression

Problem

The elastic net (combination of lasso and ridge)

Results

- Class balance has a normalization-dependent impact on the model.
- In mixed datas, choice of normalization implictly biases coefficients.

This Talk

Topic

Normalization (scaling) of binary features in regularized regression

Problem

The elastic net (combination of lasso and ridge)

Results

- Class balance has a normalization-dependent impact on the model.
- In mixed datas, choice of normalization implictly biases coefficients.

Notes

- Not yet published (and partly work-in-progress)
- Joint work with Jonas Wallin

Overview

Preliminaries

General Setup

- Data consists of a fixed matrix of features $\boldsymbol{X} \in \mathbb{R}^{n \times p}$ and a response vector $\boldsymbol{y} \in \mathbb{R}^n$.
- $\boldsymbol{y}$ comes from a linear model, that is,

$$y_i = \beta_0^* + \boldsymbol{x}_i^\intercal \boldsymbol{\beta}^* + \varepsilon_i \quad \text{for} \quad i \in 1, \ldots, n,$$

  where $\boldsymbol{\beta}^*$ is the vector of *true* coefficients.
- ε_i is the measurement noise, generated from some random variable

The Elastic Net

Linear regression plus a combination of the ℓ_1 and ℓ_2 penalties:

$$(\hat{\beta}_0, \hat{\boldsymbol{\beta}}) = \underset{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p}{\arg\min} \left(\frac{1}{2} \|\boldsymbol{y} - \beta_0 - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 + \frac{\lambda_2}{2} \|\boldsymbol{\beta}\|_2^2 \right).$$

The Elastic Net

Linear regression plus a combination of the ℓ_1 and ℓ_2 penalties:

$$(\hat{\beta}_0, \hat{\boldsymbol{\beta}}) = \underset{\beta_0 \in \mathbb{R}, \beta \in \mathbb{R}^p}{\arg\min} \left(\frac{1}{2} \|\boldsymbol{y} - \beta_0 - \boldsymbol{X}\boldsymbol{\beta}\|_2^2 + \lambda_1 \|\boldsymbol{\beta}\|_1 + \frac{\lambda_2}{2} \|\boldsymbol{\beta}\|_2^2 \right).$$

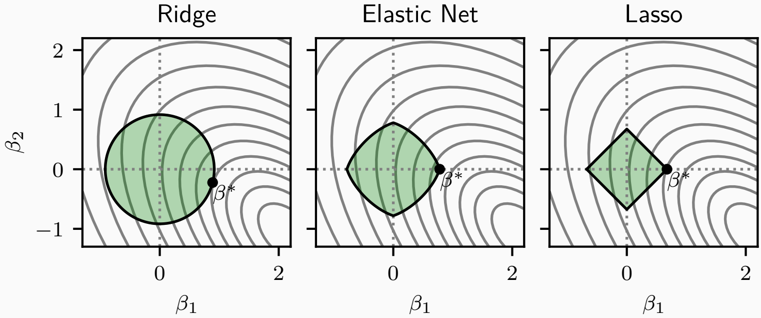


Figure 1: The elastic net penalty is a combination of the lasso and ridge penalties. Here shown as a constrained problem.

The Elastic Net Path

- Usually don't know optimal λ_1 and λ_2 in advance.

The Elastic Net Path

- Usually don't know optimal λ_1 and λ_2 in advance.
- Instead we typically hyper-optimize (e.g. cross-validate) over a grid.

The Elastic Net Path

- Usually don't know optimal λ_1 and λ_2 in advance.
- Instead we typically hyper-optimize (e.g. cross-validate) over a grid.
- Common parametrization:

$$\lambda_1 = \alpha\lambda,$$
$$\lambda_2 = (1 - \alpha)\lambda$$

with $\alpha \in [0, 1]$.

The Elastic Net Path

- Usually don't know optimal λ_1 and λ_2 in advance.
- Instead we typically hyper-optimize (e.g. cross-validate) over a grid.
- Common parametrization:

$$\lambda_1 = \alpha\lambda,$$
$$\lambda_2 = (1-\alpha)\lambda$$

  with $\alpha \in [0, 1]$.
- For each α, solve the elastic net over a sequence of λ: the **elastic net path**.

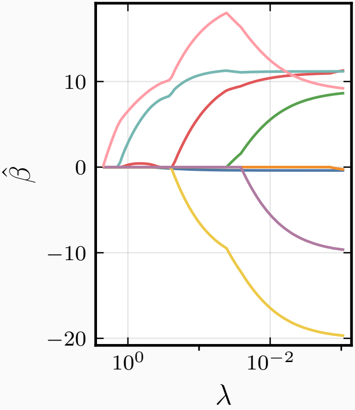


Figure 2: The elastic net path

Background on the Elastic Net

- Proposed by Zou and Hastie (2005).

Background on the Elastic Net

- Proposed by Zou and Hastie (2005).
- Lasso (ℓ_1) part:
  - Enables sparsity (interpretability, parsimony, feature selection)
  - Efficient when $p \gg n$ (due to screening rules (El Ghaoui, Viallon, and Rabbani 2010; Tibshirani et al. 2012))

Background on the Elastic Net

- Proposed by Zou and Hastie (2005).
- Lasso (ℓ_1) part:
  - Enables sparsity (interpretability, parsimony, feature selection)
  - Efficient when $p \gg n$ (due to screening rules (El Ghaoui, Viallon, and Rabbani 2010; Tibshirani et al. 2012))
- Ridge (ℓ_2) part
  - Mitigates lasso issue in correlated data
  - Better predictive performance when true signal is non-sparse

Background on the Elastic Net

- Proposed by Zou and Hastie (2005).
- Lasso (ℓ_1) part:
  - Enables sparsity (interpretability, parsimony, feature selection)
  - Efficient when $p \gg n$ (due to screening rules (El Ghaoui, Viallon, and Rabbani 2010; Tibshirani et al. 2012))
- Ridge (ℓ_2) part
  - Mitigates lasso issue in correlated data
  - Better predictive performance when true signal is non-sparse
- Very efficient solvers for the full path (coordinate descent)

Sensitivity to Scale

Since both the lasso and ridge penalize the *norm* of the coefficients, they are sensitive to the scale of the features.

