Understanding Ridge, Lasso, and Elastic Net Regression

David S. Rosenberg

Bloomberg ML EDU

February 7, 2018

Contents

Linear Regression

Linear Least Squares Regression

- Input space $\mathcal{X} = \mathbf{R}^d$
- Output space $\mathcal{Y} = \mathbf{R}$ (Regression problem)
- Training Data $\mathcal{D}_n = ((x_1, y_1), \ldots, (x_n, y_n))$ drawn i.i.d. from $\mathcal{P}_{\mathcal{X} \times \mathcal{Y}}$.

Linear Least Squares Regression

- Input space $\mathcal{X} = \mathbf{R}^d$
- Output space $\mathcal{Y} = \mathbf{R}$ (Regression problem)
- Training Data $\mathcal{D}_n = ((x_1, y_1), \ldots, (x_n, y_n))$ drawn i.i.d. from $\mathcal{P}_{\mathcal{X} \times \mathcal{Y}}$.

- Find $\hat{w} \in \mathbf{R}^d$ minimizing sum of squares error:

$$J(w) = \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i \right)^2.$$

Linear Least Squares Regression

- Input space $\mathcal{X} = \mathbf{R}^d$
- Output space $\mathcal{Y} = \mathbf{R}$ (Regression problem)
- Training Data $\mathcal{D}_n = ((x_1, y_1), \ldots, (x_n, y_n))$ drawn i.i.d. from $\mathcal{P}_{\mathcal{X} \times \mathcal{Y}}$.
- Find $\hat{w} \in \mathbf{R}^d$ minimizing sum of squares error:

$$J(w) = \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i \right)^2.$$

- Then prediction function is

$$f(x) = \hat{w}^T x,$$

  where $\hat{w}$ is a minimizer of $J(w)$.

Overfitting

- Issue: We can **overfit** if number of features d too large for data set size n.

Overfitting

- Issue: We can **overfit** if number of features d too large for data set size n.
- Control overfitting by "**regularization**"
  - fit the training data less well, in hopes that you'll generalize better

Constraint-Form Regularization (Ivanov Regularization)

- For complexity measure $\Omega : w \mapsto [0, \infty)$ and fixed $r \geqslant 0$, solve

$$\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i \right)^2$$
$$\text{s.t. } \Omega(w) \leqslant r.$$

Constraint-Form Regularization (Ivanov Regularization)

- For complexity measure $\Omega : w \mapsto [0, \infty)$ and fixed $r \geqslant 0$, solve

$$\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i \right)^2$$
$$\text{s.t. } \Omega(w) \leqslant r.$$

- Today we'll discuss 3 complexity measures:
  - $w \mapsto \|w\|^2 = w_1^2 + \cdots + w_d^2$ (ℓ^2 or "**ridge**" penalty)

Constraint-Form Regularization (Ivanov Regularization)

- For complexity measure $\Omega : w \mapsto [0, \infty)$ and fixed $r \geqslant 0$, solve

$$\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i \right)^2$$
$$\text{s.t. } \Omega(w) \leqslant r.$$

- Today we'll discuss 3 complexity measures:
  - $w \mapsto \|w\|^2 = w_1^2 + \cdots + w_d^2$ (ℓ^2 or "**ridge**" penalty)
  - $w \mapsto \|w\|_1 = |w_1| + \cdots + |w_d|$ (ℓ^1 or "**lasso**" penalty)

Constraint-Form Regularization (Ivanov Regularization)

- For complexity measure $\Omega : w \mapsto [0, \infty)$ and fixed $r \geqslant 0$, solve

$$\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i \right)^2$$
$$\text{s.t. } \Omega(w) \leqslant r.$$

- Today we'll discuss 3 complexity measures:
  - $w \mapsto \|w\|^2 = w_1^2 + \cdots + w_d^2$ (ℓ^2 or "**ridge**" penalty)
  - $w \mapsto \|w\|_1 = |w_1| + \cdots + |w_d|$ (ℓ^1 or "**lasso**" penalty)
  - $w \mapsto (1-\alpha)\|w\|^2 + \alpha\|w\|_1$ ("**elastic net**" penalty)

Regularization Paths and Lasso Sparsity

Ridge Regression (Constraint Form)

The ridge regression solution for complexity parameter $r \geqslant 0$ is

$$\hat{w} \in \underset{\|w\|_2^2 \leqslant r^2}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2.$$

Ridge Regression

Ridge Regression (Constraint Form)

The ridge regression solution for complexity parameter $r \geqslant 0$ is

$$\hat{w} \in \underset{\|w\|_2^2 \leqslant r^2}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2.$$

Ridge Regression (Penalty Form)

The ridge regression solution for regularization parameter $\lambda \geqslant 0$ is

$$\hat{w} \in \underset{w \in \mathbf{R}^d}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda \|w\|_2^2,$$

where $\|w\|_2^2 = w_1^2 + \cdots + w_d^2$ is the square of the ℓ_2-norm.

Ridge Regression

Ridge Regression (Constraint Form)

The ridge regression solution for complexity parameter $r \geqslant 0$ is

$$\hat{w} \in \underset{\|w\|_2^2 \leqslant r^2}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2.$$

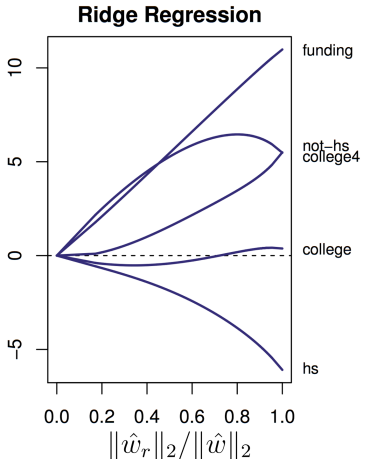
Ridge Regression (Penalty Form)

The ridge regression solution for regularization parameter $\lambda \geqslant 0$ is

$$\hat{w} \in \underset{w \in \mathbf{R}^d}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda \|w\|_2^2,$$

where $\|w\|_2^2 = w_1^2 + \cdots + w_d^2$ is the square of the ℓ_2-norm.

