

Regularization for Deep Learning 7.1-7.4

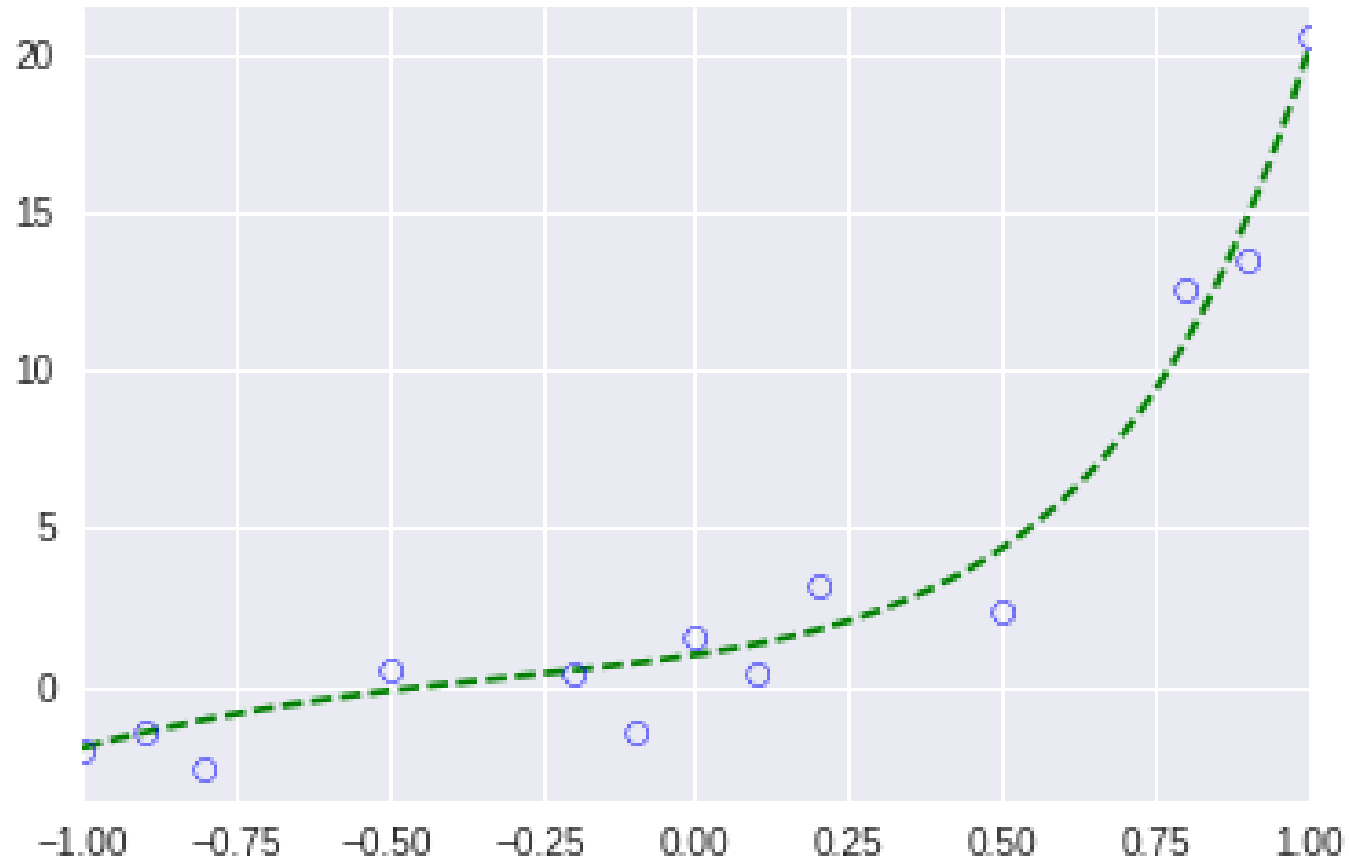
Deep Learning輪講会(2017-06-21 Wednesday)

