MATH466/MATH766
Math of machine learning
01/22 Lecture 4 HW1 and Sparsity in Machine Learning

References:

- Ch2, 3 of High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications by John Wright and Yi Ma
- Ch3.4 of The Elements of Statistical Learning by Trevor Hastie, Robert Tibshirani and Jerome Friedman
- Ch6.3 of Linear Algebra and its Applications (Fourth Edition) by Gilbert Strang

Todays contents:

- regularization
- l1 norm and sparsity
- sparse signal modeling

Important concepts:

- regularization
- sparsity

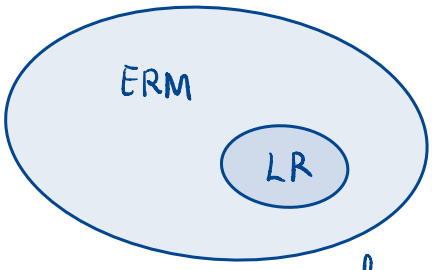
Recommend reading:

- Ch1 of High-Dimensional Data Analysis with Low-Dimensional Models: Principles, Computation, and Applications by John Wright and Yi Ma
- Ch3.4 of The Elements of Statistical Learning by Trevor Hastie, Robert Tibshirani and Jerome Friedman
- Ch3.3, 3.6, 3.8 of The Elements of Statistical Learning by Trevor Hastie, Robert Tibshirani and Jerome Friedman

Review.

Lec2

Regression:



+ Regularization

LR + l_2-regularization → RR

l_1-regularization → LASSO

① sparsity (today)

③ linear model → nonlinear model (HW1-2)

parametric → non-parametric

Lec3

Classification: (linear) parameterize p + MLE → logistic regression

② binary → multiclass (HW1-1)

parametric ↑ maximize marginal distance → SVM

non-parametric model - kNN

④ compare parametric v.s. non-parametric (HW1-4)

HW 1 prob 1. 2. go-through

Prob 3 linearly separable data / two-ring data

training accuracy

testing accuracy

# of training data	LR	kNN	MLP
100			
500			
1000			

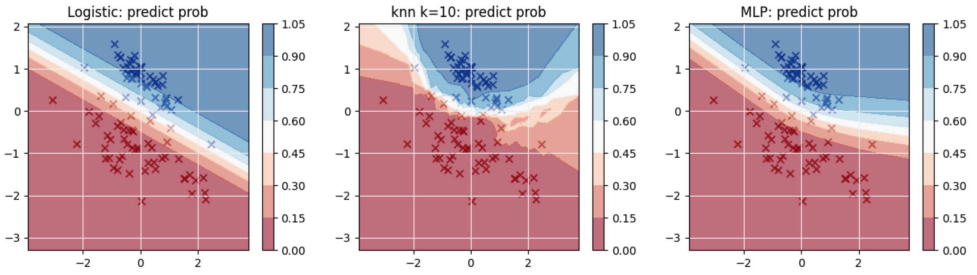
4 tables. ① compare performance of 3 methods on the same dataset w/ different # of training samples

② compare the same method on different dataset

③ compare different methods on different dataset

---

about decision boundary plot



predict_prob

$p(x) = 0.9, 0.8, \cdots, 0.1$

about warning on maximum iteration

iterative algorithm.

$\min_{\theta} \mathcal{L}(\theta)$

e.g. For $k = 1, 2, \cdots, K$

  if $\|\nabla \mathcal{L}(\theta^k)\| >$ tolerance

    $\theta^{k+1} = \theta^k - \tau \nabla \mathcal{L}(\theta^k)$

Inverse Problem and Regularization

An oversimplified example: given a mapping $\mathcal{L}: A \mapsto B = \mathcal{L}(A)$

forward problem: given A, find $\mathcal{L}(A)$

e.g. 1. A is some function f

$B = \mathcal{L}(A) = \{(x_1, f(x_1)), (x_2, f(x_2)), \cdots, (x_n, f(x_n))\}$

e.g. 2. A is a clean image

$\mathcal{L}$ is some blurring / degenerating process

inverse problem: observe B (partial, noisy)

want to find A such that $\mathcal{L}(A) = B$

e.g. 1. B: training data set, supervised learning

e.g. 2. B is the blurred / degenerated image

want to find the clean image

Inverse problems are usually harder than forward problem.

