

Discussion on Regularization

He He

Slides based on Lecture 3a from David Rosenberg's [course material](#).

CDS, NYU

Feb 16, 2021

- Thanks for the course feedback!
- Piazza posting instructions
 - Search for similar questions
 - Describe your progress and clarify confusion points
- Feel free to turn on video (when talking)
- Tutorial for convex optimization (preparation for SVM) on **Thursday 9:30am–10:30am** during Marylou's OH
 - Convex functions
 - Primal/Dual problem, strong/weak duality
 - Complementary slackness, KKT conditions

Model Selection

Feature Selection

- **Goal:** Select the “best” subset of features according to some **score**
 - Can also be formulated as ℓ_0 regularization
 - ℓ_0 “norm”: number of non-zero elements
 - **Forward/Backward selection** is a greedy method often used in practice
- Pitfalls in feature selection
 - Is it possible to include irrelevant features (false positives)?
 - What happens when we have dependence among features (e.g. colinearity)?

Model Selection

- Feature selection is a special case of **model selection**:
 - Degree of the polynomial function
 - Decision tree vs kNN
 - More broadly, hyperparameters of learning algorithms
- We need to assess the performance of the model in order to select the “best” one
 - Can we use the training error?
 - What is the ideal performance measure?

- **Test error** (or **generalization error**) of a predictor \hat{f} :

$$\mathbb{E}_{P_{\mathcal{X} \times \mathcal{Y}}} \left[\ell(\hat{f}(x), y) \right].$$

- Note that this is just the risk of \hat{f} .
- What we really care about is the test error, not the error on the test set!
- But we can use the test set error to estimate the test error.
- **Important:** the test set cannot influence training in *any* way.
 - Is it okay to look at the test set as long as the label is hidden?
- For model selection, our goal is to estimate the test error of each model

Estimate Test Error for Model Selection

In order to do model selection,

- We need to **estimate test error**, but we cannot use the true test set.
- Best approach is to use a **validation set** (if we have enough data).

Other methods to estimate test error:

- Re-use training samples: create multiple train/test sets
 - Cross validation, bootstrap
- Training error + penalty
 - AIC, BIC, MDL

Bias-Variance Decomposition

- Note that the test error is a random variable. Why?
- Assume the true model is $y = f(x) + \epsilon$ and $\mathbb{E}\epsilon = 0$ and $\text{Var}(\epsilon) = \sigma^2$
- Consider the expected square loss over *training sets*:

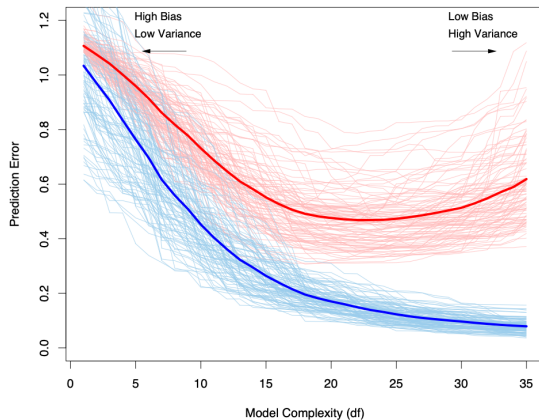
$$\text{err}(x) = \mathbb{E} \left[\left(y - \hat{f}(x) \right)^2 \right] \quad (1)$$

$$= \sigma^2 + \mathbb{E} \left[\hat{f}(x) - \mathbb{E}\hat{f}(x) \right]^2 + \left[f(x) - \mathbb{E}\hat{f}(x) \right]^2 \quad (2)$$

- Both excess risk decomposition and bias-variance decomposition analyze different sources of the test error and they lead to similar conclusions.
- What's the relation between complexity and bias/variance?