河野 晋策 @lapis_zero09

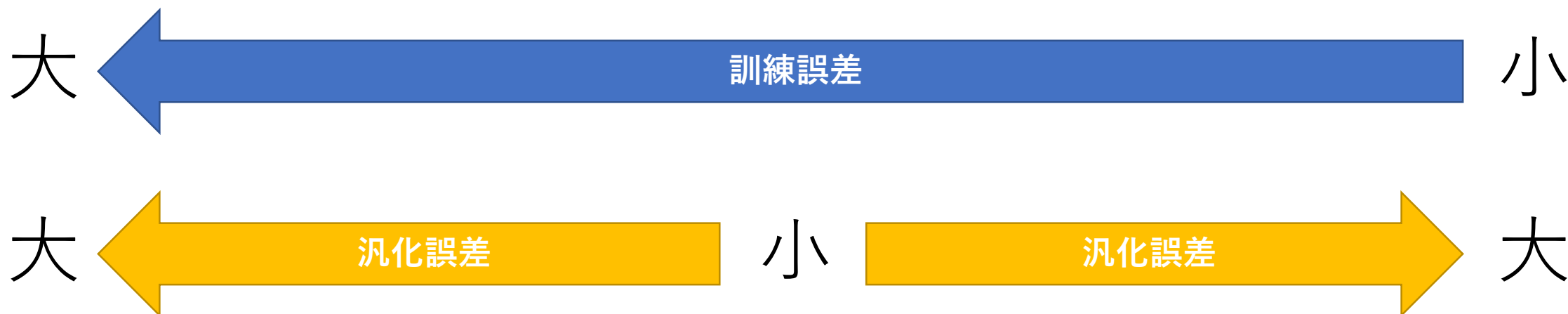
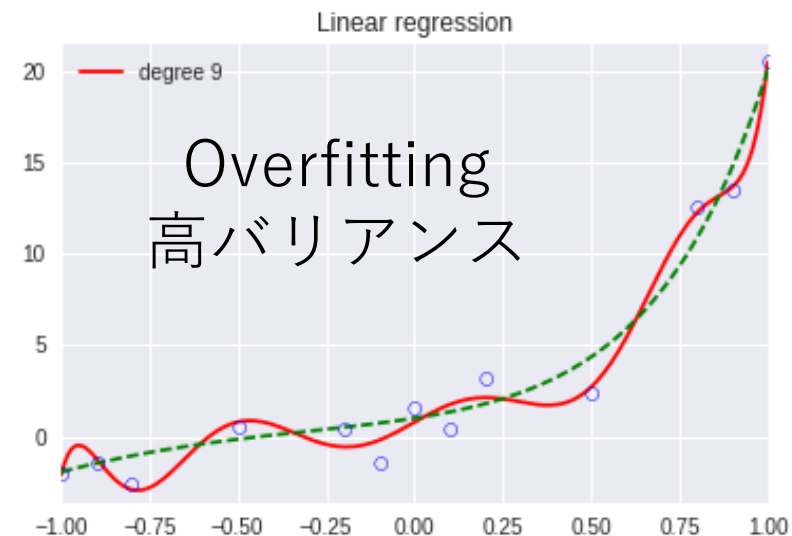
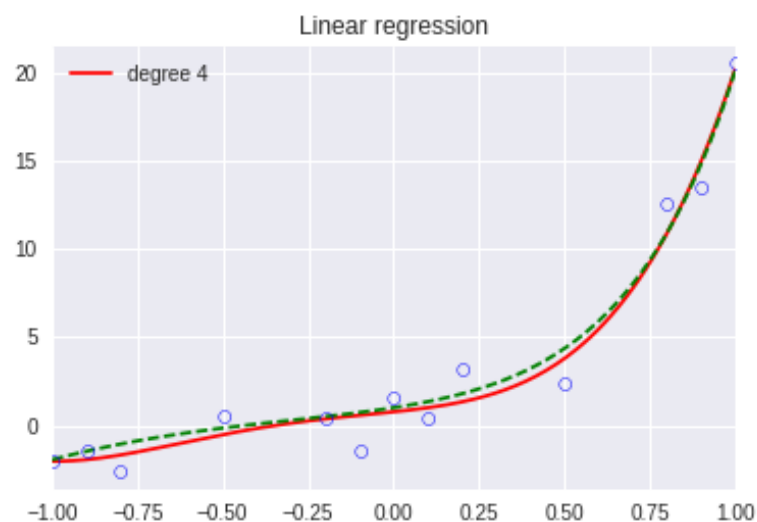
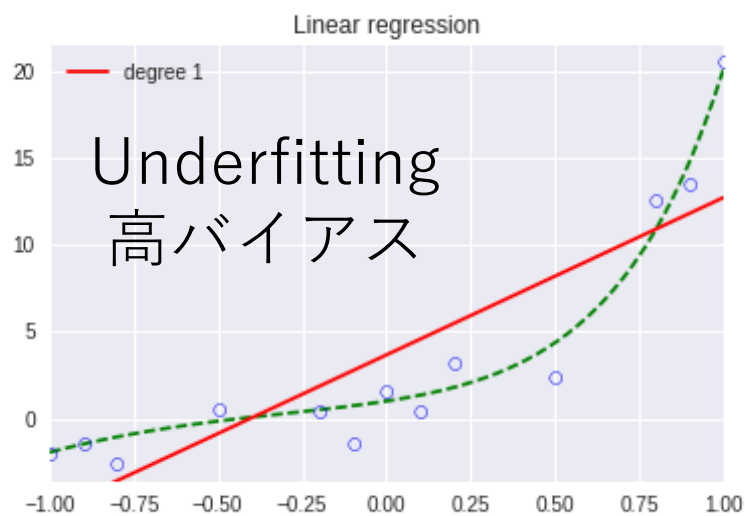
Regularization and Under-Constrained Problems