Sensitivity to Scale

Since both the lasso and ridge penalize the *norm* of the coefficients, they are sensitive to the scale of the features.

Example

Assume

$$\boldsymbol{X} \sim \mathrm{Normal}\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}\right), \qquad \boldsymbol{\beta}^* = \begin{bmatrix}\frac{1}{2}\\1\end{bmatrix}.$$

Sensitivity to Scale

Since both the lasso and ridge penalize the *norm* of the coefficients, they are sensitive to the scale of the features.

Example

Assume

$$\boldsymbol{X} \sim \mathrm{Normal}\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}\right), \qquad \boldsymbol{\beta}^* = \begin{bmatrix}\frac{1}{2}\\1\end{bmatrix}.$$

Model	$\hat{\boldsymbol{\beta}}$	$\hat{\boldsymbol{\beta}}_{\text{std}}$
OLS	$\begin{bmatrix}0.50 & 1.00\end{bmatrix}^\intercal$	$\begin{bmatrix}1.00 & 1.00\end{bmatrix}^\intercal$
Lasso	$\begin{bmatrix}0.38 & 0.50\end{bmatrix}^\intercal$	$\begin{bmatrix}0.74 & 0.50\end{bmatrix}^\intercal$
Ridge	$\begin{bmatrix}0.37 & 0.41\end{bmatrix}^\intercal$	$\begin{bmatrix}0.74 & 0.41\end{bmatrix}^\intercal$

Sensitivity to Scale

Since both the lasso and ridge penalize the *norm* of the coefficients, they are sensitive to the scale of the features.

Example

Assume

$$\boldsymbol{X} \sim \mathrm{Normal}\left(\begin{bmatrix}0\\0\end{bmatrix}, \begin{bmatrix}2 & 0\\0 & 1\end{bmatrix}\right), \qquad \boldsymbol{\beta}^* = \begin{bmatrix}\frac{1}{2}\\1\end{bmatrix}.$$

Model	$\hat{\boldsymbol{\beta}}$	$\hat{\boldsymbol{\beta}}_{\text{std}}$
OLS	$\begin{bmatrix}0.50 & 1.00\end{bmatrix}^\intercal$	$\begin{bmatrix}1.00 & 1.00\end{bmatrix}^\intercal$
Lasso	$\begin{bmatrix}0.38 & 0.50\end{bmatrix}^\intercal$	$\begin{bmatrix}0.74 & 0.50\end{bmatrix}^\intercal$
Ridge	$\begin{bmatrix}0.37 & 0.41\end{bmatrix}^\intercal$	$\begin{bmatrix}0.74 & 0.41\end{bmatrix}^\intercal$

Large scale means less penalization because the size of β_j can be smaller for an equivalent effect (on $\boldsymbol{y}$).

Normalization

- The solution to the scale sensitivity is to normalized the features (before fitting).

Normalization

- The solution to the scale sensitivity is to normalized the features (before fitting).
- Let $\tilde{\boldsymbol{X}}$ be the normalized feature matrix, with elements

$$\tilde{x}_{ij} = \frac{x_{ij} - c_j}{s_j},$$

where x_{ij} is an element of the (unnormalized) feature matrix and c_j and s_j are the *centering* and *scaling* factors respectively.

Normalization

- The solution to the scale sensitivity is to normalized the features (before fitting).
- Let $\tilde{\boldsymbol{X}}$ be the normalized feature matrix, with elements

$$\tilde{x}_{ij} = \frac{x_{ij} - c_j}{s_j},$$

  where x_{ij} is an element of the (unnormalized) feature matrix and c_j and s_j are the *centering* and *scaling* factors respectively.
- Usage of key terms ambiguous in literature.

Normalization

- The solution to the scale sensitivity is to normalized the features (before fitting).
- Let $\tilde{\boldsymbol{X}}$ be the normalized feature matrix, with elements

$$\tilde{x}_{ij} = \frac{x_{ij} - c_j}{s_j},$$

  where x_{ij} is an element of the (unnormalized) feature matrix and c_j and s_j are the *centering* and *scaling* factors respectively.
- Usage of key terms ambiguous in literature.
- After fitting, we transform the coefficients back to their original scale via

$$\hat{\beta}_j = \frac{\hat{\beta}_j^{(n)}}{s_j} \quad \text{for} \quad j = 1, 2, \dots, p.$$

Table 1: Common ways to normalize $\boldsymbol{X}$

Normalization	Centering (c_j)	Scaling (s_j)
Standardization	$\frac{1}{n}\sum_{i=1}^{n} x_{ij}$	$\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_{ij}-\bar{x}_j)^2}$
Min–Max	$\min_i(x_{ij})$	$\max_i(x_{ij}) - \min_i(x_{ij})$
Unit Vector (L2)	0	$\sqrt{\sum_{i=1}^{n} x_{ij}^2}$
Max–Abs	0	$\max_i(\lvert x_{ij}\rvert)$
Adaptive Lasso	0	β_j^{OLS}

Motivation

The Type of Normalization Matters

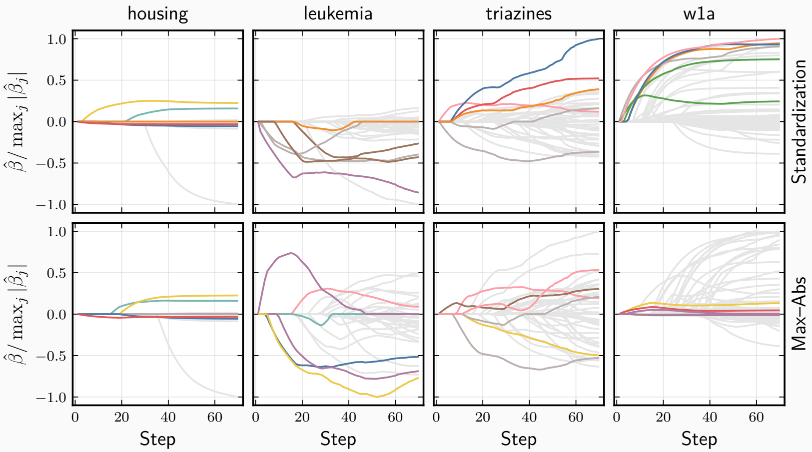


Figure 3: Lasso paths under two different types of normalization (standardization and max–abs normalization). The union of the first ten features selected in any of the schemes are colored.