Bias-Variance Trade-off

Training set error (blue) and test set error (red)



Regularization and Dependent Features

ℓ_p Regularization

ℓ_0 regularization (subset selection)

$$f(w) = \|Xw - y\|^2 + \lambda \|w\|_0$$

ℓ_1 regularization (Lasso)

$$f(w) = \|Xw - y\|^2 + \lambda \|w\|_1$$

ℓ_2 regularization (Ridge)

$$f(w) = \|Xw - y\|^2 + \lambda \|w\|^2$$

- Which one(s) can be used for feature selection?
- Which one(s) is fast to solve?
- Which one(s) gives unique solution?

Repeated features

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

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

- What is the ERM solution if we get a new feature x_2 , but we always have $x_2 = x_1$?

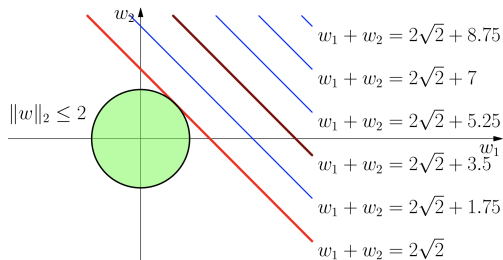
Duplicate Features: ℓ_1 and ℓ_2 norms

- $\hat{f}(x_1, x_2) = w_1 x_1 + w_2 x_2$ is an ERM iff $w_1 + w_2 = 4$.
- What if we introduce the ℓ_1 and ℓ_2 regularization:

| 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 |

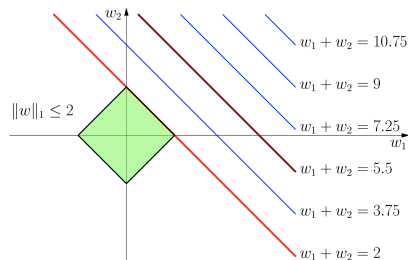
- $\|w\|_1$ doesn't discriminate, as long as all have same sign
- $\|w\|_2^2$ minimized when weight is spread equally
- Picture proof: What does the level sets of ERM look like?

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 $\|w\|_2 \leq 2$ is ridge solution.
- Note that $w_1 = w_2$ at the solution

Equal Features, ℓ_1 Constraint

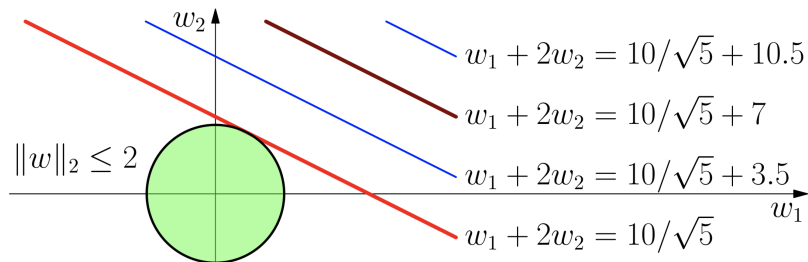


- Suppose the line $w_1 + w_2 = 5.5$ corresponds to the empirical risk minimizers.
- Intersection of $w_1 + w_2 = 2$ and the norm ball $\|w\|_1 \leq 2$ is lasso solution.
- Note that the solution set is $\{(w_1, w_2) : w_1 + w_2 = 2, w_1, w_2 \geq 0\}$.

Linearly Related Features

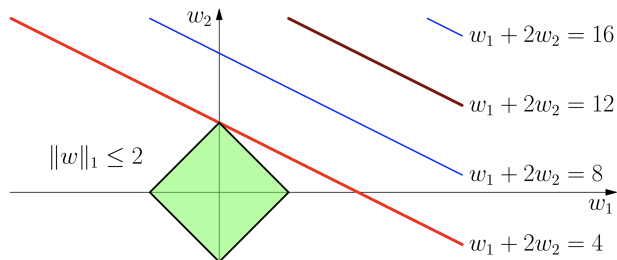
- Linear prediction functions: $f(x) = w_1x_1 + w_2x_2$
- Same setup, now suppose $x_2 = 2x_1$.
- Then all functions with $w_1 + 2w_2 = k$ have the same empirical risk.
- What function will we select if we do ERM with ℓ_1 or ℓ_2 constraint?