Chap 7.1-7.3

Underfitting Overfitting



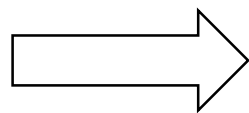
Underfitting Overfitting



Regularization

- “複雑度”の高いモデル

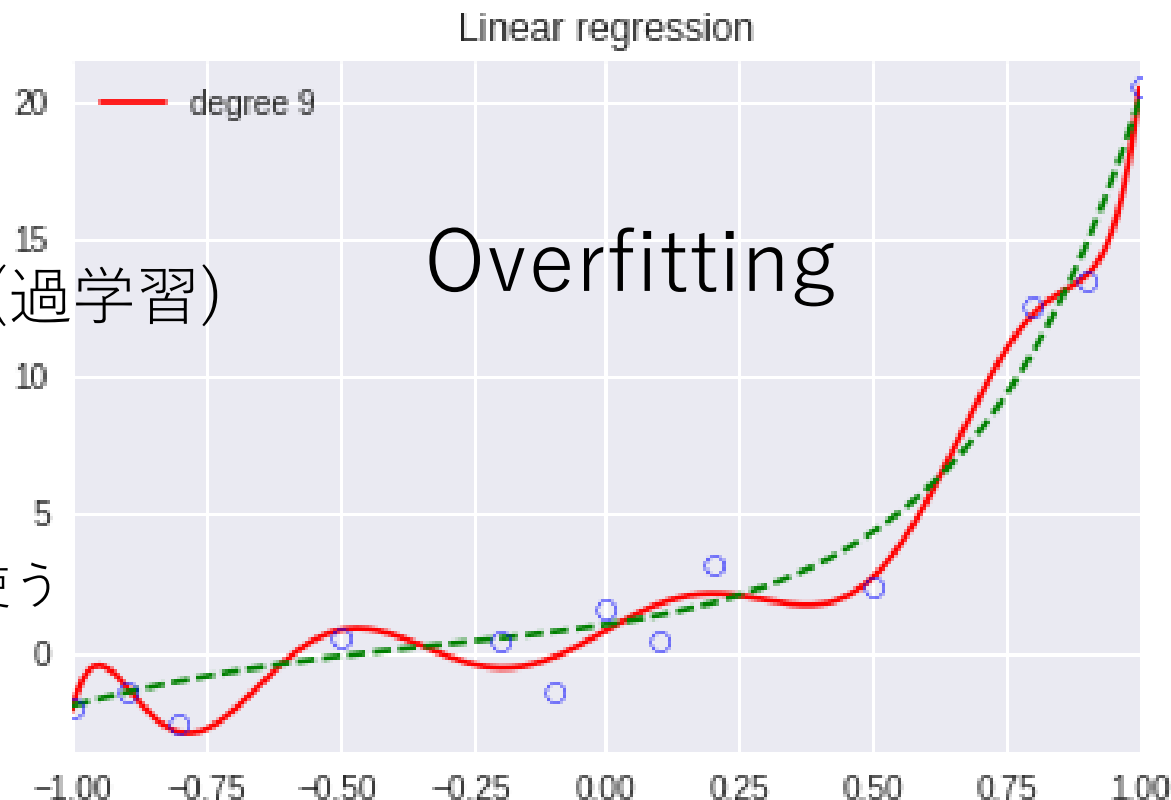
- 訓練誤差は低い
- 汎化誤差は高い



Overfitting(過学習)

- 汎化誤差を減らす方法

- より多くのデータの収集(Chap7.4)
- パラメータ数が少ない, 単純なモデルを使う
- データの次元数を減らす
- **Regularization**(正則化)
 - モデルの“複雑さ”を制御



Deep Learning book. p.229より抜粋

“the true generation process essentially involves simulating the entire universe”

“we are always trying to fit a square peg into a round hole”

Regularization

元の目的関数

正則化項

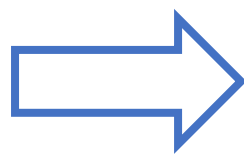
- 罰則付き最適化(Chap7.2)

- $minimize \tilde{J}(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y}) = J(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y}) + \alpha \Omega(\boldsymbol{\theta})$

- $J(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y})$ と $\Omega(\boldsymbol{\theta})$ 両方を最適化したい

例

- $J(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y})$: 年金制度の充実
 - $\Omega(\boldsymbol{\theta})$: 国民の税負担の増加



回帰

- $J(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y})$: 訓練誤差を最小化
 - $\Omega(\boldsymbol{\theta})$: "複雑さ"を最小化