Motivation

- So normalization matters and you might expect that there should be lot of literature on the topic.

Motivation

- So normalization matters and you might expect that there should be lot of literature on the topic.
- But the short summary of the literature on the topic is that there is no literature on the topic.

Motivation

- So normalization matters and you might expect that there should be lot of literature on the topic.
- But the short summary of the literature on the topic is that there is no literature on the topic.
- Everyone agrees that you need to normalize (for most data), but how to do so is not discussed and often motivated by being “standard”.

Motivation

- So normalization matters and you might expect that there should be lot of literature on the topic.
- But the short summary of the literature on the topic is that there is no literature on the topic.
- Everyone agrees that you need to normalize (for most data), but how to do so is not discussed and often motivated by being "standard".
- Documentation for popular machine learning packages advocate different normalization strategies when data is sparse.

Motivation

- So normalization matters and you might expect that there should be lot of literature on the topic.
- But the short summary of the literature on the topic is that there is no literature on the topic.
- Everyone agrees that you need to normalize (for most data), but how to do so is not discussed and often motivated by being "standard".
- Documentation for popular machine learning packages advocate different normalization strategies when data is sparse.
- Consensus for approximately normal features but little discussion on binary features and choice seems domain-specific. (Statisticians standardize, machine learning people scale to $[0, 1]$ or $[-1, 1]$.)

Aims

We focus on the following aspects of normalization in the context of the elastic net:

- Binary features, particularly with respect to the class balance thereof
- A mix of binary and normally distributed features
- Interactions

Results

Orthogonal Features

There is not an explicit solution to the elastic net problem in general (unless $\lambda_1 = 0$).

[1] We have also assumed that the features are mean-centered here.

Orthogonal Features

There is not an explicit solution to the elastic net problem in general (unless $\lambda_1 = 0$).

But if we assume that the features are orthogonal, that is

$$\tilde{\boldsymbol{X}}^\intercal \tilde{\boldsymbol{X}} = \operatorname{diag}(\tilde{\boldsymbol{x}}_1^\intercal \tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_p^\intercal \tilde{\boldsymbol{x}}_p),$$

then there is an explicit solution to the elastic net problem[1]:

$$\hat{\beta}_j = \frac{\mathrm{S}_{\lambda_1}\left(\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{y}\right)}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)},$$

where

$$\mathrm{S}_\lambda(z) = \operatorname{sign}(z) \max(|z| - \lambda, 0).$$

[1]We have also assumed that the features are mean-centered here.

Orthogonal Features

There is not an explicit solution to the elastic net problem in general (unless $\lambda_1 = 0$).

But if we assume that the features are orthogonal, that is

$$\tilde{\boldsymbol{X}}^\intercal \tilde{\boldsymbol{X}} = \operatorname{diag}(\tilde{\boldsymbol{x}}_1^\intercal \tilde{\boldsymbol{x}}_1, \ldots, \tilde{\boldsymbol{x}}_p^\intercal \tilde{\boldsymbol{x}}_p),$$

then there is an explicit solution to the elastic net problem[1]:

$$\hat{\beta}_j = \frac{\mathrm{S}_{\lambda_1}\left(\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{y}\right)}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)},$$

where

$$\mathrm{S}_\lambda(z) = \operatorname{sign}(z)\max(|z| - \lambda, 0).$$

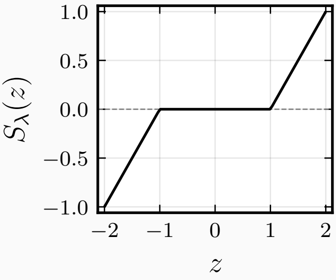


Figure 4: Soft thresholding

[1]We have also assumed that the features are mean-centered here.

Bias and Variance of the Elastic Net Estimator

The goal is computing the expected value of the elastic net estimator,

$$\operatorname{E} \hat{\beta}_j = \frac{\operatorname{E} \mathrm{S}_{\lambda_1}\left(\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{y}\right)}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)},$$

since we treat $\boldsymbol{X}$ as fixed.

Bias and Variance of the Elastic Net Estimator

The goal is computing the expected value of the elastic net estimator,

$$\operatorname{E} \hat{\beta}_j = \frac{\operatorname{E} \operatorname{S}_{\lambda_1}\left(\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{y}\right)}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)},$$

since we treat $\boldsymbol{X}$ as fixed.

Letting $Z = \tilde{\boldsymbol{x}}^\intercal \boldsymbol{y}$ and assuming that ε_i is i.i.d. with mean zero and finite variance σ_ε^2, we have

$$\begin{aligned}
\operatorname{E} Z = \mu &= \operatorname{E}\left(\tilde{\boldsymbol{x}}_j^\intercal(\boldsymbol{x}_j\beta_j + \boldsymbol{\varepsilon})\right) = \tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{x}_j \beta_j,\\
\operatorname{Var} Z = \sigma^2 &= \operatorname{Var}\left(\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{\varepsilon}\right) = \sigma_\varepsilon^2 \|\tilde{\boldsymbol{x}}_j\|_2^2.
\end{aligned}$$

Next, will will turn to $\operatorname{E} \operatorname{S}_{\lambda_1}(Z)$.

Bias of Soft-Thresholding

The expected value of the soft-thresholding estimator is

$$
\begin{aligned}
&\operatorname{E} \operatorname{S}_\lambda(Z) \\
&= \int_{-\infty}^{\infty} \operatorname{S}_\lambda(z) f_Z(z) \,\mathrm{d}z \\
&= \int_{-\infty}^{-\lambda} (z+\lambda) f_Z(z) \,\mathrm{d}z \\
&\quad + \int_{\lambda}^{\infty} (z-\lambda) f_Z(z) \,\mathrm{d}z.
\end{aligned}
$$

And so the bias of $\hat{\beta}_j$ is

$$\operatorname{E} \hat{\beta}_j - \beta_j^* = \frac{1}{d_j} \operatorname{E} \operatorname{S}_\lambda(Z) - \beta_j^*.$$

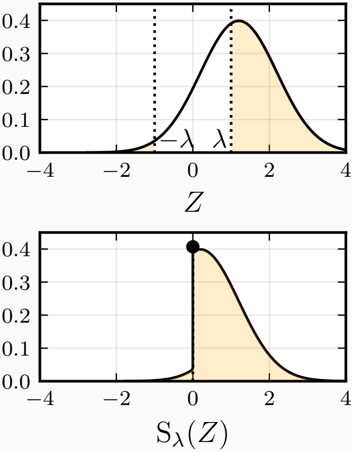


Figure 5: Distributions of Z and the its value after soft-thresholding.