One difficulty: non-unique solution

Regularization is a common strategy in inverse problem

Idea / Goal: include prior knowledge / belief / preference to make the solution unique

Outcome: solution of certain property

e.g. image denoising. Rudin - Osher - Fatemi (ROF) model (or total variation denoising)

edge-preserving

"sparsity" is a common belief
- e.g. matrix completion (Netflix problem)
- in many applications

an important property people pursue
- e.g. LASSO, compressed sensing

Sparsity

For $w \in \mathbb{R}^d$, $\|w\|_0 := \sum_{j=1}^{d} 1\{w_j \neq 0\}$ = # of non-zero entries in w

sparsity wants small $\|\cdot\|_0$ ($\|\cdot\|_0$ is called "ℓ_0-norm" but is not a norm)

$$\text{LASSO}: \quad \min_w \ \frac{1}{2}\|Xw - y\|^2 + \lambda \|w\|_1 \qquad (*_\ell)$$

where $\|w\|_1 := \sum_{j=1}^{d} |w_j|$

Q: Why not using ℓ_0-norm as a regularization ?
What's the benefit of ℓ_1-regularization ?

Q: Does ℓ_1-regularization encourages sparsity ? Why ?
Answer Today !

Q: How to solve $(*_\ell)$?

Explanation 1

Consider $d = 2$ Assume $\hat{w} = \operatorname{argmin}_w \frac{1}{2}\|Xw - y\|^2$

contour of $R(w) = \frac{1}{2}\|Xw - y\|^2$

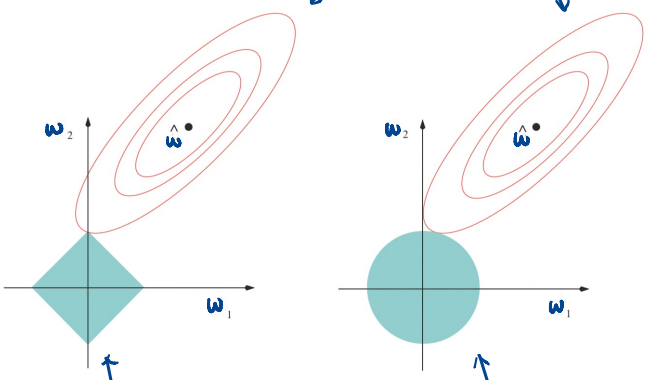


FIGURE 3.11. *Estimation picture for the lasso (left) and ridge regression (right). Shown are contours of the error and constraint functions. The solid blue areas are the constraint regions $|w_1| + |w_2| \le t$ and $w_1^2 + w_2^2 \le t^2$, respectively, while the red ellipses are the contours of the least squares error function.*

Contour of $\|w\|_1$ Contour of $\|w\|_2$

grow the contour of R and $\|\cdot\|_1$ $\|\cdot\|_2$ at the same "speed" until they meet

→ solution under ℓ_1 ℓ_2 regularization

ℓ_2-norm contour grows equally fast in all directions (isotropic)

ℓ_1-norm ——— faster on axes (anisotropic)

Explanation 2

Consider $x = (1, \varepsilon)$, $\varepsilon > 0$, $\varepsilon \ll 1$

then $\|x\|_1 = 1+\varepsilon$, $\|x\|_2^2 = 1+\varepsilon^2$

If we want to reduce $\|\cdot\|_1$ or $\|\cdot\|_2^2$, reducing which entry is more "efficient"?

	$(1-\delta, \varepsilon)$	$(1, \varepsilon-\delta)$
$\|\cdot\|_2$	$1+\varepsilon-\delta$	$1+\varepsilon-\delta$
$\|\cdot\|_2^2$	$1+\varepsilon^2-(2\delta-\delta^2)$	$1+\varepsilon^2-(2\varepsilon\delta-\delta^2)$

(assume $\delta < \varepsilon$, $\delta < 1$)

To obtain the same amount of diminish in $\|\cdot\|_2^2$,

reducing the "large" entry is more "efficient".

l_2-norm square "discourage" sparsity.