- トレードオフパラメータ α で優先度を定める

Regularization

元の目的関数

正則化項

Memo: ω の p ノルム

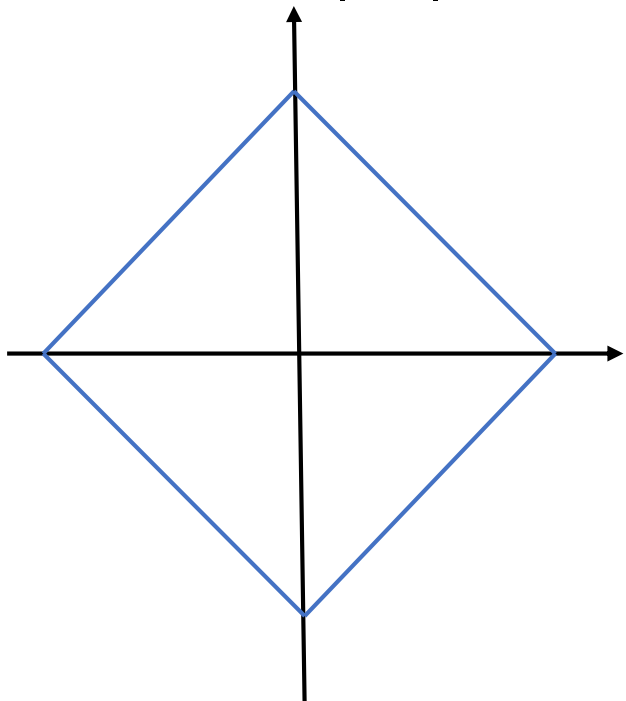
$$\boldsymbol{\omega}^T = (\omega_1, \omega_2, \dots, \omega_D)$$

$$\|\boldsymbol{\omega}\|_p = \left(\sum_{i=1}^D |\omega^i|^p \right)^{\frac{1}{p}}$$

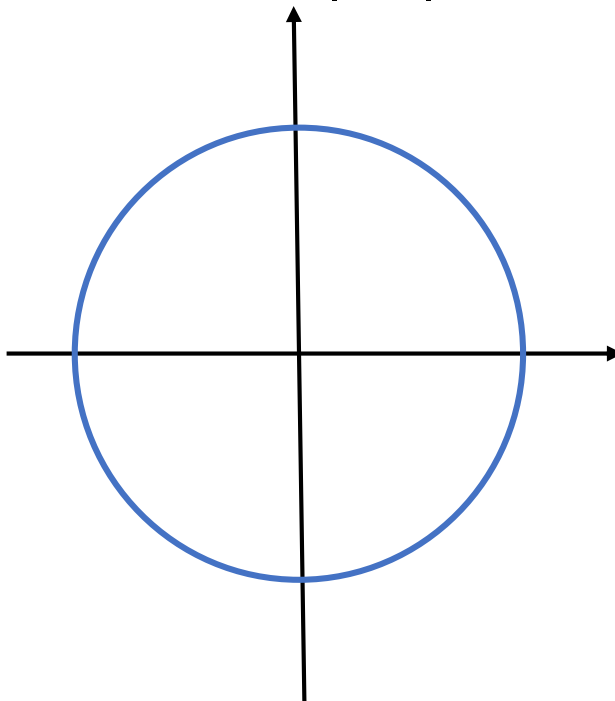
正則化された目的関数

$$\tilde{J}(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y}) = J(\boldsymbol{\theta}; \mathbf{X}, \mathbf{y}) + \alpha \Omega(\boldsymbol{\theta})$$

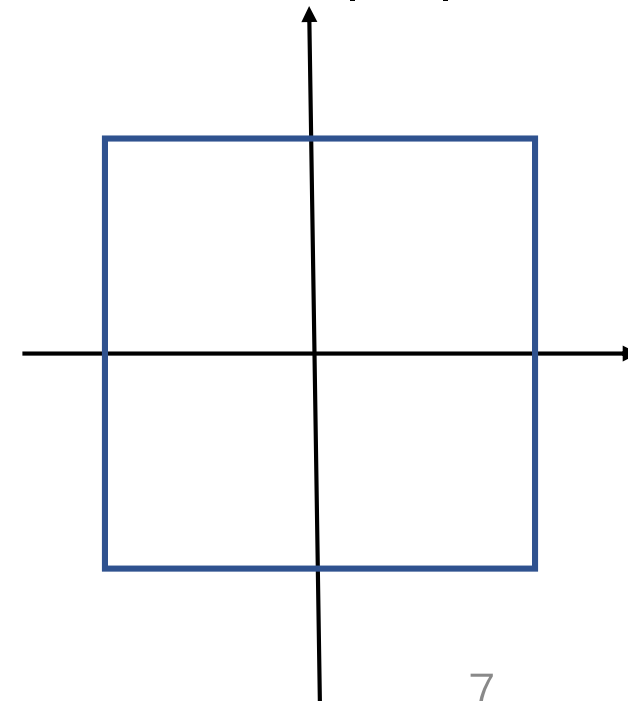
L1 ノルム
 $\Omega(\theta) = \|\boldsymbol{\omega}\|_1$



L2 ノルム
 $\Omega(\theta) = \|\boldsymbol{\omega}\|_2$



Max ノルム
 $\Omega(\theta) = \|\boldsymbol{\omega}\|_\infty$



L^2 Regularization (Chap 7.1.1)

