Asymptotically Optimal Regularization for Smooth Parametric Models

Percy Liang — UC Berkeley
Francis Bach — INRIA - École Normale Supérieure
Guillaume Bouchard — Xerox Research Centre Europe
Michael I. Jordan — UC Berkeley

Motivation

Regularization is important to prevent overfitting (in theory and practice)

Many regularizers used in machine learning:

- Penalize norms of parameter vector (L_2, L_1, block norms, etc.)
- Regularize discriminative model with generative model
- Multi-task learning: shrink related tasks towards each other
- Semi-supervised learning: entropy reg., posterior reg., Gen. Expect. Criteria

Questions:

- Given a regularizer, how well does it perform?
- What is the best regularizer?

Setup

Loss function:

$\ell(z;\theta)$ model parameters $\theta \in \mathbb{R}^d$

Example (linear regression): $\ell((x,y);\theta) = \frac{1}{2}(y - \theta^\top x)^2$

Regularizer:

$R_n(\lambda,\theta)$ (e.g., $= \frac{\lambda}{n} r(\theta)$) regularization parameters $\lambda \in \mathbb{R}^k$

Example (L_2 regularization): $r(\theta) = \frac{1}{2}\|\theta\|^2$

Training:

Training data: $Z_1,\dots,Z_n \overset{\text{i.i.d.}}{\sim} p^*$ $\quad Z_i = (X_i, Y_i)$

Estimator: $\hat\theta_n^\lambda \overset{\text{def}}{=} \operatorname{argmin}_\theta \frac{1}{n}\sum_{i=1}^n \ell(Z_i;\theta) + R_n(\lambda,\theta)$

Evaluation:

Expected risk: $\mathbf{L}_n(\lambda) \overset{\text{def}}{=} \mathbb{E}_{Z_1,\dots,Z_n\sim p^*}\mathbb{E}_{Z\sim p^*}[\ell(Z;\hat\theta_n^\lambda)]$

Assumptions:

- Loss function ℓ is smooth (not necessarily squared loss or includes $\log p^*$)
- Regularizer R_n is smooth

Coverage of our analysis:

- Included: linear regression, logistic regression; L_2 regularization
- Excluded: SVMs; L_1 regularization

Outline of approach

Wishful thinking: find reg. parameters λ that minimize the expected risk

$$\operatorname{argmin}_\lambda \mathbf{L}_n(\lambda)$$

Part 1:

Problem: $\mathbf{L}_n(\lambda)$ is very complicated, can't be minimized directly
Solution: minimize Taylor approximation of $\mathbf{L}_n(\lambda)$
Significance: provides insight into loss-regularizer interactions

Part 2:

Problem: Unimplementable since best λ depends on p^* (through $\theta_\infty = \operatorname{argmin}_\theta \mathbb{E}_{Z\sim p^*}[\ell(Z,\theta)]$)
Solution: plugin $\hat\theta_n^0$ (preliminary unregularized estimate) for θ_∞
Significance: get practical algorithm

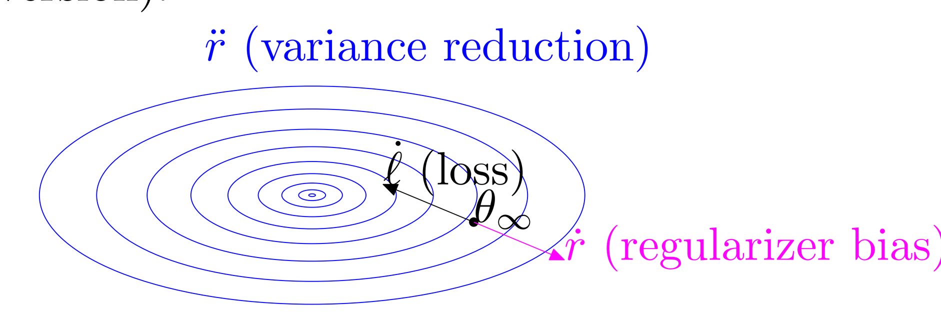
Part 1 (oracle regularizer)

Main theorem:

$$\mathbf{L}_n(\lambda) = \mathbf{L}_n(0) + \mathbf{L}(\lambda)\cdot n^{-2} + \cdots$$

Asymptotic risk (simplified version):

$$\mathbf{L}(\lambda) \overset{\text{def}}{=} \frac{1}{2}\lambda^2\|\dot r\|^2 - \lambda \operatorname{tr}(\ddot r)$$



Oracle regularizer (solve for λ):

$\lambda^* = \operatorname{argmin}_\lambda \mathbf{L}(\lambda) = \frac{\operatorname{tr}(\ddot r)}{\|\dot r\|^2}$ $\qquad \mathbf{L}(\lambda^*) = -\frac{\operatorname{tr}(\ddot r)^2}{2\|\dot r\|^2}$

(Note that optimal regularization λ^* could be negative!)