Variance of Soft-Thresholding

The variance of the soft-thresholding estimator is

$$\operatorname{Var} S_\lambda(Z) = \int_{-\infty}^{-\lambda} (z+\lambda)^2 f_Z(z)\,\mathrm{d}z + \int_{\lambda}^{\infty} (z-\lambda)^2 f_Z(z)\,\mathrm{d}z - \left(\operatorname{E} \mathrm{S}_\lambda(Z)\right)^2$$

and consequently the variance of the elastic net estimator is therefore

$$\operatorname{Var} \hat{\beta}_j = \frac{1}{d_j^2} \operatorname{Var} \mathrm{S}_\lambda(Z).$$

Normally Distributed Noise

We now assume that $\varepsilon_i \sim \mathrm{Normal}\left(0, \sigma_\varepsilon^2\right)$, which means that

$$Z \sim \mathrm{Normal}\left(\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{x}_j \beta_j, \sigma_\varepsilon^2 \|\tilde{\boldsymbol{x}}_j\|_2^2\right).$$

Let $\theta = -\mu - \lambda_1$ and $\gamma = \mu - \lambda_1$. Then the expected value of soft-thresholding of Z is

$$\begin{aligned}
\mathrm{E}\,\mathrm{S}_{\lambda_1}(Z) &= \int_{-\infty}^{\frac{\theta}{\sigma}} (\sigma u - \theta)\,\phi(u)\,\mathrm{d}u + \int_{-\frac{\gamma}{\sigma}}^{\infty} (\sigma u + \gamma)\,\phi(u)\,\mathrm{d}u \\
&= -\theta\,\Phi\left(\frac{\theta}{\sigma}\right) - \sigma\,\phi\left(\frac{\theta}{\sigma}\right) + \gamma\,\Phi\left(\frac{\gamma}{\sigma}\right) + \sigma\,\phi\left(\frac{\gamma}{\sigma}\right)
\end{aligned}$$

where $\phi(u)$ and $\Phi(u)$ are the pdf and cdf of the standard normal distribution, respectively.

Normally Distributed Noise

We now assume that $\varepsilon_i \sim \operatorname{Normal}(0, \sigma_\varepsilon^2)$, which means that

$$Z \sim \operatorname{Normal}\left(\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{x}_j \beta_j, \sigma_\varepsilon^2 \|\tilde{\boldsymbol{x}}_j\|_2^2\right).$$

Let $\theta = -\mu - \lambda_1$ and $\gamma = \mu - \lambda_1$. Then the expected value of soft-thresholding of Z is

$$\begin{aligned}
\operatorname{E} \operatorname{S}_{\lambda_1}(Z) &= \int_{-\infty}^{\frac{\theta}{\sigma}} (\sigma u - \theta)\, \phi(u) \, \mathrm{d}u + \int_{-\frac{\gamma}{\sigma}}^{\infty} (\sigma u + \gamma)\, \phi(u) \, \mathrm{d}u \\
&= -\theta\, \Phi\left(\frac{\theta}{\sigma}\right) - \sigma\, \phi\left(\frac{\theta}{\sigma}\right) + \gamma\, \Phi\left(\frac{\gamma}{\sigma}\right) + \sigma\, \phi\left(\frac{\gamma}{\sigma}\right)
\end{aligned}$$

where $\phi(u)$ and $\Phi(u)$ are the pdf and cdf of the standard normal distribution, respectively.

Similar, but more complicated, expression can be derived for $\operatorname{Var} \operatorname{S}_{\lambda_1}(Z)$.

Binary Features

Let's say we have a binary feature $\boldsymbol{x}_j$, such that $x_{ij} \in \{0, 1\}$. Let $q \in [0, 1]$ be the class balance of this feature, that is: $q = \bar{\boldsymbol{x}}_j$.

In this case, we observe that

$$\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j = \frac{1}{s_j^2}(\boldsymbol{x}_j - \mathbf{1}c_j)^\intercal(\boldsymbol{x}_j - \mathbf{1}c_j) = \frac{1}{s_j^2}(nq - 2nq^2 + nq^2) = \frac{n(q - q^2)}{s_j^2},$$

$$\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{x}_j = \frac{1}{s_j}(\boldsymbol{x}_j^\intercal \boldsymbol{x}_j - \boldsymbol{x}_j^\intercal \mathbf{1}c_j) = \frac{n(q - q^2)}{s_j}.$$

Binary Features

Let's say we have a binary feature $\boldsymbol{x}_j$, such that $x_{ij} \in \{0, 1\}$. Let $q \in [0, 1]$ be the class balance of this feature, that is: $q = \bar{\boldsymbol{x}}_j$.