Ridge Regression: Regularization Path



$$\hat{w}_r = \arg\min_{\|w\|_2^2 \leq r^2} \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i\right)^2$$

$$\hat{w} = \hat{w}_\infty = \text{Unconstrained ERM}$$

- For $r = 0$, $\|\hat{w}_r\|_2/\|\hat{w}\|_2 = 0$.
- For $r = \infty$, $\|\hat{w}_r\|_2/\|\hat{w}\|_2 = 1$

Modified from Hastie, Tibshirani, and Wainwright's *Statistical Learning with Sparsity*, Fig 2.1. About predicting crime in 50 US cities.

Lasso Regression (Constraint Form)

The lasso regression solution for complexity parameter $r \geqslant 0$ is

$$\hat{w} = \underset{\|w\|_1 \leqslant r}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2.$$

Lasso Regression

Lasso Regression (Constraint Form)

The lasso regression solution for complexity parameter $r \geqslant 0$ is

$$\hat{w} = \underset{\|w\|_1 \leqslant r}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2.$$

Ridge Regression (Penalty Form)

The ridge regression solution for regularization parameter $\lambda \geqslant 0$ is

$$\hat{w} = \underset{w \in \mathbf{R}^d}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda \|w\|_2^2,$$

where $\|w\|_2^2 = w_1^2 + \cdots + w_d^2$ is the square of the ℓ_2-norm.

Lasso Regression

Lasso Regression (Constraint Form)

The lasso regression solution for complexity parameter $r \geqslant 0$ is

$$\hat{w} = \underset{\|w\|_1 \leqslant r}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2.$$

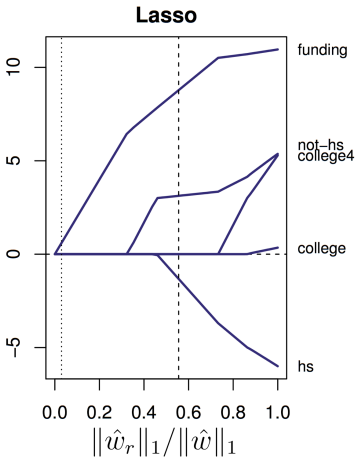
Ridge Regression (Penalty Form)

The ridge regression solution for regularization parameter $\lambda \geqslant 0$ is

$$\hat{w} = \underset{w \in \mathbf{R}^d}{\arg\min} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda \|w\|_2^2,$$

where $\|w\|_2^2 = w_1^2 + \cdots + w_d^2$ is the square of the ℓ_2-norm.

Lasso Regression: Regularization Path



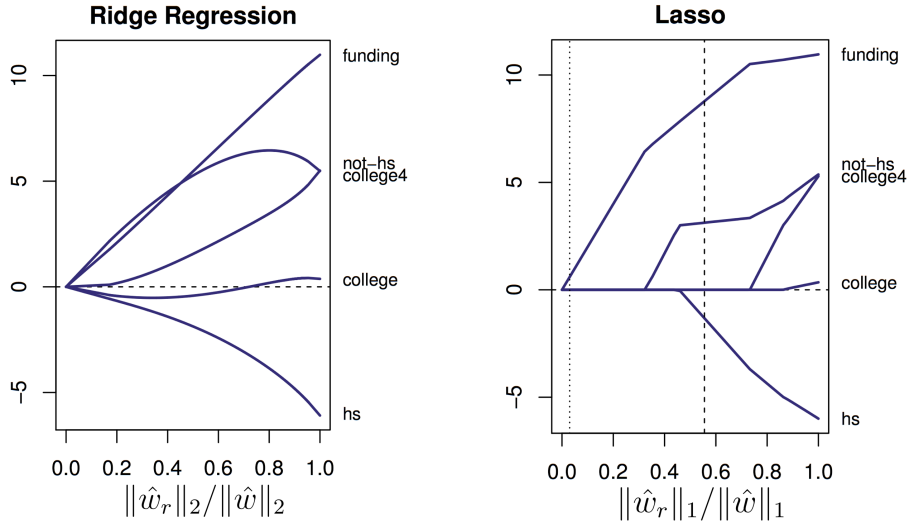
$$\hat{w}_r = \arg\min_{\|w\|_1 \le r} \frac{1}{n} \sum_{i=1}^{n} \left(w^T x_i - y_i\right)^2$$

$$\hat{w} = \hat{w}_\infty = \text{Unconstrained ERM}$$

- For $r = 0$, $\|\hat{w}_r\|_1/\|\hat{w}\|_1 = 0$.
- For $r = \infty$, $\|\hat{w}_r\|_1/\|\hat{w}\|_1 = 1$

Modified from Hastie, Tibshirani, and Wainwright's *Statistical Learning with Sparsity*, Fig 2.1. About predicting crime in 50 US cities.

Ridge vs. Lasso: Regularization Paths



Modified from Hastie, Tibshirani, and Wainwright's *Statistical Learning with Sparsity*, Fig 2.1. About predicting crime in 50 US cities.

Why does Lasso regression give sparse solutions?

Parameter Space

- Illustrate affine prediction functions in parameter space.

The ℓ_1 and ℓ_2 Norm Constraints

- For visualization, restrict to 2-dimensional input space
- $\mathcal{F} = \{f(x) = w_1 x_1 + w_2 x_2\}$ (linear hypothesis space)

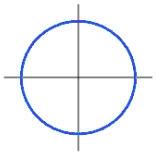
The ℓ_1 and ℓ_2 Norm Constraints

- For visualization, restrict to 2-dimensional input space
- $\mathcal{F} = \{f(x) = w_1 x_1 + w_2 x_2\}$ (linear hypothesis space)
- Represent $\mathcal{F}$ by $\left\{(w_1, w_2) \in \mathbf{R}^2\right\}$.

