

Regularizations

Shusen Wang

The ℓ_2 -Norm Regularization

Linear Regression

Input: feature matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ and labels $\mathbf{y} \in \mathbb{R}^n$.

Output: vector $\mathbf{w} \in \mathbb{R}^d$ such that $\mathbf{X}\mathbf{w} \approx \mathbf{y}$.

Task

Linear Regression

Input: feature matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$ and labels $\mathbf{y} \in \mathbb{R}^n$.

Output: vector $\mathbf{w} \in \mathbb{R}^d$ such that $\mathbf{X}\mathbf{w} \approx \mathbf{y}$.

Task

- Least squares regression:

$$\min_{\mathbf{w}} \frac{1}{n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2.$$

- Ridge regression:

$$\min_{\mathbf{w}} \frac{1}{n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \gamma \|\mathbf{w}\|_2^2.$$



Loss Function



Regularization

Methods

Ridge Regression:

Algorithms

- **Analytical solution:** $\mathbf{w}^{\star} = (\mathbf{X}^T \mathbf{X} + n\gamma \mathbf{I}_d)^{-1} \mathbf{X}^T \mathbf{y}$.
 - Time complexity: $O(nd^2 + d^3)$.

Ridge Regression:

Algorithms

- **Analytical solution:** $\mathbf{w}^* = (\mathbf{X}^T \mathbf{X} + n\gamma \mathbf{I}_d)^{-1} \mathbf{X}^T \mathbf{y}$.
 - Time complexity: $O(nd^2 + d^3)$.
- **Derivations:**
 - The objective function is $f(\mathbf{w}) = \frac{1}{n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \gamma \|\mathbf{w}\|_2^2$.
 - The gradient is $\nabla f(\mathbf{w}) = \frac{2}{n} \mathbf{X}^T (\mathbf{X}\mathbf{w} - \mathbf{y}) + 2\gamma \mathbf{w}$.
 - Set $\nabla f(\mathbf{w}) = 0$ leads to $\frac{2}{n} (\mathbf{X}^T \mathbf{X} + n\gamma \mathbf{I}_d) \mathbf{w} = \frac{2}{n} \mathbf{X}^T \mathbf{y}$.
- **Time complexity:**
 - $O(nd^2)$ time for the multiplication $\mathbf{X}^T \mathbf{X}$.
 - $O(d^3)$ time for the inversion of the $d \times d$ matrix $\mathbf{X}^T \mathbf{X} + n\gamma \mathbf{I}_d$.

Ridge Regression:

Algorithms

- **Conjugate gradient (CG)**

- $O\left(\sqrt{\kappa} \log \frac{1}{\epsilon}\right)$ iterations to reach ϵ precision.
- $O(nd)$ per-iteration time complexity (for computing the gradient).
- Hessian matrix: $\nabla^2 f(\mathbf{w}) = \frac{2}{n}(\mathbf{X}^T \mathbf{X} + n\gamma \mathbf{I}_d)$.
- $\kappa = \frac{\lambda_{\max}(\mathbf{X}^T \mathbf{X}) + n\gamma}{\lambda_{\min}(\mathbf{X}^T \mathbf{X}) + n\gamma}$ is the condition number of the Hessian.

Usefulness of Regularization

Question: Why do we use the ℓ_2 -norm regularization?

Usefulness of Regularization

Question: Why do we use the ℓ_2 -norm regularization?

- Reason 1: easier to optimize.
 - Conjugate gradient (CG) requires $O\left(\sqrt{\kappa} \log \frac{1}{\epsilon}\right)$ iterations to reach ϵ precision.
 - Least squares: $\kappa = \frac{\lambda_{\max}(\mathbf{X}^T \mathbf{X})}{\lambda_{\min}(\mathbf{X}^T \mathbf{X})}$.
 - Ridge regression: $\kappa = \frac{\lambda_{\max}(\mathbf{X}^T \mathbf{X}) + n\gamma}{\lambda_{\min}(\mathbf{X}^T \mathbf{X}) + n\gamma}$ (decrease as γ increases).
 - ➡ CG converges faster as γ increases.