- 元の目的関数 : $J(\boldsymbol{\omega}; \mathbf{X}, \mathbf{y})$

- L2正則化項入り : $\tilde{J}(\boldsymbol{\omega}; \mathbf{X}, \mathbf{y}) = J(\boldsymbol{\omega}; \mathbf{X}, \mathbf{y}) + \frac{\alpha}{2} \boldsymbol{\omega}^T \boldsymbol{\omega}$

正則化パラメータ



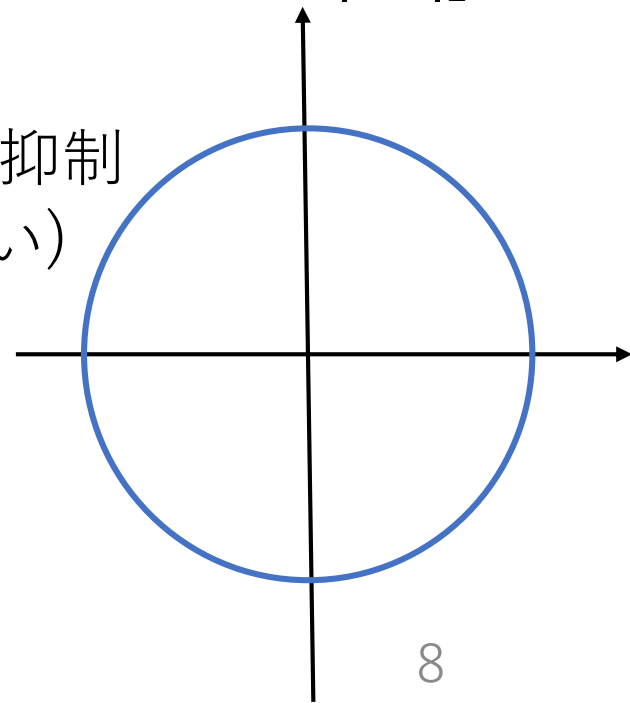
L2正則化項

- L2正則化項

- $\boldsymbol{\omega}$ の各要素が大きい値をとると大きくなる→”複雑さ”を抑制
- 凸関数の和は凸関数→唯一の局所最適解が求まる(嬉しい)
- 微分可能→解析解が求まる(嬉しい)

- 二乗誤差項 + L2正則化項 = リッジ回帰

L2ノルム
 $\Omega(\theta) = \|\boldsymbol{\omega}\|_2$



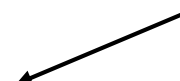
L^2 Regularization (Chap 7.1.2)

- 元の目的関数 : $J(\boldsymbol{\omega}; \mathbf{X}, \mathbf{y})$
- L1正則化項入り : $\tilde{J}(\boldsymbol{\omega}; \mathbf{X}, \mathbf{y}) = J(\boldsymbol{\omega}; \mathbf{X}, \mathbf{y}) + \alpha \sum_i |\omega_i|$

正則化パラメータ



L2正則化項

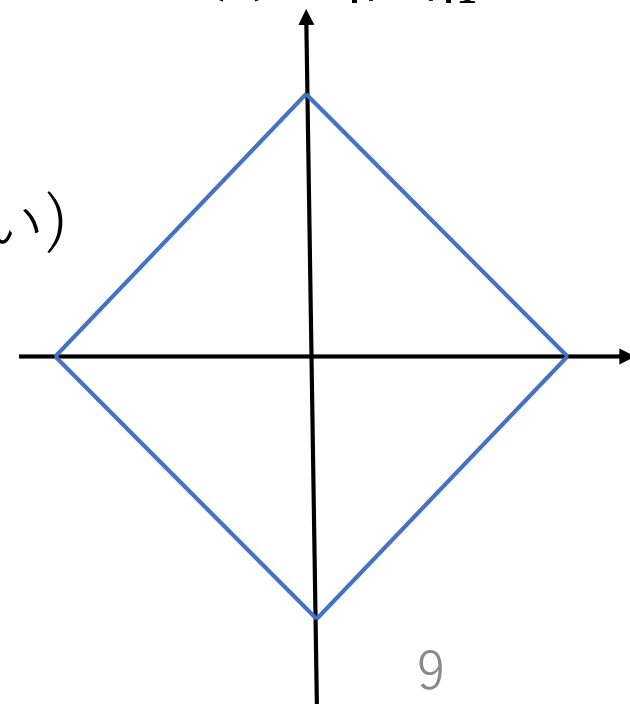


- L1正則化項

- 原点からの遠さに対して線形で罰則
- 凸関数の和は凸関数 → 唯一の局所最適解が求まる (嬉しい)
- 原点付近で微分不可能 → 解析解は求まらない (悲しい)