In this case, we observe that

$$\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j = \frac{1}{s_j^2}(\boldsymbol{x}_j - \mathbf{1}c_j)^\intercal(\boldsymbol{x}_j - \mathbf{1}c_j) = \frac{1}{s_j^2}(nq - 2nq^2 + nq^2) = \frac{n(q - q^2)}{s_j^2},$$

$$\tilde{\boldsymbol{x}}_j^\intercal \boldsymbol{x}_j = \frac{1}{s_j}(\boldsymbol{x}_j^\intercal \boldsymbol{x}_j - \boldsymbol{x}_j^\intercal \mathbf{1}c_j) = \frac{n(q - q^2)}{s_j}.$$

And consequently

$$\mu = \frac{\beta_j^* n(q - q^2)}{s_j}, \qquad \sigma^2 = \frac{\sigma_\varepsilon^2 n(q - q^2)}{s_j^2}, \qquad d_j = \frac{n(q - q^2)}{s_j} + \lambda_2 s_j.$$

Noiseless Case

In the noiseless case, we have

$$\hat{\beta}_j = \frac{\mathrm{S}_{\lambda_1}(\tilde{\boldsymbol{x}}^\intercal \boldsymbol{y})}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)} = \frac{\mathrm{S}_{\lambda_1}\left(\frac{\beta_j^* n(q-q^2)}{s_j}\right)}{s_j\left(\frac{n(q-q^2)}{s_j^2} + \lambda_2\right)}.$$

Noiseless Case

In the noiseless case, we have

$$\hat{\beta}_j = \frac{\mathrm{S}_{\lambda_1}(\tilde{\boldsymbol{x}}^\intercal \boldsymbol{y})}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)} = \frac{\mathrm{S}_{\lambda_1}\left(\frac{\beta_j^* n(q-q^2)}{s_j}\right)}{s_j\left(\frac{n(q-q^2)}{s_j^2} + \lambda_2\right)}.$$

- Means that the elastic net estimator depends on class balance (q).

Noiseless Case

In the noiseless case, we have

$$\hat{\beta}_j = \frac{\mathrm{S}_{\lambda_1}(\tilde{\boldsymbol{x}}^\intercal \boldsymbol{y})}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)} = \frac{\mathrm{S}_{\lambda_1}\left(\frac{\beta_j^* n(q-q^2)}{s_j}\right)}{s_j\left(\frac{n(q-q^2)}{s_j^2} + \lambda_2\right)}.$$

- Means that the elastic net estimator depends on class balance (q).
- To remove the effect of q, an intuitive setting would be $s_j = q - q^2$ (the variance of $\boldsymbol{x}_j$) in the case of the lasso and $s_j = \sqrt{q - q^2}$ (the standard deviation of $\boldsymbol{x}_j$) in the case of the ridge.

Noiseless Case

In the noiseless case, we have

$$\hat{\beta}_j = \frac{\mathrm{S}_{\lambda_1}(\tilde{\boldsymbol{x}}^\intercal \boldsymbol{y})}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)} = \frac{\mathrm{S}_{\lambda_1}\left(\frac{\beta_j^* n(q-q^2)}{s_j}\right)}{s_j\left(\frac{n(q-q^2)}{s_j^2} + \lambda_2\right)}.$$

- Means that the elastic net estimator depends on class balance (q).
- To remove the effect of q, an intuitive setting would be $s_j = q - q^2$ (the variance of $\boldsymbol{x}_j$) in the case of the lasso and $s_j = \sqrt{q - q^2}$ (the standard deviation of $\boldsymbol{x}_j$) in the case of the ridge.
- Suggests the parametrization

$$s_j = (q - q^2)^\delta, \qquad \delta \geq 0.$$

Noiseless Case

In the noiseless case, we have

$$\hat{\beta}_j = \frac{\mathrm{S}_{\lambda_1}(\tilde{\boldsymbol{x}}^\intercal \boldsymbol{y})}{s_j\left(\tilde{\boldsymbol{x}}_j^\intercal \tilde{\boldsymbol{x}}_j + \lambda_2\right)} = \frac{\mathrm{S}_{\lambda_1}\left(\frac{\beta_j^* n(q-q^2)}{s_j}\right)}{s_j\left(\frac{n(q-q^2)}{s_j^2} + \lambda_2\right)}.$$

- Means that the elastic net estimator depends on class balance (q).
- To remove the effect of q, an intuitive setting would be $s_j = q - q^2$ (the variance of $\boldsymbol{x}_j$) in the case of the lasso and $s_j = \sqrt{q - q^2}$ (the standard deviation of $\boldsymbol{x}_j$) in the case of the ridge.
- Suggests the parametrization

$$s_j = (q - q^2)^\delta, \qquad \delta \geq 0.$$

- Indicates there might be no (simple) s_j that will work for the elastic net.

Probability of Selection

Since $\boldsymbol{X}$ is fixed and ε is normal, it is straightforward to compute the probability of selection:

$$\Pr(\hat{\beta}_j \neq 0) = \Phi\left(\frac{\mu - \lambda_1}{\sigma}\right) + \Phi\left(\frac{-\mu - \lambda_1}{\sigma}\right).$$

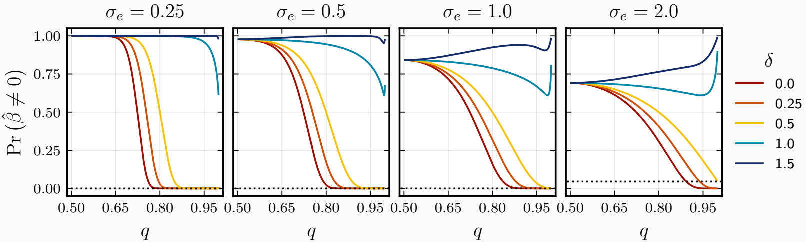


Figure 6: Probability that the elastic net selects a feature across different noise levels (σ_ε), types of normalization (δ), and class balance (q). The dashed line is asymptotic behavior for $\delta = 1/2$.

Asymptotic Results for Bias and Variance

Theorem

If $\boldsymbol{x}_j$ is a binary feature with class balance $q \in (0,1)$ and $\lambda_1, \lambda_2 \in (0, \infty)$, $\sigma_\varepsilon > 0$, and $s_j = (q - q^2)^\delta$, $\delta \geq 0$, then

$$\lim_{q \to 1^+} \mathrm{E}\, \hat{\beta}_j = \begin{cases} 0 & \textit{if } 0 \leq \delta < \frac{1}{2}, \\ \frac{2n\beta_j^*}{n+\lambda_2}\, \Phi\left(-\frac{\lambda_1}{\sigma_\varepsilon \sqrt{n}}\right) & \textit{if } \delta = \frac{1}{2}, \\ \beta_j^* & \textit{if } \delta \geq \frac{1}{2}. \end{cases}$$