Usefulness of Regularization

Question: Why do we use the ℓ_2 -norm regularization?

- Reason 1: easier to optimize.
- Reason 2: better generalization.
 - Least squares has better training error (due to the optimality).
 - Ridge regression makes better prediction on test set (due to *bias-variance decomposition*).

The ℓ_1 -Norm Regularization

Motivations

$$\mathbf{x} \in \mathbb{R}^d \xrightarrow{\text{prediction}} y \in \mathbb{R}$$

Fact 1: y can be independent of some of the d feature.

Fact 2: if $d \gg n$, linear models are likely to overfit.

Motivations

$$\mathbf{x} \in \mathbb{R}^d \xrightarrow{\text{prediction}} y \in \mathbb{R}$$

Fact 1: y can be independent of some of the d feature.

Fact 2: if $d \gg n$, linear models are likely to overfit.

Example: Use genomic data to predict disease.

- d is huge: human have 20K protein-coding genes.
- n is small: tens or hundreds of human participants in an experiment.
- Most genes are irrelevant to a specific disease.

Motivations

$$\mathbf{x} \in \mathbb{R}^d \xrightarrow{\text{prediction}} y \in \mathbb{R}$$

Fact 1: y can be independent of some of the d feature.

Fact 2: if $d \gg n$, linear models are likely to overfit.

Goal 1: Select the features relevant to y .

Motivations

$$\mathbf{x} \in \mathbb{R}^d \xrightarrow{\text{prediction}} y \in \mathbb{R}$$

Fact 1: y can be independent of some of the d feature.

Fact 2: if $d \gg n$, linear models are likely to overfit.

Goal 1: Select the features relevant to y .

Goal 2: Prevent overfitting for **large d , small n** problems.

The ℓ_1 -Norm Constraint

• LASSO: $\min_{\mathbf{w}} \frac{1}{2n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2;$ s. t. $\|\mathbf{w}\|_1 \leq t.$

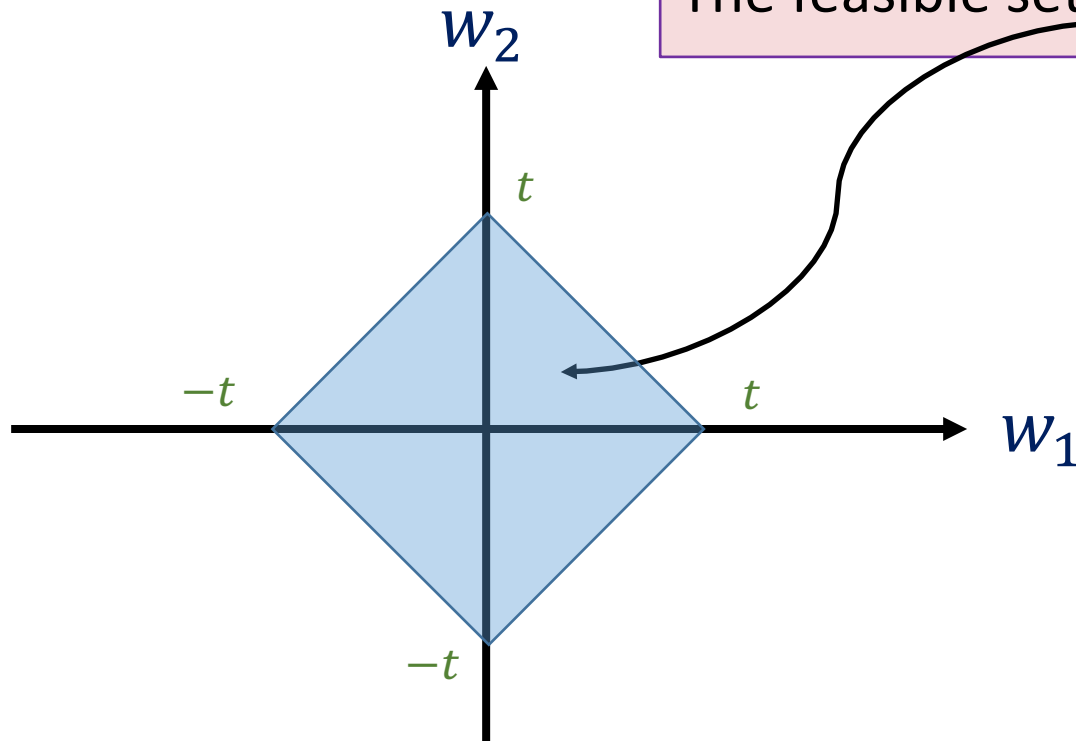


The feasible set $\{\mathbf{w}: \|\mathbf{w}\|_1 \leq t\}$ is convex.