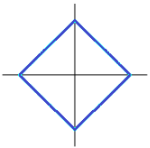
The ℓ_1 and ℓ_2 Norm Constraints

- For visualization, restrict to 2-dimensional input space
- $\mathcal{F} = \{f(x) = w_1 x_1 + w_2 x_2\}$ (linear hypothesis space)
- Represent $\mathcal{F}$ by $\left\{(w_1, w_2) \in \mathbf{R}^2\right\}$.

- ℓ_2 contour: $w_1^2 + w_2^2 = r$



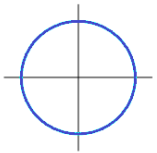
- ℓ_1 contour: $|w_1| + |w_2| = r$



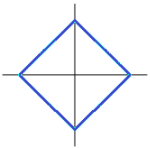
The ℓ_1 and ℓ_2 Norm Constraints

- For visualization, restrict to 2-dimensional input space
- $\mathcal{F} = \{f(x) = w_1 x_1 + w_2 x_2\}$ (linear hypothesis space)
- Represent $\mathcal{F}$ by $\left\{(w_1, w_2) \in \mathbf{R}^2\right\}$.

- ℓ_2 contour: $w_1^2 + w_2^2 = r$



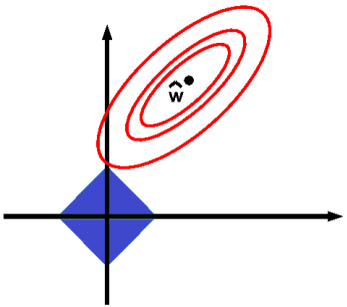
- ℓ_1 contour: $|w_1| + |w_2| = r$



Where are the "sparse" solutions?

The Famous Picture for ℓ_1 Regularization

- $f_r^* = \arg\min_{w \in \mathbf{R}^2} \frac{1}{n} \sum_{i=1}^n \left(w^T x_i - y_i\right)^2$ subject to $|w_1| + |w_2| \leqslant r$



- Blue region: Area satisfying complexity constraint: $|w_1| + |w_2| \leqslant r$
- Red lines: contours of $\hat{R}_n(w) = \sum_{i=1}^n \left(w^T x_i - y_i\right)^2$.

KPM Fig. 13.3

The Average Loss on a Training Set

- Denote the average training loss or "**empirical risk**" of $f(x) = w^T x$ by

$$\hat{R}_n(w) = \frac{1}{n}\|Xw - y\|^2,$$

where X is the **design matrix** (examples in rows, features in columns).

The Average Loss on a Training Set

- Denote the average training loss or "**empirical risk**" of $f(x) = w^T x$ by

$$\hat{R}_n(w) = \frac{1}{n}\|Xw - y\|^2,$$

  where X is the **design matrix** (examples in rows, features in columns).
- $\hat{R}_n$ is minimized by $\hat{w} = \left(X^T X\right)^{-1} X^T y$, the OLS solution. (For X full rank.)

The Average Loss on a Training Set

- Denote the average training loss or "**empirical risk**" of $f(x) = w^T x$ by

$$\hat{R}_n(w) = \frac{1}{n}\|Xw - y\|^2,$$

  where X is the **design matrix** (examples in rows, features in columns).
- $\hat{R}_n$ is minimized by $\hat{w} = \left(X^T X\right)^{-1} X^T y$, the OLS solution. (For X full rank.)
- So $\hat{R}_n(w)$ gets bigger as w moves away from $\hat{w}$.

The Average Loss on a Training Set

- Denote the average training loss or "**empirical risk**" of $f(x) = w^T x$ by

$$\hat{R}_n(w) = \frac{1}{n}\|Xw - y\|^2,$$

  where X is the **design matrix** (examples in rows, features in columns).
- $\hat{R}_n$ is minimized by $\hat{w} = \left(X^T X\right)^{-1} X^T y$, the OLS solution. (For X full rank.)
- So $\hat{R}_n(w)$ gets bigger as w moves away from $\hat{w}$.
- **Level sets** of $\hat{R}_n(w)$ (i.e. all w's achieving the same training error) are **ellipsoids**:

$$\left\{ w \mid (w - \hat{w})^T X^T X (w - \hat{w}) = k \right\}.$$

Level Sets are Ellipsoids (A few more details)

- By "completing the square", we can show for any $w \in \mathbf{R}^d$:

$$\hat{R}_n(w) = \frac{1}{n}(w - \hat{w})^T X^T X (w - \hat{w}) + \hat{R}_n(\hat{w})$$

Level Sets are Ellipsoids (A few more details)

- By "completing the square", we can show for any $w \in \mathbf{R}^d$:

$$\hat{R}_n(w) = \frac{1}{n}(w - \hat{w})^T X^T X (w - \hat{w}) + \hat{R}_n(\hat{w})$$

- Set of w with $\hat{R}_n(w)$ exceeding $\hat{R}_n(\hat{w})$ by $c > 0$ is

$$\left\{ w \mid \hat{R}_n(w) = c + \hat{R}_n(\hat{w}) \right\} = \left\{ w \mid (w - \hat{w})^T X^T X (w - \hat{w}) = nc \right\},$$

  which is an **ellipsoid centered at** $\hat{w}$.