Asymptotic Results for Bias and Variance

Theorem

If $\boldsymbol{x}_j$ is a binary feature with class balance $q \in (0,1)$ and $\lambda_1, \lambda_2 \in (0,\infty)$, $\sigma_\varepsilon > 0$, and $s_j = (q - q^2)^\delta$, $\delta \geq 0$, then

$$\lim_{q \to 1^+} \operatorname{E} \hat{\beta}_j = \begin{cases} 0 & \text{if } 0 \leq \delta < \frac{1}{2}, \\ \frac{2n\beta_j^*}{n+\lambda_2} \, \Phi\left(-\frac{\lambda_1}{\sigma_\varepsilon \sqrt{n}}\right) & \text{if } \delta = \frac{1}{2}, \\ \beta_j^* & \text{if } \delta \geq \frac{1}{2}. \end{cases}$$

and

$$\lim_{q \to 1^+} \operatorname{Var} \hat{\beta}_j = \begin{cases} 0 & \text{if } 0 \leq \delta < \frac{1}{2}, \\ \infty & \text{if } \delta \geq \frac{1}{2}. \end{cases}$$

A Bias–Variance Tradeoff

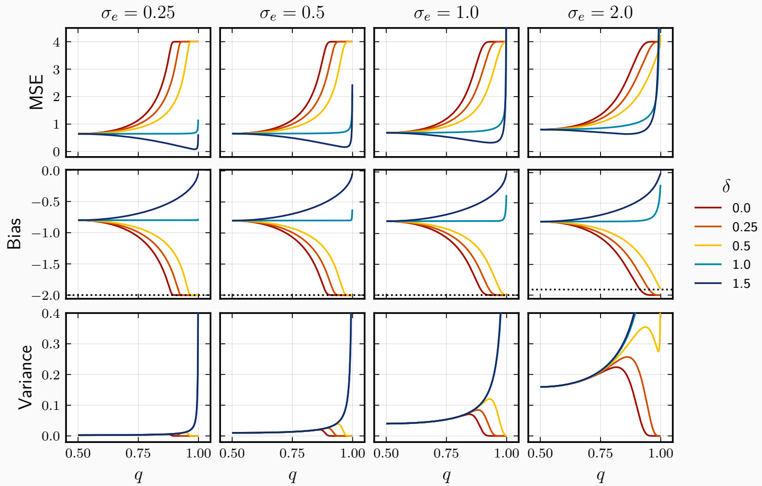
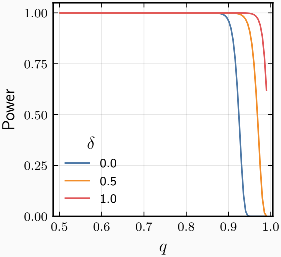


Figure 7: Bias, variance, and mean-squared error for a one-dimensional lasso problem. Theoretical result for orthogonal features. Dotted line is asymptotic result or $\delta = 1/2$.

Multiple Features: Power, FDR, and NMSE

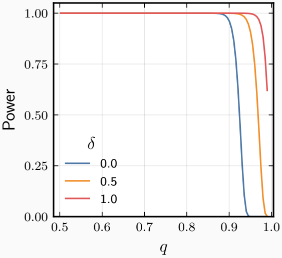
Lasso example with 10 true signals and varying q and p.



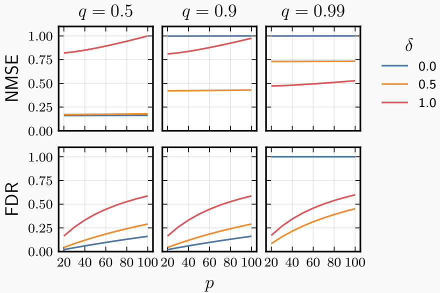
(a) Power in the sense of detecting all the true signals. Constant p.

Multiple Features: Power, FDR, and NMSE

Lasso example with 10 true signals and varying q and p.



(a) Power in the sense of detecting all the true signals. Constant p.



(b) False discovery rate (FDR) and normalized mean-squared error (NMSE).

Figure 8: Mean squared error (MSE), false discovery rate (FDR), and power.

Mixed Data

So far: all binary features. What about mixing binary and continuous (normal) features?

How to put binary features and normal features on the "same" scale?

Mixed Data

So far: all binary features. What about mixing binary and continuous (normal) features?

How to put binary features and normal features on the "same" scale?

Our Definition of Comparability

The effects of a binary feature and a normally distributed feature are comparable if a flip in the binary feature has the same effect as a two-standard deviation change in the normal feature (Gelman 2008).

Mixed Data

So far: all binary features. What about mixing binary and continuous (normal) features?

How to put binary features and normal features on the "same" scale?

Our Definition of Comparability

The effects of a binary feature and a normally distributed feature are comparable if a flip in the binary feature has the same effect as a two-standard deviation change in the normal feature (Gelman 2008).

Examples

Assume entries in $\boldsymbol{x}_1$ are binary and $\boldsymbol{x}_2$ come from a random variable X_2. The effects are comparable in the following cases:

- $X_2 \sim \mathrm{Normal}(\mu, 1/2)$, $\beta_1^* = 1$, and $\beta_2^* = 1$.
- $X_2 \sim \mathrm{Normal}(\mu, 2)$, $\beta_1^* = 1$, and $\beta_2^* = 0.25$.

Mixed Data

So far: all binary features. What about mixing binary and continuous (normal) features?

How to put binary features and normal features on the "same" scale?

Our Definition of Comparability

The effects of a binary feature and a normally distributed feature are comparable if a flip in the binary feature has the same effect as a two-standard deviation change in the normal feature (Gelman 2008).

Examples

Assume entries in $\boldsymbol{x}_1$ are binary and $\boldsymbol{x}_2$ come from a random variable X_2. The effects are comparable in the following cases:

- $X_2 \sim \mathrm{Normal}(\mu, 1/2)$, $\beta_1^* = 1$, and $\beta_2^* = 1$.
- $X_2 \sim \mathrm{Normal}(\mu, 2)$, $\beta_1^* = 1$, and $\beta_2^* = 0.25$.

Additional Scaling

To account for this, we need to invoke additional scaling.

Choice of Scaling in Mixed Data

For the two-standard deviation notion of comparability to hold, we need to modify our scaling factor s_j.