The ℓ_1 -Norm Constraint

• LASSO: $\min_{\mathbf{w}} \frac{1}{2n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2;$ s. t. $\|\mathbf{w}\|_1 \leq t.$

The feasible set $\{\mathbf{w}: \|\mathbf{w}\|_1 \leq t\}$ is convex.



The ℓ_1 -Norm Constraint

- LASSO: $\min_{\mathbf{w}} \frac{1}{2n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2; \quad \text{s. t. } \|\mathbf{w}\|_1 \leq t.$
 - It is a convex optimization model.
 - The optimal solution \mathbf{w}^* is **sparse** (i.e., most entries are zeros).
 - Smaller $t \rightarrow$ sparser \mathbf{w}^* .

The ℓ_1 -Norm Constraint

- LASSO: $\min_{\mathbf{w}} \frac{1}{2n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2; \quad \text{s. t.} \quad \|\mathbf{w}\|_1 \leq t.$
 - It is a convex optimization model.
 - The optimal solution \mathbf{w}^* is **sparse** (i.e., most entries are zeros).
 - Smaller $t \rightarrow$ sparser \mathbf{w}^* .
 - Sparsity \longleftrightarrow feature selection. Why?
 - Let \mathbf{x}' be a test feature vector.
 - The prediction is $\mathbf{x}'^T \mathbf{w}^*$.
 - If $w_1^* = 0$, then the prediction is independent of x_1' .

The ℓ_1 -Norm Regularization

- LASSO: $\min_{\mathbf{w}} \frac{1}{2n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2; \quad \text{s. t. } \|\mathbf{w}\|_1 \leq t.$

- Another form: $\min_{\mathbf{w}} \frac{1}{2n} \|\mathbf{X}\mathbf{w} - \mathbf{y}\|_2^2 + \gamma \|\mathbf{w}\|_1.$



Loss Function



Regularization

Summary

Regularized ERM

- Regularized empirical risk minimization:

$$\min_{\mathbf{w} \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^n L(\mathbf{w}; \mathbf{x}_i, y_i) \quad + \quad R(\mathbf{w}).$$

Regularized ERM

- Regularized empirical risk minimization:

$$\min_{\mathbf{w} \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^n L(\mathbf{w}; \mathbf{x}_i, y_i) \quad + \quad R(\mathbf{w}).$$



Loss Function

- Linear regression: $L(\mathbf{w}; \mathbf{x}_i, y_i) = \frac{1}{2} (\mathbf{w}^T \mathbf{x}_i - y_i)^2$
- Logistic regression: $L(\mathbf{w}; \mathbf{x}_i, y_i) = \log(1 + \exp(-y_i \mathbf{w}^T \mathbf{x}_i))$
- SVM: $L(\mathbf{w}; \mathbf{x}_i, y_i) = \max\{0, 1 - y_i \mathbf{w}^T \mathbf{x}_i\}$

Regularized ERM

- Regularized empirical risk minimization:

$$\min_{\mathbf{w} \in \mathbb{R}^d} \quad \frac{1}{n} \sum_{i=1}^n L(\mathbf{w}; \mathbf{x}_i, y_i) \quad + \quad R(\mathbf{w}).$$



Regularization

- ℓ_1 -norm: $R(\mathbf{w}) = \gamma ||\mathbf{w}||_1$
- ℓ_2 -norm: $R(\mathbf{w}) = \gamma ||\mathbf{w}||_2^2$
- Elastic net: $R(\mathbf{w}) = \gamma_1 ||\mathbf{w}||_1 + \gamma_2 ||\mathbf{w}||_2^2$