Properties of Ellipsoids

- **Level sets** of $\hat{R}_n(w)$ (i.e. all w's achieving the same training error) are **ellipsoids**:

$$\left\{ w \mid (w - \hat{w})^T X^T X (w - \hat{w}) = k \right\}.$$

Properties of Ellipsoids

- **Level sets** of $\hat{R}_n(w)$ (i.e. all w's achieving the same training error) are **ellipsoids**:

$$\left\{ w \mid (w - \hat{w})^T X^T X (w - \hat{w}) = k \right\}.$$

- The **principal axes** of the ellipsoid are in the directions of the **eigenvectors** of $X^T X$.

Properties of Ellipsoids

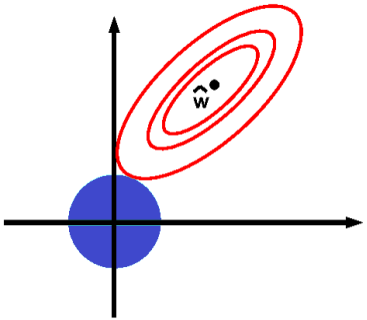
- **Level sets** of $\hat{R}_n(w)$ (i.e. all w's achieving the same training error) are **ellipsoids**:

$$\left\{ w \mid (w - \hat{w})^T X^T X (w - \hat{w}) = k \right\}.$$

- The **principal axes** of the ellipsoid are in the directions of the **eigenvectors** of $X^T X$.
- If λ is the eigenvalue for eigenvector v, then corresponding radius is $1/\sqrt{\lambda}$.

The Famous Picture for ℓ_2 Regularization

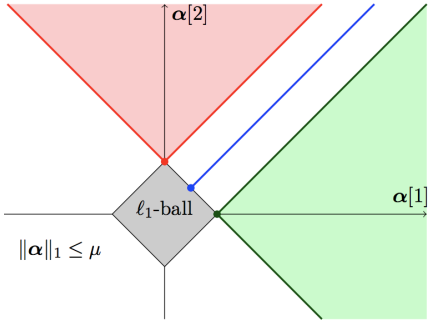
- $f_r^* = \arg\min_{w \in \mathbf{R}^2} \sum_{i=1}^n \left(w^T x_i - y_i\right)^2$ subject to $w_1^2 + w_2^2 \leqslant r$



- Blue region: Area satisfying complexity constraint: $w_1^2 + w_2^2 \leqslant r$
- Red lines: contours of $\hat{R}_n(w) = \sum_{i=1}^n \left(w^T x_i - y_i\right)^2$.

KPM Fig. 13.3

Why are Lasso Solutions Often Sparse?



- Suppose design matrix X is orthogonal, so $X^T X = I$, and contours are circles.
- Then OLS solution in green or red regions implies ℓ_1 constrained solution will be at corner

Fig from Mairal et al.'s Sparse Modeling for Image and Vision Processing Fig 1.6

Repeated Features

A Very Simple Model

- Suppose we have one feature $x_1 \in \mathbf{R}$.
- Response variable $y \in \mathbf{R}$.

A Very Simple Model

- Suppose we have one feature $x_1 \in \mathbf{R}$.
- Response variable $y \in \mathbf{R}$.
- Got some data and ran least squares linear regression.
- The ERM is

$$\hat{f}(x_1) = 4x_1.$$

A Very Simple Model

- Suppose we have one feature $x_1 \in \mathbf{R}$.
- Response variable $y \in \mathbf{R}$.
- Got some data and ran least squares linear regression.
- The ERM is

$$\hat{f}(x_1) = 4x_1.$$

- What happens if we get a new feature x_2,
  - but we always have $x_2 = x_1$?

Duplicate Features

- New feature x_2 gives no new information.

Duplicate Features

- New feature x_2 gives no new information.
- ERM is still

$$\hat{f}(x_1, x_2) = 4x_1.$$

- Now there are some more ERMs:

$$\begin{aligned} \hat{f}(x_1, x_2) &= 2x_1 + 2x_2 \\ \hat{f}(x_1, x_2) &= x_1 + 3x_2 \\ \hat{f}(x_1, x_2) &= 4x_2 \end{aligned}$$

Duplicate Features

- New feature x_2 gives no new information.
- ERM is still

$$\hat{f}(x_1, x_2) = 4x_1.$$

- Now there are some more ERMs:

$$\begin{aligned} \hat{f}(x_1, x_2) &= 2x_1 + 2x_2 \\ \hat{f}(x_1, x_2) &= x_1 + 3x_2 \\ \hat{f}(x_1, x_2) &= 4x_2 \end{aligned}$$

- What if we introduce ℓ_1 or ℓ_2 regularization?

Duplicate Features: ℓ_1 and ℓ_2 norms

- $\hat{f}(x_1, x_2) = w_1 x_1 + w_2 x_2$ is an ERM iff $w_1 + w_2 = 4$.

Duplicate Features: ℓ_1 and ℓ_2 norms

- $\hat{f}(x_1, x_2) = w_1 x_1 + w_2 x_2$ is an ERM iff $w_1 + w_2 = 4$.
- Consider the ℓ_1 and ℓ_2 norms of various solutions:

w_1	w_2	$\|w\|_1$	$\|w\|_2^2$
4	0	**4**	16
2	2	**4**	**8**
1	3	**4**	10
-1	5	6	26

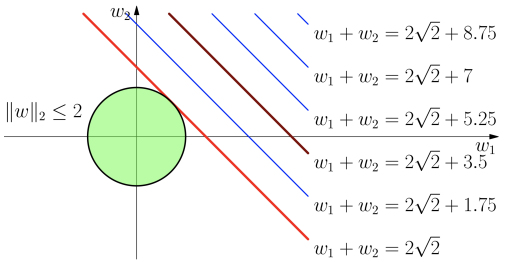
Duplicate Features: ℓ_1 and ℓ_2 norms

- $\hat{f}(x_1, x_2) = w_1 x_1 + w_2 x_2$ is an ERM iff $w_1 + w_2 = 4$.
- Consider the ℓ_1 and ℓ_2 norms of various solutions:

w_1	w_2	$\|w\|_1$	$\|w\|_2^2$
4	0	**4**	16
2	2	**4**	**8**
1	3	**4**	10
-1	5	6	26