Explanation 3

Let $B_p := \{ x : \|x\|_p \le 1 \}$

In d-dimension, $\mathrm{vol}(B_\infty) = 2^d$

$$\mathrm{vol}(B_2) = \begin{cases} d \text{ is odd} & \pi^{(d-1)/2} \Big/ \left(\frac{d}{2}\right)\left(\frac{d-2}{2}\right)\left(\frac{d-4}{2}\right)\cdots\frac{1}{2} \\ d \text{ is even} & \frac{\pi^{d/2}}{(d/2)!} \end{cases}$$

$$\mathrm{vol}(B_1) = 2^d / d!$$

$d=2$, $\mathrm{vol}(B_1) = 0.5 \times \mathrm{vol}(B_\infty) \approx 0.63 \times \mathrm{vol}(B_2)$

$d=1000$, $\mathrm{vol}(B_1) \approx 10^{-2568} \times \mathrm{vol}(B_\infty) \approx 10^{-1485} \times \mathrm{vol}(B_2)$

Note.

l_1-norm regularization does NOT always give the sparsest solution

The sparsity of the solution depends on the contour of $\|Xw-y\|_2^2$.

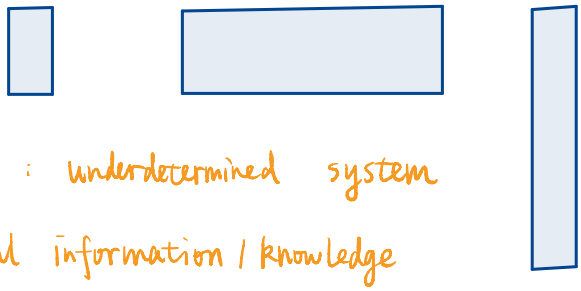
(or other fidelity loss we may see in the future)

Sparse signal modeling

$\underline{y} \in \mathbb{R}^m$ = $D \in \mathbb{R}^{m \times n}$ $\underline{x} \in \mathbb{R}^n$ $\quad m << n$

$\underline{y}$: observation

$\underline{x}$: unknown



solve x from y : underdetermined system

need additional information / knowledge

e.g. biological vision system

$y \sim$ visual sensor

$D = [d_1, d_2, \cdots, d_n] \sim$ dictionary of elementary patterns

$x \sim$ sparse coefficients

e.g. Compressed sensing

$x \sim$ signal. want to use less measurement (sampling) to recover x

Dx : encoding / sampling

$y \to x$: decoding

Before compressed sensing : Nyquist - Shannon sampling theorem

# of samples depends on signal frequency

can be expensive if signal contains both low-freq and high-freq components.

compressed sensing : around 2004.

if x is sparse, fewer samples are needed for perfectly recovering x

Problem formulation 1 : ℓ_0 - minimization

$$\min_x \quad \|x\|_0 \qquad \text{s.t.} \quad Dx = y$$

How to solve : exhaustive search

For $k = 0, 1, \cdots, n$

  For each $I \subseteq \{1, 2, \cdots, n\}$ of size k

    If the system $A_I z = y$ has a solution z

      Set $x_I = z$, $x_{I^c} = 0$

      return x

Issue : worst case complexity $n^{\|x_0\|_0}$ (x_0 ground truth)

Problem formulation 2 : ℓ_1 - minimization (relaxation)

$$\min_x \quad \|x\|_1 \qquad \text{s.t.} \quad Dx = y$$

Problem formulation 3 : Noisy observations or Approximate sparsity

Basis Pursuit : $\min_x \|x\|_1 \quad \text{s.t.} \quad \|y - Dx\|_2 \le \varepsilon$

LASSO $\min_x \lambda \|x\|_1 + \frac{1}{2}\|y - Dx\|_2^2$

Thm. Suppose $D_{ij} \sim$ iid $N(0, \frac{1}{m})$, $y = Dx_0 + z$

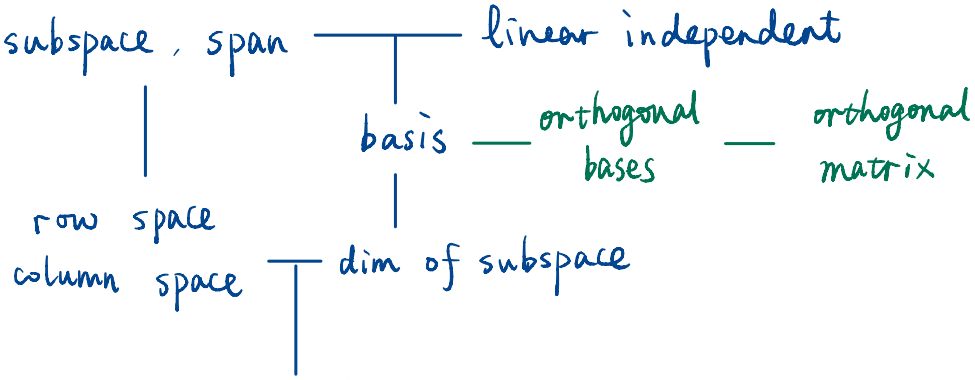
where x_0 is k-sparse, $z_i \sim$ iid $N(0, \frac{\sigma^2}{m})$