- 二乗誤差項 + L2正則化項 = ラッソ回帰

L1ノルム
 $\Omega(\theta) = \|\boldsymbol{\omega}\|_1$



L2 vs L1

- **L2の方が嬉しいポイント高いけどなぜL1使うのか**

- 特徴量(予測に有用か不明)が大量にある場面：

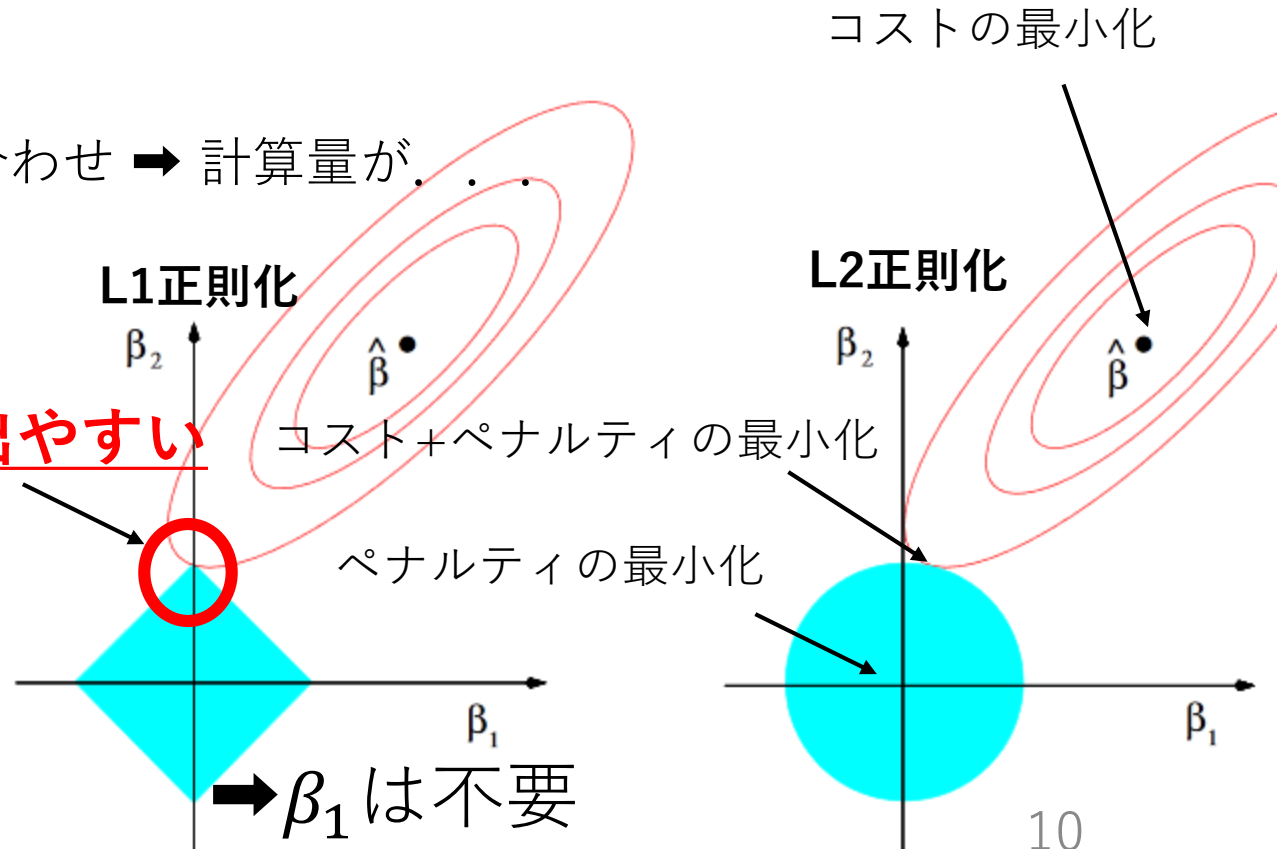
Q.どうやって特徴選択する？

A.交差検定

特徴量D個とすると、 2^D 通りの組み合わせ \rightarrow 計算量が...

L1正則化はその形状から軸上に解が出やすい

\rightarrow 自然な特徴選択が可能(嬉しい)