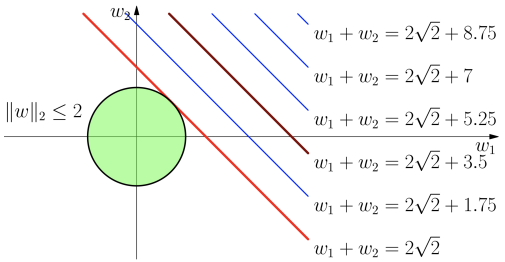
- $\|w\|_1$ doesn't discriminate, as long as all have same sign
- $\|w\|_2^2$ minimized when weight is spread equally
- Picture proof: Level sets of loss are lines of the form $w_1 + w_2 = 4$...

Equal Features, ℓ_2 Constraint



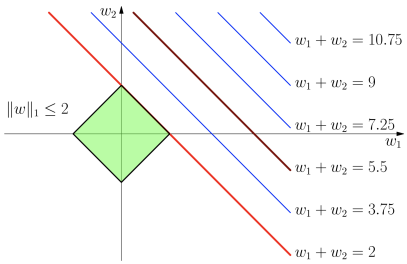
- Suppose the line $w_1 + w_2 = 2\sqrt{2} + 3.5$ corresponds to the empirical risk minimizers.

Equal Features, ℓ_2 Constraint



- Suppose the line $w_1 + w_2 = 2\sqrt{2} + 3.5$ corresponds to the empirical risk minimizers.
- Empirical risk increase as we move away from these parameter settings
- Intersection of $w_1 + w_2 = 2\sqrt{2}$ and the norm ball $\|w\|_2 \leqslant 2$ is ridge solution.
- Note that $w_1 = w_2$ at the solution

Equal Features, ℓ_1 Constraint



- Suppose the line $w_1 + w_2 = 5.5$ corresponds to the empirical risk minimizers.
- Intersection of $w_1 + w_2 = 2$ and the norm ball $\|w\|_1 \leqslant 2$ is lasso solution.
- Note that the solution set is $\{(w_1, w_2) : w_1 + w_2 = 2, w_1, w_2 \geqslant 0\}$.

Linearly Dependent Features

Linearly Related Features

- Linear prediction functions: $f(x) = w_1 x_2 + w_2 x_2$
- Same setup, now suppose $x_2 = 2x_1$.

Linearly Related Features

- Linear prediction functions: $f(x) = w_1 x_2 + w_2 x_2$
- Same setup, now suppose $x_2 = 2x_1$.
- Then all functions with $w_1 + 2w_2 = k$ are the same.
  - give same predictions and have same empirical risk

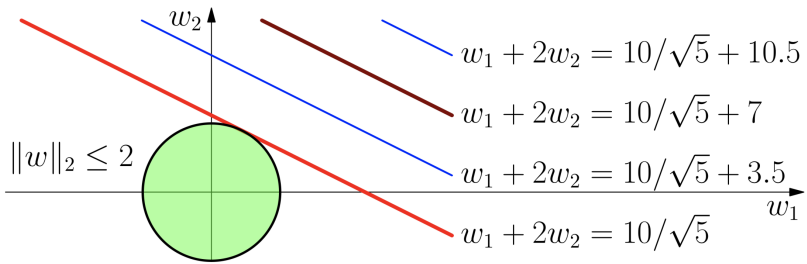
Linearly Related Features

- Linear prediction functions: $f(x) = w_1 x_2 + w_2 x_2$
- Same setup, now suppose $x_2 = 2x_1$.
- Then all functions with $w_1 + 2w_2 = k$ are the same.
  - give same predictions and have same empirical risk
- What function will we select if we do ERM with ℓ_1 or ℓ_2 constraint?

Linearly Related Features

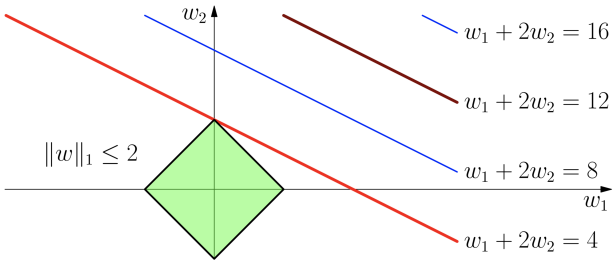
- Linear prediction functions: $f(x) = w_1 x_2 + w_2 x_2$
- Same setup, now suppose $x_2 = 2x_1$.
- Then all functions with $w_1 + 2w_2 = k$ are the same.
  - give same predictions and have same empirical risk
- What function will we select if we do ERM with ℓ_1 or ℓ_2 constraint?
- Compare a solution that just uses w_1 to a solution that just uses w_2...

Linearly Related Features, ℓ_2 Constraint



- $w_1 + 2w_2 = 10/\sqrt{5} + 7$ corresponds to the empirical risk minimizers.
- Intersection of $w_1 + 2w_2 = 10\sqrt{5}$ and the norm ball $\|w\|_2 \leqslant 2$ is ridge solution.
- At solution, $w_2 = 2w_1$.

Linearly Related Features, ℓ_1 Constraint



- Intersection of $w_1 + 2w_2 = 4$ and the norm ball $\|w\|_1 \leqslant 2$ is lasso solution.
- Solution is now a corner of the ℓ_1 ball, corresonding to a sparse solution.