Choice of Scaling in Mixed Data

For the two-standard deviation notion of comparability to hold, we need to modify our scaling factor s_j.

As before, we assume that $\boldsymbol{x}_1$ is binary and $X_2 \sim \mathrm{Normal}(\mu, 1/2)$, $\beta_1^* = \beta_2^* = 1$ so that they have *comparable* effects. Also assume we standardize $\boldsymbol{x}_2$.

We want $\hat{\beta}_1 = \hat{\beta}_2$. That is,

$$\underbrace{\frac{\mathrm{S}_{\lambda_1}\left(\frac{n(q-q^2)}{s_j}\right)}{s_1\left(\frac{n(q-q^2)}{s_1^2} + \lambda_2\right)}}_{\hat{\beta}_1} = \underbrace{\frac{\mathrm{S}_{\lambda_1}\left(\frac{n}{2}\right)}{\frac{1}{2}\left(n + \lambda_2\right)}}_{\hat{\beta}_2}.$$

Choice of Scaling in Mixed Data

For the two-standard deviation notion of comparability to hold, we need to modify our scaling factor s_j.

As before, we assume that $\boldsymbol{x}_1$ is binary and $X_2 \sim \mathrm{Normal}(\mu, 1/2)$, $\beta_1^* = \beta_2^* = 1$ so that they have *comparable* effects. Also assume we standardize $\boldsymbol{x}_2$.

We want $\hat{\beta}_1 = \hat{\beta}_2$. That is,

$$\underbrace{\frac{\mathrm{S}_{\lambda_1}\left(\frac{n(q-q^2)}{s_j}\right)}{s_1\left(\frac{n(q-q^2)}{s_1^2} + \lambda_2\right)}}_{\hat{\beta}_1} = \underbrace{\frac{\mathrm{S}_{\lambda_1}\left(\frac{n}{2}\right)}{\frac{1}{2}\left(n + \lambda_2\right)}}_{\hat{\beta}_2}.$$

The choice $s_1 = (2(q - q^2))^\delta$ works when classes are balanced ($q = 0.5$). But no clear chocie for the elastic net case.

Experiments

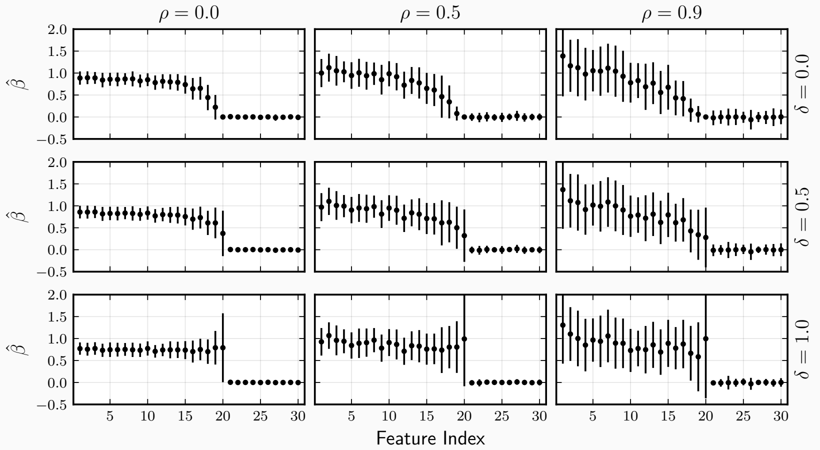


Figure 9: Lasso estimates for first 30 coefficients. First 20 features are true signals with a geometrically decreasing class balance from 0.5 to 0.99.

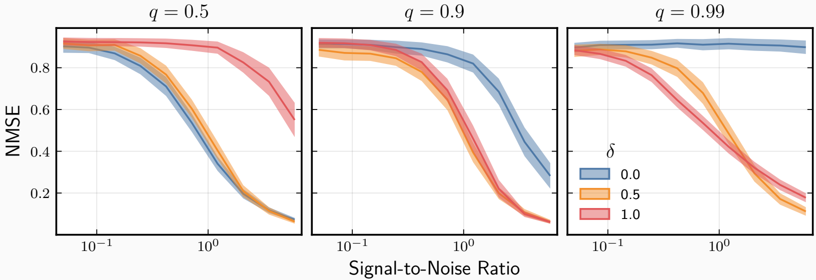


Figure 10: Normalized mean-squared test set error (NMSE).

Mixed Data

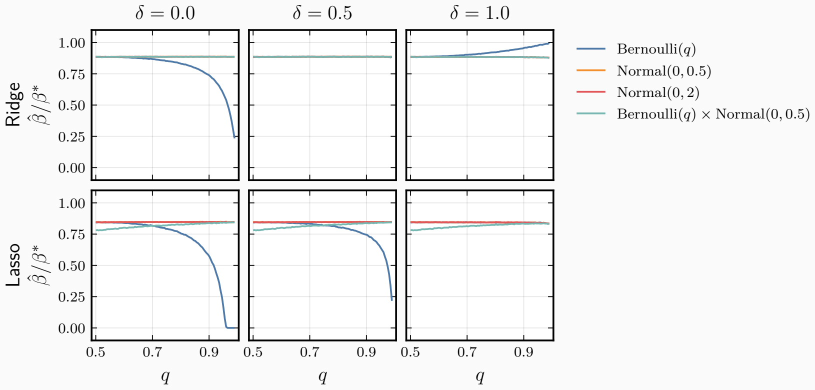


Figure 11: Comparison between lasso and ridge estimators for features generated to resemble features from various distributions.

Hyperparameter Optimization

Idea: The choice of δ affects the model, so let's optimize over it.

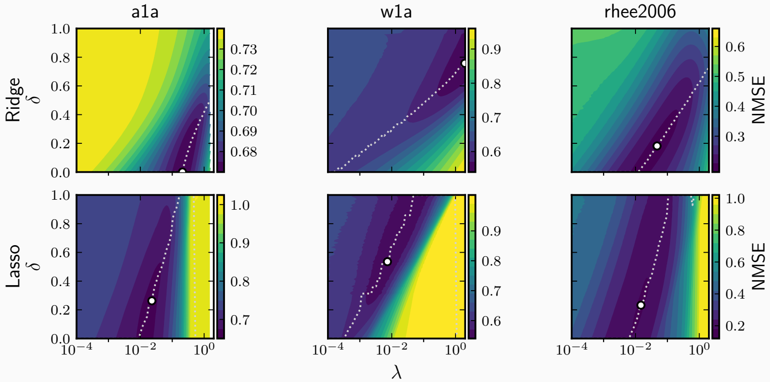


Figure 12: Contour plots of hold-out (validation set) error across a grid of δ and λ values for the lasso and ridge.