L2 vs L1

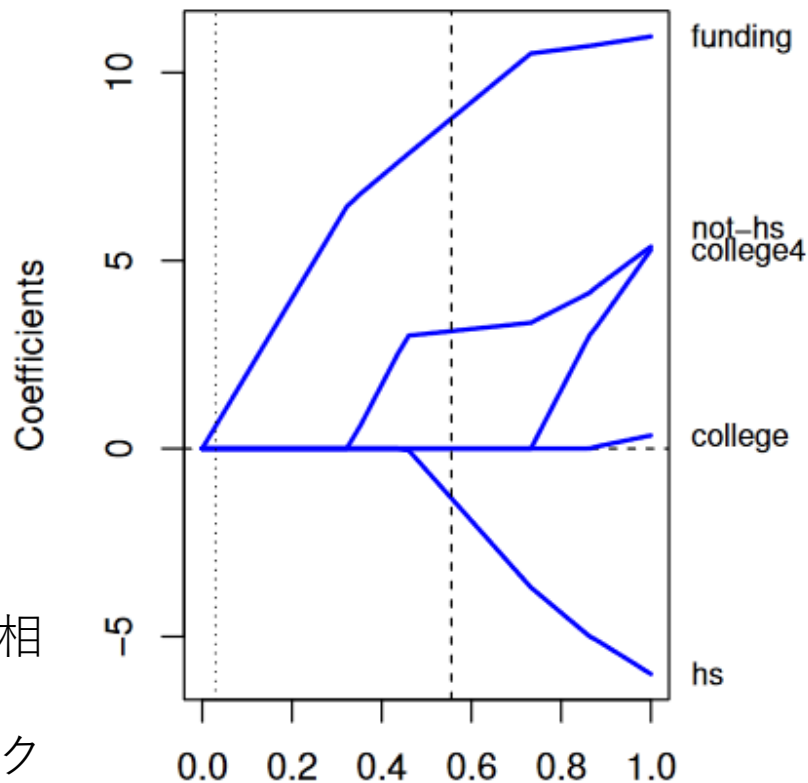
特徴選択の観察

- 予測：犯罪率
- 特徴量：アメリカ街別 警察予算, 高校卒業率, 大学卒業率など

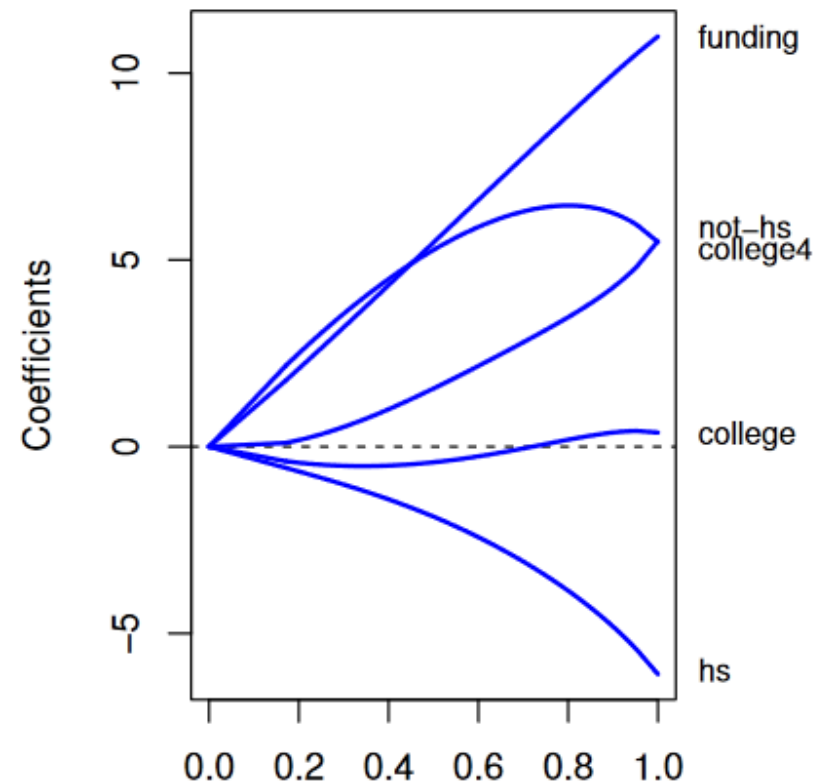
Table 2.1 Crime data: Crime rate and five predictors, for $N = 50$ U.S. cities.

city	funding	hs	not-hs	college	college4	crime rate
1	40	74	11	31	20	478
2	32	72	11	43	18	494
3	57	70	18	16	16	643
4	31	71	11	25	19	341
5	67	72	9	29	24	773
⋮	⋮	⋮	⋮	⋮		
50	66	67	26	18	16	940