Example (ridge regression): $r(\theta) = \frac{1}{2}\|\theta\|^2$

Regularizer bias: $\dot r = \theta_\infty$
Variance reduction: $\ddot r = \operatorname{tr}(I) = d$ $\Rightarrow$ Oracle regularizer: $\lambda^* = \frac{d}{\|\theta_\infty\|^2}$

Part 2 (plugin regularizer)

Oracle regularizer:

$\lambda^* = f(\theta_\infty)$ [depends on θ_∞, not implementable]

Plugin regularizer:

$\hat\lambda_n = f(\hat\theta_n^0)$ [plug unregularized estimate $\hat\theta_n^0$ in for θ_∞]

Plugin algorithm (motivated by oracle analysis):

$\hat\theta_n^0 = \operatorname{argmin}_\theta \frac{1}{n}\sum_{i=1}^n \ell(Z_i,\theta)$ [unregularized]
$\hat\lambda_n = f(\hat\theta_n^0)$ [compute regularization parameter **adaptively**]
$\hat\theta_n^{\hat\lambda_n} = \operatorname{argmin}_\theta \frac{1}{n}\sum_{i=1}^n \ell(Z_i,\theta) + \hat\lambda_n r(\theta)$ [plugin]

Analysis:

Re-analyze new regularizer $\hat\lambda_n r(\theta)$ in our framework
Result: expected risk of $\hat\theta_n^{\hat\lambda_n}$ is $\mathbf{L}(\lambda^*) - \dot f^\top \dot r$

Example: Stein's paradox

Question:

Given $X_1,\dots,X_n \sim \mathcal{N}(\theta_\infty, I_{d\times d})$ [all independent]
What is the best estimator $\hat\theta_n$ (minimizes $\mathcal{L}(\hat\theta_n) = \mathbb{E}\|\hat\theta_n - \theta_\infty\|^2$)?
Maximum likelihood: $\hat\theta_n^{\text{ML}} = \bar X = \frac{1}{n}\sum_{i=1}^n X_i$ Not optimal!

Stein paradox (1961):

James-Stein estimator: $\hat\theta_n^{\text{JS}} = (1 - \frac{d-2}{n\|\bar X\|^2})\bar X$
Surprising result: $\mathcal{L}(\hat\theta_n^{\text{JS}}) < \mathcal{L}(\hat\theta_n^{\text{ML}})$ for all $\theta_\infty, d \geq 3$

Relationship to our work:

With $r(\theta) = \frac{1}{2}\|\theta\|^2$, plugin estimator $\Rightarrow$ James-Stein estimator

Example: Hybrid generative/discriminative learning

Setup:

Discriminative model: $p_\theta(y \mid x)$ $\qquad$ Generative model: $p_\theta(x,y)$

Past asymptotic analysis [Liang & Jordan, '08]:

If model well-specified, generative better (provides more stability)
If model mis-specified, discriminative better (achieves lower risk)

Leverage both and analyze in our framework:

Loss $\ell(x,y;\theta) = -\log p_\theta(y \mid x)$ [discriminative]
Regularizer $R_n(\lambda,\theta) = -\frac{\lambda}{n^2}\sum_{i=1}^n \log p_\theta(x,y)$ [generative]

Theorem formalizes our intuitions:

$\dot r$: asymptotic misspecification
$\ddot r$: extra Fisher information provided by x

Example: Multitask learning

Setup:

Loss $\ell(x,y;\theta) = \sum_{k=1}^K (y_k - x_k^\top\theta_k)^2$ (K linear regression tasks)
Regularizer: $r(\theta;\Lambda) = \frac{1}{2}\theta^\top(\Lambda\otimes I)\theta$ (shrink similar tasks towards each other)

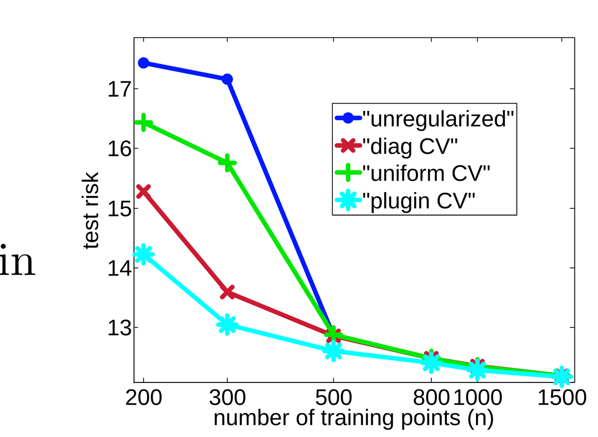
Plugin regularizer: $\hat\Lambda_n = d\cdot((\hat\Theta_n^0)^\top\hat\Theta_n^0)^{-1}$, where $\Theta = (\theta_1,\dots,\theta_K) \in \mathbb{R}^{d\times K}$

- Intuition: shrink tasks more if they are close according to $\hat\Theta_n^0$
- Allow setting K^2 regularization parameters Λ, not just one

Experiment: predict binding affinity of MHC-I molecules
5 tasks, one for each molecule; 20 features

Three regularizers:
diag CV: independent regularization
uniform CV: fixed sharing
plugin CV: sharing determined by plugin

Cross-validate regularizer strength



Conclusion

Summary:

- Minimize Taylor expansion of expected risk $\Rightarrow$ oracle regularizer
- Yields simple algorithm based on plugin regularizer

Asymptotic analysis:

- Offers a new perspective to risk bounds (more common in machine learning)
- Get exact higher-order term (not just bound)
- Advantage: can **compare** different regularizers