Then solving the LASSO with $\lambda = C \cdot 2\sigma\sqrt{\frac{\log n}{m}}$ for a large enough C

gives $\|\tilde{x} - x_0\|_2 \le C'\sigma\sqrt{\frac{k \log n}{m}}$ with high probability (w.h.p.)

Low-rank : "sparsity" of matrices

Review 1:



we care about rank b/c it tells at most how many dims of info is contained in the matrix or can be preserved in the linear transformation

Example: Netflix problem, low-rank matrix completion

	user 1	user 2	...
movie 1	9.8	?	
movie 2	7.6	5.4	
⋮		*	
			*
	?	3.2	

observing *'s

want to fill in ?'s

and make recommandations

Assumption: the matrix has low-rank (comparing to # of row/col)

need some better characterizations of rank for computational/theoretical studies.

Review 2: eigen-decomp of real symmetric matrices $A \in \mathbb{R}^{n\times n}$

$A = U \Lambda U^T$, $U \in \mathbb{R}^{n\times n}$ orthogonal

$\Lambda \in \mathbb{R}^{n\times n}$ diagonal

rank(A) = # of nonzero entries of diag(Λ)

How about general real matrices?

Singular Value Decomposition (SVD)

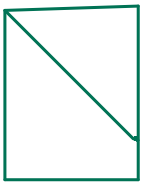
Any $A \in \mathbb{R}^{m\times n}$ can be factored into

$$A = U \Sigma V^T$$

$U \in \mathbb{R}^{m\times m}$ orthogonal, $\Sigma \in \mathbb{R}^{m\times n}$ diagonal, $V^T \in \mathbb{R}^{n\times n}$ orthogonal

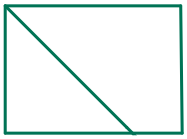
$m > n$

Σ



$m < n$

Σ



$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\min(m,n)} \geq 0$$

rank(A) = # of nonzero entries of diag(Σ)

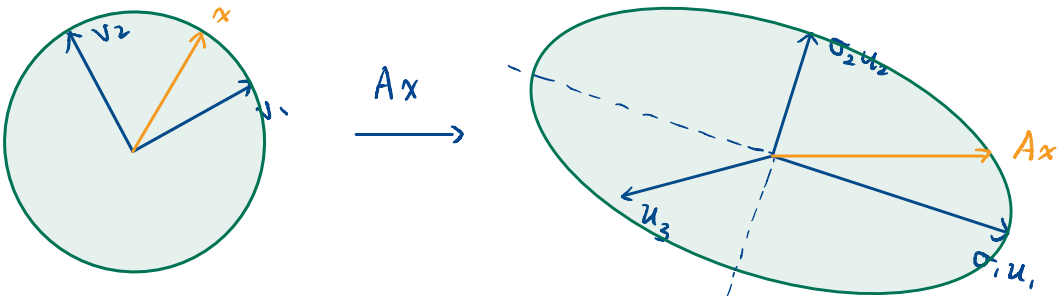
understanding SVD (take $m > n$ as an example)

Let $U = [u_1, u_2, \cdots, u_m]$, $V = [v_1, v_2, \cdots, v_n]$, $\Sigma =$

$$\begin{bmatrix} \sigma_1 & & \\ & \sigma_2 & \\ & & \ddots \\ & & \sigma_n \\ & 0 & \end{bmatrix}$$

then $AV = U\Sigma$

$$[Av_1 \quad Av_2 \quad \cdots \quad Av_n] = [\sigma_1 u_1 \quad \sigma_2 u_2 \quad \cdots \quad \sigma_n u_n]$$



Analogue: rank of matrices $\sim$ $\|\cdot\|_0$ of vectors

? $\sim$ $\|\cdot\|_1$ of vectors

$$\text{rank}(A) = \sum_{i=1}^{\min(m,n)} 1\{\sigma_i > 0\}$$

$$\text{nuclear norm} \quad \|A\|_* = \sum_{i=1}^{\min(m,n)} |\sigma_i|$$