Linearly Dependent Features: Take Away

- For identical features
  - ℓ_1 regularization spreads weight arbitrarily (all weights same sign)
  - ℓ_2 regularization spreads weight evenly
- Linearly related features
  - ℓ_1 regularization chooses variable with larger scale, 0 weight to others
  - ℓ_2 prefers variables with larger scale – spreads weight proportional to scale

Correlated Features

Correlated Features – Same Scale

- Suppose x_1 and x_2 are highly correlated and the same scale.
- This is quite typical in real data, after normalizing data.

Correlated Features – Same Scale

- Suppose x_1 and x_2 are highly correlated and the same scale.
- This is quite typical in real data, after normalizing data.
- Nothing degenerate here, so level sets are ellipsoids.

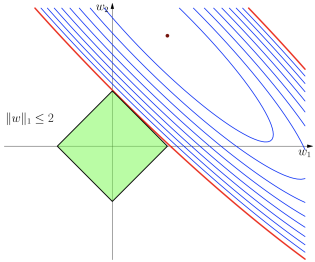
Correlated Features – Same Scale

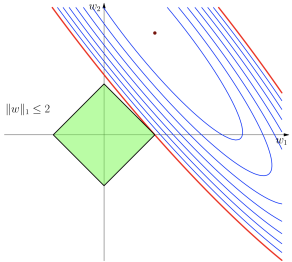
- Suppose x_1 and x_2 are highly correlated and the same scale.
- This is quite typical in real data, after normalizing data.
- Nothing degenerate here, so level sets are ellipsoids.
- But, the higher the correlation, the smaller the eigenvalues (and the closer to degenerate we get).

Correlated Features – Same Scale

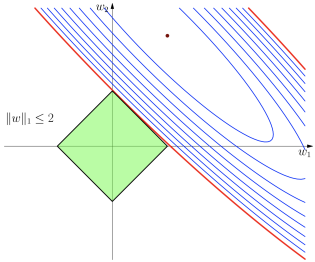
- Suppose x_1 and x_2 are highly correlated and the same scale.
- This is quite typical in real data, after normalizing data.
- Nothing degenerate here, so level sets are ellipsoids.
- But, the higher the correlation, the smaller the eigenvalues (and the closer to degenerate we get).
- That is, ellipses keep stretching out, getting closer to two parallel lines.

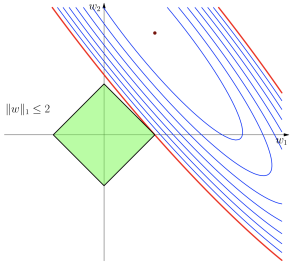
Correlated Features, ℓ_1 Regularization





Correlated Features, ℓ_1 Regularization





- Intersection could be anywhere on the top right edge.
- Minor perturbations (in data) can drastically change intersection point – very unstable solution.
- Makes division of weight among highly correlated features (of same scale) seem arbitrary.
  - If $x_1 \approx 2x_2$, ellipse changes orientation and we hit a corner. (Which one?)

The Case Against Sparsity

A Case Against Sparsity

- Suppose there's some unknown value $\theta \in \mathbf{R}$.

A Case Against Sparsity

- Suppose there's some unknown value $\theta \in \mathbf{R}$.
- We get 3 noisy observations of θ:

$$x_1, x_2, x_3 \sim \mathcal{N}(\theta, 1) \text{ (i.i.d)}$$

A Case Against Sparsity

- Suppose there's some unknown value $\theta \in \mathbf{R}$.
- We get 3 noisy observations of θ:

$$x_1, x_2, x_3 \sim \mathcal{N}(\theta, 1) \text{ (i.i.d)}$$

- What's a good estimator $\hat{\theta}$ for θ?

A Case Against Sparsity

- Suppose there's some unknown value $\theta \in \mathbf{R}$.
- We get 3 noisy observations of θ:

$$x_1, x_2, x_3 \sim \mathcal{N}(\theta, 1) \text{ (i.i.d)}$$

- What's a good estimator $\hat{\theta}$ for θ?
- Would you prefer $\hat{\theta} = x_1$ or $\hat{\theta} = \frac{1}{3}(x_1 + x_2 + x_3)$?

Estimator Performance Analysis

- $\mathbb{E}[x_1] = \theta$ and $\mathbb{E}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \theta$. So both unbiased.

Estimator Performance Analysis

- $\mathbb{E}[x_1] = \theta$ and $\mathbb{E}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \theta$. So both unbiased.
- $\mathrm{Var}[x_1] =$

Estimator Performance Analysis

- $\mathbb{E}[x_1] = \theta$ and $\mathbb{E}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \theta$. So both unbiased.
- $\text{Var}[x_1] = 1$.

Estimator Performance Analysis

- $\mathbb{E}[x_1] = \theta$ and $\mathbb{E}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \theta$. So both unbiased.
- $\text{Var}[x_1] = 1$.
- $\text{Var}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] =$

Estimator Performance Analysis

- $\mathbb{E}[x_1] = \theta$ and $\mathbb{E}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \theta$. So both unbiased.
- $\text{Var}[x_1] = 1$.
- $\text{Var}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \frac{1}{9}(1+1+1) = \frac{1}{3}$.

Estimator Performance Analysis

- $\mathbb{E}[x_1] = \theta$ and $\mathbb{E}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \theta$. So both unbiased.
- $\text{Var}[x_1] = 1$.
- $\text{Var}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \frac{1}{9}(1+1+1) = \frac{1}{3}$.
- Average has a smaller variance — the independent errors cancel each other out.

Estimator Performance Analysis

- $\mathbb{E}[x_1] = \theta$ and $\mathbb{E}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \theta$. So both unbiased.
- $\text{Var}[x_1] = 1$.
- $\text{Var}\left[\frac{1}{3}(x_1 + x_2 + x_3)\right] = \frac{1}{9}(1+1+1) = \frac{1}{3}$.
- Average has a smaller variance — the independent errors cancel each other out.
- Similar thing happens in regression with correlated features:
  - e.g. If 3 features are correlated, we could keep just one of them.
  - But we can potentially do better by using all 3.