Lasso



Ridge Regression



- 横軸：最小二乗推定量に対する相対ノルム
- 縦軸：各特徴量に対する係数ベクトル

$$\|\hat{\beta}\|_1 / \|\tilde{\beta}\|_1$$

$$\|\hat{\beta}\|_2 / \|\tilde{\beta}\|_2 \quad 11$$

Lassoで特徴選択の仕組み

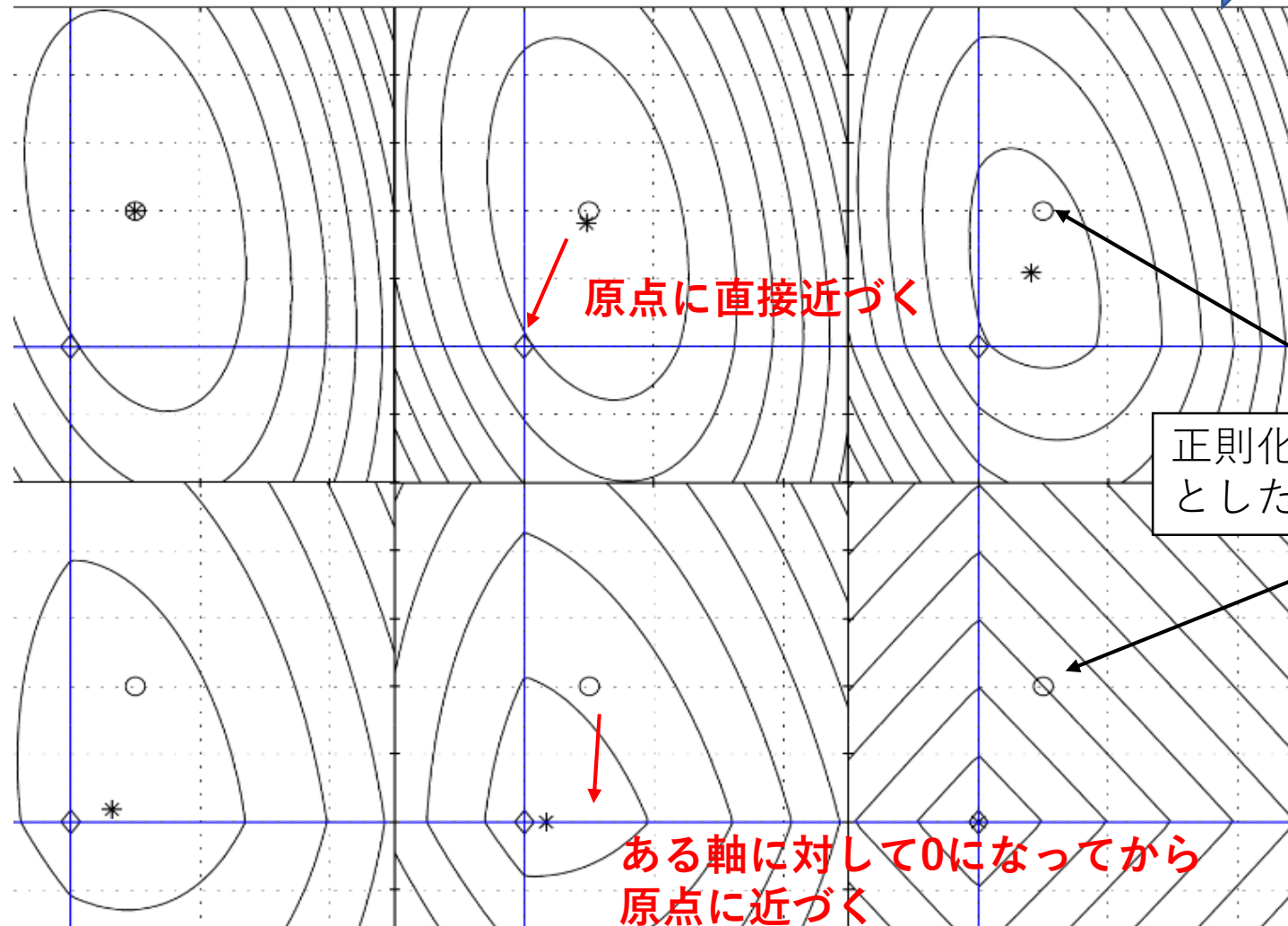
弱

正則化

強

Ridge

Lasso



Dataset Augmentation

Chap 7.4

Dataset Augmentation

- モデルをより一般化するにはより多くのデータで学習するのがいい
 - データの量は限られる ➡ 疑似データを作成
- object recognition領域で有効
 - 画像の回転や拡大, etc...
 - Vincent et al. Extracting and Composing Robust Features with Denoising Autoencoders. ICML, 2008.
- speech recognition領域で有効
 - Jaitly and Hinton. Vocal Tract Length Perturbation (VTLP) improves speech recognition. ICML, 2013.

Dataset Augmentation(蛇足な知見)

- Imbalanced Data
 - データのクラスに偏りがあって学習が困難なデータ
- 提案手法
 - Under-sampling 多いクラスを減らす
 - Over-sampling 少ないクラスを増やす(疑似データの生成)
 - N. V. Chawla, K. W. Bowyer, L. O.Hall, W. P. Kegelmeyer, “**SMOTE: synthetic minority over-sampling technique**,” Journal of artificial intelligence research, 321-357, 2002.
 - He, Haibo, Yang Bai, Eduardo A. Garcia, and Shutao Li. “**ADASYN: Adaptive synthetic sampling approach for imbalanced learning**,” In IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence), pp. 1322-1328, 2008.
 - その他 SMOTEの変形など

Dataset Augmentation(蛇足な知見)

- Model Compression

- 課題

- データが少ない問題に対して, DeeeeeeeepなNNを適用すると過学習する
- 浅いNNだと精度が足りない
- アンサンブルだと精度は出るけど大きいし, 遅い

- 提案手法

1. 元データに対してアンサンブルを訓練
2. 元データから大量の正解ラベル無し疑似データを作成
3. 訓練したアンサンブルで疑似データに正解ラベル付け
4. 大量の疑似データでDeeeeeeeepNNを訓練 ➡ アンサンブルの近似と同義

- Buciluă, Cristian; Caruana, Rich et al. **Model compression**. ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. 2006, p. 535-541.