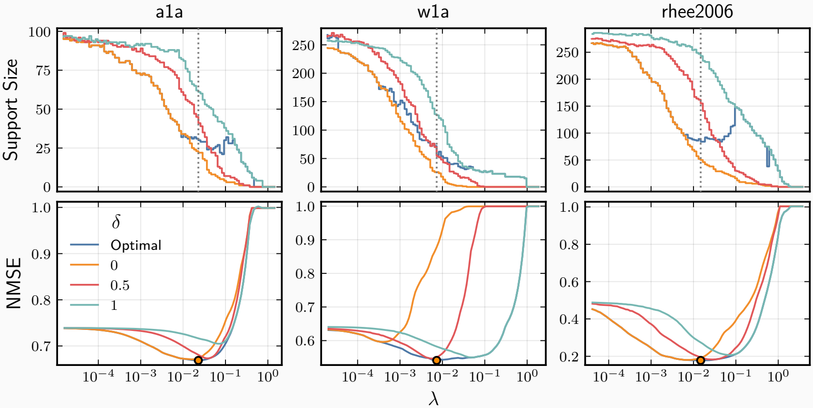


Figure 13: Support and NMSE of the lasso for different values of δ and λ.

Interaction Effects

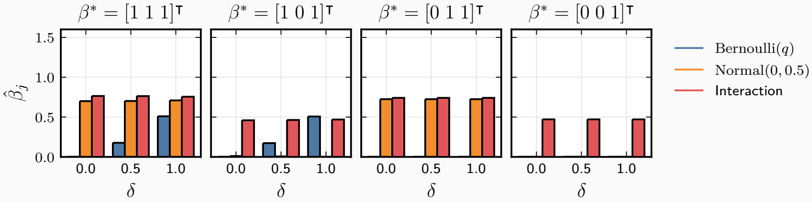


Figure 14: The effect of different normalization strategies for mixed data with interactions.

Interaction Effects

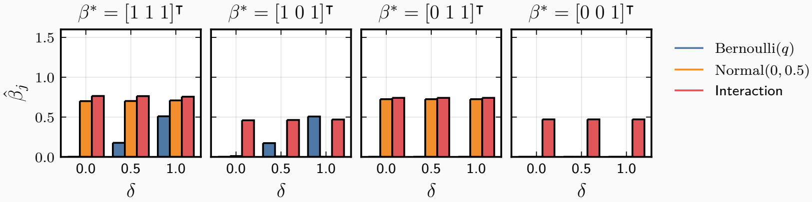


Figure 14: The effect of different normalization strategies for mixed data with interactions.

Open Questions

- How to deal with features with different locations?
- Should the interaction features be normalized conditionally?

Conclusions

Conclusions

- Class balance plays a crucial role for binary features.
- Effect depends on penalty
- Normalization mediates this effect at the cost of increased variance.
- Need to consider the notion of comparability between normal and binary features in mixed data.

Conclusions

Conclusions

- Class balance plays a crucial role for binary features.
- Effect depends on penalty
- Normalization mediates this effect at the cost of increased variance.
- Need to consider the notion of comparability between normal and binary features in mixed data.

Future Research

- Random $\boldsymbol{X}$
- Theory for $\boldsymbol{X}$ with correlation structure
- Non-Gaussian continuous features
- Other loss functions (GLMs, hinge loss, neural networks)
- Other penalties (group lasso, SCAD, MCP, SLOPE)

Thank you!

References i

El Ghaoui, Laurent, Vivian Viallon, and Tarek Rabbani (Sept. 21, 2010). *Safe Feature Elimination in Sparse Supervised Learning*. Technical report UCB/EECS-2010-126. Berkeley: EECS Department, University of California. URL: http://www.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-126.html.

Gelman, Andrew (July 10, 2008). "Scaling Regression Inputs by Dividing by Two Standard Deviations". In: *Statistics in Medicine* 27.15, pp. 2865–2873. ISSN: 02776715, 10970258. DOI: 10.1002/sim.3107. URL: https://onlinelibrary.wiley.com/doi/10.1002/sim.3107 (visited on 09/27/2023).

References ii

Tibshirani, Robert et al. (Mar. 2012). “Strong Rules for Discarding Predictors in Lasso-Type Problems”. In: *Journal of the Royal Statistical Society. Series B: Statistical Methodology* 74.2, pp. 245–266. ISSN: 1369-7412. DOI: 10/c4bb85. URL: https://iths.pure.elsevier.com/en/publications/strong-rules-for-discarding-predictors-in-lasso-type-problems (visited on 03/16/2018).

Zou, Hui and Trevor Hastie (2005). “Regularization and Variable Selection via the Elastic Net”. In: *Journal of the Royal Statistical Society. Series B (Statistical Methodology)* 67.2, pp. 301–320. ISSN: 1369-7412. URL: www.jstor.org/stable/3647580 (visited on 03/12/2018).

Extras

Max–Abs Scaling of Continuous Features

- Min–max normalization is sometimes used in continuous data
- Very sensitive to outliers
- But also depend on sample size!
- In other words, results in model validation with varying sample sizes can yield very strange results.

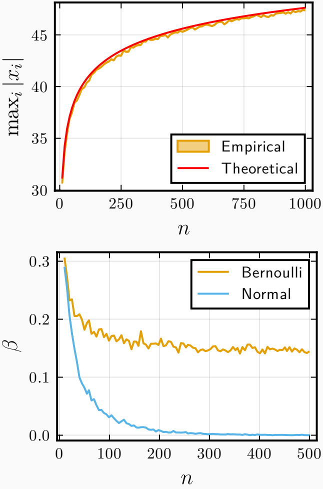


Figure 15: Effects of maximum absolute value scaling.