Example with highly correlated features

- Model in words:
  - y is some unknown linear combination of z_1 and z_2.
  - But we don't observe z_1 and z_2 directly.

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Model in words:
  - y is some unknown linear combination of z_1 and z_2.
  - But we don't observe z_1 and z_2 directly.
  - We get 3 noisy observations of z_1, call them x_1, x_2, x_3.
  - We get 3 noisy observations of z_2, call them x_4, x_5, x_6.

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Model in words:
  - y is some unknown linear combination of z_1 and z_2.
  - But we don't observe z_1 and z_2 directly.
  - We get 3 noisy observations of z_1, call them x_1, x_2, x_3.
  - We get 3 noisy observations of z_2, call them x_4, x_5, x_6.
- We want to predict y from our noisy observations.

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Model in words:
  - y is some unknown linear combination of z_1 and z_2.
  - But we don't observe z_1 and z_2 directly.
  - We get 3 noisy observations of z_1, call them x_1, x_2, x_3.
  - We get 3 noisy observations of z_2, call them x_4, x_5, x_6.
- We want to predict y from our noisy observations.
- That is, we want an estimator $\hat{y} = f(x_1, x_2, x_3, x_4, x_5, x_6)$ for estimating y.

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Suppose (x, y) generated as follows:

$$\begin{aligned} z_1, z_2 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ \varepsilon_0, \varepsilon_1, \ldots, \varepsilon_6 &\sim \mathcal{N}(0,1) \text{ (independent)} \end{aligned}$$

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Suppose (x, y) generated as follows:

$$\begin{aligned} z_1, z_2 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ \varepsilon_0, \varepsilon_1, \ldots, \varepsilon_6 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ y &= 3z_1 - 1.5z_2 + 2\varepsilon_0 \end{aligned}$$

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Suppose (x, y) generated as follows:

$$
\begin{aligned}
z_1, z_2 &\sim \mathcal{N}(0,1) \text{ (independent)} \\
\varepsilon_0, \varepsilon_1, \ldots, \varepsilon_6 &\sim \mathcal{N}(0,1) \text{ (independent)} \\
y &= 3z_1 - 1.5z_2 + 2\varepsilon_0 \\
x_j &= \begin{cases} z_1 + \varepsilon_j/5 & \text{for } j = 1,2,3 \\ z_2 + \varepsilon_j/5 & \text{for } j = 4,5,6 \end{cases}
\end{aligned}
$$

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Suppose (x, y) generated as follows:

$$\begin{aligned} z_1, z_2 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ \varepsilon_0, \varepsilon_1, \ldots, \varepsilon_6 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ y &= 3z_1 - 1.5z_2 + 2\varepsilon_0 \\ x_j &= \begin{cases} z_1 + \varepsilon_j/5 & \text{for } j = 1,2,3 \\ z_2 + \varepsilon_j/5 & \text{for } j = 4,5,6 \end{cases} \end{aligned}$$

- Generated a sample of $((x_1, \ldots, x_6), y)$ pairs of size $n = 100$.

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Suppose (x, y) generated as follows:

$$\begin{aligned} z_1, z_2 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ \varepsilon_0, \varepsilon_1, \ldots, \varepsilon_6 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ y &= 3z_1 - 1.5z_2 + 2\varepsilon_0 \\ x_j &= \begin{cases} z_1 + \varepsilon_j/5 & \text{for } j = 1,2,3 \\ z_2 + \varepsilon_j/5 & \text{for } j = 4,5,6 \end{cases} \end{aligned}$$

- Generated a sample of $((x_1, \ldots, x_6), y)$ pairs of size $n = 100$.
- That is, we want an estimator $\hat{y} = f(x_1, x_2, x_3, x_4, x_5, x_6)$ that is good for estimating y.

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Suppose (x, y) generated as follows:

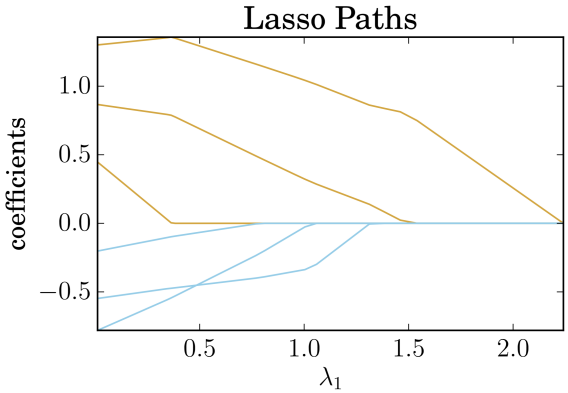
$$\begin{aligned} z_1, z_2 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ \varepsilon_0, \varepsilon_1, \ldots, \varepsilon_6 &\sim \mathcal{N}(0,1) \text{ (independent)} \\ y &= 3z_1 - 1.5z_2 + 2\varepsilon_0 \\ x_j &= \begin{cases} z_1 + \varepsilon_j/5 & \text{for } j = 1,2,3 \\ z_2 + \varepsilon_j/5 & \text{for } j = 4,5,6 \end{cases} \end{aligned}$$

- Generated a sample of $((x_1, \ldots, x_6), y)$ pairs of size $n = 100$.
- That is, we want an estimator $\hat{y} = f(x_1, x_2, x_3, x_4, x_5, x_6)$ that is good for estimating y.
- **High feature correlation**: Correlations within the groups of x's is around 0.97.

Example from Section 4.2 in Hastie et al's *Statistical Learning with Sparsity*.

Example with highly correlated features

- Lasso regularization paths:



- Lines with the same color correspond to features with essentially the same information
- Distribution of weight among them seems almost arbitrary

Hedge Bets When Variables Highly Correlated

- When variables are highly correlated (and same scale – assume we've standardized features),
  - we want to give them roughly the same weight.