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 $\|w\|_2 \leq 2$ 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 $\|w\|_1 \leq 2$ is lasso solution.
- Solution is now a corner of the ℓ_1 ball, corresponding 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**
- In practice, **feature standardization** is important.
- How to standardize the test set?

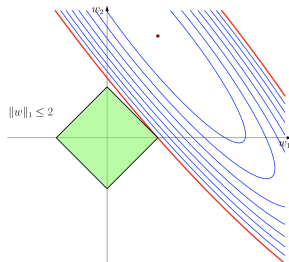
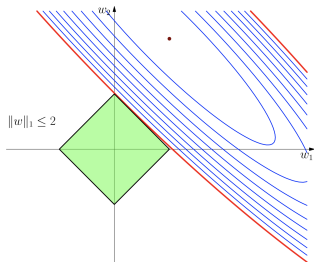
Correlated Features on Same Scale

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

What do the level sets look like?

- Nothing degenerate here, so level sets are ellipsoids.
- But, the higher the correlation, the closer to degenerate we get.
- That is, ellipsoids keep stretching out, getting closer to two parallel lines.

Correlated Features, ℓ_1 Regularization



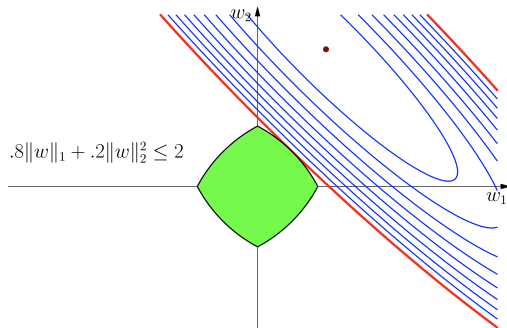
- Intersection could be anywhere on the top right edge.
- Minor perturbations (in data) can drastically change intersection point – very unstable solution.
- Makes division of weight among highly correlated features (of same scale) seem arbitrary.
 - If $x_1 \approx 2x_2$, ellipse changes orientation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

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

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

What are the coefficients for correlated variables?

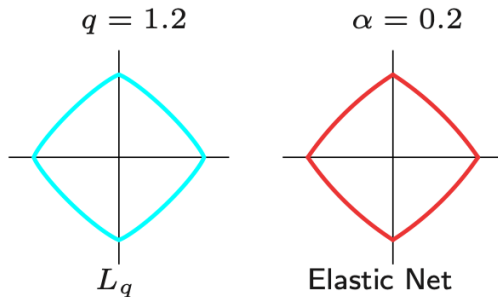
Highly Correlated Features, Elastic Net Constraint



- Elastic net solution is closer to $w_2 = w_1$ line, despite high correlation.
- Elastic net selects variables like Lasso
- And shrinks coefficients of correlated variables like Ridge

Elastic Net vs ℓ_q Constraints

What if we use ℓ_q penalty where $q \in (1, 2)$?



Sparsity

Why doesn't ℓ_2 give sparsity

Consider ℓ_2 regularized least squares:

$$L(w) = \frac{1}{2} \|Xw - y\|^2 + \frac{1}{2} \|w\|^2. \quad (3)$$

Let w^* be the optimal solution. What's the condition for $w_j^* = 0$?

Why does ℓ_1 give sparsity

Consider ℓ_1 regularized least squares:

$$L(w) = \frac{1}{2} \|Xw - y\|^2 + \|w\|_1. \quad (4)$$

Let w^* be the optimal solution. What's the condition for $w_j^* = 0$?

Do we always want sparsity or simpler models?

Do we always want sparsity or simpler models?

- Subjective desire for parsimony: Occam's razor
- Avoid overfit: approximation/estimation error trade-off
- No free lunch theorem