Hedge Bets When Variables Highly Correlated

- When variables are highly correlated (and same scale – assume we've standardized features),
  - we want to give them roughly the same weight.
- Why?
  - Let their errors cancel out

Hedge Bets When Variables Highly Correlated

- When variables are highly correlated (and same scale – assume we've standardized features),
  - we want to give them roughly the same weight.
- Why?
  - Let their errors cancel out
- How can we get the weight spread more evenly?

Elastic Net

Elastic Net

- The **elastic net** combines lasso and ridge penalties:

$$\hat{w} = \arg\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda_1 \|w\|_1 + \lambda_2 \|w\|_2^2$$

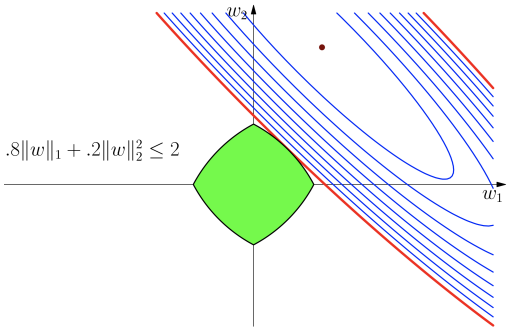
Elastic Net

- The **elastic net** combines lasso and ridge penalties:

$$\hat{w} = \arg\min_{w \in \mathbf{R}^d} \frac{1}{n} \sum_{i=1}^{n} \left\{ w^T x_i - y_i \right\}^2 + \lambda_1 \|w\|_1 + \lambda_2 \|w\|_2^2$$

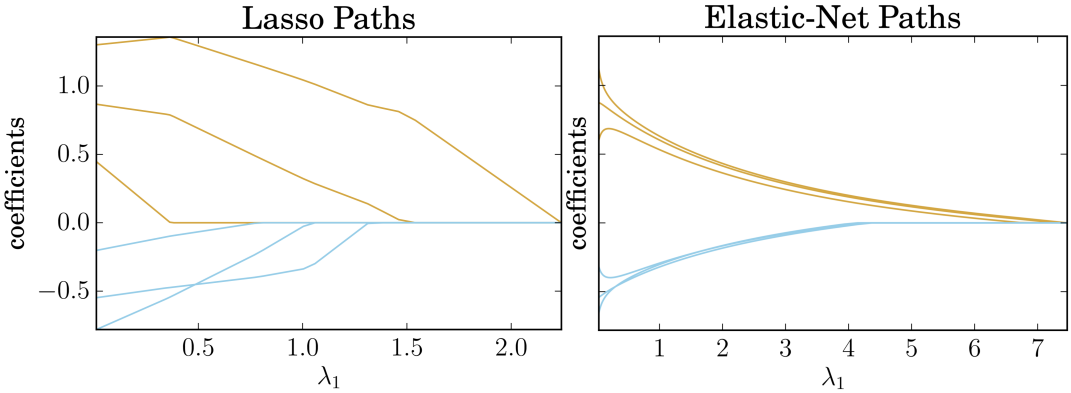
- We expect correlated random variables to have similar coefficients.

Highly Correlated Features, Elastic Net Constraint



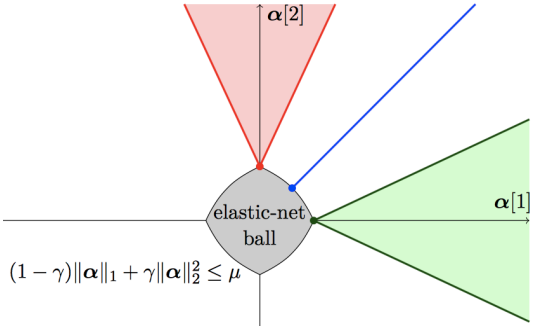
- Elastic net solution is closer to $w_2 = w_1$ line, despite high correlation.

Elastic Net Results on Model



- Lasso on left; Elastic net on right.
- Ratio of ℓ_2 to ℓ_1 regularization roughly $2:1$.

Elastic Net - "Sparse Regions"



- Suppose design matrix X is orthogonal, so $X^T X = I$, and contours are circles (and features uncorrelated)
- Then OLS solution in green or red regions implies elastic-net constrained solution will be at corner

Fig from Mairal et al.'s Sparse Modeling for Image and Vision Processing Fig 1.9

Elastic Net – A Theorem for Correlated Variables

Theorem

Let $\rho_{ij} = \widehat{corr}(x_i, x_j)$. *Suppose features* $x_1, \ldots, x_d$ *are standardized and* $\hat{w}_i$ *and* $\hat{w}_j$ *are selected by elastic net, with* $\hat{w}_i \hat{w}_j > 0$. *Then*

$$|\hat{w}_i - \hat{w}_j| \leqslant \frac{\|y\|_2 \sqrt{2}}{\sqrt{n}\lambda_2} \sqrt{1 - \rho_{ij}}.$$

Elastic Net – A Theorem for Correlated Variables

Theorem

Let $\rho_{ij} = \widehat{corr}(x_i, x_j)$. Suppose features $x_1, \ldots, x_d$ are standardized and $\hat{w}_i$ and $\hat{w}_j$ are selected by elastic net, with $\hat{w}_i \hat{w}_j > 0$. Then

$$|\hat{w}_i - \hat{w}_j| \leqslant \frac{\|y\|_2 \sqrt{2}}{\sqrt{n}\lambda_2} \sqrt{1 - \rho_{ij}}.$$

Proof.

See Theorem 1 in Zou and Hastie's 2005 paper "Regularization and variable selection via the elastic net." Or see these notes that adapt their proof to our notation. □